(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5100493号
(P5100493)

(45)発行日　平成24年12月19日(2012.12.19)　(24)登録日　平成24年10月5日(2012.10.5)

(51)Int.Cl.		F I		
G06F 3/12	(2006.01)	G06F	3/12	C
B41J 29/38	(2006.01)	B41J	29/38	Z
H04N 1/00	(2006.01)	H04N	1/00	C
		G06F	3/12	D
		G06F	3/12	K

請求項の数 18　(全 38 頁)

(21)出願番号　特願2008-121986 (P2008-121986)
(22)出願日　平成20年5月8日 (2008.5.8)
(65)公開番号　特開2009-271745 (P2009-271745A)
(43)公開日　平成21年11月19日 (2009.11.19)
審査請求日　平成23年4月27日 (2011.4.27)

(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　100145827
弁理士　水垣　親房
(72)発明者　菅野　勝一
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　内田　正和

最終頁に続く

(54)【発明の名称】画像形成装置、ジョブ処理方法及びプログラム

(57)【特許請求の範囲】
【請求項１】
情報処理装置から取得するジョブを印刷処理する画像形成装置であって、
前記ジョブの印刷処理を中断した場合、前記ジョブの構造をヘッダ情報と、ページ実体情報とを含む構造化文章形式に変換する変換手段と、
前記変換手段により変換された前記構造化文章形式のジョブの前記ヘッダ情報を前記ジョブの印刷処理を中断した状況に応じて修正する修正手段と、
前記修正手段により前記ヘッダ情報を修正したジョブを保持する保持手段と、
を備えることを特徴とする画像形成装置。
【請求項２】
前記情報処理装置から取得するジョブの構造が構造化文章形式であるかどうかを判断する判断手段と、
前記判断手段により前記情報処理装置から取得するジョブの構造が構造化文章形式であると判断した場合は、前記修正手段は、前記変換手段による変換を行うことなく、前記情報処理装置から取得するジョブのヘッダ情報を前記ジョブの印刷処理を中断した状況に応じて修正することを特徴とする請求項１記載の画像形成装置。
【請求項３】
前記情報処理装置から取得するジョブの構造が構造化文章形式であるかどうかを判断する判断手段と、
前記判断手段により前記情報処理装置から取得するジョブの構造が構造化文章形式でな

いと判断した場合は、前記変換手段は、前記ジョブの印刷処理を中断した場合、前記ジョブの構造をヘッダ情報と、ページ実体情報とを含む構造化文章形式に変換することを特徴とする請求項１記載の画像形成装置。

【請求項４】

前記情報処理装置から取得するジョブの構造が構造化文章形式であるかどうかを判断する判断手段と、

前記判断手段が前記情報処理装置から取得するジョブの構造が構造化文章形式であると判断した場合、前記ヘッダ情報にコメントを示す情報が含まれているかどうかを判別する判別手段とを備え、

前記判別手段により前記ヘッダ情報にコメントを示す情報が含まれていると判別した場合、前記修正手段は、前記コメントと識別可能なコメントを前記ヘッダ情報に追加する修正を行い、

前記判別手段により前記ヘッダ情報にコメントを示す情報が含まれていないと判別した場合、前記修正手段は、前記ヘッダ情報にコメントを追加する修正を行うことを特徴とする請求項１記載の画像形成装置。

【請求項５】

前記保持手段に保持されたジョブのヘッダ情報に基づいて、当該ジョブの状態を修正前の状態に復元する復元手段を備えることを特徴とする請求項１記載の画像形成装置。

【請求項６】

前記保持手段に保持されるジョブに対して、中断箇所から再開印刷するか、最初から再印刷するかを指示する指示手段と、

前記指示手段により再開印刷が指示されている場合は、前記保持手段に保持されているジョブを中断した箇所からページ実体情報の印刷を再開し、前記指示手段により再印刷が指示されている場合は、前記復元手段により前記保持手段に保持されているジョブのヘッダ情報を中断前の状態に復元してから、前記ジョブのページ実体情報の再印刷を開始する再開手段と、

を備えることを特徴とする請求項５記載の画像形成装置。

【請求項７】

前記指示手段により再開印刷又は再印刷が指示された場合、前記保持手段に保持されているジョブの出力先を選択する選択手段と、

前記選択手段により選択される出力先に前記保持手段により保持されているジョブを転送する転送手段と、

を有することを特徴とする請求項６記載の画像形成装置。

【請求項８】

前記修正手段は、前記ヘッダ情報に設定されている前記ページ実体情報へのリンクのうち、印刷済みのページへのリンクを削除して印刷済み状態に修正することを特徴とする請求項１記載の画像形成装置。

【請求項９】

前記復元手段は、前記ヘッダ情報に設定されている前記ページ実体情報へのリンクのうち、印刷済みのページへのリンクを復元して印刷可能な状態に復元することを特徴とする請求項５記載の画像形成装置。

【請求項１０】

ジョブ処理方法であって、

ジョブの印刷処理を中断した場合、構造化文章形式のジョブのヘッダ情報を前記ジョブの印刷処理を中断した状況に応じて修正する修正ステップと、

前記修正ステップにより前記ヘッダ情報を修正したジョブを保持手段に保持する保持ステップと、

ジョブの構造が構造化文章形式であるかどうかを判断する判断ステップと、

前記判断ステップによりジョブの構造が構造化文章形式でないと判断した場合に前記ジョブの印刷処理を中断した場合、前記ジョブの構造をヘッダ情報と、ページ実体情報とを

含む構造化文章形式に変換する変換ステップと、
を備えることを特徴とするジョブ処理方法。
【請求項１１】
ジョブの構造が構造化文章形式であるかどうかを判断する判断ステップを備え、
前記判断ステップによりジョブの構造が構造化文章形式であると判断した場合は、前記修正ステップは、前記ジョブのヘッダ情報を前記ジョブの印刷処理を中断した状況に応じて修正することを特徴とする請求項１０記載のジョブ処理方法。
【請求項１２】
ジョブの構造が構造化文章形式であるかどうかを判断する判断ステップと、
前記判断ステップがジョブの構造が構造化文章形式であると判断した場合、前記ヘッダ情報にコメントを示す情報が含まれているかどうかを判別する判別ステップとを備え、
前記判別ステップにより前記ヘッダ情報にコメントを示す情報が含まれていると判別した場合、前記修正ステップは、前記コメントと識別可能なコメントを前記ヘッダ情報に追加する修正を行い、
前記判別ステップにより前記ヘッダ情報にコメントを示す情報が含まれていないと判別した場合、前記修正ステップは、前記ヘッダ情報にコメントを追加する修正を行うことを特徴とする請求項１０記載のジョブ処理方法。
【請求項１３】
前記保持手段に保持されたジョブのヘッダ情報に基づいて、当該ジョブの状態を修正前の状態に復元する復元ステップを備えることを特徴とする請求項１０記載のジョブ処理方法。
【請求項１４】
前記保持手段に保持されるジョブに対して、中断箇所から再開印刷するか、最初から再印刷するかを指示する指示ステップと、
前記指示ステップにより再開印刷が指示されている場合は、前記保持手段に保持されているジョブを中断した箇所からページ実体情報の印刷を再開し、前記指示ステップにより再印刷が指示されている場合は、前記復元ステップにより前記保持手段に保持されているジョブのヘッダ情報を中断前の状態に復元してから、前記ジョブのページ実体情報の再印刷を開始する再開ステップと、
を備えることを特徴とする請求項１３記載のジョブ処理方法。
【請求項１５】
前記指示ステップにより再開印刷又は再印刷が指示された場合、前記保持手段に保持されているジョブの出力先を選択する選択ステップと、
前記選択ステップにより選択される出力先に前記保持手段により保持されているジョブを転送する転送ステップと、
を有することを特徴とする請求項１４記載のジョブ処理方法。
【請求項１６】
前記修正ステップは、前記ヘッダ情報に設定されている前記ページ実体情報へのリンクのうち、印刷済みのページへのリンクを削除して印刷済み状態に修正することを特徴とする請求項１０記載のジョブ処理方法。
【請求項１７】
前記復元ステップは、前記ヘッダ情報に設定されている前記ページ実体情報へのリンクのうち、印刷済みのページへのリンクを復元して印刷可能な状態に復元することを特徴とする請求項１３記載のジョブ処理方法。
【請求項１８】
ジョブの印刷処理を中断した場合、構造化文章形式のジョブのヘッダ情報を前記ジョブの印刷処理を中断した状況に応じて修正する修正ステップと、
前記修正ステップにより前記ヘッダ情報を修正したジョブを保持手段に保持する保持ステップと、
ジョブの構造が構造化文章形式であるかどうかを判断する判断ステップと、

前記判断ステップによりジョブの構造が構造化文章形式でないと判断した場合に前記ジョブの印刷処理を中断した場合、前記ジョブの構造をヘッダ情報と、ページ実体情報とを含む構造化文章形式に変換する変換ステップと、
をコンピュータに実行させることを特徴とするプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、情報処理装置から取得するジョブを処理する画像形成装置に関するものである。
【背景技術】
【０００２】
情報処理装置と画像形成装置とが通信して取得するジョブ（以下、単にＪｏｂと呼ぶ場合がある）を処理する従来の印刷システムにおいて、印刷中のＪｏｂをなんらかの理由（エラー事象の発生や割り込み印刷Ｊｏｂなどにより）で中断し再開印刷する場合がある。この場合において、従来は中断Ｊｏｂと、何処まで印刷したかといった管理情報の２種類の情報で管理し、再開印刷あるいは最初から再印刷などを制御していた。
【０００３】
そして、中断処理を行うデバイス（あるいはPull Print の場合はサーバも含む）が再開あるいは再印刷などを特定する必要があった。
【０００４】
また、ＰＤＦやＰＤＬなどＰａｇｅ毎に独立でない印刷Ｊｏｂに関しては両者の切り替え制御は困難であり、中断したデバイスが中間データを活用して続きを行う等しか実現出来なかった。
【０００５】
そのため、中断時のＪｏｂアーカイブや割り込み時に異なったデバイスへの印刷Ｊｏｂ転送など必ずしも中断したデバイスで再開するとは限らないケースでは再開が困難であり、転送された側がSolutionや状況に応じて途中からの印刷再開印刷か、最初から再印刷などの制御はできなかった。
【特許文献１】特開２００６－９９７２５号公報
【発明の開示】
【発明が解決しようとする課題】
【０００６】
このように従来の印刷システムでは、Ｊｏｂを中断した時や、その中断したＪｏｂを異なったデバイスへ転送する場合などにおいて、中断Ｊｏｂの再開時に中断箇所から印刷するのか、最初から再印刷するのかという課題がある。
【０００７】
また、Ｊｏｂを転送する場合には転送された側の印刷機のSolutionによってはＪｏｂの印刷すら行えない場合がある。
【０００８】
さらに、中断したデバイスそのものでＪｏｂの再開をしようと、転送されたデバイスでＪｏｂの再開をしようとも、中断箇所からの印刷も、最初からの全項再印刷にも対応できないという課題があった。
【０００９】
本発明は、上記の課題を解決するためになされたもので、本発明の目的は、印刷処理が中断したジョブの構造を再開印刷または再印刷を指示可能な状態に修正できる仕組みを提供することである。
【課題を解決するための手段】
【００１０】
上記目的を達成する本発明の画像形成装置は以下に示す構成を備える。
【００１１】

情報処理装置から取得するジョブを印刷処理する画像形成装置であって、前記ジョブの印刷処理を中断した場合、前記ジョブの構造をヘッダ情報と、ページ実体情報とを含む構造化文章形式に変換する変換手段と、前記変換手段により変換された前記構造化文章形式のジョブの前記ヘッダ情報を前記ジョブの印刷処理を中断した状況に応じて修正する修正手段と、前記修正手段により前記ヘッダ情報を修正したジョブを保持する保持手段とを備えることを特徴とする。

【発明の効果】

【００１２】

本発明によれば、印刷処理が中断したジョブの構造を再開印刷または再印刷を指示可能な状態に修正できる。

【発明を実施するための最良の形態】

【００１３】

次に本発明を実施するための最良の形態について図面を参照して説明する。

【００１４】

＜システム構成の説明＞

〔第１実施形態〕

図１は、本実施形態を示す印刷システムの一例を示すブロック図である。なお、特に断らない限り、本発明の機能が実行されるのであれば、単体の機器であっても、複数の機器からなるシステムであっても、ＬＡＮ,ＷＡＮ等のネットワークを介して接続が為され処理が行われるシステムであっても本発明を適用できることは言うまでもない。また、本印刷システムは、情報処理装置と画像形成装置とが双方向通信可能に接続されている。ここで、双方向通信手段は、ネットワーク、双方向通信インタフェースのいずれでもよい。

【００１５】

さらに、本実施形態では、画像形成装置の例としてプリンタの例を示すが、画像形成装置としては、プリント機能を備える複合機（MFP：Ｍｕｌｔｉ Ｆｕｎｃｔｉｏｎ Ｐｅｒｉｐｈｅｒａｌ）や、ファクシミリ装置等であってもよい。ここで、画像形成装置は、情報処理装置からジョブを取得して、後述する構造化文章であるかどうかの判断処理や、当該判断処理に基づいて、中断したジョブの状況、既に印刷済みのページが存在する等のように中断した時点の状況に適応して印刷を再開するために、ジョブを構造化文章への変換処理、変換した構造化文章の修正処理を行う。

【００１６】

図１において、４０００はホストコンピュータで、ＲＯＭ３のプログラム用ＲＯＭあるいは外部メモリ１１に記憶された文書処理プログラム等に基づいて図形、イメージ、文字、表（表計算等を含む）等が混在した文書処理を実行するＣＰＵ１を備える。さらに、ＣＰＵ１は、後述するシステムバス４に接続される各デバイスを総括的に制御する。

【００１７】

また、このＲＯＭ３のプログラム用ＲＯＭあるいは外部メモリ１１には、ＣＰＵ１の制御プログラムであるオペレーティングシステム（以下ＯＳ）等を記憶する。また、ＲＯＭ３のフォント用ＲＯＭあるいは外部メモリ１１には上記文書処理の際に使用するフォントデータ等を記憶する。さらに、ＲＯＭ３のデータ用ＲＯＭあるいは外部メモリ１１には上記文書処理等を行う際に使用する各種データを記憶する。なお、外部メモリ１１には、アプリケーションからの印刷要求に従い、プリンタに出力する印刷情報を生成するプリンタドライバもインストールされている。

【００１８】

また、本実施形態に示すプリンタドライバは、ＸＰＳ（XML Paper Specification）という構造化されたＪｏｂをプリンタに出力する機能を備えている。なお、アプリケーションからいう構造化されたＪｏｂをプリンタに出力する機能を備える構成としてもよい。

【００１９】

２はＲＡＭで、ＣＰＵ１の主メモリ、ワークエリア等として機能する。５はキーボードＩ／Ｆ（ＫＢＣ）で、キーボード（ＫＢ）９や不図示のポインティングデバイスからのキ

ー入力を制御する。

【００２０】

６はディプレイＩ／Ｆ（ＣＲＴＣ）で、ディスプレイ（ＣＲＴ）１０の表示を制御する。７は外部メモリＩ／Ｆ（ＤＫＣ）で、ハードディスク（ＨＤ）、フレキシブルディスク（ＦＤ）等の外部メモリ１１とのアクセスを制御する。なお、外部メモリ１１には、ブートプログラム、各種のアプリケーション、フォントデータ、ユーザファイル、編集ファイル、プリンタドライバ等を記憶している。

【００２１】

８はプリンタＩ／Ｆ（ＰＲＴＣ）で、所定の双方向性インタフェース２１を介してプリンタ５０００に接続されて、プリンタ５０００との通信制御処理を実行する。

【００２２】

なお、ＣＰＵ１は、例えばＲＡＭ２上に設定された表示情報ＲＡＭへのアウトラインフォントの展開（ラスタライズ）処理を実行し、ディスプレイ１０上でのＷＹＳＩＷＹＧを可能としている。

【００２３】

また、ＣＰＵ１は、ＣＲＴ１０上の不図示のマウスカーソル等で指示されたコマンドに基づいて登録された種々のウインドウを開き、種々のデータ処理を実行する。ユーザは印刷を実行する際、印刷の設定に関するウインドウを開き、プリンタの設定や、印刷モードの選択を含むプリンタドライバに対する印刷処理方法の設定を行える。

【００２４】

プリンタ５０００において、１２はプリンタＣＰＵ（ＣＰＵ）で、ＲＯＭ１３のプログラム用ＲＯＭに記憶された制御プログラム等あるいは外部メモリ１４に記憶された制御プログラム等に基づいてシステムバス１５に接続される印刷部（プリンタエンジン）１７に出力情報としての画像信号を出力する。なお、プリンタ５０００は、後述するフローチャートの処理を実行することで、構造化文章形式のジョブをクライアントコンピュータ４０００から受信して処理する。ここで、構造化文章形式として、ＸＰＳ（XML Paper Specification）形式のジョブ処理を行うことが可能であるが、構造化文章形式としてはＸＰＳに限定されるものではない。また、クライアントコンピュータ４０００から受信するジョブが上記構造化文章形式でない場合は、後述ＣＰＵ１２がジョブの構造を構造化文章形式に変換して外部メモリ１４に保持させる。つまり、プリンタ５０００は、プリンタ５０００自身で、中断したジョブを再開する場合に、ジョブを扱い易い形式に自ら変換することで、ＰＤＦやＰＤＬ等のＰａｇｅ毎に独立でない印刷ジョブでも、プリンタ５０００の判断で、中断したジョブの印刷を開始するジョブ処理が可能となる。ここで、印刷処理を再開する場合に、継続印刷と、全項再印刷とのいずれにも対応可能であるが、その処理については後述する。また、全項再印刷とは、中断したページからではなく、最初のページから全てのページを印刷する処理であり、継続印刷とは、中断する前に正常に印刷されたページを除く、未印刷ページから印刷を再開する処理である。

【００２５】

また、このＲＯＭ１３のプログラムＲＯＭには、ＣＰＵ１２の制御プログラム等を記憶する。ＲＯＭ１３のフォント用ＲＯＭには上記出力情報を生成する際に使用するフォントデータ等を記憶し、ＲＯＭ１３のデータ用ＲＯＭはハードディスク等の外部メモリ１４がないプリンタの場合には、ホストコンピュータ上で利用される情報等を記憶している。

【００２６】

ＣＰＵ１２は、入力部１８を介してホストコンピュータとの通信処理が可能となっており、プリンタ内の情報等をクライアントコンピュータ４０００に通知可能に構成されている。

【００２７】

１９はＣＰＵ１２の主メモリ、ワークエリア等として機能するＲＡＭで、図示しない増設ポートに接続されるオプションＲＡＭによりメモリ容量を拡張することができるように構成されている。

【００２８】
なお、ＲＡＭ１９は、出力情報展開領域、環境データ格納領域、ＮＶＲＡＭ等に用いられる。前述したハードディスク（ＨＤ）、ＩＣカード等の外部メモリ１４は、メモリコントローラ２０によりアクセスを制御される。１６は印刷部インタフェースである。
【００２９】
外部メモリ１４は、オプションとして接続され、フォントデータ、エミュレーションプログラム、フォームデータ等を記憶する。また、２２は操作部で、印刷機能を設定したりするスイッチ、ステータスを表示するためのＬＥＤ表示器等が配されている。なお、操作部２２は、大型液晶表示装置を備えたタッチパネルで構成されていてもよい。また、前述した外部メモリは１個に限らず、少なくとも１個以上備え、内蔵フォントに加えてオプションフォントカード、言語系の異なるプリンタ制御言語を解釈するプログラムを格納した外部メモリを複数接続できるように構成されていてもよい。
【００３０】
さらに、プリンタ本体内部のＲＡＭ１９上に格納されているデータをコピーして格納できるようになっていても良い。また図示しないＮＶＲＡＭを有し、操作部２２からのプリンタモード設定情報を記憶するようにしてもよい。
【００３１】
図２は、本実施形態を示す画像形成装置におけるジョブ中断処理を説明する図である。以下、図３に示すジョブステータスと構造化文書例との対応を参照しながら、ジョブ中断処理について説明する。
【００３２】
図２において、２０１は中断箇所記憶Stepで、プリンタ５０００上で印刷中のＪｏｂがその印刷機上でのエラーや、緊急優先度を持った別の後続Ｊｏｂの割り込み等による何らかの理由により中断され、その時点の中断箇所を印刷機上のＲＡＭ１９で記憶しておくStepである。
【００３３】
図３に示す状態３０９は、Ｊｏｂの投入時点のＪｏｂ管理情報の状態を示している。このＪｏｂ管理情報は、プリンタ５０００上のＲＡＭ１９上で管理される。Ｊｏｂ名３１０はJob1でこれから印刷すべきページは１ページ目、ステータスは未印刷という情報を管理する。もちろんこれ以外の情報を管理しても良い。このこれから印刷すべきページが図２の中断箇所情報２０４として、図３に示す印刷ページ３１１やステータス３１２で示されている。なお、図２に示す中断箇所情報２０４はＲＡＭ１９に管理される。
【００３４】
図３に示す状態３１３は、２ページ目印刷完了した時点で中断されたＪｏｂ管理情報の状態を示している。このＪｏｂ管理情報は図３に示す状態３０９が遷移したものである。Ｊｏｂ名３１４はJob1でこれから印刷すべきページは３ページ目、ステータスは中断という情報を管理する。もちろんこれ以外の情報を管理しても良い。
【００３５】
また、図２に示す中断Ｊｏｂ更新Step２０２は、中断された構造化データ形式のＪｏｂの構造部分を修正し、Ｊｏｂの内部に中断したページ位置を記憶させるStepである。
【００３６】
さらに、図２に示すＪｏｂステータスの更新２０５は、図３に示すＪｏｂステータス３１２を図３で示されるようなＪｏｂステータス３１６に遷移させることを示している。なお、３０２、３０６はヘッダ部で、構造部分３０３、３０７を指示している。また、実施形態に示す構造化文章では、構造部分３０３、３０７はページ実体部３０４，３０８へのリンクが設定される構成となっている。本実施形態では、ヘッダ部と構造部分とをまとめてヘッダ情報とし、ページ実体部を構成する各ページの情報をページ実体情報とする。
【００３７】
また、図２に示す構造部分の操作２０６で未印刷部分を示すＪｏｂに更新というのは、構造化データ形式のＪｏｂの構造部分のリンクを手直しすることにより、ページ実体部分

はそのまま残した状態で中断された箇所以降を示すＪｏｂに変換することを示す。図３に示すＪｏｂ３０１は、Ｊｏｂ投入時点の状態で、Ｊｏｂ３０５が途中で中断された例を示している。

【００３８】

Ｊｏｂ３０１において、構造部分３０３からページ実体部３０４へのリンクは全てのページにおいてリンクされているが、Ｊｏｂ３０１が遷移したＪｏｂ３０５では、構造部分３０７からページ実体３０８へのリンクは、印刷済みのページにおいて切れている。

【００３９】

しかし、印刷済みのページ実態はそのまま残しておく。このように構造部分のページリンクを中断したページ以降のみを指し示すように組み替えることにより、データそのものに中断箇所情報を埋め込むことが可能となるのである。

【００４０】

図２に示す中断Ｊｏｂ再開Step２０３は中断したＪｏｂを再度印刷するStepであり、構造部分を操作したＪｏｂを印刷２０７では、構造化されたＪｏｂの構造部分を組み替えて中断箇所以降を指し示すように修正されたＪｏｂを印刷する。

【００４１】

Ｊｏｂ自身が何ページ目から印刷すべきかを記憶しているので、そのまま新規Ｊｏｂと同じ扱いで再印刷を行うことで、中断箇所以降のページの印刷を行うことが可能となる。

【００４２】

図４は、本実施形態を示す画像形成装置で処理されるジョブの構造を説明する図である。

【００４３】

図４において、４０１は一般的な構造化されたＪｏｂを示している。Documentsという構造部分と、n.fpageという複数のページ実体を示す部分で構成される。ここで、nはページ番号と一致し、順次カウントアップされてゆく。なお、本発明では、構造部分とページ実体部分を、ページリンクを用いてまとめた構造化文章で構成されたＪｏｂについて言及するものであり、構造化された文章であれば、図４に示す例以外にも適用されるのは言うまでもない。

【００４４】

４０２はDocumentsの一例を示している。ここではXML形式の文章でDocumentsを表現しているが、この形式だけに限定されるものではない。

【００４５】

ここで、PageContent Sourceで始まる行が各ページ実体へのリンクとなっている。1.fpageから4.fpageまでがページ実体そのものが格納されているファイルである。この図４では合計４ページで構成されるドキュメントを構造化文章の形で示している。

【００４６】

図５は、図４に示したDocumentを修正することによりページリンクを切るところを示した図である。

【００４７】

図５において、５０１、５０２はDocumentで、Document５０２はページリンクを切るための独自のTagが追加されている。

【００４８】

Tag５１１のPrint status="Done"は、Tag５１２の／Printと１つの対になっており、Tag５１３のPrint status="None"はTag５１４の／Printと１つの対になっている。

【００４９】

Tag５２１とTag５２２の間にあるPageContent Source行が、印刷済みのページリンクを示しており、Tag５２３とTag５２４の間にあるPageContent Source行が未印刷のページリンクを示している。

【００５０】

このDocument５０１の例は、まだ印刷していないDocumentの例である。一方、Document

５０２は、Tag５２１のPrint status="Done"とTag５２２の／Printとの間に1.fpageと2.fpageへのリンクが張られているPageContent Source行があるので、１、２ページ目が印刷済みであることを示している。

【００５１】

更に、Tag５２３のPrint status="None"及びTag５２４の／Printの間には3.fpageと4.fpageへのリンクが張られているPageContent Source行があるので３、４ページ目が未印刷であることを示している。

【００５２】

なお、図４に示したDocument４０２と図５に示すDocument５０１は全ページに対してリンクが張られているという意味において等価である。

【００５３】

また、図４の例においてDocument４０２をDocument５０２で置き換えたDocument４０１を作成すると、そのデータは３ページ以降しか印刷しなくなる。

【００５４】

一方、図４の例においてDocument４０２をDocument５０１で置き換えたDocument４０１を作成すると、そのデータは全ページ印刷するように戻すことが可能となる。本実施形態では、いずれの場合でもページ実体部分４０３～４０６はそのまま残し構造部分の操作だけで実現させる。

【００５５】

図６は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、図５に示したDocumentの修正処理に対応する。なお、Ｓ６０２～Ｓ６０４は各ステップを示す。図６の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【００５６】

Ｊｏｂ更新からDocumentの更新処理がスタートすると、Ｓ６０２で、ＣＰＵ１２は、印刷したページ実体を示す行をサーチする。具体的には、例えば１，２ページの印刷が完了した時点で、ｊｏｂ更新処理が起動されていたならば、その１、２ページにリンクを張っている行を探し出す。

【００５７】

なお、ページ実体はn.fpage（nはページ番号）という形で表現されているので、具体的には1.fpageと2.fpageにリンクを貼っている行を見つけることになる。

【００５８】

次に、Ｓ６０３で、ＣＰＵ１２は、Ｓ６０２で見つかったページ実体を示す行を印刷済みに変更する。具体的には、図５に示したDocument５０２において、Tag５２１のPrint status="Done"とTag５２２の／Printを1.fpageと2.fpageへのリンク行を包含する形で挿入する。

【００５９】

そして、Ｓ６０４で、ＣＰＵ１２は、残りのページ実体を示す行を未印刷に変更して、本処理を終了する。具体的には、Tag５２３のPrint status="None"及びTag５２４の／Printを3.fpageと4.fpageへのリンク行を包含する形で挿入する。

【００６０】

そして次は６０５終了となる。

【００６１】

以上の処理により、図５に示すDocument５０２の作成が可能となる。これにより、印刷処理を中断したジョブ状態から、既に印刷済みのページを除いた状態から中断ジョブを再開することができる。

【００６２】

図７は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、Document５０２の形式のDocumentをDocument５０１の形式のDocu

mentに戻すに対応する。なお、Ｓ６０７、Ｓ６０８は各ステップを示す。図６の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【００６３】

まず、Ｓ６０７で、ＣＰＵ１２は、印刷済みのページ実体を示す行をサーチする。具体的には、Document５０２の内部を検索してPrint status="Done"と／Printの対を探し出す。次に、Ｓ６０８で、ＣＰＵ１２は、見つかったページ実体を示す行を未印刷に変更して、本処理を終了する。このように、ＣＰＵ１２は、ジョブの状態を"Done"の状態を修正前の"None"の状態に復元することができる。

【００６４】

具体的には、Print status="Done"と／Printの対の間にはページリンクを示す行を入れないようにして、全てのページリンクを示す行をPrint status="None"と／Printの対の間に入れるように変換する。この操作により、Document５０２はDocument５０１に戻すことが可能となる。つまり、Document５０２はDocument５０１に戻すことにより、印刷処理を中断する前のジョブ状態に復元して、中断したジョブの印刷を再開することができる。

【００６５】

なお、図４に示す例では構造を示すDocumentsにXML文章を採用したので、図５においても未印刷及び印刷済みを示す行に当発明独自のXMLのTagを使った構文を例に挙げて説明した。しかしながら、はじめに述べたとおり、構造部分はXMLに限定されるものではない。

【００６６】

従って、当然未印刷及び印刷済みを示す方法はこれに限定されるものではなく、構造部分のフォーマットにしたがった最適な方法で表現することも本発明に含んでいる。以上のことにより、構造部分とページ実体部分をページリンクで接続する構造化文章において、構造部分の操作により印刷済みページへのリンクを切ることにより、未印刷部分のみを印刷するような構造化文章への変換が可能となる。また、ページ実体はそのまま残っているのでページリンクを元に戻すだけで、元の全ページ印刷する構造化文章に戻すことが可能となる。

【００６７】

更にこのシステムの特徴として、印刷済みページがどこなのかという情報は構造化文章そのものに記憶されており、他のシステム等は一切必要としていない点にも言及しておく。

【００６８】

〔第２実施形態〕

図８、図９は、本実施形態を示す画像形成装置で処理される構造化文章の内部構造を説明する図である。

【００６９】

図８において、７０１は例えば米国Microsoft社が規定しているＸＰＳという構造化されたＪｏｂの例を示している。

【００７０】

ＸＰＳ（XML Paper Specification）では、.relsがFixedDocumentSequence.fdseq をリンクし、これが更に FixedDocument.fdocをリンクし、これが更にn.fpageというページ実体部分をリンクする構造をとっている。７０２はFixedDocumentSequence.fdseqの一例である。この中で、後述するFixedDocument.fdocへのリンクが張られている。７０３はFixedDocument.fdocの一例である。この中のTag７３１及びTag７３２がページ実体へのリンクとなっている。

【００７１】

７０４及び７０５はページ実体を示している。このページ実体７０４、７０５は主にTextやJPEGファイルなどで構成されるがこれだけに限定されるものではない。

【００７２】

図９において、８０１は、例えばＸＰＳという構造化された他のＪｏｂの例を示している。ＸＰＳは作成するアプリケーションによっては構造部分の作りが異なってくる。この一例では、複数のFixedDocument.fdocがある例を示している。８０２はFixedDocumentSequence.fdseqの一例である。この中で複数のFixedDocument.fdocへのリンクが張られている。８０３及び８０４はFixedDocument.fdocの一例である。この各々の中、Tag８３１、Tag８３２、Tag８４１、Tag８４２がページ実体へのリンクとなっている。

【００７３】

８０５～８０８はページ実体を示している。このページ実体８０５～８０８は、主にTextやJPEGファイルなどで構成されるがこれだけに限定されるものではない。

【００７４】

図１０は、図８、図９に示したＸＰＳにおけるFixedDocument.fdocの例を示す図である。本例は、修正によって、印刷済みページに対するページリンクを切る例を示している。ここでは全５ページDocumentの中で、１、２ページが印刷済の例として以下に述べる。

【００７５】

クライアントコンピュータ４０００のアプリケーション又はプリンタドライバがＸＰＳ文章を作成した直後のFixedDocument.fdocは、図１０に示すFixedDocument.fdoc９０３もしくはFixedDocument.fdoc９０４で示した例のような形式をしている。

【００７６】

FixedDocument.fdoc９０３とFixedDocument.fdoc９０４の違いはFixedDocument.fdoc９０３の方にはTag９３１で示したコメント行が入っている。これはＸＰＳ文章を作成したドライバやアプリケーションが追加するコメントの例である。

【００７７】

一方、FixedDocument.fdoc９０４にはドライバやアプリケーションが作成したコメントは一切入っていない。

【００７８】

図１０に示すFixedDocument.fdoc９０２は１、２ページ目が印刷済みなため、１、２ページ目へのリンクをTag９２１、９２２で示したようにコメント化して切ったところを示している。これは、FixedDocument.fdoc９０４を修正した結果できあがるFixedDocument.fdocであり、この中に存在するコメントはページリンクを切っているTag９２１、９２２のコメントのみである。図１０に示すFixedDocument.fdoc９０１は１、２ページ目が印刷済みなため、１、２ページ目へのリンクをTag９１２、９１３で示したようにコメント化して切ったところを示している。これは、FixedDocument.fdoc９０３を修正した結果できあがるFixedDocument.fdocである。Tag９１２とTag９１３のコメントの中には、本実施形態で追加したコメントであることを示すC-LBPという文字列が、Tag９１２、９１３のコメントの先頭に追加されている。これにより、プリケーションが追加したTag９１１のコメントと、Tag９１２とTag９１３とを区別できるようになっている。なお、C-LBPという文字列はあくまでも一例であり、この文字列に限定するものではない。ここで、C-LBPは、プリンタドライバ等が追加したコメントと識別可能とするためのコメントである。

【００７９】

このFixedDocument.fdoc９０１の中に存在するコメントは、ＸＰＳ文章を作成したドライバやアプリケーションが追加したTag９１１のコメントと、本実施形態が追加したTag９１２、９１３のコメントである。しかしながら、これらは実施形態がコメント中に追加した文字列により区別することが可能である。以上の様に構造部分へのコメントを操作することにより、印刷済みページに対するページリンクを切り、未印刷部分のみを示すＪｏｂへと変換することが可能となる。つまり、一度切り離したリンクを復元することで、ページ実体部を構成する全てのページにリンクを付け加えて、リンクを修正する前の状態に復元することが可能となる。

【００８０】

図１１は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、Ｊｏｂ中のFixedDocument.fdocの修正処理である。また、本処

理は、図１０に示すFixedDocument.fdoc９０４、９０３のいずれの形式にも対応し、更に、図８のようにFixedDocument.fdocが１つだけの場合も図９のようにFixedDocument.fdocが複数ある場合にも対応する。

【００８１】

なお、Ｓ１００２～Ｓ１０１２は各ステップを示す。図１１の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【００８２】

FixedDocument.fdocの修正処理が開始すると、Ｓ１００２で、複数のFixedDocument.fdocへのポインタとなる変数Ｎを「１」に初期化しておく。そして、Ｓ１００３でＮが「１」の場合、まずはじめにＪｏｂ中にFixedDocument.fdocが有るか否かをＣＰＵ１２が判断する。ここで、ＣＰＵ１２がＪｏｂ中にFixedDocument.fdocがあると判断した場合、Ｓ１００４に進む。一方、Ｓ１００３で、Ｊｏｂ中にFixedDocument.fdocが無いとＣＰＵ１２が判断した場合は、続いて、Ｊｏｂ中にFixedDocument1.fdocがあるか否かをＣＰＵ１２が判断する。ここで、ＣＰＵ１２がFixedDocument1.fdocがあると判断した場合は、１００４に進む。それでもFixedDocument1.fdocが無いとＣＰＵ１２が判断した場合は、本処理を終了となる。以降、FixedDocument.fdocやFixedDocumentN.fdoc（Ｎが1から順にカウントアップされて行く）のことをまとめてFixedDocumentN.fdocと表現する。

【００８３】

次に、Ｓ１００４では、ＣＰＵ１２は、Ｓ１００３で存在を確認したFixedDocumentN.fdocを開く。そして、Ｓ１００５で開いたFixedDocumentN.fdocのコメントの存在をＣＰＵ１２が確認する。このCheckにより、ＸＰＳ文章を作成したクライアントコンピュータ４０００のドライバやアプリケーションが追加したコメントが入っているかいないかの判断を行うことができる。

【００８４】

次に、Ｓ１００６で、FixedDocumentN.fdocのコメントがあったか否かをＣＰＵ１２が判断する。ここで、コメントが無かったとＣＰＵ１２が判断した場合は、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが無いことを意味し、Ｓ１００７に進む。

【００８５】

一方、Ｓ１００６で、追加したコメントがあるとＣＰＵ１２が判断した場合には、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントがあることを意味し、Ｓ１０１１に進む。

【００８６】

そして、Ｓ１００７では、印刷済みとなっているn.fpageのある行をＣＰＵ１２が検索する。ここで、n.fpageのnはページ番号を示しているので、ここの部分による判断で印刷済みのn.fpageのある行の検索が行える。

【００８７】

そして、Ｓ１００８では、検索できたn.fpageのある行にコメントをＣＰＵ１２が挿入する。例えば図１０に示すFixedDocument.fdoc９０４の形式のFixedDocument.fdocにおいて、２ページまで印刷済みであった場合、1.fpageのある行と2.fpageのある行をコメント化する。その結果として図１０に示すFixedDocument.fdoc９０２におけるTag９２１、９２２のような行を作成する。

【００８８】

なお、図１０に示すFixedDocument.fdoc９０２におけるTag９２１、９２２の例では、１行１行にコメントの開始と終了を埋めているが、複数行まとめて１つのコメント開始と終了でくるんでも良い。本実施形態では、1.fpageと2.fpageのある行をコメント化できれば、その方法は問わない。

【００８９】

そして、Ｓ１００８の結果、FixedDocument.fdocに記述されているコメントは、本実施形態で追加したコメントだけであることが明白となっている。なお、本実施形態で追加したことを示す目印はない。

【００９０】

次に、Ｓ１００９では、ＣＰＵ１２は、コメントを追加・修正したFixedDocument.fdocを、例えば外部メモリ１４に保存し、FixedDocument.fdoc をCloseする。

【００９１】

そして、Ｓ１０１０では、複数のFixedDocument.fdocへのポインタとなる変数Ｎを１つカウントアップして、その後、Ｓ１００３に戻る。

【００９２】

一方、Ｓ１０１１では、Ｓ１００７と同様に印刷済みとなっているn.fpageのある行をＣＰＵ１２が検索する。ただし、ＣＰＵ１２は、FixedDocument.fdoc内部にＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが入っていることが分かっているので、その範囲を避けて検索を行う。

【００９３】

そして、Ｓ１０１２では、Ｓ１００８と同様に検索できたn.fpageのある行にコメントを挿入する。具体的には、本実施形態で追加するコメントがＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントと区別できるように、コメント内部に独自の目印を埋め込んだ形でコメント追加を行う。

【００９４】

図１０に示すFixedDocument.fdoc９０３の形式のFixedDocument.fdocにおいて２ページまで印刷済みであった場合、1.fpageのある行と2.fpageのある行をコメント化する。その結果として図１０に示すFixedDocument.fdoc９０１におけるTag９１２、９１３のような行を作成する。なお、図１０に示したFixedDocument.fdoc９０１の例では、この目印としてC-LBPという文字列を埋め込んでいるが、本実施形態ではこの文字列に限定されるものではない。

【００９５】

そして、コメント挿入後はＳ１００９に進み、コメントを追加・修正したFixedDocument.fdocを保存し、FixedDocument.fdoc をCloseする。

【００９６】

この図１１に示すＳ１０１２の結果、FixedDocument.fdocに記述されているコメントは、本実施形態が追加したコメント（本実施形態が追加したことを示す目印あり）とＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが入っていることが明白となっている。

【００９７】

以上の操作により、FixedDocument.fdocにページリンクを切るコメントを追加することができる。

【００９８】

また、この手順では、構造部が複数のFixedDocument.fdocで構成される場合や、ＸＰＳ文章を作成したドライバやアプリケーションがコメントを追加している場合にも対応することができる。

【００９９】

図１２は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、図１１に示す処理により印刷済ページへのページリンクが切られたFixedDocument.fdocを回復して、元の全ページ印刷するＪｏｂに戻す処理である。なお、Ｓ１１０２～Ｓ１１１０は各ステップを示す。図１２の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。また、本処理は、図１０のFixedDocument.fdoc９０１、９０２いずれの形式にも対応し、更に、図８のようにFixedDocument.fdocが１つだけの場合も図９のようにFixedDocumen

t.fdocが複数ある場合にも対応する。

【０１００】

FixedDocument.fdoc更新処理（回復処理）がスタートすると、Ｓ１１０２で、ＣＰＵ１２は、複数のFixedDocument.fdocへのポインタとなる変数Ｎ（ＲＡＭ２に保持される）を「１」に初期化する。

【０１０１】

次に、Ｓ１１０３で、変数Ｎが「１」の場合、まずはじめにFixedDocument.fdocが有るか否かをＣＰＵ１２が判断する。ここで、ＣＰＵ１２がFixedDocument.fdocがあると判断した場合、Ｓ１１０４に進む。また、ＣＰＵ１２がFixedDocument.fdocが無いと判断した場合は、続いて、ＣＰＵ１２はFixedDocument1.fdocがあるか否かを判断する（図示しない）。ここで、FixedDocument1.fdocがあるとＣＰＵ１２が判断した場合は、Ｓ１１０４に進み、それでも無いと判断した場合は、本処理を終了する。以降、FixedDocument.fdocやFixedDocumentN.fdoc（Ｎが1から順にカウントアップされて行く）のことをまとめてFixedDocumentN.fdocと表現する。

【０１０２】

次に、Ｓ１１０４で、ＣＰＵ１２は、Ｓ１１０３で存在を確認したFixedDocumentN.fdocを開く。次に、Ｓ１１０５で、ＣＰＵ１２は、開いたFixedDocumentN.fdocのコメントの中身をサーチする。具体的には、本実施形態で追加したコメントがあるか無いかをサーチする。このCheckにより、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが混在しているのかいないのかの判断を行うことができる。

【０１０３】

次に、Ｓ１１０６でコメントサーチの結果、本実施形態で追加したコメントがあるか無いかをＣＰＵ１２が判断する。ここで、ＣＰＵ１２が追加したコメントが無いと判断した場合には、Ｓ１１０７に進み、追加したコメントが有ると判断した場合にはＳ１１１０に進む。

【０１０４】

そして、Ｓ１１０７で、ＣＰＵ１２は、FixedDocument.fdoc内部の全てのコメントを解除して、コメントで囲まれていた場所を再び有効な記述に戻す処理を行う。

【０１０５】

なお、Ｓ１１０７に処理が移った時点で、FixedDocument.fdoc内部にあるコメントは全て本実施形態で追加したコメントであることが明白である。なお、このコメントには本実施形態で示した追加したことを示す目印はない。したがって、全てのコメントを解除すると全ページのドキュメントを印刷する構造に戻すことができることも明白である。

【０１０６】

よって、ここでは全てのコメントを解除し、コメントで囲まれていた場所を再び有効な記述に戻す処理を行う。

【０１０７】

そして、Ｓ１１０８で、ＣＰＵ１２は、修正処理の終わったFixedDocument.fdocを保存してCloseする処理を行う。次に、Ｓ１１０９では複数のFixedDocument.fdocへのポインタとなる変数Ｎを１つカウントアップして、その後、Ｓ１１０３に戻る。

【０１０８】

一方、Ｓ１１１０では、Ｓ１１０７と同様に、ＣＰＵ１２がFixedDocument.fdoc内部のコメントを解除するが、対象は本実施形態で追加したことを示す目印のついているコメントだけを解除する。

【０１０９】

なお、Ｓ１１１０に処理が移った時点で、FixedDocument.fdoc内部にあるコメントには本実施形態で追加したコメントとＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが混在していることが明白である。つまり、本実施形態で追加したことを示す目印を含むコメント行が存在している。

【０１１０】

したがって、全てのコメントを解除してしまうと、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントも解除することになってしまい、予期せぬ事態をもたらす可能性がある。

【０１１１】

そこで、この場合には本実施形態で追加したことを示す目印を含むコメントだけを解除して、コメントで囲まれていた場所を再び有効な記述に戻す処理を行う。この処理により、ページリンクを切っていたコメントだけ解除することができ、結果として全ページのドキュメントを印刷する構造に戻すことができる。以上の処理により、印刷済ページへのページリンクが切られたFixedDocument.fdocを回復して、元の全ページ印刷するＪｏｂに戻すことが可能となる。

【０１１２】

また、この手順では、複数のFixedDocument.fdocで構成される場合や、ＸＰＳ文章を作成したドライバやアプリケーションがコメントを追加する場合にも対応することができる。

【０１１３】

以上のことにより、構造部分とページ実体部分をページリンクで接続する構造化されたＸＰＳ文章において、構造部分の操作により印刷済みページへのリンクを切ることにより、未印刷部分のみを印刷するような構造化文章への変換が可能となる。

【０１１４】

また、ページ実体はそのまま残っているのでページリンクを元に戻すだけで、元の全ページ印刷を行う構造化されたＸＰＳ文章に戻すことが可能となる。

【０１１５】

更にこのシステムの特徴として、印刷済みページがどこなのかという情報は構造化文章そのものに記憶されており、他のシステム等は一切必要としていない点にも言及しておく。

【０１１６】

また、第２実施形態は、第１実施形態のように独自Tagの追加など行わず、構造部分にコメントの追加・削除を行うことで構造部分を操作している。従って、第２実施形態で作成されたＸＰＳ文章はＸＰＳの文法に準拠しているという点についても言及しておく。

【０１１７】

〔第３実施形態〕

図１３は、本実施形態を示す画像形成装置で処理されるＸＰＳにおけるFixedDocument.fdocの例を示す図である。本例は、後述するFixedDocumentSequence.fdseqの修正と一緒に行うことによって、印刷済みページに対するページリンクを切る例を示している。ここでは全５ページDocumentの中で、１、２ページが印刷済の例として以下に述べる。

【０１１８】

ＸＰＳ文章を作成した直後のFixedDocument.fdocは、図１３に示すFixedDocument.fdoc９５３もしくはFixedDocument.fdoc９５４で示した例のような形式をしている。

【０１１９】

FixedDocument.fdoc９５３とFixedDocument.fdoc９５４の違いはFixedDocument.fdoc９５３の方にはコメント行９５６が入っている。これはＸＰＳ文章を作成したドライバやアプリケーションが追加するコメントの例である。

【０１２０】

一方、FixedDocument.fdoc９５４にはドライバやアプリケーションが作成したコメントは一切入っていない。

【０１２１】

９５２はオリジナルのFixedDocument.fdocをコピーしたFixedDocument_add.fdocを修正した結果を表している。本例は、１、２ページ目が印刷済みな為、１、２ページ目へのリンク行を全て削除してリンクを切っている状態に対応する。９５１はオリジナルのFixedDocument.fdocをコピーしたFixedDocument_add.fdocを修正した結果を表している。本例は

、１、２ページ目が印刷済みな為、１、２ページ目へのリンク行を全て削除してリンクを切っている状態に対応する。修正した結果９５２との違いは、唯一コメント行９５５が入っている点である。これはＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントの例である。

【０１２２】

図１４は、本実施形態を示す画像形成装置で処理されるＸＰＳにおけるFixedDocumentSequence.fdseqの例を示す図である。本例は、先に説明したFixedDocument.fdocの修正と一緒に修正することによって、印刷済みページに対するページリンクを切る例を示している。

【０１２３】

図１４において、ＸＰＳ文章を作成した直後のFixedDocumentSequence.fdseqは、FixedDocumentSequence.fdseq９６４もしくはFixedDocumentSequence.fdseq９６３で示した例のような形式をしている。

【０１２４】

FixedDocumentSequence.fdseq９６４とFixedDocumentSequence.fdseq９６３の違いはFixedDocumentSequence.fdseq９６３の方にはコメント行９７０が入っている。これはＸＰＳ文章を作成したドライバやアプリケーションが追加するコメントの例である。一方、FixedDocumentSequence.fdseq９６４にはドライバやアプリケーションが作成したコメントは一切入っていない。

【０１２５】

FixedDocumentSequence.fdseq９６２はオリジナルのFixedDocument.fdocへのリンク（Tag９６８参照）をコメントにして、そのコピーであるFixedDocument_add.fdocへのリンクを追加している（Tag９６９参照）。

【０１２６】

先に説明したようにオリジナルのFixedDocument.fdocのコピーのFixedDocument_add.fdocの中身は印刷済みページへのリンクが削除されている。したがって、このような構造部分の修正により印刷済ページへのリンクを切ることが可能となる。

【０１２７】

FixedDocumentSequence.fdseq９６１はオリジナルのFixedDocument.fdocへのリンク（Tag９６６参照）をコメントにして、そのコピーであるFixedDocument_add.fdocへのリンクを追加している（Tag９６７参照）。

【０１２８】

FixedDocumentSequence.fdseq９６２との違いはコメント行９６５が入っている点である。これはＸＰＳ文章を作成したドライバやアプリケーションが追加するコメントの例である。

【０１２９】

コメント行９６６では、本実施形態で追加したことを示す目印となる文字列はコメント中に入っていない。これは、元に戻す際に、FixedDocument_add.fdoc 行を削除し、FixedDocument.fdoc行を戻すように処理すればよいためで、つまりは.fdoc行のファイル名で判別（_addが追加されているか否かにより）できるからである。

【０１３０】

図１５は、図１３、図１４に示したＸＰＳ構成をまとめたＸＰＳ文章９８０の構造を示す図である。本例は、図１４に示すFixedDocumentSequence.fdseqが新規に追加した図１３に示すixedDocument_add.fdocをリンクしてオリジナルのFixedDocument.fdocへのリンクは消えた状態である。

【０１３１】

各ページ実体はFixedDocument_add.fdocから印刷済ページを除いた分がリンクされており、FixedDocument.fdocからは全項リンクされている。この状態の構造化文章を印刷すると、印刷済ページを除いた分が印刷されることになる。本構造の内部では、FixedDocument.fdocからページ実体へのリンクは残っているが、FixedDocument.fdocに対する上位階層

からのリンクが無いので、こちらは有効にはならない。この状態の構造化文章を全項印刷Ｊｏｂに変換するには、FixedDocumentSequence.fdseqにおいてFixedDocument.fdocへのリンクを復活させ、代わりにFixedDocument_add.fdocへのリンクを切ればよい。

【０１３２】

図１６は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、FixedDocumentSequence.fdseqの更新と、FixedDocument.fdocの複製処理である。また、本処理は、図８に示した７０１の形式（FixedDocument.fdocが１つだけ）にも図９に示した８０１の形式（FixedDocument.fdocが複数）にも対応する。なお、Ｓ１０２２～Ｓ１０２８は各ステップを示す。図１６の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【０１３３】

Ｊｏｂ更新処理が開始されると、Ｓ１０２２で、ＣＰＵ１２はFixedDocumentSequence.fdseqを開く。そして、Ｓ１０２３で、ＣＰＵ１２は、FixedDocumentSequence.fdseqの内部を検索し、FixedDocument.fdocを順次見つけ出す。なお、最初はFixedDocument.fdocを検索し、次いで、FixedDocument1.fdoc、FixedDocument2.fdoc という具合にFixedDocumentN.fdoc（Ｎが１から順にカウントアップ） を検索して行く。以降FixedDocumentN.fdocと表現する。

【０１３４】

次に、Ｓ１０２４で検索対象のFixedDocumentN.fdocがあったか否かをＣＰＵ１２が判断する。ここで、ＣＰＵ１２があると判断した場合は、Ｓ１０２５に進み、無いと判断した場合は、Ｓ１０２６に進む。

【０１３５】

そして、Ｓ１０２５では、ＣＰＵ１２は、見つけたFixedDocumentN.fdocを複製してFixedDocumentN_add.fdocという名称で作成して、Ｓ１０２３へ戻る。

【０１３６】

なお、ファイル名に付け足した_addは複製されたFixedDocumentN.fdocであることを示すものである。本実施形態は、複製されたFixedDocumentN.fdocの識別の為に追加する名称を_addだけに限定するものではなく、名称で区別できるようになっていれば、これ以外の文字列が採用可能であることはいうまでもない。

【０１３７】

次に、Ｓ１０２６では、ＣＰＵ１２は、FixedDocumentSequence.fdseq内にあるFixedDocumentN.fdocをリンクしている行をコメント化してリンクを切る作業を行う。これでオリジナルの全項印刷へのリンクである、FixedDocumentN.fdocへのリンクが切れることになる。

【０１３８】

そして、Ｓ１０２７では、ＣＰＵ１２は、新規に複製したFixedDocumentN_add.fdocへのリンクを追加する作業を行う。これで、印刷を中断箇所以降の文章へのリンクであるFixedDocumentN_add.fdocへのリンクが新たに追加されることになる。

【０１３９】

その後、Ｓ１０２８で、ＣＰＵ１２は、追加処理の終わったFixedDocumentSequence.fdseqを保存してCloseする処理を行い、本処理を終了する。

【０１４０】

図１７は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、オリジナルのFixedDocumentN.fdocを複製して作成されたFixedDocumentN_add.fdocの修正処理である。

【０１４１】

また、本処理は、図１３に示したFixedDocument.fdoc９５４、９５３いずれの形式にも対応し、更に、図８のようにFixedDocument.fdocが１つだけの場合も図９のようにFixedDoc

ument.fdocが複数ある場合にも対応する。なお、Ｓ１０３２～Ｓ１０４２は各ステップを示す。図１６の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【０１４２】

FixedDocumentN_add.fdoc修正処理がスタートすると、Ｓ１０３２で、ＣＰＵ１２は複数のFixedDocumentN_add.fdocへのポインタとなる変数Ｎを「１」に初期化する。次に、Ｓ１０３３で、Ｎが「１」の場合まずはじめにFixedDocument_add.fdocが有るか否かをＣＰＵ１２が判断する。ここで、ＣＰＵ１２があると判断した場合、Ｓ１０３４に進み、無いと判断した場合は、続いてFixedDocument1_add.fdocがあるか否かをＣＰＵ１２が判断する（図示しない）。そして、FixedDocument1_add.fdocがあるとＣＰＵ１２が判断した場合、Ｓ１０３４に進み、それでも無いとＣＰＵ１２が場合は、本処理を終了する。以降、FixedDocument_add.fdocやFixedDocumentN_add.fdoc（Ｎが1から順にカウントアップされて行く）のことをまとめてFixedDocumentN_add.fdocと表現する。

【０１４３】

次に、Ｓ１０３４では、ＣＰＵ１２は、Ｓ１０３３で存在を確認したFixedDocumentN_add.fdocを開く。そして、Ｓ１０３５で、ＣＰＵ１２は、開いたFixedDocumentN_add.fdocのコメント行の存在を確認する。このCheckにより、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが入っているかいないかの判断を行うことができる。

【０１４４】

そして、Ｓ１０３６で、コメントがあったか否かをＣＰＵ１２が判断する。ここで、コメントが無いとＣＰＵ１２が判断した場合は、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが無いことを意味し、Ｓ１０３７に進む。

【０１４５】

一方、Ｓ１０３６で、コメントがあるとＣＰＵ１２が判断した場合には、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントがあることを意味し、Ｓ１０４１に進む。

【０１４６】

そして、Ｓ１０３７では、ＣＰＵ１２は、印刷済みとなっているn.fpageのある行を検索する。ここで、n.fpageのnはページ番号を示しているので、ここの部分による判断で印刷済みのn.fpageのある行の検索が行える。

【０１４７】

そして、Ｓ１０３８で、ＣＰＵ１２は、検索できたn.fpageのある行を削除する。

【０１４８】

図１３に示すFixedDocument.fdoc９５４の形式のFixedDocumentN_add.fdocにおいて２ページまで印刷済みであった場合、1.fpageのある行と2.fpageのある行を削除する。その結果として図１３に示すFixedDocument_add.fdoc９５２のようなFixedDocumentN_add.fdocを作成する。

【０１４９】

そして、Ｓ１０３９では、ＣＰＵ１２は、n.fpageへのリンクを削除したFixedDocumentN_add.fdocを保存しCloseする。そして、Ｓ１０４０で、ＣＰＵ１２は、複数のFixedDocumentN_add.fdocへのポインタとなる変数Ｎを１つカウントアップして、その後、Ｓ１０３３に戻る。

【０１５０】

一方、Ｓ１０４１では、ＣＰＵ１２がＳ１０３７と同様に印刷済みとなっているn.fpageのある行を検索するが、FixedDocumentN_add.fdoc内部にＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが入っていることが分かっているので、その範囲を避けて検索を行う。

【０１５１】

そして、Ｓ１０４２では、Ｓ１０３８と同様に検索できたn.fpageのある行を削除して

、Ｓ１０３９へ進む。これにより、図１３に示すFixedDocument.fdoc９５３の形式のFixedDocumentN_add.fdocにおいて２ページまで印刷済みであった場合、1.fpageのある行と2.fpageのある行を削除する。

【０１５２】

その結果として図１３のFixedDocumentN_add.fdoc９５１ようなFixedDocumentN_add.fdocを作成する。なお、FixedDocumentN_add.fdocに元々あったコメント行９５６はそのままコメント行９５５として残っている。以上の操作により、オリジナルのFixedDocument.fdocをそのまま残し、そのコピーであるFixedDocument_add.fdocを印刷済みページへのリンクを切るように構成することができる。

【０１５３】

その上で、FixedDocumentSequence.fdseqからのリンクをオリジナルのFixedDocument.fdocから切り離し、そのコピーであるFixedDocument_add.fdocへ切り替えることにより、未印刷部分のみを印刷するような構造化文章への変換が可能となる。

【０１５４】

また、この手順では、構造部が複数のFixedDocument.fdocで構成される場合や、ＸＰＳ文章を作成したドライバやアプリケーションがコメントを追加している場合にも対応することができる。

【０１５５】

図１８は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、図１５、図１６の操作により印刷済ページへのページリンクが切られたFixedDocumentSequence.fdseqを回復して、元の全ページ印刷するＪｏｂに戻す処理である。

【０１５６】

また、本処理は、図１４に示すFixedDocumentSequence.fdseq９６１、９６２いずれの形式にも対応する。なお、Ｓ１１２２～Ｓ１１２５は各ステップを示す。図１８の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【０１５７】

FixedDocumentSequence.fdseq更新処理（回復処理）がスタートすると、Ｓ１１２２で、ＣＰＵ１２は、FixedDocumentSequence.fdseqを開く。そして、Ｓ１１２３で、ＣＰＵ１２は、FixedDocumentSequence.fdseq内のFixedDocument_add.fdocへのリンクをはっている行を削除する。これで、印刷を中断した箇所以降を示すＪｏｂ構成になっている構造部分へのリンクが切れたことになる。なお、ここでは、中断箇所以降を示すＪｏｂ構成へのリンクがFixedDocument_add.fdocとなっておりオリジナルのリンクに _add が追加された形になっているが、本実施形態では、この文字列に限定するものではない。

【０１５８】

次に、Ｓ１１２４で、ＣＰＵ１２は、FixedDocumentSequence.fdseq内でオリジナルのFixedDocumentN.fdocへのコメントを削除して、リンクを復活させる。

【０１５９】

これで、オリジナルの全文印刷を行うFixedDocumentN.fdocへのリンクが復活したことになる。

【０１６０】

そして、Ｓ１１２５で、FixedDocumentSequence.fdseqを保存してCloseし、本処理が終了する。以上の処理により、印刷済ページへのページリンクが切られたFixedDocumentSequence.fdseqを回復して、元の全ページ印刷するＪｏｂに戻すことが可能となる。また、この手順では、ＸＰＳ文章を作成したドライバやアプリケーションがコメントを追加している場合にも対応することができる。

【０１６１】

以上のことにより、構造部分とページ実体部分をページリンクで接続する構造化された

ＸＰＳ文章において、構造部分の操作により印刷済みページへのリンクを切ることにより、未印刷部分のみを印刷するような構造化文章への変換が可能となる。

【０１６２】

また、ページ実体はそのまま残っているのでページリンクを元に戻すだけで、元の全ページ印刷を行う構造化されたＸＰＳ文章に戻すことが可能となる。

【０１６３】

更にこのシステムの特徴として、印刷済みページがどこなのかという情報は構造化文章そのものに記憶されており、他のシステム等は一切必要としていない点にも言及しておく。

【０１６４】

また、この第３実施形態は、第１実施形態のように独自Tagの追加など行わず、構造部分へのコメントの追加・削除や、リンクの削除を行うことで構造部分を操作している。

【０１６５】

従って、この第３実施形態で作成されたＸＰＳ文章はＸＰＳの文法に準拠しているという点についても言及しておく。

【０１６６】

図１９は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、構造化されていない文章の構造化処理である。

【０１６７】

本実施形態では、クライアントコンピュータ４０００からPCLやPostScriptといった構造化されていないＪｏｂがプリンタ５０００に投入された場合は、そのＪｏｂを構造化Ｊｏｂに変換してから処理を行う。なお、Ｓ１２０２～Ｓ１２０５は各ステップを示す。図１９の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【０１６８】

Ｊｏｂの中断が開始されると、Ｓ１２０２で、ＣＰＵ１２は、Ｊｏｂの中断処理を行い、Ｊｏｂを停止する。

【０１６９】

そして、Ｓ１２０３で、そのＪｏｂが構造化文章となっているか否かをＪｏｂのヘッダ等でＣＰＵ１２が判断する。ここで、PCLやPostScript、ＸＰＳなどは、Ｊｏｂの先頭数バイトを確認すれば、どれなのかの判断が行うことが可能である。ここで、扱うＪｏｂが、PCLやPostScript等で構造化文章でないとＣＰＵ１２が判断した場合は、Ｓ１２０４に進み、構造化文章であるとＣＰＵ１２が判断した場合には、Ｓ１２０５に進む。

【０１７０】

そして、Ｓ１２０４では、ＣＰＵ１２は、中断したＪｏｂを構造化文章に変換する処理を行う。具体的には、各ページのページ実体を、図４に示したページ実体部分４０３～４０６で示したように１つのファイル化する。このファイルの内部にはTextやJPEGなどの画像情報が含まれるが、これ以外の要素が含まれていても良い。

【０１７１】

その各ページ実体を、図４に示したようなリンク構造で構成される構造部分４０２を作成し、全体をＺｉｐなどのアーカイブツールにて一つにまとめて構造化文章を作成する。ここで、Ｚｉｐとは、フィル・カッツが考案および開発したデータ圧縮形式およびファイルフォーマットである。

【０１７２】

そして、Ｓ１２０５で、図３に示すステータス３１２をステータス３１６に変換するような、Ｊｏｂステータスの更新処理を行い、本処理を終了する。

【０１７３】

以上の処理により、クライアントコンピュータ４０００からPCLやPostScriptといった構造化されていないＪｏｂがプリンタ５０００に投入された場合でも、ＸＰＳのような構

造化文章と同一の処理に乗せることが可能となる。もちろんＪｏｂが初めからＸＰＳのような構造化文章で投入された場合には、この変換処理は不要となる。

【０１７４】

〔第４実施形態〕

ここでは、図１に示したプリンタ５０００の操作部２２に表示されるユーザインタフェースを通して、全体の処理を説明する。

【０１７５】

図２０～図２３は、図１に示した操作部２２に表示されるユーザインタフェースの一例を示す図である。本例は、上述したエラーや割り込みＪｏｂによって中断等の要因で印刷を中断したジョブに対する印刷再開指示画面の一例である。なお、操作部２２は、タッチパネルを備え、後述するボタンを指示することが可能に構成されている。もちらん、本発明はタッチパネルを採用したシステムに限定されるものではない。

【０１７６】

図２０において、１３０１は印刷再開画面で、プリンタ５０００の操作部２２に表示される。印刷再開画面１３０１は、Ｊｏｂがエラーや割り込みＪｏｂによって中断された後の画面を示している。１３０２、１３０３はボタンで、ボタン１３０２は継続印刷を指示するボタンとして機能し、ボタン１３０３は全項再印刷を指示するボタンとして機能する。

【０１７７】

本実施形態では、エラー状態の解消や割り込み印刷により優先的に印刷していたＪｏｂの終了により、中断状態が解除されると、中断していたＪｏｂの再開が始まる。この印刷再開画面１３０１は、Ｊｏｂの再開時に中断箇所から、継続して印刷するか初めから全項印刷するかのいずれであるかをユーザが選択するボタン１３０２、１３０３で決定する。つまり、このように、ＣＰＵ１２は、ジョブの状態を中断状態あるいは中断前の状態から印刷する指示をユーザが選択することができる。

【０１７８】

ここで、ボタン１３０２がユーザの操作で選択され、さらに、ＯＫを示すボタン１３０５が押されると、図１０に示したFixedDocumen.fdoc９０１やFixedDocumen.fdoc９０２のような構造をもった構造化文章をそのまま印刷することにより、中断箇所からの印刷を再開する。

【０１７９】

一方、ボタン１３０３がユーザの操作により選択され、ボタン１３０５が押されると、図１２に示す処理を起動して、図１０に示したFixedDocumen.fdoc９０１やFixedDocumen.fdoc９０２のような構造をもった構造化文章を、図１０に示したFixedDocumen.fdoc９０３やFixedDocumen.fdoc９０４のような構造をもった構造化文章に変換した後印刷を行い、全項印刷を行う。

【０１８０】

一方、戻るを示すボタン１３０４をユーザが押すと、印刷再開画面１３０１が表示される前の画面（不図示）に処理が戻る。以下の画面において、戻るとＯＫの機能はこれと同様である。

【０１８１】

図２１は、上記図２０に示した印刷再開画面に対して、同時に、出力先を選択できるようにした印刷再開画面１４０１の一例である。なお、ボタン１４０２、１４０３は、ボタン１３０２、１３０３と同じ機能を指示するボタンである。

【０１８２】

図２１において、ボタン１４０４を選択して、ＯＫを示すボタン１４０７が押されると、印刷再開画面１４０１を表示しているデバイスで、印刷再開を行う。

【０１８３】

一方、ボタン１４０５をユーザが選択して、ボタン１４０７が押されると、次に図２３に示す他デバイス選択画面１６０１を操作部２２に表示して、印刷再開画面１４０１を表

示しているデバイス以外の他のデバイスにそのＪｏｂを転送することを指示する。なお、ボタン１４０６は、ボタン１３０４と同じ戻るを指示するボタンである。

【０１８４】

図２２は、図２１に示した印刷再開画面１４０１を拡張した印刷再開画面１５０１であり、図２１に示した出力先としてデバイスに付属のメモリカードへの出力が選択できるようにしたボタン１５０６を備える画面例である。なお、デバイス付属のメモリカードは、図１に示した外部メモリ１４として示されている。また、ボタン１５０２，１５０３は、ボタン１３０２，１３０３と同じ機能を指示するボタンである。また、ボタン１５０４、１５０５は、図２１に示すボタン１４０４、１４０５と同じ機能を指示するボタンである。さらに、ボタン１５０７は、図２０に示すボタン１３０４と同じ機能を指示するボタンである。ボタン１５０８は、図２０に示すボタン１３０５と同じ機能を指示するボタンである。

【０１８５】

図２２に示すボタン１５０６を選択して、ボタン１５０８が押されると、図２５に示すメモリカード選択画面１８０１を操作部２２に表示して、複数備えるメモリカードデバイスのうち出力対象となるデバイスの選択を行う。

【０１８６】

図２３は、他デバイス選択画面１６０１の一例であり、図２１に示したボタン１４０５を選択した後で、操作部２２に表示される画面の一例である。

【０１８７】

本実施形態では、ネットワーク環境からデータを転送可能なデバイスを検索して、そのデバイス名称とＩＰアドレスを一覧１６０２として表示する。

【０１８８】

この一覧１６０２から他のデバイスを選択することも可能であるし、一覧１６０２に表示されないデバイスに出力したい場合、ボタン１６０３をユーザが選択した後、ボタン１６０５を押す。これにより、図２４に示すＩＰアドレス入力画面１７０１が操作部２２に表示され、ＩＰアドレスをユーザが直接入力することが可能となる。ボタン１６０４は、図２０に示すボタン１３０４と同じ機能を指示するボタンである。

【０１８９】

図２４に示すＩＰアドレス入力画面１７０１では、入力部１７０２にＩＰアドレスを、操作部２２のキーをユーザが操作して直接設定して、ボタン１７０４を押すことですることで、ＩＰアドレスの直接入力が可能に構成されている。ボタン１７０３は、、図２０に示すボタン１３０４と同じ機能を指示するボタンである。

【０１９０】

図２５は、図２２に示したボタン１５０６をユーザが選択して、ボタン１５０８を押した後に、操作部２２に表示される画面の一例である。ここでは、その時点で使用可能なメモリカードが選択可能である。ここに表示されているメモリカードは一例であって、ここに表示されているメモリカードに限定されるものではない。ボタン１８０３は、メモリカードとして、ＣＦカードを選択する場合に押下される。ボタン１８０２は、メモリカードとして、ＳＤカードを選択する場合に押下される。ボタン１８０４は、メモリカードとして、メモリスティックを選択する場合に押下される。ボタン１８０５は、メモリカードとして、スマートカードを選択する場合に押下される。ボタン１８０６は、メモリカードとして、ＵＳＢメモリを選択する場合に押下される。

【０１９１】

例えばボタン１８０６を選択して、ボタン１８０８を押すと、ＵＳＢメモリに対してＸＰＳに代表される構造化されたＪｏｂが出力される。なお、ＵＳＢメモリは、図示しない外部入力インタフェースに着脱可能に構成されている。ボタン１８０７は、ボタン１３０４と同じ機能を指示するボタンである。

【０１９２】

以上の画面処理により、中断した箇所から再印刷する場合や最初から全項再印刷という

選択が可能となり、また、中断Ｊｏｂを別の印刷機に転送したり、外部メモリに出力したりすることが可能となる。

【０１９３】

〔第５実施形態〕

ここでは、中断Ｊｏｂを、図２１に示したボタン１４０５などで他のデバイスに転送する時に、その転送されるＪｏｂの受け側のデバイスにも本実施形態に示す機能が搭載されている場合の例を説明する。

【０１９４】

図２６は、本実施形態を示す画像形成装置で表示可能なユーザインタフェースの一例を示す図である。本例は、中断Ｊｏｂを受信した側でPrintTicketを調査し、その結果全項印刷Ｊｏｂとして扱う必要があると判断した場合に表示されるＪｏｂ印刷画面の一例である。

【０１９５】

図２６において、２１０１は、中断Ｊｏｂを受信した側でPrintTicketを調査し、その結果全項印刷Ｊｏｂとして扱う必要があると判断した場合に表示される画面の例である。この判断処理は、図２７に示すフローチャートを参照して、詳細に説明する。

【０１９６】

例えばＪｏｂにStaple指定などがなされていた場合、中断した先行印刷しているＪｏｂのStaple状態も中途半端であるし、途中から再開印刷するＪｏｂのStaple状態も中途半端になる可能性が非常に高い。このような場合、全項印刷に切り替えて最初から印刷したほうが好ましい結果が得られる可能性が高い。

【０１９７】

そこで、図２６に示すＪｏｂ印刷画面では、理由２１０２に示すように、全項印刷に切り替えたほうが好ましい理由が表示される。本例は、「Staple指定あり」がその理由である。

【０１９８】

ボタン２１０３をユーザの操作で選択すると、好ましくはないのを承知で中断箇所から印刷再開し、ボタン２１０４を選択すると、より好ましい先頭からの全項印刷を行うことを指示する。ボタン２１０５は、図２０に示すボタン１３０４と同じ機能を指示するボタンである。ボタン２１０６は、ボタン１３０５と同じ機能を指示するボタンである。

【０１９９】

図２７は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、中断Ｊｏｂを受信した側でPrintTicketを調査し、その結果全項印刷Ｊｏｂとして扱う必要があるかどうかを判断する処理である。なお、Ｓ２３０２～Ｓ２３０８は各ステップを示す。図２７の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。

【０２００】

Print Ticket調査処理を開始すると、Ｓ２３０２で、ＣＰＵ１２は、構造化文章に付随するPrint Ticket（不図示）をＲＡＭ１９上に読み込む。同時に、ＣＰＵ１２は中断判断フラグを初期化する。これにより、中断判断フラグがＯＦＦ状態となる。

【０２０１】

次に、Ｓ２３０３で、ＣＰＵ１２は、Print Ticketを確認し、全項印刷に切り替えたほうが好ましい結果が得られ得る属性が設定されているか否か確認する。例えばStaple属性、Collate属性、両面印刷属性、など途中で印刷を中断してしまうとつじつまが合わなくなる属性がこれに該当する。本実施形態では、そのような属性を予め決めておき、それをＲＯＭ１３、ＲＡＭ１９、もしくは外部メモリ１４上に設定しておくことで、確認処理を実行する。

【０２０２】

ここで、全項印刷に切り替えたほうが好ましい結果が得られうる属性が設定されている

とＣＰＵ１２が判断した場合には、Ｓ２３０４に進み、そうでないとＣＰＵ１２が判断した場合には、Ｓ２３０８に進む。

【０２０３】

そして、Ｓ２３０４では、全項印刷に切り替えるか否かの判断を、図２６に示すＪｏｂ印刷画面２１０１を操作部２２に表示することによりユーザに問い合わせて、その結果をＣＰＵ１２が判断する。

【０２０４】

なお、ユーザに問い合わせることは、本実施形態において必須ではなく、自動的に継続印刷の処理を行っても良いし、自動的に全項印刷の処理を行っても良い。

【０２０５】

ここで、ＣＰＵ１２が切り替えると判断した場合には、Ｓ２３０５に進み、切り替えないと判断した場合には、Ｓ２３０６に進む。

【０２０６】

そして、Ｓ２３０５では、図１２に示した処理を行い、図１３に示したFixedDocument.fdoc９０１やFixedDocument.fdoc９０２の形式であった、FixedDocument.fdocをFixedDocument.fdoc９０３やFixedDocument.fdoc９０４の形式に変換する。これにより、印刷Ｊｏｂを全項印刷Ｊｏｂに変換することができる。もしくは図１８に示す処理を行い、図１４に示すFixedDocumentSequence.fdseq９６１やFixedDocumentSequence.fdseq９６２の形式であった、FixedDocumentSequence.fdseqをFixedDocumentSequence.fdseq９６３やFixedDocumentSequence.fdseq９６４の形式に変換する。これにより、印刷Ｊｏｂを全項印刷Ｊｏｂに変換することができる。

【０２０７】

一方、Ｓ２３０８では、ＣＰＵ１２は、中断調査フラグをＯＮにして、後ほど受信Ｊｏｂが中断Ｊｏｂなのか否かの判断処理を起動するための判断材料をセットして、Ｓ２３０６へ進む。

【０２０８】

次に、Ｓ２３０６では、Print Ticket（ＰＴ）上にデバイス能力に不適合な属性があるか否かの確認をＣＰＵ１２が行う。例えばＪｏｂを転送する側がＡ３印刷可能なデバイスで、Ｊｏｂを受けた側がＡ４までしか印刷可能でない場合、用紙サイズがデバイス能力に不適合な属性となりこれの変更が必要となる。

【０２０９】

また、Ｊｏｂを転送する側が両面印刷可能で、Ｊｏｂを受信する側が両面印刷できない場合は印刷方法（片面、両面）がデバイス能力に不適合となり、これの変更が必要となる。これらの属性は一例であって、これだけに限定されるものではない。

【０２１０】

ここで、デバイス能力に不適合な属性が有るとＣＰＵ１２が判断した場合にはＳ２３０７に進み、無いと判断した場合には、本処理を終了となる。

【０２１１】

そして、Ｓ２３０７では、図２８に示すＪｏｂ印刷画面２００１を操作部２２に表示して、不適合な属性を表示し、ユーザにその変更を求める。そして、変更結果はPrint Ticket（ＰＴ）に反映させ、印刷処理を続行する。

【０２１２】

この後に中断処理フラグの判断をＣＰＵ１２が行い、中断処理フラグがＯＮであるとＣＰＵ１２が判断した場合には、図２９に示すＪｏｂ調査処理に移り、そうでないとＣＰＵ１２が判断した場合には、本処理を終了する。

【０２１３】

なお、図２８に示すＪｏｂ印刷画面２００１は、Ｊｏｂを受信した側でＪｏｂ属性を修正する必要がある場合に、操作部２２に表示される。例えばＪｏｂを送信する側がＡ３印刷可能なデバイスで、Ｊｏｂを受けた側がA4までしか印刷可能でない場合、用紙サイズの変更が必要となる。

【０２１４】
　また、Ｊｏｂを送信する側が両面印刷可能で、Ｊｏｂを受信する側が両面印刷できない場合はこれも両面印刷属性の変更が必要となる。これらの属性は一例であって、これだけに限定されるものではない。
【０２１５】
図２８において、２００２は用紙サイズ選択部で、用紙サイズの変更可能な候補が表示され、印刷方法選択部２００３には、両面印刷が片面印刷に変更する必要があることが表示される。ボタン２００４は、図２０に示すボタン１３０４と同じ機能を指示するボタンである。ボタン２００５は、ボタン１３０５と同じ機能を指示するボタンである。
【０２１６】
　図２９は、本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。本例は、本実施形態に示す機能がＪｏｂの受信側にあった場合に、その受信側で受信したＪｏｂをどのように扱った中断Ｊｏｂであるか否かの判断する処理である。なお、Ｓ２２０２～Ｓ２２１８は各ステップを示す。図２９の各ステップを実行するプログラムはＲＯＭ１３に格納されており、ＲＡＭ１９を使ってＣＰＵ１２により実行される。別の方法としては、外部メモリ１４上に格納し、ＲＡＭ１９上に展開して実行しても良い。
【０２１７】
　Ｊｏｂ調査が開始されると、Ｓ２２０２で、ＣＰＵ１２は、FixedDocumentSequence.fdseqを開く。次に、Ｓ２２０３でFixedDocumentSequence.fdseqの内部にFixedDocument_add.fdocという形式の複製されたFixedDocument.fdocがあるか否かＣＰＵ１２が判断する。ここで、FixedDocument_add.fdocという形式の複製されたFixedDocument.fdocがあるとＣＰＵ１２が判断した場合は、図１４に示した形式で中断箇所を記録したＸＰＳ文章であるということが判断できるので、Ｓ２２１７に進む。そして、Ｓ２２１７で、ＣＰＵ１２は、受信Ｊｏｂは中断Ｊｏｂであると判断した後、Ｓ２２１８に進み、FixedDocumentSequence.fdseqをColseして、本処理が終了する。
【０２１８】
なお、ここでは FixedDocument.fdocのファイル名の内部に_addが有るか否かで判断を行ったが、本実施形態ではこの文字列だけに限定するものではない。
【０２１９】
　一方、Ｓ２２０３で、FixedDocument_add.fdocという形式の複製されたFixedDocument.fdocが無いとＣＰＵ１２が判断した場合は、継続調査の為、Ｓ２２０４に進み、ＣＰＵ１２は、FixedDocumentSequence.fdseqをColseして、Ｓ２２０５に進む。
【０２２０】
　そして、Ｓ２２０５で、ＣＰＵ１２は、複数のFixedDocument.fdocへのポインタとなる変数Ｎを１に初期化する。そして、Ｓ２２０６で、Ｎが１の場合まずはじめにFixedDocument.fdocが有るか否かＣＰＵ１２が判断する。ここで、FixedDocument.fdocがあるとＣＰＵ１２が判断した場合、Ｓ２２０７に進む。一方、Ｓ２２０６で、FixedDocument.fdocが無いとＣＰＵ１２が判断した場合は、続いてFixedDocument1.fdocがあるか否かをＣＰＵ１２が判断する（図示しない）。ここで、FixedDocument1.fdocがあるとＣＰＵ１２が判断した場合は、Ｓ２２０７に進み（図示しない）、それでも無いとＣＰＵ１２が判断した場合は、Ｓ２２１６へ進み、受信したＪｏｂは中断ＪｏｂではないとＣＰＵ１２が判断して、本処理を終了する。以降、FixedDocument.fdocやFixedDocumentN.fdoc（Ｎが1から順にカウントアップされて行く）のことをまとめてFixedDocumentN.fdocと表現する。
【０２２１】
　次に、Ｓ２２０７では、ＣＰＵ１２は、Ｓ２２０６で存在を確認したFixedDocumentN.fdocを開く。そして、Ｓ２２０８で、ＣＰＵ１２は、開いたFixedDocumentN.fdocのコメント行の中身をサーチする。具体的には当発明が追加したコメントがあるか無いかをサーチする。
【０２２２】

そして、Ｓ２２０９で、コメント行のサーチの結果、本実施形態で追加したコメント（本実施形態で追加したということを示す目印のついたコメント）があるか無いかをＣＰＵ１２が判断する。ここで、目印つきのコメントが無いとＣＰＵ１２が判断した場合には、Ｓ２２１０に進み、目印つきのコメントが有るとＣＰＵ１２が判断した場合には、Ｓ２２１４に進む。

【０２２３】

そして、Ｓ２２１０で、ＣＰＵ１２は、コメント行があるか無いかを判断する。ここで、コメント行が無いとＣＰＵ１２が判断した場合には、Ｓ２２１１に進み、コメント行がある場合には、Ｓ２２１３に進む。

【０２２４】

これはこのＳ２２１０に処理が移った時点で、残されている可能性として、ＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントが入っている場合と、本実施形態でコメントを追加したがそれが、追加したという目印なしの場合のいずれかになるためである。

【０２２５】

そして、Ｓ２２１３では、コメントがn.fpageをコメントしているか否かの判断をＣＰＵ１２が行う。ここで、n.fpageをコメントしているとＣＰＵ１２が判断した場合、それは本実施形態で追加したコメントであり、ページリンクを切るために追加されたコメントに他ならない。この場合は、Ｓ２２１４に進む。

【０２２６】

一方、Ｓ２２１３でn.fpageをコメントしていないとＣＰＵ１２が判断した場合は、それはＸＰＳ文章を作成したドライバやアプリケーションが追加したコメントと判断できるので、Ｓ２２１１に進む。

【０２２７】

そして、Ｓ２２１１ではコメントチェックを行ったFixedDocumentN.fdocを保存しCloseする。そして、Ｓ２２１２で、ＣＰＵ１２は、複数のFixedDocumentN.fdocへのポインタとなる変数Ｎを１つカウントアップして、その後、Ｓ２２０６に戻る。

【０２２８】

一方、Ｓ２２０９で目印つきのコメント有りと判断された場合、並びに、Ｓ２２１３で、n.fpageをコメントとしていると判断した場合は、Ｓ２２１４で、受信Ｊｏｂは中断Ｊｏｂであるとの判断する。この後、Ｓ２２１５に進み、ＣＰＵ１２は、0penしているFixedDocumentN.fdocを保存＆Closeして、本処理を終了する。

【０２２９】

なお、ＣＰＵ１２がＳ２２１４で、受信Ｊｏｂは中断Ｊｏｂであると判断した場合には、操作部２２に、図３０に示すユーザインタフェースを表示して、ユーザに中断箇所から印刷するのか、全項再印刷にするのかのを問い合わせる。

【０２３０】

図３０は、本実施形態を示す画像形成装置で表示されるユーザインタフェースの一例を示す図である。

【０２３１】

図３０において、１９０１はＪｏｂ受信画面であり、構造化文章の構造部分が更新されているか否か確認のうえ、変更されていることが明らかになった為その中断Ｊｏｂをどう扱うべきか判断をユーザに対して促すための画面である。１９０２はボタンで、継続印刷を指示する場合に選択され、ボタン１９０２が選択された後、ボタン１９０５をユーザが押下した場合には、構造部分が更新されているＪｏｂをそのまま印刷する。ボタン１９０４は、図２０に示すボタン１３０４と同じ機能を指示するボタンである。

【０２３２】

１９０３はボタンで、全項再印刷を指示する場合に選択され、ボタン１９０３が選択された後、ボタン１９０５をユーザが押下した場合には、図１２に示す処理を起動する。そして、図１０に示したFixedDocument.fdoc９０１やFixedDocument.fdoc９０２のような構

造の構造化文章を、図１０に示したFixedDocument.fdoc９０３やFixedDocument.fdoc９０４のような構造の構造化文章に変換した後印刷を行い、全項印刷を行う。

【０２３３】

なお、図２１に示すボタン１４０５をユーザが選択した場合、もしくは図２２に示すボタン１５０６を選択した場合に出力先として、本実施形態に示す機能を搭載していないデバイスが選択された場合には、ＸＰＳの文法に準拠したＸＰＳ文章が転送される。この場合に転送されるＸＰＳ文章は中断箇所以降を印刷するように設定されている文章である。

【０２３４】

以上の処理により、Ｊｏｂを受ける側でPrint Ticketを確認して、全項印刷に切り替えるべきか否かの自動判別を行い、ユーザに指示を仰ぐことができる。また、自動的に継続印刷や全項印刷を行うこともできる。

【０２３５】

また、Ｊｏｂを受けた側のデバイス能力に適合しない印刷属性が含まれている場合にもそれを自動判別して、最適な属性に変換するようにユーザに指示を仰ぐことができる。

【０２３６】

さらに、全項印刷に切り替えるべきPrint Ticketが含まれていない場合でも、受信側でそれが中断Jonなのか、否かの判断を行うことが可能となる。これにより、中断Ｊｏｂだった場合には、中断箇所から印刷するのか、最初から全項印刷するのかの判断を仰ぐことが可能となる。

【０２３７】

以下、図３１に示すメモリマップを参照して本発明に係る画像形成装置、画像形成装置で読み取り可能なデータ処理プログラムの構成について説明する。

【０２３８】

図３１は、本発明に係る画像形成装置で読み取り可能な各種データ処理プログラムを格納する記憶媒体のメモリマップを説明する図である。

【０２３９】

なお、特に図示しないが、記憶媒体に記憶されるプログラム群を管理する情報、例えばバージョン情報，作成者等も記憶され、かつ、プログラム読み出し側のＯＳ等に依存する情報、例えばプログラムを識別表示するアイコン等も記憶される場合もある。

【０２４０】

さらに、各種プログラムに従属するデータも上記ディレクトリに管理されている。また、各種プログラムをコンピュータにインストールするためのプログラムや、インストールするプログラムが圧縮されている場合に、解凍するプログラム等も記憶される場合もある。

【０２４１】

本実施形態における図６、図７、図１１、図１２、図１６～図１９、図２７、図２９に示す機能が外部からインストールされるプログラムによって、ホストコンピュータにより遂行されていてもよい。そして、その場合、ＣＤ－ＲＯＭやフラッシュメモリやＦＤ等の記憶媒体により、あるいはネットワークを介して外部の記憶媒体から、プログラムを含む情報群を出力装置に供給される場合でも本発明は適用されるものである。

【０２４２】

以上のように、前述した実施形態の機能を実現するソフトウエアの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する。そして、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出し実行することによっても、本発明の目的が達成されることは言うまでもない。

【０２４３】

この場合、記憶媒体から読み出されたプログラムコード自体が本発明の新規な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。

【０２４４】

従って、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、ＯＳに供給するスクリプトデータ等、プログラムの形態を問わない。

【０２４５】

プログラムを供給するための記憶媒体としては、例えばフレキシブルディスク、ハードディスク、光ディスク、光磁気ディスク、ＭＯ、ＣＤ－ＲＯＭ、ＣＤ－Ｒ、ＣＤ－ＲＷ、磁気テープ、不揮発性のメモリカード、ＲＯＭ、ＤＶＤなどを用いることができる。

【０２４６】

この場合、記憶媒体から読出されたプログラムコード自体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。

【０２４７】

その他、プログラムの供給方法としては、クライアントコンピュータのブラウザを用いてインターネットのホームページに接続する。そして、該ホームページから本発明のコンピュータプログラムそのもの、もしくは、圧縮され自動インストール機能を含むファイルをハードディスク等の記録媒体にダウンロードすることによっても供給できる。また、本発明のプログラムを構成するプログラムコードを複数のファイルに分割し、それぞれのファイルを異なるホームページからダウンロードすることによっても実現可能である。つまり、本発明の機能処理をコンピュータで実現するためのプログラムファイルを複数のユーザに対してダウンロードさせるＷＷＷサーバやｆｔｐサーバ等も本発明の請求項に含まれるものである。

【０２４８】

また、本発明のプログラムを暗号化してＣＤ－ＲＯＭ等の記憶媒体に格納してユーザに配布し、所定の条件をクリアしたユーザに対し、インターネットを介してホームページから暗号化を解く鍵情報をダウンロードさせる。そして、その鍵情報を使用することにより暗号化されたプログラムを実行してコンピュータにインストールさせて実現することも可能である。

【０２４９】

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はない。例えば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）等が実際の処理の一部または全部を行う。そして、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。

【０２５０】

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せる。その後、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等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。

【０２５１】

本発明は上記実施形態に限定されるものではなく、本発明の趣旨に基づき種々の変形（各実施形態の有機的な組合せを含む）が可能であり、それらを本発明の範囲から除外するものではない。

【０２５２】

本発明の様々な例と実施形態を示して説明したが、当業者であれば、本発明の趣旨と範囲は、本明細書内の特定の説明に限定されるのではない。

【図面の簡単な説明】

【０２５３】

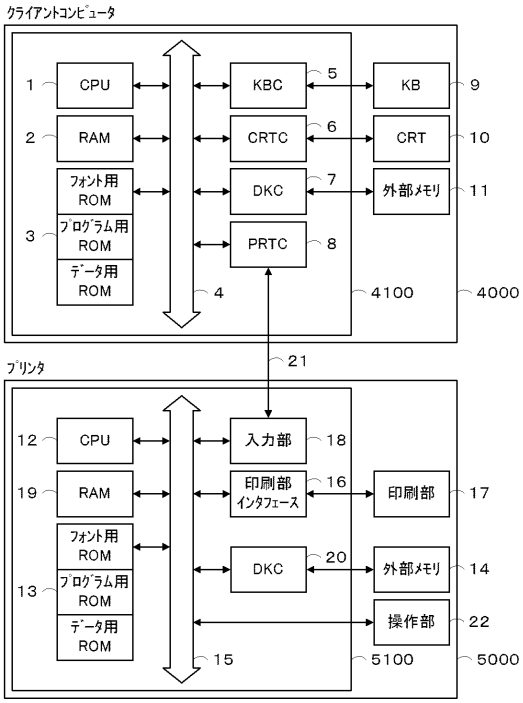
【図１】本実施形態を示す印刷システムの一例を示すブロック図である。

【図２】本実施形態を示す画像形成装置におけるジョブ中断処理を説明する図である。

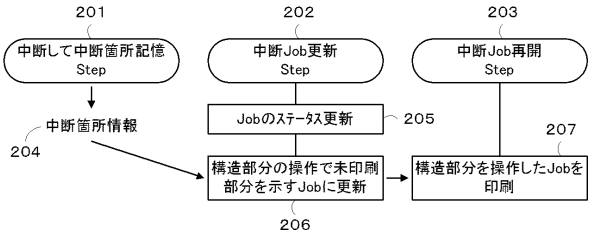
【図３】本実施形態を示す画像形成装置で処理されるジョブとそのステータスを示す図である。
【図４】本実施形態を示す画像形成装置で処理されるジョブの構造を説明する図である。
【図５】図４に示したDocumentを修正することによりページリンクを切るところを示した図である。
【図６】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図７】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図８】本実施形態を示す画像形成装置で処理される文章の内部構造を説明する図である。
【図９】本実施形態を示す画像形成装置で処理される文章の内部構造を説明する図である。
【図１０】図８、図９に示したＸＰＳにおけるFixedDocument.fdocの例を示す図である。
【図１１】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図１２】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図１３】本実施形態を示す画像形成装置で処理されるＸＰＳにおけるFixedDocument.fdocの例を示す図である。
【図１４】本実施形態を示す画像形成装置で処理されるＸＰＳにおけるFixedDocumentSequence.fdseqの例を示す図である。
【図１５】図１３、図１４に示したＸＰＳ構成をまとめたＸＰＳ文章の構造を示す図である。
【図１６】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図１７】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図１８】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図１９】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図２０】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図２１】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図２２】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図２３】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図２４】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図２５】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図２６】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図２７】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図２８】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。

【図２９】本実施形態を示す画像形成装置におけるデータ処理手順の一例を示すフローチャートである。
【図３０】図１に示した操作部に表示されるユーザインタフェースの一例を示す図である。
【図３１】本発明に係る画像形成装置で読み取り可能な各種データ処理プログラムを格納する記憶媒体のメモリマップを説明する図である。
【符号の説明】
【０２５４】
１　ＣＰＵ
２　ＲＡＭ
３　ＲＯＭ
４　システムバス
５　ＫＢＣ
６　ＣＲＴＣ
７　ＤＫＣ
８　ＰＲＴＣ
９　ＫＢ
１０　ＣＲＴ
１１　外部メモリ
１２　ＣＰＵ
１３　ＲＯＭ
１４　外部メモリ
１５　システムバス
１６　印刷部インタフェース
１７　印刷部
１８　入力部
１９　ＲＡＭ
２０　ＤＫＣ
２１　双方向性インタフェース
２２　操作パネル
４０００　クライアントコンピュータ
５０００　プリンタ

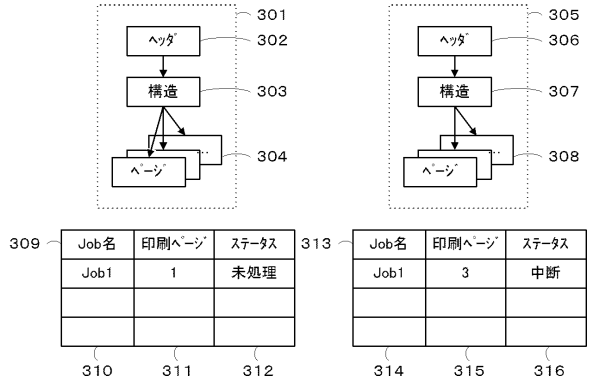
【図１】



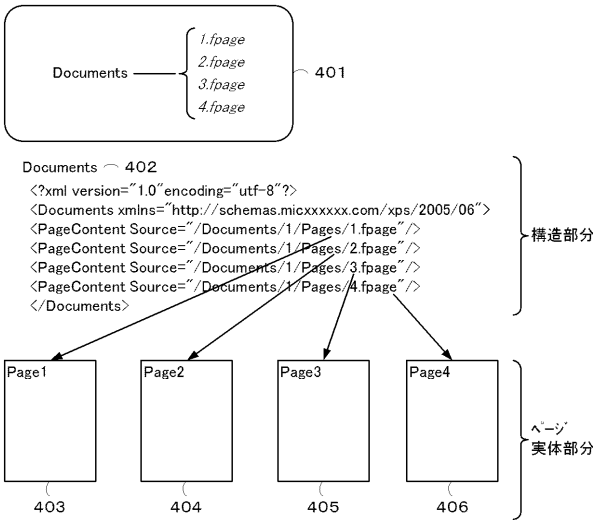
【図２】



【図３】



【図４】



【図５】

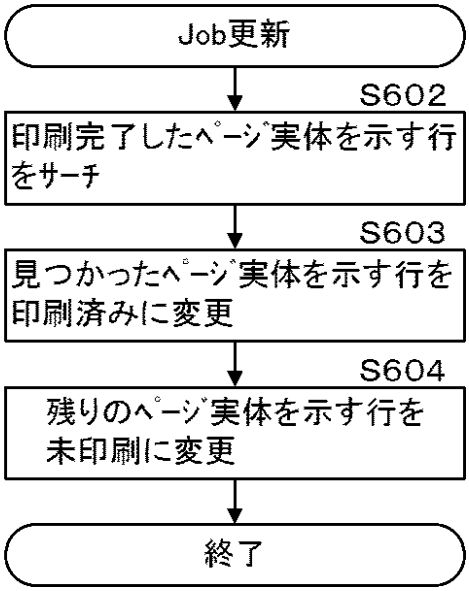
Document 501

```
<?xml version="1.0"encoding="utf-8"?>
<Documents xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<Print status="/Done"> 511
</Print> 512
<Print status="/None"> 513
<PageContent Source="/Documents/1/Pages/1.fpage"/>
<PageContent Source="/Documents/1/Pages/2.fpage"/>
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</Print> 514
</Documents>
```

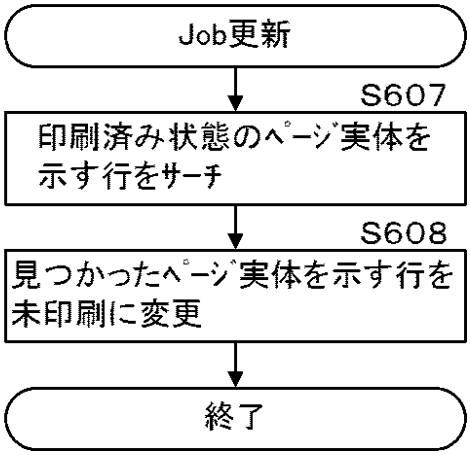
Document 502

```
<?xml version="1.0"encoding="utf-8"?>
<Documents xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<Print status="/Done"> 521
<PageContent Source="/Documents/1/Pages/1.fpage"/>
<PageContent Source="/Documents/1/Pages/2.fpage"/>
</Print> 522
<Print status="/None"> 523
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</Print> 524
</Documents>
```

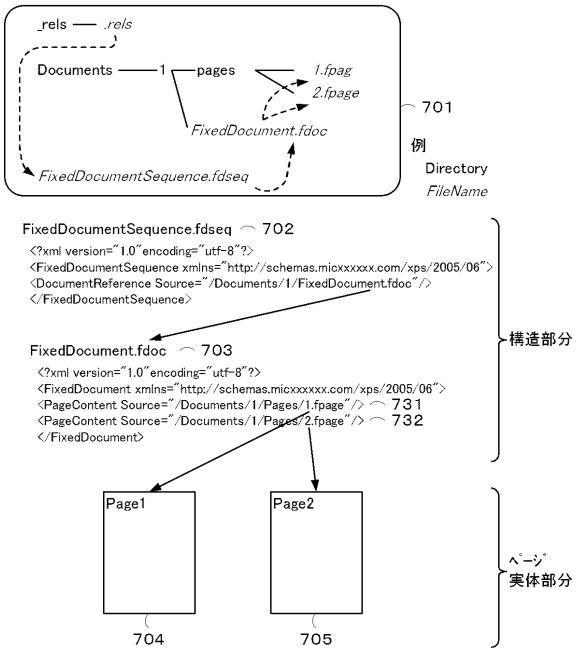
【図６】


Job更新
S602
印刷完了したﾍﾟｰｼﾞ実体を示す行をｻｰﾁ
S603
見つかったﾍﾟｰｼﾞ実体を示す行を印刷済みに変更
S604
残りのﾍﾟｰｼﾞ実体を示す行を未印刷に変更
終了

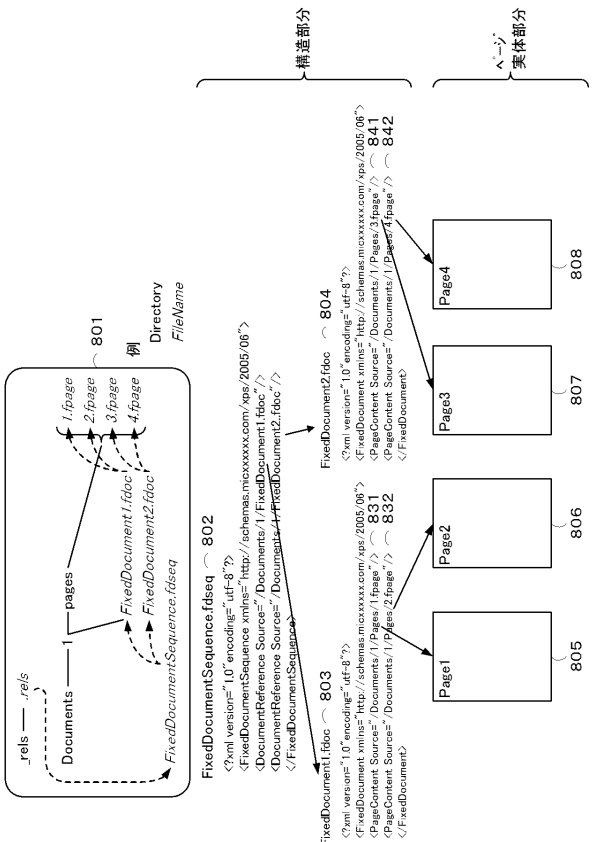
【図７】


Job更新
S607
印刷済み状態のﾍﾟｰｼﾞ実体を示す行をｻｰﾁ
S608
見つかったﾍﾟｰｼﾞ実体を示す行を未印刷に変更
終了

【図８】


_rels
.rels
Documents
1
pages
1.fpag
2.fpage
FixedDocument.fdoc
FixedDocumentSequence.fdseq
701
例
Directory
FileName
FixedDocumentSequence.fdseq 702
<?xml version="1.0"encoding="utf-8"?>
<FixedDocumentSequence xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<DocumentReference Source="/Documents/1/FixedDocument.fdoc"/>
</FixedDocumentSequence>
FixedDocument.fdoc 703
<?xml version="1.0"encoding="utf-8"?>
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/1.fpage"/> 731
<PageContent Source="/Documents/1/Pages/2.fpage"/> 732
</FixedDocument>
構造部分
Page1
Page2
704
705
ページ実体部分

【図９】


構造部分
ページ実体部分
801
例
Directory
FileName
1.fpage
2.fpage
3.fpage
4.fpage
FixedDocument1.fdoc
FixedDocument2.fdoc
FixedDocumentSequence.fdseq
_rels
.rels
Documents
1
pages
FixedDocumentSequence.fdseq 802
<?xml version="1.0"encoding="utf-8"?>
<FixedDocumentSequence xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<DocumentReference Source="/Documents/1/FixedDocument1.fdoc"/>
<DocumentReference Source="/Documents/1/FixedDocument2.fdoc"/>
</FixedDocumentSequence>
FixedDocument1.fdoc 803
<?xml version="1.0"encoding="utf-8"?>
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/1.fpage"/> 831
<PageContent Source="/Documents/1/Pages/2.fpage"/> 832
</FixedDocument>
FixedDocument2.fdoc 804
<?xml version="1.0"encoding="utf-8"?>
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/3.fpage"/> 841
<PageContent Source="/Documents/1/Pages/4.fpage"/> 842
</FixedDocument>
Page1
Page2
Page3
Page4
805
806
807
808

【図１０】

FixedDocument.fdoc 901

```
<?xml version="1.0"encoding="utf-8"?>
<!-- This File is generated by XPS Document Writer--> ⌒911
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<!--C-LBP<PageContent Source="/Documents/1/Pages/1.fpage"/>--> ⌒912
<!--C-LBP<PageContent Source="/Documents/1/Pages/2.fpage"/>--> ⌒913
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

FixedDocument.fdoc 902

```
<?xml version="1.0"encoding="utf-8"?>
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<!--<PageContent Source="/Documents/1/Pages/1.fpage"/>--> ⌒921
<!--<PageContent Source="/Documents/1/Pages/2.fpage"/>--> ⌒922
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

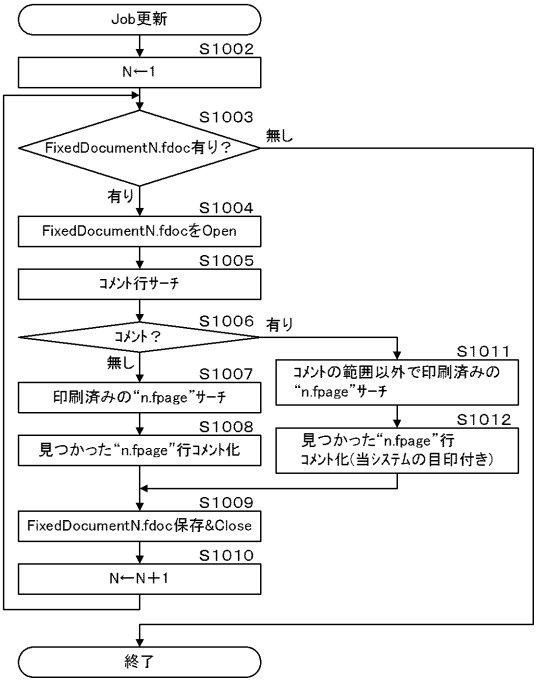
FixedDocument.fdoc 903

```
<?xml version="1.0"encoding="utf-8"?>
<!-- This File is generated by XPS Document Writer--> ⌒931
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/1.fpage"/>
<PageContent Source="/Documents/1/Pages/2.fpage"/>
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

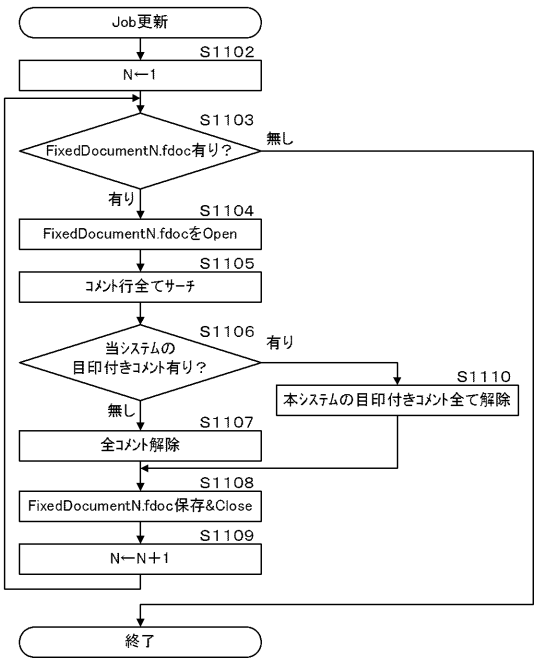
FixedDocument.fdoc 904

```
<?xml version="1.0"encoding="utf-8"?>
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/1.fpage"/>
<PageContent Source="/Documents/1/Pages/2.fpage"/>
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

【図１１】



【図１２】



【図１３】

FixedDocument_add.fdoc 951

```
<?xml version="1.0"encoding="utf-8"?>
<!-- This File is generated by XPS Document Writer--> ⌒955
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

FixedDocument_add.fdoc 952

```
<?xml version="1.0"encoding="utf-8"?>
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

FixedDocument.fdoc 953

```
<?xml version="1.0"encoding="utf-8"?>
<!-- This File is generated by XPS Document Writer--> ⌒956
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/1.fpage"/>
<PageContent Source="/Documents/1/Pages/2.fpage"/>
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

FixedDocument.fdoc 954

```
<?xml version="1.0"encoding="utf-8"?>
<FixedDocument xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<PageContent Source="/Documents/1/Pages/1.fpage"/>
<PageContent Source="/Documents/1/Pages/2.fpage"/>
<PageContent Source="/Documents/1/Pages/3.fpage"/>
<PageContent Source="/Documents/1/Pages/4.fpage"/>
<PageContent Source="/Documents/1/Pages/5.fpage"/>
</FixedDocument>
```

【図１４】

FixedDocumentSequence.fdseq 961

```
<?xml version="1.0"encoding="utf-8"?>
<!-- This File is generated by XPS Document Writer--> 965
<FixedDocumentSequence xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<!--<DocumentReference Source="/Documents/1/FixedDocument.fdoc"/>--> 966
<DocumentReference Source="/Documents/1/FixedDocument_add.fdoc"/> 967
</FixedDocumentSequence>
```

FixedDocumentSequence.fdseq 962

```
<?xml version="1.0"encoding="utf-8"?>
<FixedDocumentSequence xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<!--<DocumentReference Source="/Documents/1/FixedDocument.fdoc"/>--> 968
<DocumentReference Source="/Documents/1/FixedDocument_add.fdoc"/> 969
</FixedDocumentSequence>
```

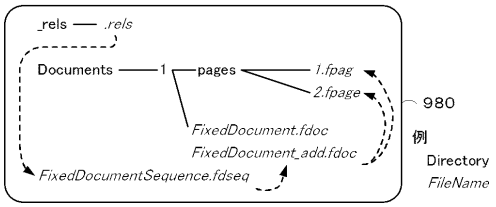
FixedDocumentSequence.fdseq 963

```
<?xml version="1.0"encoding="utf-8"?>
<!-- This File is generated by XPS Document Writer--> 970
<FixedDocumentSequence xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<DocumentReference Source="/Documents/1/FixedDocument.fdoc"/>
</FixedDocumentSequence>
```

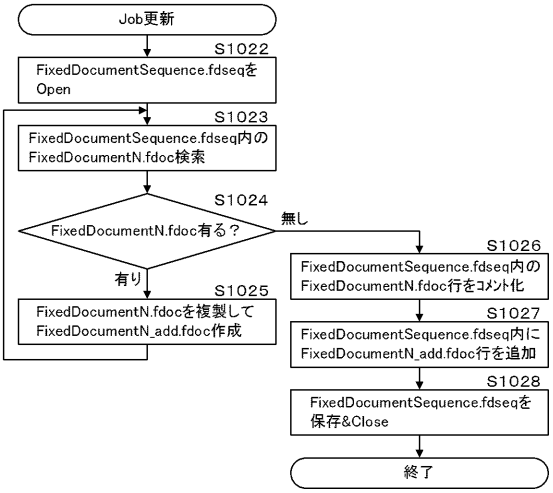
FixedDocumentSequence.fdseq 964

```
<?xml version="1.0"encoding="utf-8"?>
<FixedDocumentSequence xmlns="http://schemas.micxxxxxx.com/xps/2005/06">
<DocumentReference Source="/Documents/1/FixedDocument.fdoc"/>
</FixedDocumentSequence>
```

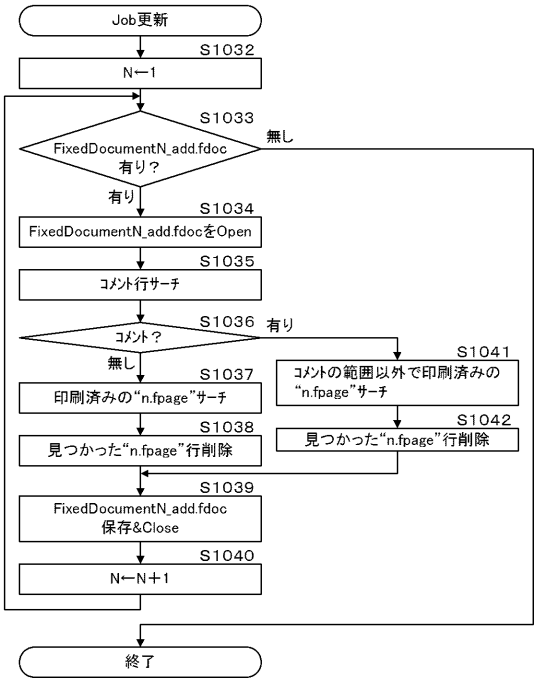
【図１５】



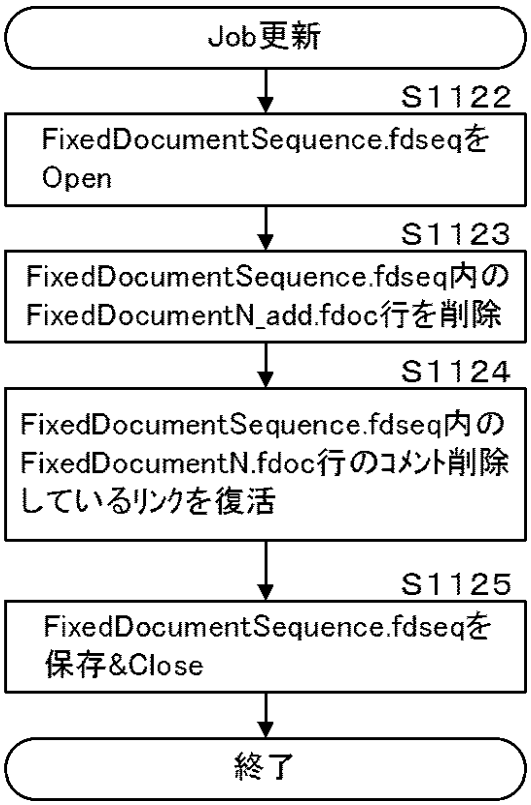
【図１６】



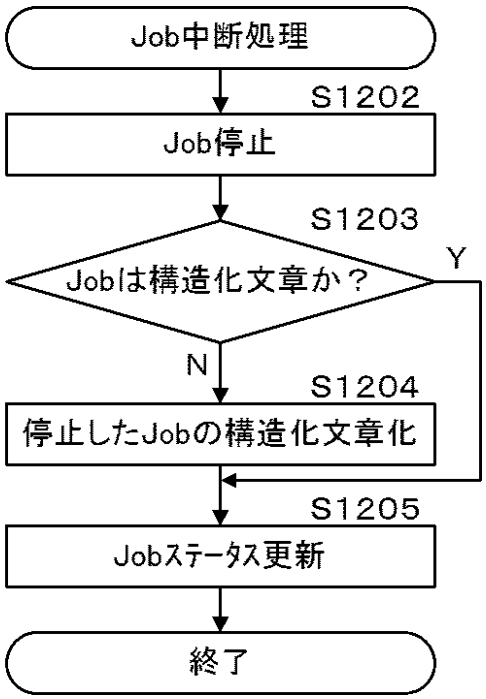
【図１７】



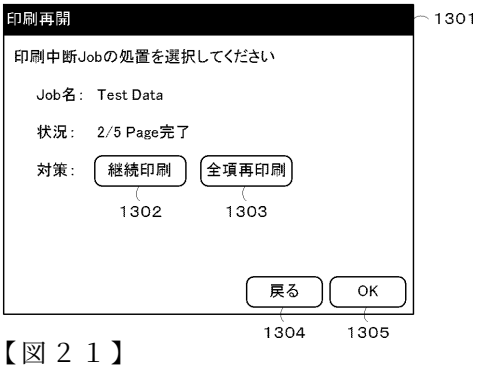
【図１８】



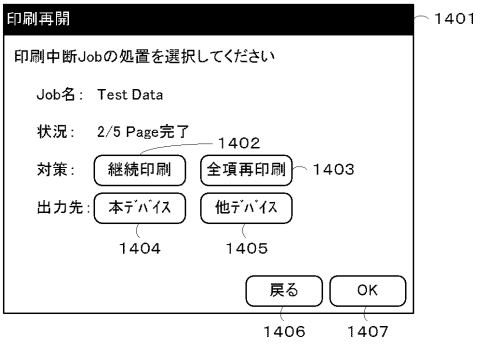
【図１９】



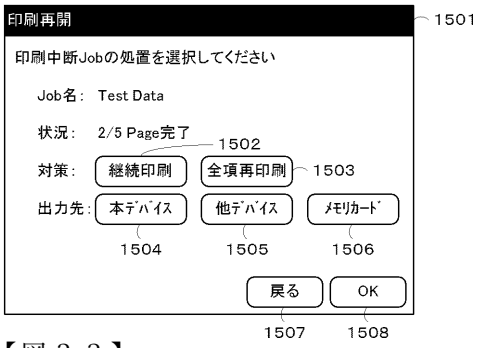
【図２０】



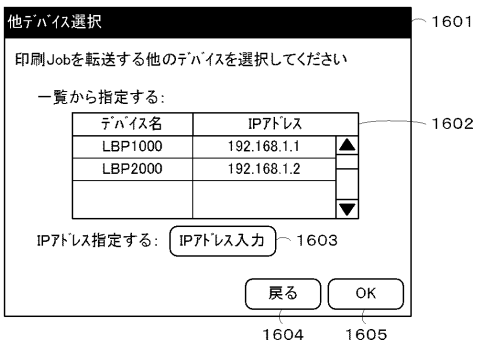
【図２１】



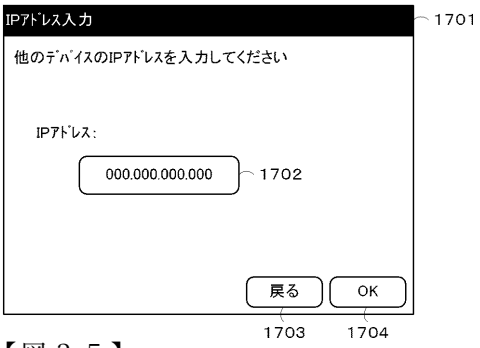
【図２２】



【図２３】



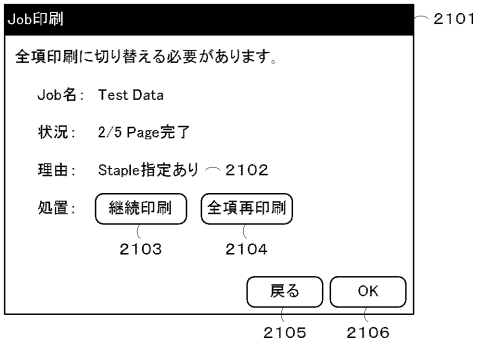
【図２４】



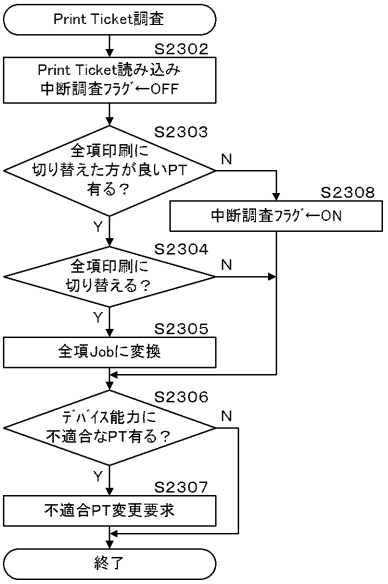
【図２５】



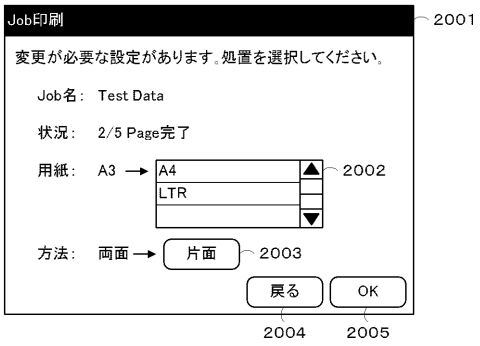
【図２６】



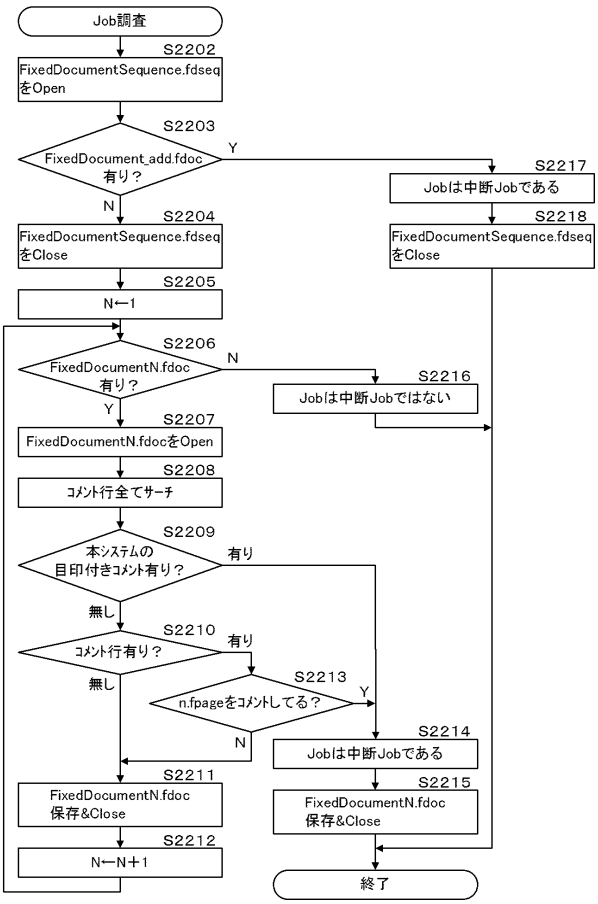
【図２７】



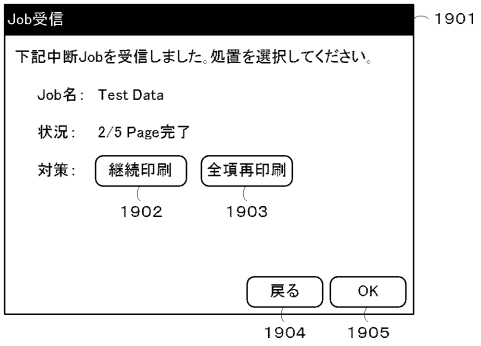
【図２８】



【図２９】



【図３０】



【図３１】

FD/CD-ROM等の記憶媒体

ディレクトリ情報
第1の処理ﾌﾟﾛｸﾞﾗﾑ 図6に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第2の処理ﾌﾟﾛｸﾞﾗﾑ 図7に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第3の処理ﾌﾟﾛｸﾞﾗﾑ 図11に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第4の処理ﾌﾟﾛｸﾞﾗﾑ 図12に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第5の処理ﾌﾟﾛｸﾞﾗﾑ 図16に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第6の処理ﾌﾟﾛｸﾞﾗﾑ 図17に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第7の処理ﾌﾟﾛｸﾞﾗﾑ 図18に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第8の処理ﾌﾟﾛｸﾞﾗﾑ 図19に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第9の処理ﾌﾟﾛｸﾞﾗﾑ 図27に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群
第10の処理ﾌﾟﾛｸﾞﾗﾑ 図29に示すﾌﾛｰﾁｬｰﾄのｽﾃｯﾌﾟに対応するﾌﾟﾛｸﾞﾗﾑｺｰﾄﾞ群

記憶媒体のﾒﾓﾘﾏｯﾌﾟ

フロントページの続き

(56)参考文献 特開平１１－３１４４３８（ＪＰ，Ａ）
特開平０９－３１１７７０（ＪＰ，Ａ）
特開２００８－０５２４５７（ＪＰ，Ａ）
特開２００７－３１０５６４（ＪＰ，Ａ）
特開２００８－０１１５６８（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　　３／１２
Ｂ４１Ｊ　２９／３８
Ｈ０４Ｎ　　１／００