

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4817959号
(P4817959)

(45) 発行日 平成23年11月16日 (2011.11.16)

(24) 登録日 平成23年9月9日 (2011.9.9)

(51) Int.Cl.

F I

G O 6 F 3/12 (2006.01)
 G O 6 F 17/21 (2006.01)
 G O 6 T 11/60 (2006.01)
 H O 4 N 1/387 (2006.01)

G O 6 F 3/12 F
 G O 6 F 17/21 5 6 6 E
 G O 6 T 11/60 1 0 0 A
 G O 6 F 17/21 5 3 0 E
 G O 6 F 17/21 5 4 0

請求項の数 18 (全 41 頁) 最終頁に続く

(21) 出願番号 特願2006-135998 (P2006-135998)
 (22) 出願日 平成18年5月16日 (2006.5.16)
 (65) 公開番号 特開2007-310432 (P2007-310432A)
 (43) 公開日 平成19年11月29日 (2007.11.29)
 審査請求日 平成21年5月14日 (2009.5.14)

(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 100076428
 弁理士 大塚 康德
 (74) 代理人 100112508
 弁理士 高柳 司郎
 (74) 代理人 100115071
 弁理士 大塚 康弘
 (74) 代理人 100116894
 弁理士 木村 秀二
 (72) 発明者 宮本 大次郎
 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

最終頁に続く

(54) 【発明の名称】 文書処理装置及び文書処理方法

(57) 【特許請求の範囲】

【請求項 1】

文書データの可変領域にデータを割り当てて印刷する可変印刷処理を行うための文書処理装置であって、

前記可変領域に対して共通属性を指定する指定手段と、

前記指定手段により前記共通属性が指定された場合、前記可変領域と同一の共通可変領域を前記可変領域が配置されていないページに配置する配置手段と、

前記共通可変領域が配置された複数ページから選択された選択ページの前記共通可変領域の位置およびサイズの少なくとも1つを変更する変更手段とを有し、

前記選択ページとは異なるページの前記共通可変領域は、前記変更手段により前記選択ページにおいて変更された位置およびサイズと同じ位置および同じサイズにて配置されることを特徴とする文書処理装置。

【請求項 2】

前記選択ページとは異なるページの前記共通可変領域が、前記変更手段により前記選択ページにおいて変更された位置およびサイズと同じ位置および同じサイズにて配置された結果、前記共通可変領域が収まらないページについて、前記変更手段は、前記ページに配置された共通可変領域の共通属性を解除し、解除された領域を前記ページ内に収まる位置に変更することを特徴とする請求項 1 に記載の文書処理装置。

【請求項 3】

前記変更手段は、前記共通属性が解除された領域を移動した結果、前記移動された領域

10

20

がページ内に収まらない場合、前記ページについて前記解除された領域がページ内に収まるように前記解除された領域の幅および高さを示すサイズ情報を変更することを特徴とする請求項 2 に記載の文書処理装置。

【請求項 4】

前記配置手段は、前記共通可変領域が配置された後、選択されたページにおいて前記共通属性が解除された場合、前記選択されたページについて前記共通可変領域を残し、それ以外のページについては前記共通可変領域を削除することを特徴とする請求項 1 乃至 3 のいずれか 1 項に記載の文書処理装置。

【請求項 5】

新たなページを追加する文書編集手段をさらに備え、

前記文書編集手段により新たにページが追加された際に、前記配置手段は、前記新たなページにも前記共有可変領域を追加することを特徴とする請求項 1 乃至 4 のいずれか 1 項に記載の文書処理装置。

【請求項 6】

ページのデータを解析し、オブジェクトの種類ごとに領域を分類する原稿解析手段をさらに備え、

前記配置手段は、ユーザの指示に従って前記ページ内のイメージオブジェクトまたは文字オブジェクト上に前記共通可変領域を配置することを特徴とする請求項 1 乃至 5 のいずれか 1 項に記載の文書処理装置。

【請求項 7】

指定手段と配置手段と変更手段とを備えた文書処理装置による、文書データの可変領域にデータを割り当てて印刷する可変印刷処理を行うための文書処理方法であって、

前記指定手段が、前記可変領域に対して共通属性を指定する指定工程と、

前記配置手段が、前記指定工程により前記共通属性が指定された場合、前記可変領域と同一の共通可変領域を前記可変領域が配置されていないページに配置する配置工程と、

前記変更手段が、前記共通可変領域が配置された複数ページから選択された選択ページの前記共通可変領域の位置およびサイズの少なくとも 1 つを変更する変更工程とを有し、

前記選択ページとは異なるページの前記共通可変領域は、前記変更工程により前記選択ページにおいて変更された位置およびサイズと同じ位置および同じサイズにて配置されることを特徴とする文書処理方法。

【請求項 8】

前記選択ページとは異なるページの前記共通可変領域が、前記変更工程により前記選択ページにおいて変更された位置およびサイズと同じ位置および同じサイズにて配置された結果、前記共通可変領域が収まらないページについて、前記変更工程では、前記ページに配置された共通可変領域の共通属性を解除し、解除された領域を前記ページ内に収まる位置に変更することを特徴とする請求項 7 に記載の文書処理方法。

【請求項 9】

前記変更工程では、前記共通属性が解除された領域を移動した結果、前記移動された領域がページ内に収まらない場合、前記ページについて前記解除された領域がページ内に収まるように前記解除された領域の幅および高さを示すサイズ情報を変更することを特徴とする請求項 8 に記載の文書処理方法。

【請求項 10】

前記配置工程では、前記共通可変領域が配置された後、選択されたページにおいて前記共通属性が解除された場合、前記選択されたページについて前記共通可変領域を残し、それ以外のページについては前記共通可変領域を削除することを特徴とする請求項 7 乃至 9 のいずれか 1 項に記載の文書処理方法。

【請求項 11】

新たなページを追加する文書編集工程をさらに有し、

前記文書編集工程により新たにページが追加された際に、前記配置工程では、前記新たなページにも前記共有可変領域を追加することを特徴とする請求項 7 乃至 10 のいずれか

10

20

30

40

50

1 項に記載の文書処理方法。

【請求項 1 2】

ページのデータを解析し、オブジェクトの種類ごとに領域を分類する原稿解析工程をさらに有し、

前記配置工程では、ユーザの指示に従って前記ページ内のイメージオブジェクトまたは文字オブジェクト上に前記共通可変領域を配置することを特徴とする請求項 7 乃至 1 1 のいずれか 1 項に記載の文書処理方法。

【請求項 1 3】

文書データの可変領域にデータを割り当てて印刷する可変印刷処理をコンピュータにより行うためのプログラムであって、

前記可変領域に対して共通属性を指定する指定手段と、

前記指定手段により前記共通属性が指定された場合、前記可変領域と同一の共通可変領域を前記可変領域が配置されていないページに配置する配置手段と、

前記共通可変領域が配置された複数ページから選択された選択ページの前記共通可変領域の位置およびサイズの少なくとも 1 つを変更する変更手段としてコンピュータを機能させ、

前記選択ページとは異なるページの前記共通可変領域は、前記変更工程により前記選択ページにおいて変更された位置およびサイズと同じ位置および同じサイズにて配置されることを特徴とするプログラム。

【請求項 1 4】

前記選択ページとは異なるページの前記共通可変領域が、前記変更手段により前記選択ページにおいて変更された位置およびサイズと同じ位置および同じサイズにて配置された結果、前記共通可変領域が収まらないページについて、前記変更手段は、前記ページに配置された共通可変領域の共通属性を解除し、解除された領域を前記ページ内に収まる位置に変更することを特徴とする請求項 1 3 に記載のプログラム。

【請求項 1 5】

前記変更手段は、前記共通属性が解除された領域を移動した結果、前記移動された領域がページ内に収まらない場合、前記ページについて前記解除された領域がページ内に収まるように前記解除された領域の幅および高さを示すサイズ情報を変更することを特徴とする請求項 1 4 に記載のプログラム。

【請求項 1 6】

前記配置手段は、前記共通可変領域が配置された後、選択されたページにおいて前記共通属性が解除された場合、前記選択されたページについて前記共通可変領域を残し、それ以外のページについては前記共通可変領域を削除することを特徴とする請求項 1 3 乃至 1 5 のいずれか 1 項に記載のプログラム。

【請求項 1 7】

新たなページを追加する文書編集手段としてさらにコンピュータを機能させ、

前記文書編集手段により新たにページが追加された際に、前記配置手段は、前記新たなページにも前記共有可変領域を追加することを特徴とする請求項 1 3 乃至 1 6 のいずれか 1 項に記載のプログラム。

【請求項 1 8】

ページのデータを解析し、オブジェクトの種類ごとに領域を分類する原稿解析手段としてさらにコンピュータを機能させ、

前記配置手段は、ユーザの指示に従って前記ページ内のイメージオブジェクトまたは文字オブジェクト上に前記共通可変領域を配置することを特徴とする請求項 1 3 乃至 1 7 のいずれか 1 項に記載のプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、文書処理方法および装置および媒体に関するもので、特にパーソナルコンピ

10

20

30

40

50

ュータ等の情報処理とプリンタからなるシステムにおける文書処理方法および装置および媒体に関するものである。

【背景技術】

【0002】

従来、PC等で作成した文書データやスキャナから読み取った画像データなどに対して、印刷部ごとに異なるデータを付加して印刷を行う可変印刷が行われている。また従来、印刷機によって電子データの印刷を行う際に「複製禁止」などの文字を原稿の可視性を損なわない程度に原稿データに重ねるスタンプが行われている。スタンプは、原稿データに重ねて文字等のデータを付加することにより、原稿のオリジナルデータの複製を抑止する効果がある。

10

【0003】

スタンプと可変印刷を組み合わせることにより、スタンプによって付加するデータを印刷部ごとに変えて印刷することができる。これにより、印刷された原稿が複製されたとき、スタンプの内容からコピー元を特定することができるようになる。スタンプは、原稿データが複数枚のページに及ぶとき、基本的に原稿データの全ページに対して付加される。全ページに対して同一のデータを付加する場合は、共通フィールドとして一つの基本となるフィールドを設定し、そのほかのフィールドは基本となるフィールドを参照することで全てのページに付加することができる（特許文献1参照）。

【0004】

また、スタンプを原稿データと重ねると原稿データが見づらくなるため、スタンプを付加する位置をユーザによって範囲指定させた後、その範囲内で原稿の内容（文書など）となるべく重ならないように配置することもできた（特許文献2参照）。

20

【特許文献1】特開2001-175654号公報

【特許文献2】特開2000-209433号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

しかしながら、上述した従来の技術では全ページに対してスタンプを付加する場合、同一のデータを同じ位置に付加することは可能であったが、フィールドの特定の属性（フィールドの位置やサイズなど）を共通で使うことができなかった。たとえば、特定のページについてのみスタンプデータの位置などを修正する場合には、共通フィールドから独立したフィールドに戻す必要があり、フィールドの特定の属性（フィールドの位置やサイズなど）を全ページ共通で使うことができなかった。そのため、原稿サイズの違うページなど特定のページのみユーザの望む位置にスタンプが配置されなかった場合、共通フィールドではなく個別のフィールドとして特定のページに対するスタンプを設定しなおさなければならず、作業に手間がかかっていた。

30

【0006】

また、スタンプとして複製抑止の効果を得るためには原稿データと重なっている必要があり、ユーザが特定のデータ（画像など）を保護したい場合に保護したい部分のみにスタンプを配置する。そのためにはユーザが個々にスタンプデータの位置などを調整しなければならず、作業に手間がかかっていた。

40

【課題を解決するための手段】

【0007】

本発明によれば、文書データの可変領域にデータを割り当てて印刷する可変印刷処理を行うための文書処理装置であって、前記可変領域に対して共通属性を指定する指定手段と、前記指定手段により前記共通属性が指定された場合、前記可変領域と同一の共通可変領域を前記可変領域が配置されていないページに配置する配置手段と、前記共通可変領域が配置された複数ページから選択された選択ページの前記共通可変領域の位置およびサイズの少なくとも1つを変更する変更手段とを有し、前記選択ページとは異なるページの前記共通可変領域は、前記変更手段により前記選択ページにおいて変更された位置およびサイ

50

ズと同じ位置および同じサイズにて配置される。

【発明の効果】

【０００８】

本発明によって、全てのページでフィールドの属性を共有しながら、特定の属性のみを個別に設定することを可能とすることにより、作業効率が改善される。

【０００９】

また、画像・テキスト等に重なるように自動的にフィールドの位置を調整するため、特定データのみにスタンプを付加する作業効率が改善される。

【発明を実施するための最良の形態】

【００１０】

以下、図面を参照しながら本発明に係る実施の形態を詳細に説明する。

【００１１】

<システム概要>

まず、本発明に係る文書処理システムの概要について、図１乃至図１２を参照して以下に説明する。尚、この文書処理システムは、一般のアプリケーションによって作成されたデータファイルを電子原稿ファイルに変換する電子原稿ライタと、その電子原稿ファイルを編集する機能を提供する製本アプリケーションとを含む。そして、製本アプリケーションは、作成されたデータを一まとめにした文書の作成及び編集を可能とし、その操作性を向上させて文書編集を効率的に行えるものである。

【００１２】

<システム構成及び動作>

図１は、本実施形態の文書処理システムのソフトウェア構成を示す図である。文書処理システムは、本発明の情報処理装置に好適な実施形態であるデジタルコンピュータ１００（以下、ホストコンピュータとも呼ぶ）によって実現されている。図１に示す一般アプリケーション１０１は、ワードプロセッシングやスプレッドシート、フォトレタッチ、ドロー、あるいはペイント、プレゼンテーション、テキスト編集などの機能を提供するアプリケーションプログラムであり、ＯＳに対する印刷機能を有している。これらのアプリケーションは、作成された文書データや画像データなどのアプリケーションデータを印刷する際に、オペレーティングシステム（ＯＳ）によって提供される所定のインタフェース（一般に、ＧＤＩと呼ばれる）を利用する。即ち、アプリケーション１０１は、作成したデータを印刷するために、上述のインタフェースを提供するＯＳの出力モジュールに対して、あらかじめ定められる、ＯＳに依存する形式の出力コマンド（ＧＤＩ関数と呼ばれる）を送信する。一方、出力コマンドを受けた出力モジュールは、その出力コマンドをプリンタ等の出力デバイスが処理可能な形式に変換して、変換されたコマンド（ＤＤＩ関数と呼ばれる）を出力する。出力デバイスが処理可能な形式はデバイスの種類やメーカ、機種などによって異なる。そのため、デバイスごとにデバイスドライバが提供されており、ＯＳではそのデバイスドライバを利用してコマンドの変換を行い、印刷データを生成し、ＪＬ（Job Language）でくくることにより印刷ジョブが生成される。ＯＳとしてマイクロソフト社のウィンドウズ（登録商標）を利用する場合には、前述した出力モジュールとしてはＧＤＩ（Graphic Device Interface）と呼ばれるモジュールが相当する。

【００１３】

電子原稿ライタ１０２は、上述のデバイスドライバを改良したものであり、本文書処理システム実現のために提供されるソフトウェアモジュールである。但し、電子原稿ライタ１０２は特定の出力デバイスを目的としておらず、詳細は後述する製本アプリケーション１０４やプリンタドライバ１０６により処理可能な形式に出力コマンドを変換する。この電子原稿ライタ１０２による変換後の形式（以後電子原稿形式と呼ぶ）は、ページ単位の原稿を詳細な書式をもって表現可能であれば特に問わない。実質的な標準形式のうち、例えばアドビシステムズによるＰＤＦ形式やＳＶＧ形式などが電子原稿形式として採用できる。

【００１４】

一般アプリケーション 101 から電子原稿ライタ 102 を利用させる場合には、出力に使用するデバイスドライバとして電子原稿ライタ 102 を指定してから印刷を実行させる。但し、電子原稿ライタ 102 によって作成されたままの電子原稿ファイルは、電子原稿ファイルとして完全な形式を備えていない。そのため、デバイスドライバとして電子原稿ライタ 102 を指定するのは製本アプリケーション 104 であり、その管理下でアプリケーションデータの電子原稿ファイルへの変換が実行される。そして、製本アプリケーション 104 は電子原稿ライタ 102 が生成した新規の不完全な電子原稿ファイルを後述する形式を備えた電子原稿ファイルとして完成させる。以下、この点を明瞭に識別する必要がある場合、電子原稿ライタ 102 によって作成されたファイルを「電子原稿ファイル」と呼び、製本アプリケーション 104 によって構造を与えられた電子原稿ファイルを「ブックファイル」と呼ぶ。また、特に区別する必要がない場合は、アプリケーションにより生成されるドキュメントファイル、電子原稿ファイル、及びブックファイルをいずれも文書ファイル（または文書データ）と呼ぶ。

10

【0015】

このようにデバイスドライバとして電子原稿ライタ 102 を指定し、一般アプリケーション 101 によりデータを印刷させる。その結果、アプリケーションデータはアプリケーション 101 によって定義されたページ（以後論理ページあるいは原稿ページと呼ぶ）を単位とする電子原稿形式に変換され、電子原稿ファイル 103 としてハードディスクなどの記憶媒体に格納される。尚、ハードディスクは、本実施形態の文書処理システムを実現するコンピュータが備えているローカルドライブであっても良く、ネットワークに接続されている場合にはネットワーク上に提供されるドライブであっても良い。

20

【0016】

製本アプリケーション 104 は電子原稿ファイル（あるいはブックファイル）103 を読み込み、それを編集するための機能を利用者に提供する。但し、製本アプリケーション 104 は、各ページの内容を編集する機能は提供しておらず、ページを最小単位として構成される、後述する章やブックの構造を編集するための機能を提供している。

【0017】

製本アプリケーション 104 によって編集されたブックファイル 103 を印刷する際には、製本アプリケーション 104 によって電子原稿デスプーラ 105 が起動される。電子原稿デスプーラ 105 は、製本アプリケーションと共にコンピュータ内にインストールされるプログラムモジュールである。電子原稿デスプーラ 105 は、製本アプリケーションで利用するドキュメント（ブックファイル）を印刷する際に、プリンタドライバへ描画データを出力するために使用されるモジュールである。電子原稿デスプーラ 105 は、指定されたブックファイルをハードディスクから読み出し、ブックファイルに記述された形式で各ページを印刷するために、前述した OS の出力モジュールに適合する出力コマンドを生成し、不図示の出力モジュールに出力する。その際に、出力デバイスとして使用されるプリンタ 107 用のプリンタドライバ 106 がデバイスドライバとして指定される。上述の出力モジュールは、受信した出力コマンドをデバイスコマンドに変換して指定されたプリンタ 107 用のプリンタドライバ 106 に出力し、そのプリンタドライバ 106 がプリンタ 107 で解釈実行可能なページ記述言語等のコマンドに変換する。そして、変換されたコマンドはプリンタドライバ 106 から不図示のシステムスプーラを介してプリンタ 107 に送信され、プリンタ 107 によってコマンドに応じた画像が印刷される。デスプーラ 105 は、プリンタドライバ 106 に対して入力されるデータ（出力データ）を生成する工程を実行する手段である。

30

40

【0018】

図 2 は、コンピュータ 100 のハードウェアブロック図である。図 2 において、CPU 201 は、ROM 203 のプログラム用 ROM に記憶された、あるいはハードディスク 211 から RAM 202 にロードされた OS や一般アプリケーション、製本アプリケーションなどのプログラムを実行し、図 1 のソフトウェア構成や、後述するフローチャートの手順を実現する。RAM 202 は、CPU 201 の主メモリ、ワークエリア等として機能す

50

る。キーボードコントローラ（ＫＢＣ）２０５は、キーボード２０９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。ＣＲＴコントローラ（ＣＲＴＣ）２０６は、ＣＲＴディスプレイ２１０の表示を制御する。ディスクコントローラ（ＤＫＣ）２０７は、ブートプログラム、種々のアプリケーション、フォントデータ、ユーザファイル、後述する編集ファイル等を記憶するハードディスク（ＨＤ）２１１やフレキシブルディスク（ＦＤ）等とのアクセスを制御する。ＰＲＴＣ２０８は、接続されたプリンタ１０７との間の信号の交換を制御する。ＮＣ２１２はネットワークに接続されて、ネットワークに接続された他の機器との通信制御処理を実行する。

【００１９】

また、本実施形態で説明するフローチャートは、ＣＰＵ２０１により各フローチャートの手順を記述したプログラムを実行することで実現される。

10

【００２０】

< 電子原稿データの形式 >

編集アプリケーション１０４の詳細を言及する前に、ブックファイルのデータ形式について説明する。ブックファイルは紙媒体の書物を模倣した３層の層構造を有する。まず、上位層は「ブック」と呼ばれ、１冊の本を模倣しており、その本全般に係る属性が定義されている。その下の中間層は、本でいう章に相当し、やはり「章」と呼ばれる。各章についても、章毎の属性が定義できる。そして、下位層は「ページ」であり、アプリケーションプログラムで定義された各ページに相当する。尚、各ページについてもページ毎の属性が定義できる。更に、一つのブックは複数の章を含んでいてよく、また一つの章は複数のページを含むこともできる。

20

【００２１】

図３に示す（Ａ）は、ブックファイルの形式の一例を模式的に示す図である。図示するように、この例では、ブックファイルにおけるブック、章、ページは、それぞれに相当するノードにより示されている。一つのブックファイルは一つのブックを含む。ブック、章は、ブックとしての構造を定義するための概念であるから、定義された属性値と下位層へのリンクとをその実体として含む。ページは、アプリケーションプログラムによって出力されたページ毎のデータを実体として有する。そのため、ページは、その属性値のほかに、原稿ページの実体（原稿ページデータ）と各原稿ページデータへのリンクを含む。尚、紙媒体等に出力する際の印刷ページは複数の原稿ページを含む場合がある。この構造に関してはリンクによって表示されず、ブック、章、ページ各階層における属性として表示される。

30

【００２２】

図３において、ブック３０１には、ブック属性が定義されていると共に、２つの章３０２Ａ、３０２Ｂがリンクされている。このリンクにより、章３０２Ａ、３０２Ｂがブック３０１に包含されていることが表示される。章３０２Ａには、ページ３０３Ａ、３０３Ｂがリンクされ、これらページが含まれることが示されている。各ページ３０３Ａ、３０３Ｂにはそれぞれ属性値が定義され、その実体である原稿ページデータ（１）、（２）へのリンクが含まれる。これらリンクは、図３に示す（Ｂ）のように原稿ページデータ３０４のデータ（１）、（２）を指し示し、ページ３０３Ａ、３０３Ｂの実体が原稿ページデータ（１）、（２）であることを表示している。

40

【００２３】

図４は、ブック属性のリストである。下位層と重複して定義可能な項目に関しては、下位層の属性値が優先採用される。そのため、ブック属性にのみ含まれる項目に関しては、ブック属性に定義された値はブック全体を通して有効な値となる。しかし、下位層と重複する項目については、下位層において定義されていない場合における既定値としての意味を有する。尚、図示された各項目は具体的に１項目に対応するのではなく、関連する複数の項目を含むものもある。

【００２４】

図５は章属性の、図６はページ属性のリストである。章属性とページ属性との関係もブ

50

ック属性と下位層の属性との関係と同様である。図4～図6から明らかなように、ブック属性に固有の項目は、印刷方法、製本詳細、表紙／裏表紙、インデックス紙、合紙、章区切りの6項目である。これらはブックを通して定義される項目である。印刷方法の属性としては、片面印刷、両面印刷、製本印刷の3つの値を指定できる。ここで、製本印刷とは、別途指定する枚数の用紙を束にして2つ折りにし、その束をつづり合わせることで製本が可能となる形式で印刷する方法である。製本詳細属性としては、製本印刷が指定されている場合に、見開き方向や、束になる枚数等が指定できる。

【0025】

表紙／裏表紙属性は、ブックとしてまとめられる電子原稿ファイルを印刷する際に、表紙及び裏表紙となる用紙を付加することの指定、及び付加した用紙への印刷内容の指定を含む。インデックス紙属性は、章の区切りとして、印刷装置に別途用意される耳付きインデックス紙の挿入の指定及びインデックス（耳）部分への印刷内容の指定を含む。この属性は、印刷用紙とは別に用意された用紙を所望の位置に挿入するインサート機能を持ったインサータが使用する印刷装置に備えられている場合か、あるいは、複数の給紙カセットを使用可能である場合に有効となる。これは合紙属性についても同様である。

10

【0026】

合紙属性は、章の区切りとして、インサータから、あるいは給紙カセットから供給される用紙の挿入の指定、及び合紙を挿入する場合には、給紙元の指定などを含む。

【0027】

章区切り属性は、章の区切り目において、新たな用紙を使用するか、新たな印刷ページを使用するか、特に何もしないか等の指定を含む。片面印刷時には、新たな用紙の使用と新たな印刷ページの使用とは同じ意味を持つ。両面印刷時には、「新たな用紙の使用」を指定すれば連続する章が1枚の用紙に印刷されることは無いが、「新たな印刷ページの使用」を指定すれば、連続する章が1枚の用紙の表裏に印刷されることがあり得る。

20

【0028】

章属性に関しては、章に固有の項目はなく、全てブック属性と重複する。従って、章属性における定義とブック属性における定義とが異なれば、章属性で定義された値が優先する。ブック属性と章属性とにのみ共通する項目は、用紙サイズ、用紙方向、N - u p印刷指定、拡大縮小、排紙方法の5項目である。これらのうち、N - u p印刷指定属性は1印刷ページに含まれる原稿ページ数を指定するための項目である。指定可能な配置としては、1×1、1×2、2×2、3×3、4×4などがある。排紙方法属性は、排出した用紙にステイブル処理を施すか否かを指定するための項目であり、この属性の有効性は使用する印刷装置がステイブル機能を有するか否かに依存する。

30

【0029】

ページ属性に固有の項目には、ページ回転属性、ズーム、配置指定、アノテーション、ページ分割などがある。ページ回転属性は、原稿ページを印刷ページに配置する際の回転角度を指定するための項目である。ズーム属性は、原稿ページの変倍率を指定するための項目である。変倍率は、仮想論理ページ領域のサイズを100%として指定される。仮想論理ページ領域とは、原稿ページを、N u p等の指定に応じて配置した場合に、1原稿ページが占める領域である。例えば、1×1であれば、仮想論理ページ領域は1印刷ページに相当する領域となり、1×2であれば、1印刷ページの各辺を約70パーセントに縮小した領域となる。

40

【0030】

ブック、章、ページについて共通な属性としては、ウォーターマーク属性及びヘッダ・フッタ属性がある。ここで、ウォーターマークとは、アプリケーションで作成されたデータに重ねて印刷される、別途指定される画像や文字列などである。ヘッダ・フッタは、それぞれ各ページの上余白及び下余白に印刷されるウォーターマークである。但し、ヘッダ・フッタには、ページ番号や日時など変数により指定可能な項目が用意されている。尚、ウォーターマーク属性およびヘッダ・フッタ属性において指定可能な内容は、章とページとは共通であるが、ブックはそれらと異なっている。ブックにおいてはウォーターマーク

50

やヘッダ・フッタの内容を設定できるし、また、ブック全体を通してどのようにウォーターマークやヘッダ・フッタを印刷するかを指定することができる。一方、章やページでは、その章やページにおいて、ブックで設定されたウォーターマークやヘッダ・フッタを印刷するか否かを指定できる。

【 0 0 3 1 】

＜ブックファイルの生成手順＞

ブックファイルは上述したような構造及び内容を有している。次に、製本アプリケーション 1 0 4 及び電子原稿ライタ 1 0 2 によるブックファイルの作成手順について説明する。ブックファイルの作成手順は、製本アプリケーション 1 0 4 によるブックファイルの編集操作の一環として実現される。

10

【 0 0 3 2 】

図 7 は、製本アプリケーション 1 0 4 によりブックファイルを開く際の手順である。まず、製本アプリケーション 1 0 4 は、開こうとするブックファイルが、新規作成すべきものであるか、それとも既存のものであるか判定する（ステップ S 7 0 1）。新規作成の場合、製本アプリケーション 1 0 4 は、章を含まないブックファイルを新規に作成する（ステップ S 7 0 2）。新規に作成されるブックファイルは、図 3 の例で示せば、ブックノード 3 0 1 のみ有し、章のノードに対するリンクが存在しないブックのノードとなる。ブック属性は、新規作成用としてあらかじめ用意された属性のセットが適用される。そして、製本アプリケーション 1 0 4 は、新規ブックファイルを編集するためのユーザインタフェース（UI）画面を表示する（ステップ S 7 0 3）。図 1 1 は、新規にブックファイルが作成された際の UI 画面の一例である。この場合は、ブックファイルは実質的な内容を持たないため、UI 画面 1 1 0 0 には何も表示されない。

20

【 0 0 3 3 】

一方、既存のブックファイルがあれば、製本アプリケーション 1 0 4 は、指定されたブックファイルを開き（ステップ S 7 0 3）、そのブックファイルの構造、属性、内容に従ってユーザインタフェース（UI）画面を表示する。図 1 0 は、既存のブックファイルから指定されたブックファイルを表示する UI 画面の一例である。UI 画面 1 1 0 0 は、ブックの構造を示すツリー部 1 1 0 1 と、印刷された状態を表示するプレビュー部 1 1 0 2 とを含む。ツリー部 1 1 0 1 には、ブックに含まれる章、各章に含まれるページが図 3 に示す（A）のような木構造で表示される。ツリー部 1 1 0 1 に表示されるページは原稿ページである。またプレビュー部 1 1 0 2 には、印刷ページの内容が縮小されて表示される。その表示順序はブックの構造を反映したものとなっている。

30

【 0 0 3 4 】

さて、開かれたブックファイルには、電子原稿ライタ 1 0 2 によって電子原稿ファイルに変換されたアプリケーションデータを、新たな章として追加することができる。この機能を電子原稿インポート機能と呼ぶ。図 7 に示す手順によって新規に作成されたブックファイルに電子原稿インポートすることで、そのブックファイルには実体が与えられる。この機能は、図 1 0 の画面にアプリケーションデータをドラッグアンドドロップ操作することで起動される。

【 0 0 3 5 】

40

図 8 に電子原稿インポートの手順を示す。まず、製本アプリケーション 1 0 4 は、指定されたアプリケーションデータを生成したアプリケーションプログラムを起動する。そして、製本アプリケーション 1 0 4 は、デバイスドライバとして電子原稿ライタ 1 0 2 を指定してアプリケーションデータを印刷出力させることで、電子原稿データに変換する（ステップ S 8 0 1）。変換を終えたなら、製本アプリケーション 1 0 4 は、変換されたデータが画像データであるか否かを判定する（ステップ S 8 0 2）。この判定は、ウィンドウズ（登録商標）OS の下であれば、アプリケーションデータのファイル拡張子に基づいて行われる。例えば、拡張子が「b m p」であればウィンドウズ（登録商標）ビットマップデータであり、「j p g」であれば j p e g 圧縮された画像データ、「t i f f」であれば t i f f 形式の画像データであると判定できる。また、このような画像データの場合は

50

S 8 0 1 0 のようにアプリケーションを起動せずに、画像データから直接電子原稿ファイルを生成することが可能であるため、S 8 0 1 0 の処理を省略することも可能である。

【 0 0 3 6 】

ここで、画像データでなかった場合、製本アプリケーション 1 0 4 は、ステップ S 8 0 1 で生成された電子原稿ファイルを、現在開かれているブックファイルのブックに、新たな章として追加する（ステップ S 8 0 3 ）。章属性としては、ブック属性と共通するものについてはブック属性の値がコピーされ、そうでないものについては、予め用意された規定値に設定される。

【 0 0 3 7 】

また、ステップ S 8 0 2 で画像データである場合、製本アプリケーション 1 0 4 は、原則として新たな章を追加せず、指定されている章に、ステップ S 8 0 1 で生成された電子原稿ファイルに含まれる各原稿ページを追加する（ステップ S 8 0 4 ）。但し、ブックファイルが新規作成されたファイルであれば、新たな章が作成されて、その章に属するページとして電子原稿ファイルの各ページが追加される。ページ属性は、上位層の属性と共通のものについてはその属性値が与えられ、アプリケーションデータにおいて定義された属性を電子原稿ファイルに引き継いでいるものについてはその値が与えられる。例えば、N u p 指定などがアプリケーションデータにおいて指定されていた場合には、その属性値が引き継がれる。このようにして、新規なブックファイルが作成され、あるいは、新規な章が追加される。

【 0 0 3 8 】

図 9 は、図 8 に示すステップ S 8 0 1 において、電子原稿ライター 1 0 2 により電子原稿ファイルを生感させる手順のフローチャートである。まず、電子原稿ライター 1 0 2 は、新たな電子原稿ファイルを作成して、開く（ステップ S 9 0 1 ）。製本アプリケーション 1 0 4 は、指定したアプリケーションデータに対応するアプリケーションを起動し、電子原稿ライターをデバイスドライバとして O S の出力モジュールに対して出力コマンドを送信させる。出力モジュールは、受信した出力コマンドを電子原稿ライター 1 0 2 によって電子原稿形式のデータに変換し、出力する（ステップ S 9 0 2 ）。その出力先はステップ S 9 0 1 で開いた電子原稿ファイルである。製本アプリケーション 1 0 4 は、指定されたデータ全てについて変換が終了したか判定し（ステップ S 9 0 3 ）、終了していれば電子原稿ファイルを閉じる（ステップ S 9 0 4 ）。電子原稿ライター 1 0 2 によって生成される電子原稿ファイルは、図 3 の（ B ）に示される原稿ページデータの実体を含むファイルである。

【 0 0 3 9 】

< ブックファイルの編集 >

以上のようにして、アプリケーションデータからブックファイルを作成することができる。生成されたブックファイルについては、章及びページに対して次のような編集操作が可能である。

- （ 1 ）新規追加
- （ 2 ）削除
- （ 3 ）コピー
- （ 4 ）切り取り
- （ 5 ）貼り付け
- （ 6 ）移動
- （ 7 ）章名称変更
- （ 8 ）ページ番号名称振り直し
- （ 9 ）表紙挿入
- （ 1 0 ）合紙挿入
- （ 1 1 ）インデックス紙挿入
- （ 1 2 ）各原稿ページに対するページレイアウト。

【 0 0 4 0 】

このほか、一旦行った編集操作を取り消す操作や、更に取り消した操作をやり直す操作

10

20

30

40

50

が可能である。これら編集機能により、例えば複数のブックファイルの統合、ブックファイル内で章やページの再配置、ブックファイル内で章やページの削除、原稿ページのレイアウト変更、合紙やインデックス紙の挿入などの編集操作が可能となる。これらの操作を行うと、図4、図5に示す属性に操作結果が反映、あるいはブックファイルの構造に反映される。例えば、ブランクページの新規追加操作を行えば、指定された箇所にブランクページが挿入される。このブランクページは原稿ページとして扱われる。また、原稿ページに対するレイアウトを変更すれば、その変更内容は、印刷方法やN - u p印刷、表紙 / 裏表紙、インデックス紙、合紙、章区切りといった属性に反映される。

【0041】

<ブックファイルの出力>

以上のように作成・編集されるブックファイルは、印刷出力を最終目的とするものである。ここで、利用者が図10に示した製本アプリケーションのUI画面1100からファイルメニューを選択し、そこから印刷を選択すると、指定した出力デバイスにより印刷出力される。この際、まず製本アプリケーション104は、現在開かれているブックファイルからジョブチケットを作成し、そのジョブチケットを電子原稿デスプーラ105に渡す。一方、電子原稿デスプーラ105は、ジョブチケットをOSの出力コマンド、例えばウインドウズ(登録商標)のGDI関数に変換し、それを出力モジュール、例えばGDIに送信する。出力モジュールは、指定されたプリンタドライバ106によってデバイスに適したコマンドを生成し、そのデバイスに送信する。

【0042】

ここで、ジョブチケットは原稿ページを最小単位とする構造を有するデータである。ジョブチケットにおける構造は、用紙上における原稿ページのレイアウトを定義している。ジョブチケットは1ジョブにつき1つ発行される。そのため、まず最上位にドキュメントというノードがあり、文書全体の属性、例えば両面印刷 / 片面印刷などが定義されている。その下には、用紙ノードが属し、用いるべき用紙の識別子やプリンタにおける給紙口の指定などの属性が含まれる。各用紙ノードには、その用紙で印刷されるシートのノードが属する。1シートは1枚の用紙に相当する。各シートには、印刷ページ(物理ページ)が属する。片面印刷ならば1シートには1物理ページが属し、両面印刷ならば1シートに2物理ページが属する。各物理ページには、その上に配置される原稿ページが属する。また物理ページの属性として、原稿ページのレイアウトが含まれる。

電子原稿デスプーラ105は、上述のジョブチケットを、出力モジュールへの出力コマンドに変換する。

【0043】

<その他のシステム構成>

本実施形態の文書処理システムの概要は以上のようなものである。これはスタンドアロン型のシステムであるが、これを拡張したサーバクライアントシステムでもほぼ同様の構成・手順でブックファイルを作成・編集できる。但し、ブックファイルや印刷処理はサーバによって管理される。

【0044】

図12は、サーバクライアント型文書処理システムの構成を示すブロック図である。クライアント文書処理システムは、スタンドアロン型システムに、クライアントモジュールであるDOMS(Document Output Management Service: 文書出力管理サービス)ドライバ109及びDOMSプリントサービスモジュール110、DS(文書サービス)クライアントモジュール108を加えた構成を有する。このクライアント文書処理システム1200に文書管理サーバ1201、印刷集中管理サーバ1202及びプリントサーバ1203が接続されている。これらサーバは、通常ネットワークによってクライアント文書処理システムと接続されるが、サーバが同時にクライアントとしても機能する場合には、ネットワーク間の通信をシミュレートするプロセス間通信によって接続される。

【0045】

尚、図12に示す例では、文書管理サーバ1201と印刷集中管理サーバ1202の両

10

20

30

40

50

方のサーバがクライアントに接続されているが、何れか一方のみネットワーク上に存在する場合もあり得る。例えば、接続されているサーバが文書管理サーバであれば、そのクライアントモジュール 108 を含む文書管理サーバクライアントシステム 1201SC がスタンドアロン型文書管理システムに追加される。また、接続されているサーバが、印刷集中管理サーバ 1202 であれば、そのクライアントモジュールを含む印刷管理サーバクライアントシステム 1202SC が、スタンドアロン型文書管理システムに追加される。

【0046】

文書管理サーバ 1201 は、製本アプリケーション 104 により作成・編集されたブックファイルを格納するサーバである。文書管理サーバ 1201 によってブックファイルを管理する場合、ブックファイルはクライアント PC のローカル HD に代わって、あるいは

10

【0047】

印刷集中管理サーバ 1202 は、クライアント文書管理システム 1200 に格納された、あるいは文書管理サーバ 1201 に格納されたブックファイルの印刷を管理するサーバである。クライアントにおける印刷要求は、DOMS ドライバ 109 および DOMS プリントサービスモジュール 110 を介して印刷集中管理サーバ 1202 の DOMSWG サーバモジュール 1221 に送信される。集中印刷管理サーバ 1202 は、クライアントのプリンタで印刷する場合、クライアントの DOMS プリントサービスモジュール 110 を介して電子原稿デスプーラ 105 に電子原稿データを渡す。一方、プリントサーバ 1203 により印刷する場合、集中印刷管理サーバ 102 は、プリントサーバ 1203 の DOMS プリントサービスモジュール 1203 に送信する。集中印刷管理サーバは、例えば保存されているブックファイルに対して印刷要求を発行した利用者の資格などについてセキュリティチェックを行ったり、印刷処理のログを保存したりする。このように、文書処理システムは、スタンドアロンとしても、クライアントサーバシステムとしても実現できる。

20

【0048】

<プレビュー表示の内容>

既に説明した通り、ブックファイルが製本アプリケーションによって開かれると、図 10 に示したユーザインタフェース画面 1100 が表示される。ツリー部 1101 には、開いているブック（以下、「注目ブック」と呼ぶ）の構造を示すツリーが表示される。プレビュー部には、利用者の指定に応じて、3通りの表示方法が用意されている。まず、第 1 は原稿ページをそのまま表示する原稿ビューと呼ばれるモードである。この原稿ビューモードでは、注目ブックに属する原稿ページの内容が縮小されて表示される。尚、プレビュー部 1102 の表示にレイアウトは反映されない。次に、第 2 は印刷ビューモードである。この印刷ビューモードでは、プレビュー部 1102 には原稿ページのレイアウトが反映された形で原稿ページが表示される。そして、第 3 は簡易印刷ビューモードである。この簡易印刷ビューモードでは、各原稿ページの内容はプレビュー部の表示には反映されず、レイアウトのみが反映される。

30

【0049】

<ステイブル制御>

次に、ステイブル機能を有するプリンタと接続するコンピュータ 100 の製本アプリケーション 104 によって行われるステイブル制御について説明する。

40

【0050】

図 13 は、ステイブル制御システムの構成を示すブロック図である。図示するように、ステイブル制御システムは、図 2 に示したホストコンピュータ 100 とステイブル機能を有するプリンタ 107 とで構成される。

【0051】

以下、プリンタ 107 の構成について説明する。尚、本発明の機能が実現されるのであれば、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

50

等のネットワークを介して接続がなされ、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。

【0052】

図示するように、プリンタ107において、1301はプリンタCPUである。CPU1301は、ROM1302内のプログラム用ROM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1304に接続される印刷部I/F1305を介して、印刷部（プリンタエンジン）1306に出力情報としての画像信号を出力する。また、このROM1302内のプログラムROMには、CPU1301の制御プログラム等が記憶され、ROM1302内のフォント用ROMには、出力情報を生成する際に使用するフォントデータ等が記憶される。また、ROM1302内のデータ用ROMには、ハードディスク等の外部メモリ1303がないプリンタの場合、ホストコンピュータ100上で利用される情報等が記憶されている。

10

【0053】

また、CPU1301は、入力部1307を介してホストコンピュータ100との通信処理が可能となっており、プリンタ107内の情報等をホストコンピュータ100に通知できる。RAM1308は、CPU1301の主メモリや、ワークエリア等として機能するRAMであり、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。

【0054】

尚、RAM1308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。上述したハードディスク（HD）、ICカード等の外部メモリ1303は、メモリコントローラ（MC）1309によりアクセスを制御される。外部メモリ1303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、操作パネル1311は操作のためのスイッチ及びLED表示器等で構成されている。

20

【0055】

また、上述の外部メモリ1303は1個に限らず、複数個備えられ、内蔵フォントに加えてオプションカード、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良い。更に、図示しないNVRAMを有し、操作パネル1311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良い。

30

【0056】

<可変印刷>

製本アプリケーションは、生成したブックファイルのページ属性として、可変印刷を実行するための可変フィールドの設定を行うことが可能である。すなわち、図7のバリエーション項目7に、可変フィールドに関する情報が設定保存される。可変フィールドは、本実施形態では文書中にオーバーレイされる。しかし、可変フィールドは文書に挿入されるものであってもよい。

【0057】

図14は可変印刷のページレイアウト例を示している図である。1400は原稿ページ、1401は可変印刷を実行するための可変フィールドを指している。可変フィールドは原稿ページ上に複数配置することも可能であるし、一つも配置しないことも可能である。可変フィールドの中に、接続したデータベースからデータを差込み、カスタマイズして印刷を行う。1402は接続しているデータベースのデータを示している。データベースにはレコードと呼ばれる識別子があり、このレコード単位で原稿ページに対してデータを差し替える。この例では、データベース1402には、(1)～(4)の4レコードのデータが保持されている（図中では丸囲み数字で示す）。1402のデータベースのレコードには、それぞれテキストのデータ（1403）とイメージのデータ（1404）が設定されている。この例では、原稿ページ1ページ目に設定されている可変フィールドにはテキストデータ（1403）、原稿ページ2ページ目に設定されている可変フィールドにはイメージ

40

50

データ(1404)を差し込むように設定されているものとする。可変フィールドの設定や、データベースデータとの関連の設定については、後述する。

【0058】

1405は実際にデータベースのデータが差し込まれた原稿ページの例を示している。レコード(1)が差し込まれると、原稿ページ1ページ目にはレコード(1)のテキストデータ"AAA"(1406)が差し込まれ、原稿ページ2ページ目にはイメージデータ(飛行機のイメージ1407)が差し込まれて原稿ページがレイアウトされる。レコード(2)が差し込まれると、原稿ページの1ページ目にはレコード(2)のテキストデータ"BBB"(1408)が差し込まれ、原稿ページ2ページ目にはイメージデータ(トラックのイメージ1409)が差し込まれて原稿ページがレイアウトされる。レコード(3)が差し込まれると、原稿ページ1ページ目にはレコード(3)のテキストデータ"CCC"(1410)が差し込まれ、原稿ページ2ページ目にはイメージデータ(船のイメージ1411)が差し込まれて原稿ページがレイアウトされる。レコード(4)が差し込まれると、原稿ページ1ページ目にはレコード(4)のテキストデータ"DDD"(1412)が差し込まれ、原稿ページ2ページ目にはイメージデータ(自動車のイメージ1413)が差し込まれて原稿ページがレイアウトされる。原稿ページ3ページ目については、可変フィールドが設定されていないので、データの差込みは実行されない。しかし、データの差込みは実行されなくても、すべてのレコードに対して原稿ページ3ページ目は印刷される。このように、可変印刷を行うと、原稿ページ×レコード数分の印刷が行われることとなる。このレコードの挿入処理は、図43のステップS4304で行われる。

【0059】

図15のウィンドウ1500は前述した可変印刷の設定を行う可変印刷エディタのUI例を示している。この可変印刷エディタは、製本アプリケーションから起動され、ブックファイルの論理ページ単位で設定を行うことが可能である。製本アプリケーションからの起動方法は、製本アプリケーションのプレビュー画面で単一の論理ページを選択して、製本アプリケーションのUI画面からツールメニューを選択して"可変印刷エディタ"を選択する方法がある。また、製本アプリケーションのプレビュー画面で単一の論理ページを選択してマウス右クリックメニューで"可変印刷エディタ"を選択する方法などでもよい。その他(たとえばツールボタンなど)の起動方法であっても良い。また、可変印刷エディタは製本アプリケーションの一部を構成しても良い。

【0060】

1500は可変印刷エディタの全体UIを示している。可変印刷エディタが起動されると全体UI1500が表示される。1501は可変印刷エディタのウィンドウを最小化、最大化、閉じることが出来るボタンを示している。1502は可変印刷エディタの各種編集操作を行うメニューバーを示している。1503は可変印刷エディタのツールボタンを示している。これらはメニューで行える操作を簡単な操作で選択できるように用意されている。1504はスクロールバーを示しており、可変印刷エディタで原稿ページを表示しきれない場合に表示され、ウィンドウ全体をスクロールして原稿ページ全体を編集することを可能にする。1505は可変印刷エディタのグrabコントロールを示しており、マウスポインタを使ってウィンドウサイズを任意に変更することを可能にする。1506は可変印刷エディタで編集対象としている原稿ページを示している。可変印刷エディタで原稿ページのページ移動機能を用意し、製本アプリケーションに戻らなくとも、ページ移動を可能にしてもよい。1507は原稿ページ上に描画されているオブジェクトである。可変印刷エディタでは前述した可変フィールドの設定のみを機能として持つので、他の原稿ページ上のオブジェクトに対しては編集作業を施すことはできない。1508はテキストデータを差し込むテキスト可変フィールドを示している。1509はイメージデータを差し込むイメージ可変フィールドを示している。この実施形態ではテキスト可変フィールド、イメージ可変フィールドを区別しない場合、可変フィールドという言葉を用いている。これらのフィールドは1502のメニューや1503のツールボタンで可変フィールド作成機能を選択して、マウスカーソルで矩形を描画することにより作成することが可能である

。このように、可変印刷エディタは、ユーザによるGUIを介した指示に応じて、可変フィールドすなわち可変領域の位置やサイズ、挿入されるオブジェクトの種類等を、図17、図19等を参照して後述する情報（属性情報）として計算し、あるいは設定し、保存する。なお、可変印刷エディタは、単にユーザの指示通りに動作するのみならず、属性情報で指定された一定の条件に従って、可変領域の位置やサイズ等を変更し、保存する処理も行う。この処理も、図35等を参照して後述する。

【0061】

図15のウィンドウ1510は、可変フィールドへ差し込むデータを設定するために使用するデータベースの接続ダイアログ（UI）の例を示している。1510は接続ダイアログの概観、1511は接続ダイアログを閉じるボタンを示している。接続ダイアログは、1502のメニューや1503のツールボタンを選択することで表示させることが可能である。1512はデータベースへの接続を行うエリアである。すでに接続されていれば、接続中のデータベースファイル名が表示される。未接続であれば空白である。接続を行うときは、参照ボタン（1513）を押下し、データベースファイル参照ダイアログ等を表示させ、データベースファイルを指定する。データベースへ接続中に参照ボタンで他のデータベースファイルを選択したときは、選択したデータベースへ接続を切り替える。製本アプリケーションで接続が出来るデータベースの数は1つである。複数のデータベースとの接続はできない。1514～1519はデータベースに接続中に有効になるコントロールであり、接続中のデータベースの内容を示すためのものである。1514は接続しているデータベースのレコードを移動させるレコード移動ボタンである。1515は接続しているデータベースの総レコード数と、現在内容を表示しているレコード番号を示すものである。1516は接続しているデータベースに設定されているカラム名を示している。カラム名1516はデータベースの各レコードデータがどのような属性を持っているかを示すもので、図15の例では、“名前”と“イメージ”（1517）という属性を持ったデータベースであることを示している。1518は接続しているデータベースの各レコードのカラム名に対する値（データ）を示している。図15では、1レコード目のデータは“名前”という属性に対して“AAA”、“イメージ”という属性に対して“XXXX.bmp”という値を持っていることを示す。1514のレコード移動ボタンでレコードを移動すると、1519のデータの内容も移動したレコード番号に合わせて変化する。1520は接続ダイアログ（1510）で設定した内容を確定するOKボタン、1521は設定内容を反映せずに終了するキャンセルボタンを示している。

【0062】

図16のウィンドウ16-1は可変フィールドにデータベースデータの関連付けを示したUI例である。1600のテキスト可変フィールドへは前述した“名前”というカラムが設定されている。設定方法としては、テキスト可変フィールドのプロパティなどで、接続しているデータベースのカラムを設定できるようにすれば良い。テキスト可変フィールドへ設定できるデータベースのカラムの数は複数であっても良い。1601のイメージ可変フィールドへは前述した“イメージ”というカラムが設定されている。設定方法としては、イメージ可変フィールドのプロパティなどで、接続しているデータベースのカラムを設定できるようにすれば良い。イメージ可変フィールドへ設定できるデータベースのカラムの数は単数である。

【0063】

図16のウィンドウ16-2は設定した内容をプレビューしたときのUI例である。プレビューでは、可変フィールドへ設定したデータベースのデータが差し込まれた状態を確認できる。1602は図16-1で示したテキスト可変フィールド（1600）へ“名前”のカラムの値“AAA”が差し込まれた状態を示している。また1603は図16のウィンドウ16-1で示したイメージ可変フィールド（1601）へ“イメージ”のカラムの値“電車のイメージ”が差し込まれた状態を示している。このときに、設定したカラムに対するデータの型にエラーが発生していたり、データに抜けが発生していたりすると、プレビュー表示することが出来ない。1604はデータベースのレコードを移動するレコード移

動ボタンである。このボタンを使ってレコード移動を行い、それぞれのレコードにおいてプレビュー状態を確認することが出来る。なお、可変印刷は製本アプリケーションの印刷ダイアログで印刷時にON/OFFの指定を可能とする。したがって、印刷ダイアログで可変印刷をしないということを指定すれば、可変印刷エディタで設定した内容は反映せずに印刷することが出来る。

【0064】

<共有可変フィールド>

本願の製本アプリケーションは、文書データの可変領域に、印刷単位ごとに付与されたデータを挿入して印刷する可変印刷処理を行うことができる。

【0065】

任意の可変フィールドを選択し、ユーザにより可変フィールドの共有化が行われると全ページに対して同一のフィールド属性値（可変データの情報、フィールドの位置、サイズ、フォントの情報など）をもつ共有可変フィールドを設定することができる。共有可変フィールドは全ページ同一のフィールド属性値を持つため、共有可変フィールドに設定された任意ページの可変フィールドに対して属性値の変更を行った場合は、その変更が全ページの共有されている共有可変フィールドに反映される。共有可変フィールドを共有可変領域とも呼ぶ。

【0066】

[可変フィールドのデータ構造]

図17に、可変フィールドのデータ構造1700を示す。フィールドIDは、各可変フィールドを識別するために全ての可変フィールドに対して一意な値で割り当てられる識別子である。共有フラグは、可変フィールドが共有されているか否かを示すフラグである。共有化されている可変フィールドにはオン、共有化されていない通常の可変フィールドにはオフが設定される。共有IDは、共有化によって紐付け（関連づけ）されている可変フィールドを識別するための値である。任意の可変フィールドの属性値を変更した際に、同じ共有IDを持つ可変フィールドの属性値が連動して変更される。

【0067】

複数の共有可変フィールドが存在する場合の共有IDの状態を図18に示す。同じ共有ID=1を持つ、1801、1802、1803、1804のフィールド同士の属性値が共有されている。同様に、共有ID=2によって、1805、1806、1807、1808のフィールド同士の属性値も共有されている。例えば、1801の可変フィールドの位置を変更すると1802、1803、1804は1801と同じ位置に移動される。1801が移動したとしても、異なる共有IDを持つ共有可変フィールド1805、1806、1807、1808の位置は移動されない。

【0068】

可変データ情報は、データベースのどのカラムのレコードが可変フィールドに差し込まれるかを示した情報である。配置情報は、可変フィールドの配置に関する情報である。配置情報は、原稿ページのどの位置に共有可変フィールドを配置するかを示した座標情報と、可変フィールドのサイズ情報を含む。また、可変フィールドを配置する位置は原稿ページのサイズによって相対的な位置(右揃え、左右中央揃え、左揃え、上揃え、上下中央揃え、下揃え)で指定することも可能であり、その情報も含まれる。フォント情報は、差し込まれたレコードを表示、印刷するためのフォントの情報である。フォントの種類、フォントサイズ、文字飾り、位置揃え等が含まれる。枠修飾情報は、枠線の色、太さ、塗りつぶしの色、可変フィールドの透過率などの情報である。

【0069】

[共有ID情報のデータ構造]

図19に共有ID情報のデータ構造を示す。共有ID情報は、一つの共有IDに対して一つあり、共有可変フィールドとして共有しながら一部のフィールド属性のみの共有を解除するための情報が格納されている。一部のフィールド属性の共有解除を行う処理については後述する。

10

20

30

40

50

【 0 0 7 0 】

共有フラグには、位置・サイズ・フォント・枠修飾のフラグがあり、それぞれオンまたはオフの値が格納される。オンの場合には、そのフィールド属性は共有されており、オフの場合には該当する項目が共有されていないことを示す。なお、例として4つのフラグの項目を挙げたが、この項目に限らず他の項目の共有フラグを追加してもよいし、より項目を細分化して共有フラグを設定してもよい。このように図19の情報は、文書データにオーバーレイまたは挿入される可変領域の、少なくとも位置およびサイズを定義する可変領域情報を有している。なお、製本アプリケーション104は、ユーザの指示に従い、編集指示に応じて追加または変更または削除を含む編集処理を行う。また、製本アプリケーション104は、可変領域情報にしたがって、可変領域が原稿ページごとにオーバーレイまたは挿入された出力データを作成する。さらに、可変領域情報には、可変領域を文書内の各原稿ページに共通する共通可変領域として設定する旨の共通属性情報が含まれる。

10

【 0 0 7 1 】

共有フラグがオフの場合には、各共有値に共有化されていたときの値が格納されている。これにより、一度共有化が解除されたものが再度共有化された場合に、共有化されていたときのフィールド属性値に復元することができる。図20に共有可変フィールドの各属性値を設定するためのユーザインタフェースを示す。

【 0 0 7 2 】

2001は、データベースのどのカラムのデータが可変フィールドに差し込まれるかを指定するユーザインタフェースである。2001で指定した情報が、可変フィールドの可変データ情報に設定される。

20

【 0 0 7 3 】

2002は、フォントの情報を指定するユーザインタフェースである。2002で指定したフォントの種類、フォントサイズ等の情報が、可変フィールドのフォント情報に設定される。なお、ここでは省略してあるが、このほかにも文字飾りやテキストの位置揃え等の設定を指定するユーザインタフェースを付けることもできる。

【 0 0 7 4 】

2003は、可変フィールドを原稿ページ上のどの位置に配置するかを可変フィールドの左上の座標で指定するユーザインタフェースである。座標は原稿ページの左上を基準に(0, 0)から始まり、ミリメートル単位で指定することができる。なお、座標の基準は原稿ページの左上でなくても構わないし、指定単位もミリメートルではなくインチなどの単位を用いてもよい。2003で指定した位置情報が、可変フィールドの配置情報に含まれる座標情報に設定される。

30

【 0 0 7 5 】

2004は、可変フィールドを配置する位置を原稿ページの相対位置で指定するか否かを指定するためのユーザインタフェースである。相対位置は、X方向に右揃え、左右中央揃え、左揃えで指定することができ、Y方向に上揃え、上下中央揃え、下揃えで指定することができる。相対位置で指定した場合の可変フィールドの配置位置を図21に示す。

【 0 0 7 6 】

2100は、原稿ページである。左揃え2101の場合、可変フィールドの左側が原稿ページの左側(座標で示すとXが0)の位置に配置される。左右中央揃え2102の場合、可変フィールドの中央($\{\text{可変フィールドの左側座標} + \text{可変フィールドの右側座標}\} \div 2$)が原稿ページのX方向の中央(原稿ページの幅 $\div 2$)になるように配置される。右揃え2103の場合、可変フィールドの右側座標が原稿ページの右側(原稿ページの幅)に接するように配置される。

40

【 0 0 7 7 】

上揃え2104の場合、可変フィールドの上側が原稿ページの上側(座標で示すとYが0)の位置に配置される。上下中央揃え2105の場合、可変フィールドの中央($\{\text{可変フィールドの上側座標} + \text{可変フィールドの下側座標}\} \div 2$)が原稿ページのY方向の中央(原稿ページの高さ $\div 2$)になるように配置される。下揃え2106の場合、可変フィールド

50

の下側座標が原稿ページの下側(原稿ページの高さ)に接するように配置される。

【 0 0 7 8 】

可変フィールドの位置を相対位置を用いて指定をしたとき、2 0 0 5 の指定を行うことにより相対位置で指定した配置に連動してテキストの位置揃えを変更する。例えば、左揃えを指定した場合にはテキスト揃えも左揃えになり、上下中央揃えを指定した場合にはテキスト揃えも上下中央揃えに設定される。

【 0 0 7 9 】

2 0 0 6 は、可変フィールドの幅と高さを指定するためのユーザインタフェースである。2 0 0 6 で指定した情報が、可変フィールドの配置情報に含まれるサイズ情報に設定される。

10

【 0 0 8 0 】

2 0 0 7 は、各設定を同一の共有 I D を持つ共有可変フィールド同士で共有するか否かを指定するためのユーザインタフェースである。ここで指定された情報は、共有 I D 情報の共有フラグに設定される。

【 0 0 8 1 】

2 0 0 8 のボタンを押すことにより、ここで指定した変更を適用する事ができる。2 0 0 9 のボタンを押すと、指定した情報は適用されない。

【 0 0 8 2 】

[可変フィールドの共有化]

図 2 2 は、可変フィールドを共有化したときの動作を示した図である。可変フィールドの共有化は、共有化したい可変フィールドをユーザが選択し、共有化の指示を行うことによって処理が行われる。共有化の指示の例としては、可変フィールドを選択した状態で、メニュー項目から共有化を選択する方法や、キーボードで任意キーを押下する方法などがある。2 2 - 1 の状態で、ユーザによって可変フィールド 2 2 0 1 が選択され、共有化が指示される。それにより、共有化の処理が行われ、全ページに対して同一のフィールド属性値を持つ共有可変フィールドが貼り付けられ 2 2 - 2 の状態となる。つまり、可変フィールド 2 2 0 1 と同様のフィールドが 2 2 0 2 のように全ページに対して配置される。製本アプリケーション 1 0 4 は、可変(フィールド)領域を共通領域とする旨の編集指示がされた場合、編集指示の対象である可変領域情報を、文書内の各原稿ページの可変領域情報として反映する。さらに、製本アプリケーション 1 0 4 は、各ページの可変領域情報に、共通する共通属性情報を設定して互いに関連づける。

20

30

【 0 0 8 3 】

図 2 3 は、可変フィールドを共有化する処理のフローチャートである。図 2 3 の処理により、あるページの可変フィールドについて共有化可変フィールドが定義されると、その共有可変フィールドを持つ文書ファイルの全ページについて共通の編集処理が適用される。この処理は、可変フィールドエディタによる処理であり、C P U 2 0 1 により実行される。ただし本実施形態では可変フィールドエディタは製本アプリケーション 1 0 4 の一部であるので、製本アプリケーション 1 0 4 の処理として説明する。本実施形態で説明するフローは、図 4 3 を除いて図 2 3 と同様に可変フィールドエディタによる処理であり、C P U 2 0 1 により実行される。なお、フローの説明ではあたかも製本アプリケーション 1 0 4 が実行主体であるかのように記載するが、プログラムにより手順が実行される場合には、C P U が主体となる。

40

【 0 0 8 4 】

共有化がユーザによって指示されると、製本アプリケーション 1 0 4 は、新規に共有 I D を発行する(S 2 3 0 1)。共有 I D は、存在する共有 I D 情報の共有 I D を参照していき、使用されていない任意の数字を割り当てる。なお、共有 I D は一意に識別できるのであれば数字でなくてもかまわない。新規に付加する共有 I D が決定したらその共有 I D を持つ共有 I D 情報を作成する。その際に、製本アプリケーション 1 0 4 は、共有フラグを全てオンにし、選択されている共有フィールドの属性値が各共有値に格納される。

【 0 0 8 5 】

50

製本アプリケーション 104 は、共有 ID を発行したら、その共有 ID を選択されている可変フィールドの共有 ID に格納する (S 2302)。

【0086】

次に、製本アプリケーション 104 は、変数 wP に共有化が指示されたフィールドを含む文書ファイルの先頭ページを格納し (S 2303)、先頭ページから共有可変フィールドを貼り付ける (S 2304)。貼り付ける共有可変フィールドの属性値は、共有 ID 情報から取得した共有値である。なお、先頭ページが選択した可変フィールドを含む場合、2 番目のページから共有可変フィールドの貼り付け処理 (S 2304) を行う。

【0087】

共有可変フィールドを貼り付けたら、製本アプリケーション 104 は、wP に次のページを格納することで次のページに移動し (S 2305)、そのページに対して共有可変フィールドを貼り付ける。なお、次のページが共有化を指示した可変フィールドを含む場合は、そのページを飛ばして次のページに対して処理を行う。

【0088】

製本アプリケーション 104 は、最終ページまで貼り付け処理を繰り返す (S 2306) ことで、全ページに対して共有可変フィールドの貼り付けが行われる。

【0089】

以上、図 23 の処理を実行することで、製本アプリケーション 104 は、可変フィールドの共有化が指示されたページを含む文書ファイルの全ページに対して、共有可変フィールドを設定することが可能となる。

【0090】

[可変フィールドの属性値の変更]

図 24 は、共有可変フィールドの属性値を変更したときの動作を示した図である。共有可変フィールドは全ページ同一の属性値を持つため、任意ページの共有可変フィールド 2401 の属性値を変更したときに (24-1)、全ページの共有可変フィールドに対して同様の変更が成される (24-2)。図 24 では、全ページに対して設定されている共有可変フィールドのうち、共有化変フィールド 2401 の位置を修正した。その結果、2401 の移動処理に連動して、残りのページの共有化変フィールド 2402 も同様の位置に修正される。

製本アプリケーション 104 は、ユーザによる位置の変更指示に従い、可変領域の原稿ページ上の位置を示す位置情報を変更する。

【0091】

図 25 は、可変フィールドの属性値を変更したときの処理を示したフローチャートである。図 25 の処理により、あるページについて定義された共有化可変フィールドが編集されると、その共有可変フィールドを持つ全ページについて共通の編集処理が適用される。

【0092】

まず、製本アプリケーション 104 は、ユーザからの属性値変更指示のあった可変フィールドの属性値を変更する (S 2501)。

【0093】

次に、製本アプリケーション 104 は、属性値を変更したフィールドの共有フラグを参照し (S 2502)、オフであった場合、共有されていないので変更処理を終了する。共有フラグがオンの場合、製本アプリケーション 104 は、変更のあった属性値の共有 ID 情報のフラグの状態を参照する (S 2503)。変更のあった属性値の共有フラグがオフであった場合、製本アプリケーション 104 は、変更処理を終了する。オンであった場合、製本アプリケーション 104 は、属性値の変更が行われた可変フィールドの共有 ID (cID) と変更された属性値を記憶する (S 2504、S 2505)。属性値は、図 17 および図 19 に示した可変フィールドを定義する種々の属性情報の値である。また、変更された属性値は、RAM 202 等に保持される。

【0094】

次に全ページの共有可変フィールドに対して変更を反映させるために、製本アプリケー

10

20

30

40

50

ション104は、先頭ページから(S2506)可変フィールドを参照する。そして、製本アプリケーション104は、ページ内の可変フィールドを参照していき、属性値を変更した共有可変フィールドの共有IDと同じ共有IDを持ち、共有フラグがオンの共有可変フィールドを探す(S2507)。製本アプリケーション104は、共有IDが同じ共有可変フィールドが見つかった場合(S2508)、その共有可変フィールドの属性値をS2504、S2505において記憶した属性値に変更する(S2509)。つまり、製本アプリケーション104は、S2507の処理によって検索された共有化変フィールドの属性値に対して、S2501の処理によって変更された属性値を設定する。ただし、可変フィールドが相対位置で指定されていた場合は原稿ページの幅、高さ、可変フィールドの幅、高さから計算し、割り出した位置に変更する。そして、製本アプリケーション104は、次のページに移動し(S2510)、同様に同一の共有IDを持つ共有可変フィールドを探し属性値を変更する。共有IDが同じ共有可変フィールドが見つからなかったときは、そのページの可変フィールドに対して変更は行わず、次のページに移動する。つまり、製本アプリケーション104は、可変領域の位置を、原稿ページに対する相対的な位置で指示することが可能であり、ユーザによる位置の変更指示に応じて、可変領域の位置を変更する。

【0095】

製本アプリケーション104は、S2507からS2510の処理を最終ページまで行い(S2511)、属性値の変更があった共有可変フィールドと同一の共有IDを持つ、全ての共有可変フィールドの属性値を変更する。

【0096】

図25の処理によって、製本アプリケーション104は、選択された可変フィールドの属性値が変更された場合、同じ共有IDを有する可変フィールドの属性値を変更可能となる。そのため、例えば同じ共有IDを有する1つの可変フィールドに対する属性値の変更が、他の共有可変フィールドにも同様に適用される。また、製本アプリケーション104は、ユーザによって共有可変フィールドの任意の1つのサイズが変更された場合、同じ共有IDを有する可変フィールドのサイズ情報を変更する。

【0097】

[共有化の解除]

図26は、共有可変フィールドの共有化を解除する処理の動作を示した図である。26-1は可変フィールドの共有化によって、共有されている状態である。共有可変フィールド2601は共有されており、同一の共有IDが付加されている。この状態から、選択された共有可変フィールド2602に対して共有化の解除を行った後の状態が26-2である。共有化の解除を指定した可変フィールドのみが通常の可変フィールド2603として残り、他のページの共有化されていた可変フィールドは削除される。つまり、製本アプリケーション104は、記可変領域を共通領域としない旨の編集指示がされた場合、編集対象の原稿ページについて可変領域を残し、それ以外の原稿ページについて、前記可変領域と関連付けられた可変領域を削除する。

【0098】

図27は、共有可変フィールドの共有化の解除における処理を示したフローチャートである。まず、製本アプリケーション104は、共有化の解除が実行されると解除を指定した共有可変フィールドの共有フラグをオフにし、共有フラグがオフとされた共有可変フィールドを有するページ情報を記憶する(S2701)。次に、製本アプリケーション104は、共有化を解除した共有可変フィールドのIDをcIDに格納する(S2702)。そして、製本アプリケーション104は、wPに先頭ページのページ番号を格納する(S2703)。製本アプリケーション104は、wPが共有化を解除した共有可変フィールドであるか判定し(S2704)、そうであればwPを次のページに変更する(S2708)。なお、S2704の判定処理は、判定対象ページとS2701にて記憶したページ情報を比較することにより実行可能である。製本アプリケーション104は、wPが共有化を解除したページでなかった場合、ページwP内の可変フィールドの中で共有IDがcIDである可変フィールドを探す(S2705)。製本アプリケーション104は、共有IDがcIDである可

変フィールドを検出した場合、その可変フィールドを削除する(S 2 7 0 6、S 2 7 0 7)。製本アプリケーション 1 0 4 は、S 2 7 0 4 から S 2 7 0 8 の処理を先頭ページから 1 ページずつ処理を行っていき、最終ページまで終了したら処理を終了する(S 2 7 0 9)。

【 0 0 9 9 】

図 2 7 の処理により、共有解除が指示された共有可変フィールドを残すことが可能となる。例えば、同じ共有 ID が付与された可変フィールドを有するページ数が膨大な量であり、本処理を用いることで、共有解除が指示されなかったフィールドを一括で削除することが可能となる。

【 0 1 0 0 】

[通常フィールドに変更]

図 2 8 は、共有可変フィールドを通常の変フィールドに変更する処理を示した図である。2 8 - 1 は可変フィールドの共有化によって、共有されている状態である。共有可変フィールド 2 8 0 1 は共有されており、同一の共有 ID が付加されている。この状態から、共有可変フィールド 2 8 0 2 に対して通常フィールドに変更を行った後の状態が 2 8 - 2 である。通常フィールドに変更を指定した可変フィールドのみが通常の変フィールド 2 8 0 3 に変更され、その他の共有可変フィールド 2 8 0 4 は共有された状態で残る。以後、可変フィールド 2 8 0 3 に対して行われる処理(属性値の変更など)は、共有されず可変フィールド 2 8 0 3 に対して反映される。共有可変フィールド 2 8 0 4 は、共有されているため 2 8 0 4 のうち任意一つの可変フィールドに対して行われた操作は、共有可変フィールド 2 8 0 4 全てに適用される。つまり、製本アプリケーション 1 0 4 は、可変領域を共通領域としない旨の編集指示がされた場合、編集対象の原稿ページの可変領域情報を、当該ページの可変領域が他の可変領域と関連付けられていない独立した可変領域とするよう変更する。

【 0 1 0 1 】

図 2 9 は、共有可変フィールドを通常フィールドに変更する処理を示したフローチャートである。製本アプリケーション 1 0 4 は、通常フィールドに変更を実行した共有可変フィールドの共有フラグをオフにする(S 2 9 0 1)。以上で、共有化が解除され通常の変フィールドに変更される。この処理により、製本アプリケーション 1 0 4 は、共有化されている複数の可変フィールドの一部を独立して属性値を変更することが可能となる。

【 0 1 0 2 】

[共有可変フィールドのコピー・アンド・ペースト]

図 3 0 は、共有可変フィールドをコピー(3 0 - 1)し、その後ペースト処理を行った(3 0 - 2、3 0 - 3)ときの状態を示した図である。貼り付けられた可変フィールドは、共有可変フィールドとして全ページに貼り付ける方法(3 0 - 2)と、通常の変フィールドとして貼り付ける方法(3 0 - 3)がある。可変フィールド 3 0 0 3 が、共有可変フィールドとして貼り付けられた可変フィールドであり、可変フィールド 3 0 0 4 が通常の変フィールドとして貼り付けられた可変フィールドである。それぞれの方法は、ユーザによってどちらの動作にするか選択することができる。製本アプリケーションは、ユーザの指示に応じて共有可変領域情報を複製する。また、複製方法は、共有可変領域情報として全ページに複製する方法と、共有されていない可変領域情報として指示されたページに複製する方法がある。

【 0 1 0 3 】

図 3 1 は、共有可変フィールドをコピーし、ペーストする時のペースト処理を示すフローチャートである。

【 0 1 0 4 】

まず、製本アプリケーション 1 0 4 は、通常のパースト処理と同様に指定されたページにコピーされている可変フィールドを貼り付ける(S 3 1 0 1)。次に、製本アプリケーション 1 0 4 は、ユーザによって選択されたペースト処理内容に従ってペースト動作を行う。ユーザが共有可変フィールドとして貼り付けることを選択していた場合、製本アプリケーション 1 0 4 は、可変フィールドの共有化の処理(図 2 3 参照)を行い、選択されたペ

10

20

30

40

50

ージに対してペースト処理を実行する(S 3 1 0 3)。通常の可変フィールドとして貼り付けることを選択していた場合、製本アプリケーション 1 0 4 は、貼り付けられた可変フィールドの共有フラグをオフにして、選択されたページに対してペースト処理を実行する(S 3 1 0 4)。

【 0 1 0 5 】

ペースト動作の設定は、ユーザがコピーした段階で設定してもよいし、ペースト時にユーザに問い合わせ、選択させてもよい。

【 0 1 0 6 】

[原稿ページの追加による共有可変フィールドの追加]

図 3 2 は、原稿ページが追加されたときの追加ページへの共有可変フィールドの追加を行う動作を示した図である。共有可変フィールド 3 2 0 1 が付加されている状態で、新しいページ 3 2 0 2 が追加された状態が 3 2 - 1 である。新しくページ 3 2 0 2 が追加されると、製本アプリケーション 1 0 4 は、共有可変フィールド 3 2 0 1 と同様の可変フィールドをページ 3 2 0 2 にも設定する。その結果、3 2 - 2 に示すように、全ページに対して共有可変フィールド 3 2 0 3 が配置されることとなる。なお、ユーザの選択により追加しないこともできる。

【 0 1 0 7 】

原稿ページ追加時の共有可変フィールド追加設定は、ユーザが原稿ページ追加前に設定してもよいし、原稿ページ追加時にユーザに問い合わせ、選択させてもよい。このように製本アプリケーション 1 0 4 は、文書データに、新たな原稿ページを追加挿入し、新たに原稿ページが追加された際に、追加前の文書データに共有可変領域情報が含まれている場合は、追加された原稿ページに共有可変フィールドを追加する。

図 3 3 は、原稿ページが追加されたときの追加ページへの共有可変フィールドの追加を行う処理のフローチャートである。

【 0 1 0 8 】

まず、製本アプリケーション 1 0 4 は、原稿ページが追加されると追加先となる文書ファイルに含まれるいずれかの原稿ページに、共有 ID 情報があるかどうかを調べ共有可変フィールドの有無を調べる(S 3 3 0 1)。製本アプリケーション 1 0 4 は、共有可変フィールドがあると判断した場合、共有 ID 情報からフィールドの共有値を取得し、その属性値を持つ可変フィールドを追加された原稿ページに付加する(S 3 3 0 2)。

【 0 1 0 9 】

図 3 3 の処理により、新たに原稿ページを追加した場合も、既に設定されている共有可変フィールドが同様に配置されるため、新規原稿ページを追加するたびに、可変フィールドを配置する手間が省け、ユーザの操作性を向上させる。

【 0 1 1 0 】

[ページ外に配置された共有可変フィールドの自動再配置]

可変フィールドの共有化、および可変フィールドの属性値の変更直後に、共有可変フィールドが原稿ページ外に配置されることがある。そのとき、原稿ページからはみ出た共有可変フィールドを原稿ページ内に収まるように再配置する。その処理の一例を図 3 4 に示す。なお原稿ページのサイズは、電子原稿ファイル内に保存されており、電子原稿ファイルを参照することで知得できる。

【 0 1 1 1 】

可変フィールド 3 4 0 1 に対して可変フィールドの共有化を行うと、全ページに対して同じ属性を持つ可変フィールド(3 4 0 2、3 4 0 3)が作成されるため、3 4 - 1 の状態となる。座標は原稿ページの左上を基準としているため、同じ位置に配置すると原稿ページのサイズが違うページではフィールドの一部、もしくは全体が原稿ページ外にはみ出す可能性がある(3 4 0 2、3 4 0 3 の可変フィールドの状態)。ここで製本アプリケーション 1 0 4 は、共有可変フィールドが原稿ページからはみ出していると判定した場合、自動的に原稿ページ内にフィールド全体が収まる位置に共有可変フィールドを移動する(その

結果が、3 4 - 2 の状態である)。移動してもはみ出るサイズのフィールドがあった場合(原稿ページの幅、もしくは高さより大きいサイズの幅、もしくは高さを持つフィールドが配置された場合)、製本アプリケーション 1 0 4 は、フィールドサイズを原稿ページの幅、高さまで縮める。X 方向(横)にはみ出た可変フィールドは X 方向にのみ移動し、Y 方向(縦)にはみ出た可変フィールドは Y 方向にのみ移動する。例えば、可変フィールド 3 4 0 2 は X 方向のみにはみ出したので X 方向のみの移動を行い、可変フィールド 3 4 0 3 は X 方向、および Y 方向にはみ出ているので、X 方向および Y 方向に移動する。製本アプリケーション 1 0 4 は、原稿ページの中央に可変フィールドの中央が配置されるように移動する。ただし、移動する位置は原稿ページの端に移動してもよいし、任意位置に移動してもよい。

10

【 0 1 1 2 】

つまり、製本アプリケーション 1 0 4 は、互いに関連付けられた可変領域情報の中に、原稿ページ外に配置される可変領域を定義した可変領域情報があった場合、可変領域が原稿ページ内に収まる位置に、可変領域の位置情報を変更する。また、可変領域の位置情報を変更しても可変領域が原稿ページ内に収まらない場合、製本アプリケーション 1 0 4 は、原稿ページ内に収まるサイズに、可変領域の幅および高さを示すサイズ情報を変更する。

【 0 1 1 3 】

移動した共有可変フィールドは、位置が変更されるため、通常の可変フィールドに変更する、位置情報の共有を解除する、共有は解除せず位置だけを移動するという動作の中からいずれかの処理を行う。どの動作を行うかは、ユーザによって選択することができる。

20

【 0 1 1 4 】

図 3 5 は、原稿ページ外に配置された共有可変フィールドを自動的に再配置する処理を示したフローチャートである。

【 0 1 1 5 】

まず、製本アプリケーション 1 0 4 は、先頭ページから調べるために wP に先頭ページのページ番号を格納する(S 3 5 0 1)。次に、製本アプリケーション 1 0 4 は、可変フィールドの X 座標と幅を加算した値が原稿ページの幅より大きいかを調べることにより(S 3 5 0 2)、可変フィールドが原稿ページからはみ出しているかどうかを知ることができる(大きいとはみ出ている)。例えば、可変フィールドの左端を基準として可変フィールドの水平方向の位置を管理する場合、可変フィールドの座標値に対して可変フィールドの幅の値を加算すると、可変フィールドの右端の位置を特定できる。その位置が、原稿ページの右端を越える場合、製本アプリケーション 1 0 4 は、S 3 5 0 2 において Y e s と判定する。製本アプリケーションは、可変フィールドが原稿ページの水平方向に対してはみ出すと判定した場合(S 3 5 0 2 - Y e s)、可変フィールドの幅が原稿ページの幅より大きいかを調べる(S 3 5 0 3)。S 3 5 0 3 の判定により、製本アプリケーション 1 0 4 は、可変フィールドが原稿ページの幅に収まるかどうかを判定できる。大きかった場合(S 3 5 0 3 - Y e s)、製本アプリケーション 1 0 4 は、可変フィールドの幅を原稿ページの幅に変更する(S 3 5 0 4)。つまり、製本アプリケーションは現在対象としている可変フィールドの属性値の 1 つである幅の値を、原稿ページ幅の値に変更する。

30

40

次に、製本アプリケーション 1 0 4 は、原稿ページの中央に可変フィールドの中央が配置されるように可変フィールドの X 座標を決定する(S 3 5 0 5)。

【 0 1 1 6 】

S 3 5 0 6 から S 3 5 0 9 までの処理は S 3 5 0 2 から S 3 5 0 5 までの処理を Y 方向に対して行った処理のため省略する。

【 0 1 1 7 】

次に、製本アプリケーション 1 0 4 は、可変フィールドの位置、サイズに変更があった場合(S 3 5 1 0)、その可変フィールドの共有情報を変更する(S 3 5 1 1)。ここでは、ユーザの指定により、通常の可変フィールドに変更する、位置情報の共有を解除する、共有は解除せず位置だけを移動する動作のいずれかを行う。通常の可変フィールドに変更の

50

場合、図 29 の処理が行われる。位置情報の共有を解除の場合、製本アプリケーション 104 は、可変フィールドの共有 ID の共有 ID 情報の位置フラグをオフにする(サイズ変更があった場合はサイズフラグもオフ)。共有を解除しない場合は何もしない。製本アプリケーションは、以上の処理を最終ページと判定(S3512、S3513)されるまで繰り返す。なお、本願では、X 座標は水平方向に関する座標である。よって、X 方向は、水平方向を指す。一方、Y 座標は垂直方向に関する座標であり、Y 方向は、垂直方向を指す。

【0118】

[可変フィールドを原稿ページの内容に合わせて自動配置]

図 36 は、共有可変フィールドを原稿ページの内容に合わせて自動配置する動作を示した図である。まず、原稿ページの内容を解析し、原稿ページ内で画像が配置されている領域とテキストが配置されている領域を割り出す。領域を割り出す方法は、原稿ページのデータの内部構造からテキストとして持っているオブジェクトを一定の規則でグループ化し、一まとまりの領域であると認定し、その領域の範囲を保存する。イメージについても同様に一定の規則でグループ化し認識する方法がある。なお、一定の規則とは、同一の X 座標を持つテキスト、一定の距離に配置されているテキスト等の規則等である。そのほかに、原稿ページの内容を一つの画像に変換し、OCR をかけてテキストの領域とイメージの領域を割り出す方法などがあるが、テキストの領域とイメージの領域を割り出せる方法であればどのような方法を用いてもよい。

【0119】

原稿ページ内にイメージ領域 3601 とテキスト領域 3602 と共有可変フィールド 3603 を持つ原稿ページの状態が 36-1 である。この状態から共有可変フィールド 3603 を選択し、画像に重ねるように共有可変フィールドを移動する処理を行うと 36-2 の状態となる。これは、製本アプリケーション 104 が、原稿ページ内の全イメージ領域の中で共有可変フィールドと最も重ねることができるイメージ領域の中央に共有可変フィールドの中央を配置した結果、3604 の位置に可変フィールドが移動した状態である。

【0120】

テキストに重ねるように共有可変フィールドを移動する処理を行った場合は、36-3 の状態になる。これは、原稿ページ内の全テキスト領域の中で共有可変フィールドと最も重ねることができるテキスト領域の中央に共有可変フィールドの中央を配置した結果、3605 の位置に可変フィールドが移動した状態である。つまり、製本アプリケーション 104 は、原稿ページのデータを解析し、オブジェクトの種類ごとに領域を分類する。そして、製本アプリケーション 104 は、原稿ページ全体あるいは指示に応じたオブジェクトの種類の領域に、共有可変領域が配置されるように、可変領域の位置情報を設定する。なお、オブジェクトの種類とは、テキストおよびイメージを含む。

【0121】

図 37 は、共有可変フィールドを原稿ページの内容に合わせて自動配置するためのユーザインタフェースを示した図である。ユーザが共有可変フィールドを選択した後、3701 を選択することにより、製本アプリケーション 104 は、共有可変フィールドを画像と重ねるように移動させる。また、3702 を選択することにより、製本アプリケーション 104 は、共有可変フィールドをテキストと重ねるように移動させる。

【0122】

図 38 は、共有可変フィールドを画像と重ねるように移動させる処理のフローチャートである。まず、製本アプリケーション 104 は、wP に先頭ページを格納し、先頭ページのオブジェクトを調べる(S3801)。次に、製本アプリケーション 104 は、原稿ページ wP を解析しイメージ領域を割り出す(S3802)。その結果、製本アプリケーション 104 は、原稿ページ wP にイメージ領域が存在しなかった場合、原稿ページ wP が最終ページでないかを確認し(S3821)、最終ページでなかった場合は次のページを wP とする(S3822)。原稿ページ wP 内にイメージ領域が存在した場合、製本アプリケーション 104 は、一つのイメージ領域を対象イメージ領域として以降の処理を行う。次に、製本アプリ

ーション 104 は、MINに所定の値を入れる (S 3804)。所定の値とはその後の処理で最小値となりえない数である。MINには、イメージ領域に選択された共有可変フィールドを重ねたときにはみ出る面積が格納されるため、原稿ページサイズの面積以上の値にしておけば最小値になることはない。

【0123】

次に、製本アプリケーション 104 は、共有可変フィールドの幅から対象イメージ領域の幅を減じたもの(以下Wとする)が0以下になったらWに0を格納する (S 3806)。同様に、製本アプリケーション 104 は、共有可変フィールドの高さから対象イメージ領域の高さを減じたもの(以下Hとする)が0以下になったらHに0を格納する (S 3808)。

【0124】

次に、製本アプリケーション 104 は、Wが0でありHも0であった場合 (S 3809 - Yes、S 3810 - Yes) は対象イメージ領域に共有可変フィールドを包含するように配置することが可能である。そこで、製本アプリケーション 104 は、対象イメージ領域の中央に共有可変フィールドの中央が配置されるように位置を変更し (S 3820)、wP が最終ページでなかったら次のページを処理する (S 3822)。詳細には、製本アプリケーション 104 は、対象イメージ領域の座標とサイズを用いて対象イメージ領域の中心を算出する。製本アプリケーションは、算出された中心点と共有可変フィールドの中心点が一致するように、共有可変フィールドの位置を決定する。

【0125】

一方、Wが0でHが0でなかった場合 (S 3809 - Yes、S 3810 - No)、製本アプリケーション 104 は、Wに共有可変フィールドの幅を格納する (S 3811)。また、Wが0でなくHが0であった場合 (S 3809 - No、S 3810 - Yes)、製本アプリケーション 104 は、Hに共有可変フィールドの高さを格納する (S 3813)。そして、製本アプリケーション 104 は、共有可変フィールドが対象イメージ領域からはみ出る面積を算出する (S 3814)。具体的には、製本アプリケーション 104 は、はみ出る面積AにH×Wを格納する。W,H共に0でなかった場合 (S 3809 - No、S 3810 - No)、製本アプリケーション 104 は、Aに対象イメージ領域の面積(幅×高さ)から共有可変フィールドの面積(幅×高さ)を減じた値を格納する。

【0126】

そして、製本アプリケーション 104 は、AとMINを比較しAのほうが小さかった場合、MINの値をAとする (S 3817)。S 3805 から S 3817 までの処理を全イメージ領域に対して行い、最も共有可変フィールドと重ねることができるイメージ領域の中央に共有可変フィールドの中央が配置されるように位置を変更する (S 3820)。つまり、イメージ領域に対する共有可変フィールドのはみ出し量が最も少なくなるイメージ領域の中央に、共有可変フィールドを配置する。その処理を全ページに対して繰り返す。

【0127】

テキストに重ねるように共有可変フィールドを移動する場合も、同様の処理のため省略する。

【0128】

[原稿ページに合わせて共有可変フィールドを配置]

図 39 は、原稿ページサイズに合わせて共有可変フィールドを配置した状態を示す図である。

【0129】

共有可変フィールド 3901 の可変データに応じてフィールドサイズを変更する。フィールドサイズは、可変データを原稿ページの幅、もしくは高さに達するサイズまで大きくして配置する(テキストの場合、フォントサイズ、イメージの場合、画像の拡大)。39-2 は、原稿ページに合わせて共有可変フィールドを配置した状態である。3902 は1レコード目であり、3903 は2レコード目の状態を示す。共有可変フィールドは、レコードごとに原稿ページの幅、もしくは高さに収まる最大のフォントサイズを選択し、表示する。イメージの場合も同様に、原稿ページの幅、もしくは高さに収まる最大サイズまで拡

10

20

30

40

50

大される。

【 0 1 3 0 】

図 4 0 は、原稿ページサイズに合わせて共有可変フィールド(テキスト可変フィールド)を配置する処理を示したフローチャートである。

【 0 1 3 1 】

まず、フォントサイズとなる値Nに1を格納する(S 4 0 0 1)。そして、現在表示されているレコードの可変データがフォントサイズNのときの幅をTWに格納し、高さをTHに格納する(S 4 0 0 2)。TW、またはTHが原稿ページの幅、高さに達していないときはフォントサイズを1増やす(S 4 0 0 4)。TW、またはTHが原稿ページの幅、高さより大きくなった場合は、フォントサイズを1減らし(S 4 0 0 5)、そのときのTW、THを共有可変フィールドの幅、高さに設定する(S 4 0 0 6)。そして、原稿ページの幅に合わせた場合はX座標、高さに合わせた場合はY座標の値を0にする(S 4 0 0 7)。イメージ可変フィールドの場合は、原稿ページのサイズまで画像を拡大もしくは縮小することで原稿ページに合わせる。

10

【 0 1 3 2 】

[特定の原稿内容のみに共有可変フィールドを重ねる]

図 4 1 は、特定の原稿内容(画像)のみに共有可変フィールド(テキスト可変フィールド)を重ねた状態を示した図である。

【 0 1 3 3 】

全ページのイメージ領域に対して、領域内に収まるサイズで共有可変フィールドが配置される。そのため、イメージ領域の存在しない原稿ページ 4 1 0 3 には、共有可変フィールドは配置されない。

20

【 0 1 3 4 】

図 4 2 は、特定の原稿内容(画像)のみに共有可変フィールド(テキスト可変フィールド)を重ねる処理を示したフローチャートである。

まず、wPに先頭ページを格納し、先頭ページのオブジェクトを調べる(S 4 2 0 1)。次に、原稿ページwPを解析しイメージ領域を割り出す(S 4 2 0 2)。その結果、原稿ページwPにイメージ領域が存在しなかった場合は、原稿ページwPが最終ページでないかを確認し(S 4 2 1 3)、最終ページでなかった場合は次のページをwPとする(S 4 2 1 4)。原稿ページwP内にイメージ領域が存在した場合は、一つのイメージ領域を対象イメージ領域として以降の処理を行う。

30

【 0 1 3 5 】

次に、フォントサイズとなる値Nに1を格納する(S 4 2 0 4)。そして、現在表示されているレコードの可変データがフォントサイズNのときの幅をTWに格納し、高さをTHに格納する(S 4 2 0 5)。TW、またはTHがイメージ領域の幅、高さに達していないときはフォントサイズを1増やす(S 4 2 0 7)。TW、またはTHがイメージ領域の幅、高さより大きくなった場合は、フォントサイズを1減らし(S 4 2 0 8)、そのときのTW、THを共有可変フィールドの幅、高さに設定する(S 4 2 0 9)。そして、原稿ページの幅に合わせた場合はX座標、高さに合わせた場合はY座標の値を0にする(S 4 2 1 0)。S 4 2 0 4 から S 4 2 1 0 までの処理を原稿ページ中の全イメージ領域に対して行う。その処理を全原稿ページに対して行う。

40

【 0 1 3 6 】

テキスト領域に重ねる場合も同様の処理のため省略する。イメージ可変フィールドの場合は、原稿ページのサイズまで画像を拡大もしくは縮小することでイメージ領域およびテキスト領域に合わせる。

【 0 1 3 7 】

図 4 3 は、電子原稿デスプーラ 1 0 5 により、電子原稿ファイル 1 0 3 から原稿ページ単位でデータを読み出し、プリンタドライバ 1 0 6 に読み出したデータを送信する手順のフローチャートである。まず文書データ(電子原稿ファイル 1 0 3)が読み出されて(S 4 3 0 1)、その属性情報(ブック属性、章属性)に基づいて、印刷設定情報が作成され

50

る（S 4 3 0 2）。この印刷設定情報は電子原稿ファイル 1 0 3 に保存された設定であり、たとえばジョブチケットと呼ばれる規格化された形式で作成されている。次に原稿ページのデータを配置順に読み出す（S 4 3 0 3）。そしてその原稿ページを印刷設定情報に応じて編集する（S 4 3 0 4）。それから編集した原稿ページを出力データとしてプリンタドライバに送信する（S 4 3 0 5）。ステップ S 4 3 0 4 では、たとえば可変フィールドがページ属性として定義されている場合、その可変フィールドの属性にしたがってデータベースからレコードをフェッチし、フェッチしたレコードを可変フィールドに相当する位置およびサイズで、ページ上にオーバーレイする。このオーバーレイは、画像データ自体を合成しても良いが、オーバーレイする画像と、それがオーバーレイされるページおよび位置と関連づけて 1 ページ分のデータとしてプリンタドライバに送信しても良い。後者の場合、合成はプリンタドライバあるいはプリンタで実行される。また、データベースからの可変レコードのフェッチは、ページ単位ではなくたとえば部単位で行っても良い。その場合には、共有可変フィールドに挿入されるレコードは、1 部ごとに共通したレコードである。たとえばレコードに文書の配布先となる人物の氏名等を定義しておけば、1 部分の文書の前ページに各受取人の氏名を印刷することができる。

10

【 0 1 3 8 】

最後に、出力対象の文書データに含まれるすべての原稿ページの出力が終了したか判定し（S 4 3 0 6）、終了でなければ次の原稿ページについてステップ S 4 3 0 3 から繰り返す。

【 0 1 3 9 】

20

プリンタドライバ 1 0 6 は、受信した出力データをプリンタが解釈実行可能な形式に変換してプリンタに送信する。プリンタでは文書データを印刷出力する。

【 0 1 4 0 】

以上のようにして、本実施形態の文書処理システムによれば、可変フィールド（可変領域）を原稿ページに挿入またはオーバーレイされる領域として、ページごとに設定保存できる。そして、その属性は、文書データを通じて共通にすることができる。可変フィールドを共通可変フィールドと指定することで、ひとつのページについて定義された可変フィールドの属性を変更すれば、同一の共通可変フィールドが挿入あるいはオーバーレイされるすべてのページについて、可変フィールドの属性の変更が適用される。

【 0 1 4 1 】

30

[他の実施形態]

上記実施形態では、共有可変フィールドは文書を単位として定義されているが、さらに小さな単位、すなわち章を単位として定義することもできる。その場合、共有可変フィールドの追加、変更、削除は、章を単位として実行される。

【 0 1 4 2 】

また、可変フィールドのオブジェクトがページ外へはみ出す場合には、ページ内に収まる（サイズが大きすぎる場合には、できるだけ）ように可変フィールドの位置が移動されている。しかし、この移動の対象は可変フィールドに限らず、その他のフィールドたとえばウォーターマークなどにも適用可能である。

【 0 1 4 3 】

40

また、可変フィールドは原稿ページにオーバーレイされているが、ページに挿入されるものであってもよい。

【 0 1 4 4 】

また、本実施形態では、可変フィールドへのデータの挿入はデスクトップ 1 0 5 で行うものとして説明したが、製本アプリケーション 1 0 4 で行ってもよい。また、プリンタドライバで行うこともできる。

【 0 1 4 5 】

また、本実施形態では、可変フィールドを指定された種類のオブジェクト領域にオーバーレイする指示は、可変フィールドエディタのユーザインタフェースからされ、可変フィールドの属性情報としては保存されない。しかし属性情報として保存しても良い。この場合

50

、原稿ページの読み直し等による変更があっても、原稿ページの変更に応じて可変フィールドを再配置することができる。

【 0 1 4 6 】

なお本発明は、複数の機器（例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど）から構成されるシステムに適用しても、一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）に適用してもよい。また本発明の目的は、前述の実施形態の機能を実現するプログラムコードを記録した記録媒体を、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自体および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。

10

【 0 1 4 7 】

また、本発明には、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オペレーティングシステム(OS)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。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カ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込まれた場合についても、本発明は適用される。その場合、書き込まれたプログラムコードの指示に基づき、その機能拡張カ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。

20

【図面の簡単な説明】

【 0 1 4 8 】

【図 1】本実施形態のスタンドアロン型文書処理システムのソフトウェア構成例を示すブロック図である。

【図 2】本実施形態の文書処理システムを実現するハードウェア構成例を示すブロック図である。

【図 3】ブックファイルの構造の一例を示す図である。

【図 4】ブック属性の一覧を示す図である。

【図 5】章属性の一覧を示す図である。

30

【図 6】ページ属性の一覧を示す図である。

【図 7】ブックファイルを開く手順例を示すフローチャートである。

【図 8】電子原稿ファイルをブックファイルにインポートする手順例を示すフローチャートである。

【図 9】図 8 のステップ 8 0 1 に示すアプリケーションデータを電子原稿ファイルに変換する手順例を示すフローチャートである。

【図 1 0】製本アプリケーションのUI画面の図である。

【図 1 1】新規にブックファイルが作成された際のUI画面の図である。

【図 1 2】クライアントサーバ型の文書処理システムのソフトウェア構成例を示すブロック図である。

40

【図 1 3】ステイブル制御システムの構成を示すブロック図である。

【図 1 4】可変印刷の印刷例を示す図である。

【図 1 5】可変印刷エディタのユーザインタフェース例とデータベース設定を行うダイアログのユーザインタフェース例の図である。

【図 1 6】可変フィールドの設定例を示す図である。

【図 1 7】可変フィールドのデータ構造を示す図である。

【図 1 8】複数の共有可変フィールドが存在する場合の共有IDの状態を示す図である。

【図 1 9】共有ID情報のデータ構造を示す図である。

【図 2 0】共有可変フィールドの各属性値を設定するためのユーザインタフェース例の図である。

50

【図 2 1】可変フィールドの配置を相対位置で指定した場合の位置を示す図である。
【図 2 2】可変フィールドを共有化したときの動作を示した図である。
【図 2 3】可変フィールドを共有化する処理を示すフローチャートである。
【図 2 4】共有可変フィールドの属性値を変更したときの動作を示した図である。
【図 2 5】可変フィールドの属性値を変更したときの処理を示したフローチャートである。

【図 2 6】共有可変フィールドの共有化を解除する処理の動作を示した図である。
【図 2 7】共有可変フィールドの共有化の解除における処理を示したフローチャートである。

【図 2 8】共有可変フィールドを通常の変フィールドに変更する処理の動作を示した図 10
である。

【図 2 9】共有可変フィールドを通常フィールドに変更する処理を示したフローチャート
である。

【図 3 0】共有可変フィールドをコピー(3 0 - 1)し、その後ペースト処理を行った(3
0 - 2、3 0 - 3)ときの状態を示した図である。

【図 3 1】共有可変フィールドをコピーし、ペーストする時のペースト処理を示すフロー
チャートである。

【図 3 2】原稿ページが追加されたときの追加ページへの共有可変フィールドの追加を行
う動作を示した図である。

【図 3 3】原稿ページが追加されたときの追加ページへの共有可変フィールドの追加を行
う処理のフローチャートである。 20

【図 3 4】原稿ページ外に配置された共有可変フィールドを原稿ページ内に収まるように
再配置する動作を示した図である。

【図 3 5】原稿ページ外に配置された共有可変フィールドを自動的に再配置する処理を示
したフローチャートである。

【図 3 6】共有可変フィールドを原稿ページの内容に合わせて自動配置する動作を示した
図である。

【図 3 7】共有可変フィールドを原稿ページの内容に合わせて自動配置するためのユーザ
インタフェースを示した図である。

【図 3 8】共有可変フィールドを画像と重ねるように移動させる処理のフローチャートで
ある。 30

【図 3 9】原稿ページサイズに合わせて共有可変フィールドを配置した状態を示す図であ
る。

【図 4 0】原稿ページサイズに合わせて共有可変フィールド(テキスト可変フィールド)を
配置する処理を示したフローチャートである。

【図 4 1】特定の原稿内容(画像)のみに共有可変フィールド(テキスト可変フィールド)を
重ねた状態を示した図である。

【図 4 2】特定の原稿内容(画像)のみに共有可変フィールド(テキスト可変フィールド)を
重ねる処理を示したフローチャートである。

【図 4 3】文書データを出力する処理の手順を示すフローチャートである。 40

【符号の説明】

【0 1 4 9】

1 0 0 コンピュータ

1 1 0 3 論理ページ

1 4 0 0 原稿ページ

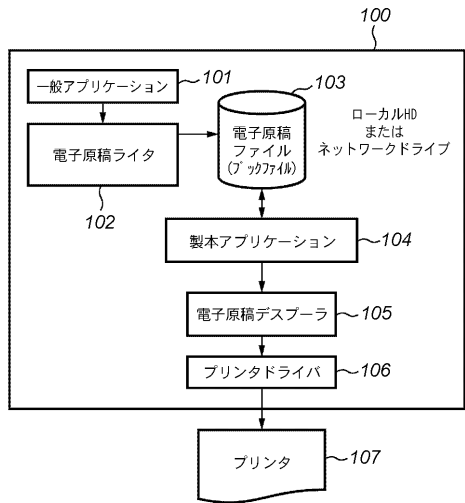
1 4 0 1 可変フィールド

2 2 0 2 共有可変フィールド

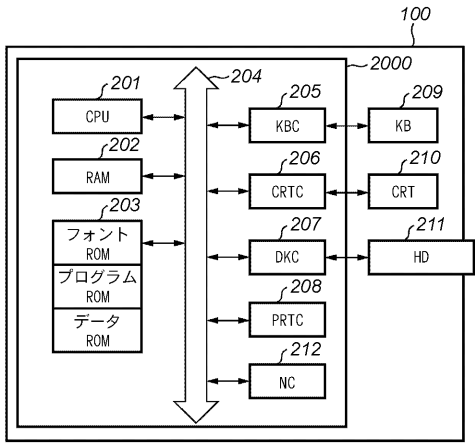
3 6 0 1 イメージ領域

3 6 0 2 テキスト領域

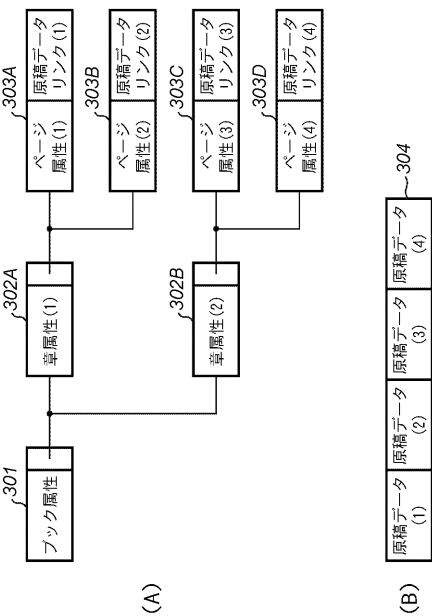
【図 1】



【図 2】



【図 3】



【図 4】

NO	属性情報	備考
1	印刷方法	
2	用紙サイズ	・ [M4493]「レタ-サイズ」(11×17)指定の場合は7折指定 ・ 製本印刷またはup印刷が指定された場合は1巻/ページ目の原稿サイズが自動で選択される
3	用紙方向	・ 固定サイズの場合のみ選択可能
4	どじ代/じ方向	・ シフト/振動指定可能
5	N-up印刷	・ 配置位置は9パターン ・ 等倍印刷指定可能
6	拡大縮小	用紙サイズに固定サイズのまたはN-up印刷を選択した場合は自動でONに指定、OFFに指定可能
7	ウォーターマーク	・ 論理ページ単位、物理ページ単位に個別の指定が可能 ・ 全巻/全ページが対象
8	ヘッダ・フッタ	・ 論理ページ単位、物理ページ単位に個別の指定が可能 ・ 全巻/全ページが対象
9	排紙方法	・ スライフル/パンチ穴は片面/両面印刷のみ ・ スライフルは箇所/箇所2箇所
10	製本詳細	・ 表紙/裏表紙は表紙/裏表紙1/2に対する印刷指定 ・ インデックス部分への文字列印字、インデックス紙へのアノテーションを設定可能 ・ 製本印刷は指定不可
11	表紙/裏表紙	・ 給紙口(インサートを含む)指定 ・ 挿入用紙に原稿データを印刷可能
12	インデックス紙	・ インデックス紙、合紙が指定された場合は「用紙がえ」固定
13	合紙	・ 片面印刷は「用紙がえ」
14	章区切り	・ 「なし」/「ページがえ」/「用紙がえ」

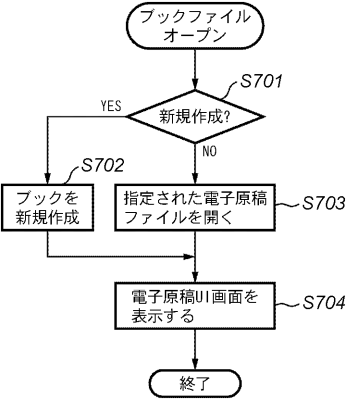
【図 5】

No	属性情報	備考	
		原稿サイズ	固定サイズ
1	用紙サイズ		固定サイズ/固定サイズ
2	用紙方向		縦/横
3	Yup印刷指定		ページ数/配置順/境界線/配置位置等
4	拡大縮小		ON/OFF
5	ウォーターマーク		表示/非表示
6	ヘッダー・フッター		表示/非表示
7	排紙方法		ステイプル

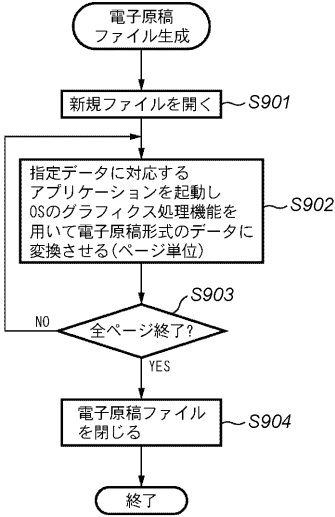
【図 6】

No	属性情報	備考	
		原稿サイズ	固定サイズ
1	ページ回数指定		0/90/180/270度の指定が可能
2	ウォーターマーク		ブロックで指定された全ウォーターマークを表示するか しないかの指定
3	ヘッダー・フッター		ブロックで指定された全ヘッダー・フッターを表示するか しないかの指定
4	ズーム		50%-200% ・ 仮想論理ページ領域にフィットしたサイズを 100%とした相対比率指定
5	配置位置		・ 固定のパターンおよび位置指定
6	アニメーション		
7	Variable項目		
8	ページ分割		

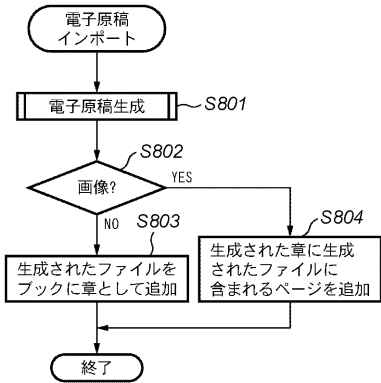
【図 7】



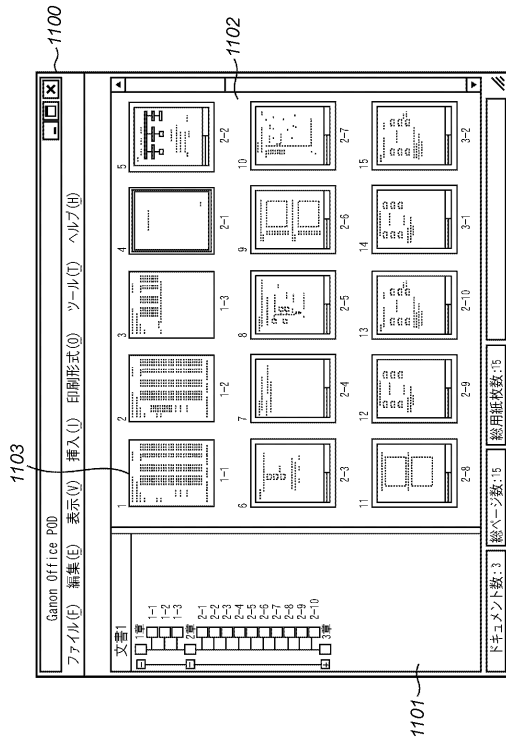
【図 9】



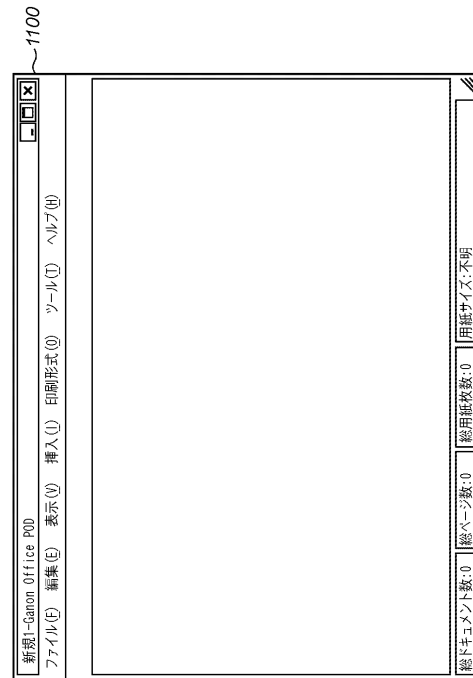
【図 8】



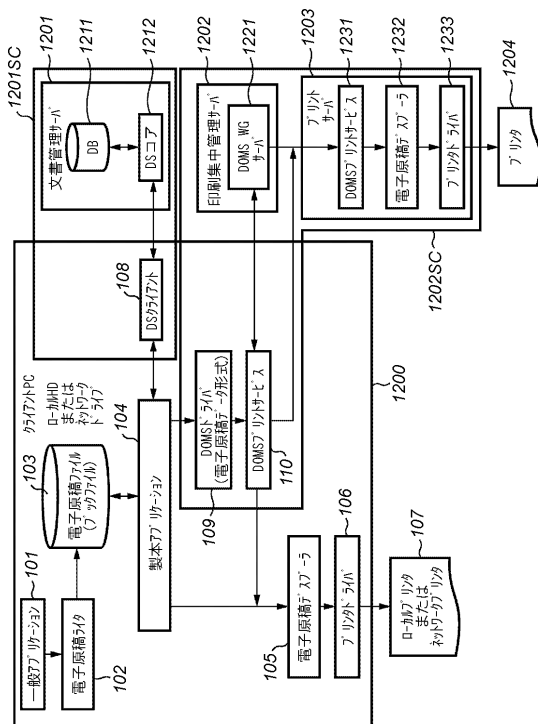
【 図 1 0 】



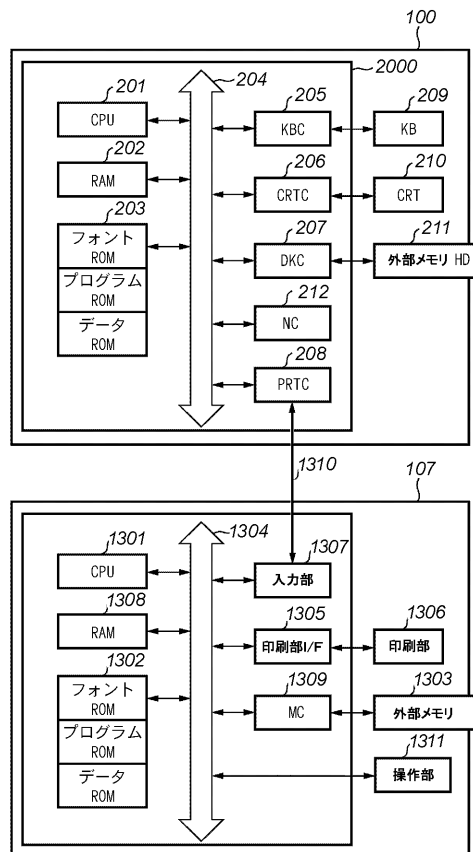
【 図 1 1 】



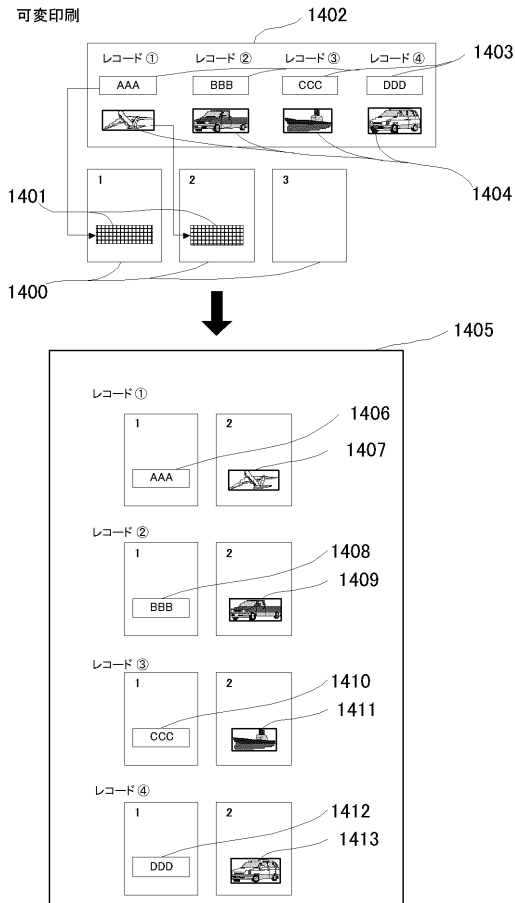
【 図 1 2 】



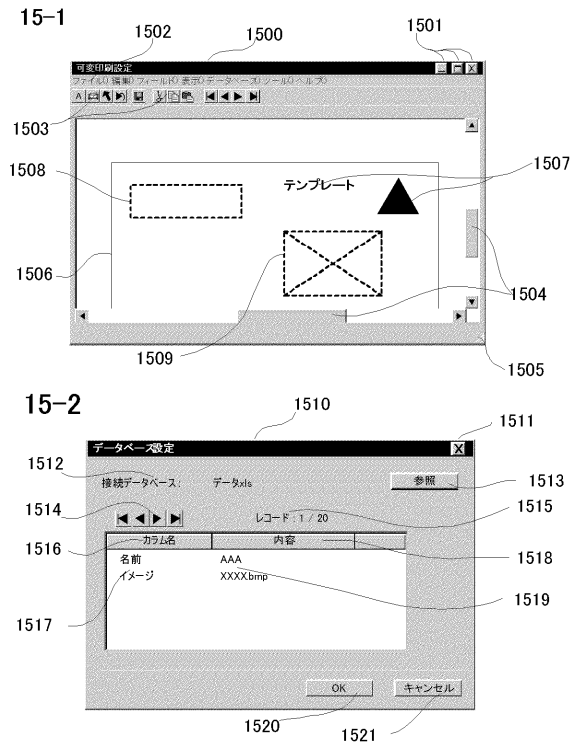
【 図 1 3 】



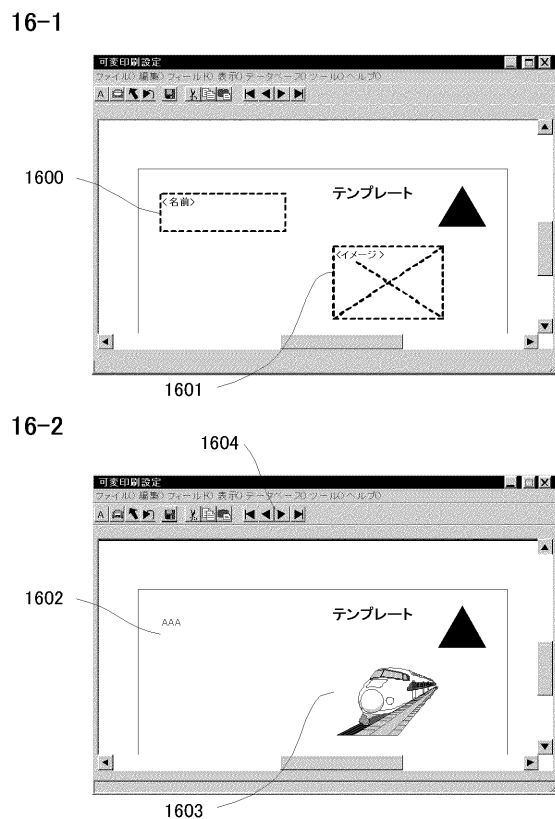
【図 14】



【図 15】



【図 16】

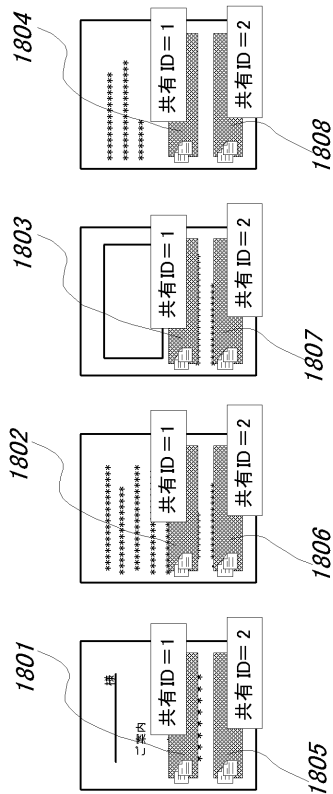


【図 17】

可変フィールド

- ・フィールドID
- ・共有フラグ
- ・共有ID
- ・可変データ情報
- ・配置情報
- ・フォント情報(イメージ情報)
- ・枠修飾情報

【図 18】



【図 19】

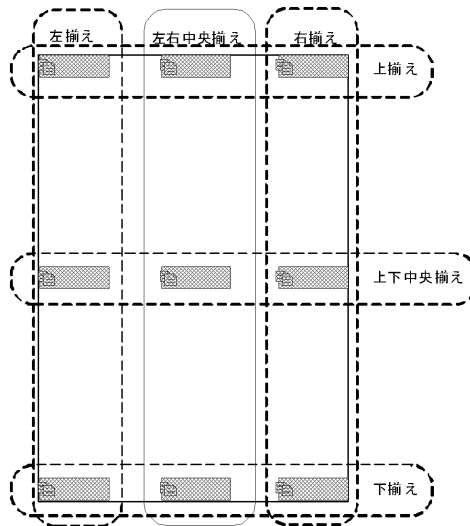
共有ID情報

- ・共有ID
- ・共有フラグ
- ・位置
- ・サイズ
- ・フォント
- ・枠修飾
- ・共有値
- ・位置
- ・サイズ
- ・フォント
- ・枠修飾

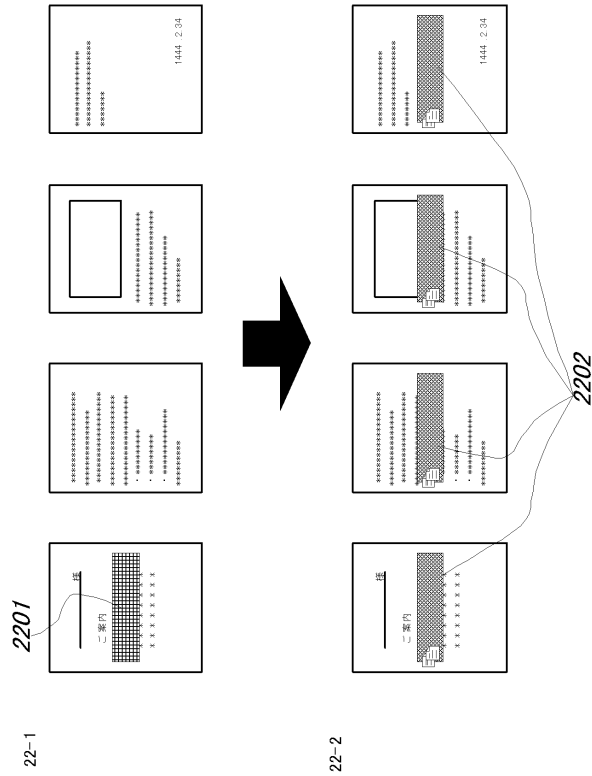
【図 20】

Figure 20 is a screenshot of a software interface for setting shared ID information. The interface includes fields for '可変データ' (Variable Data), 'フォント' (Font), '位置' (Position), and 'サイズ' (Size). It also has checkboxes for '共有' (Shared) and '共有フラグ' (Shared Flag).

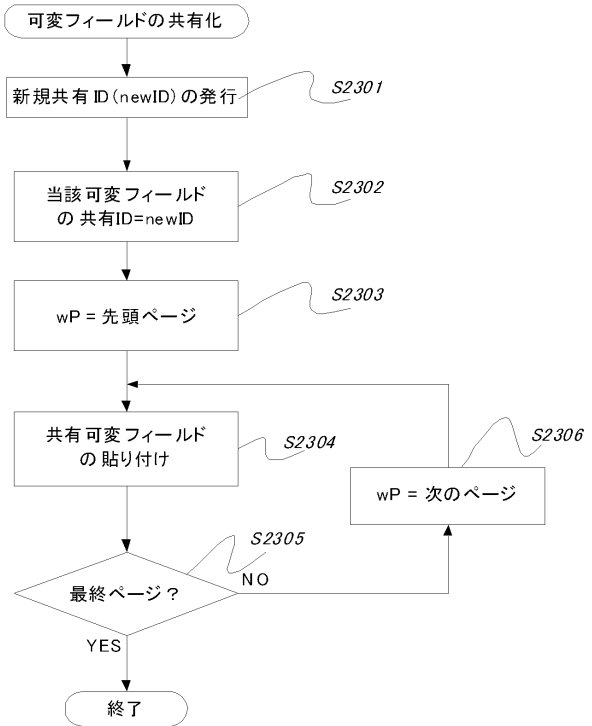
【図 21】



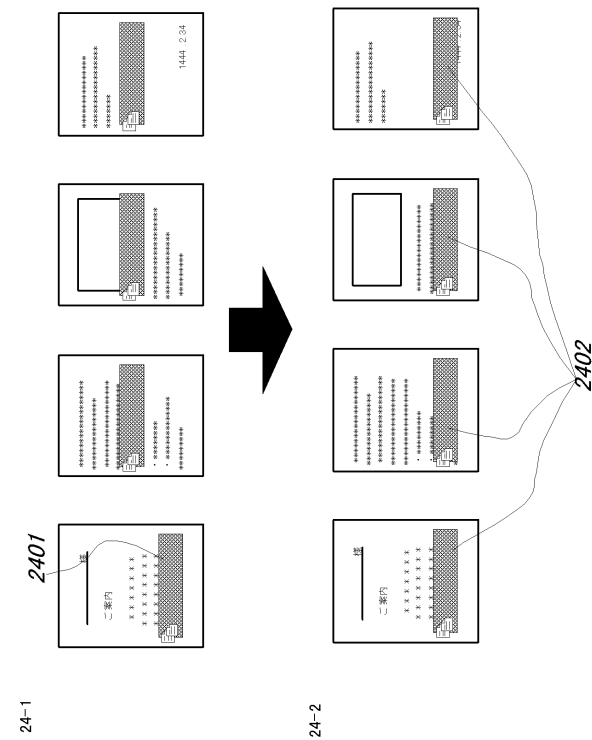
【図 22】



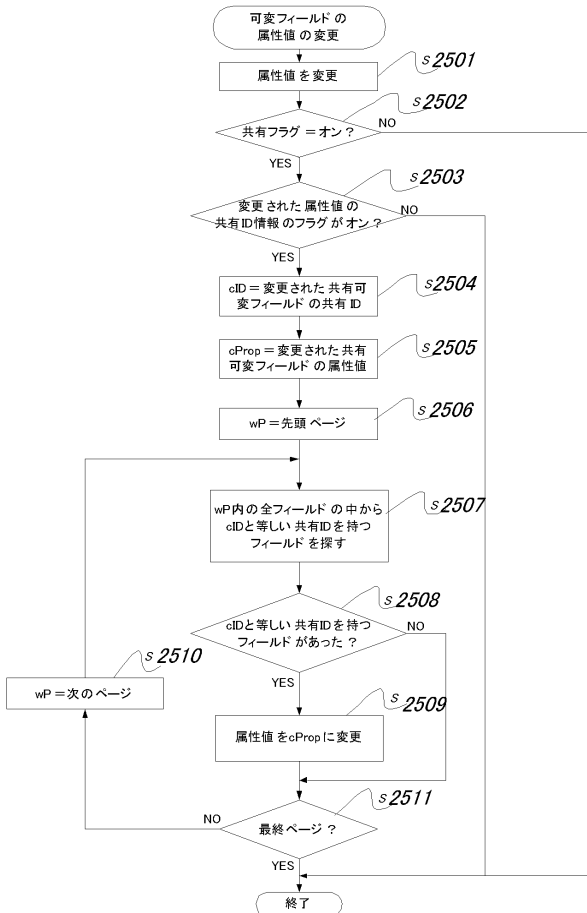
【図 23】



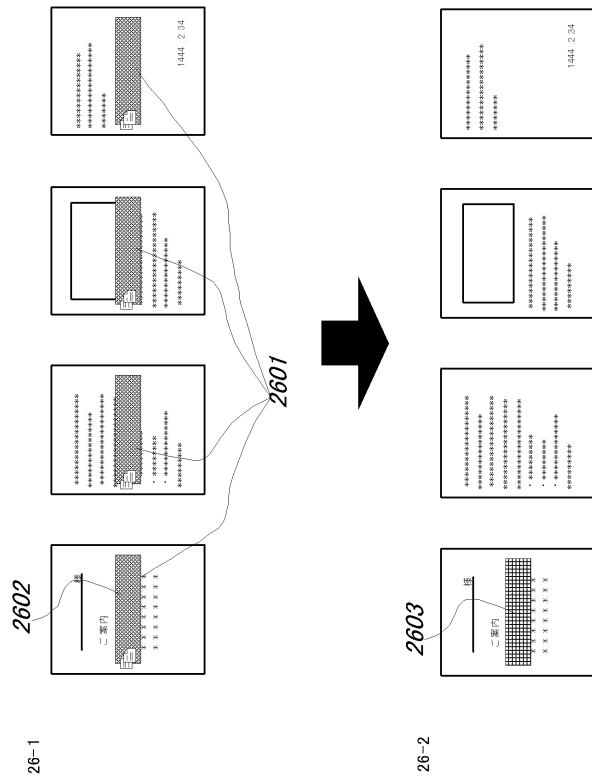
【図 24】



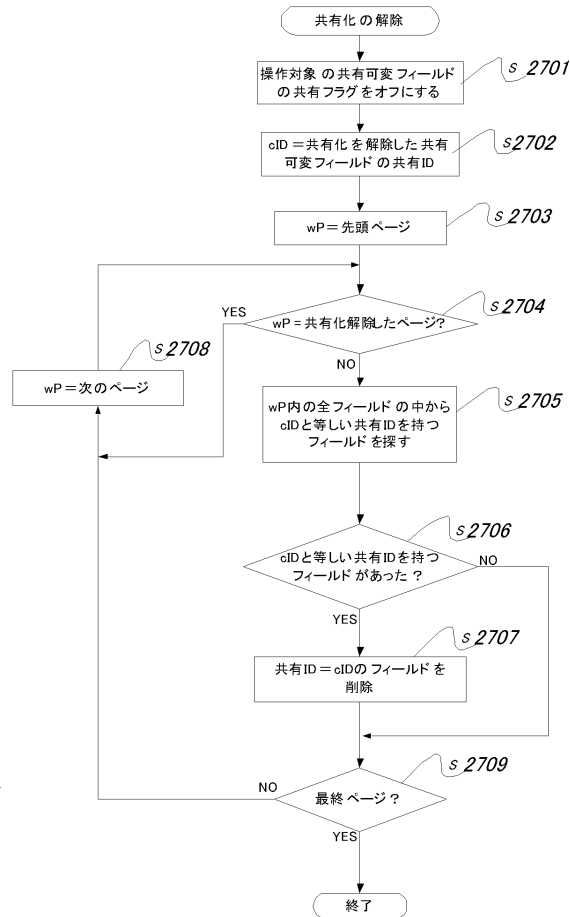
【図 25】



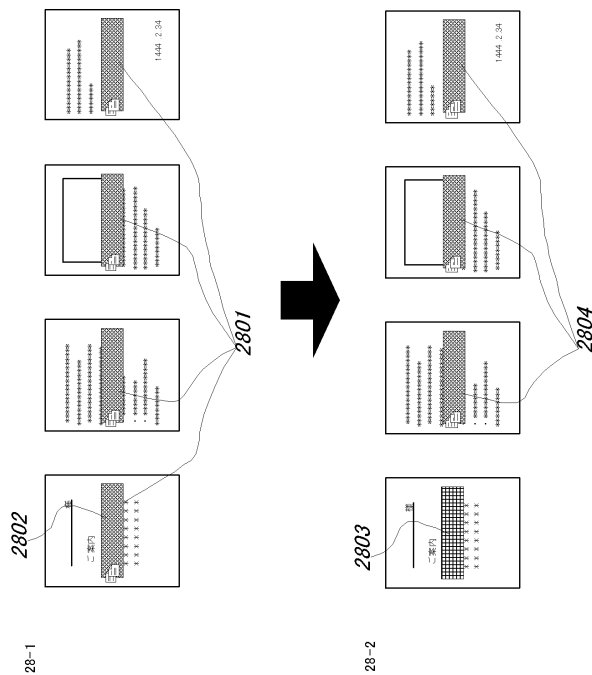
【図 26】



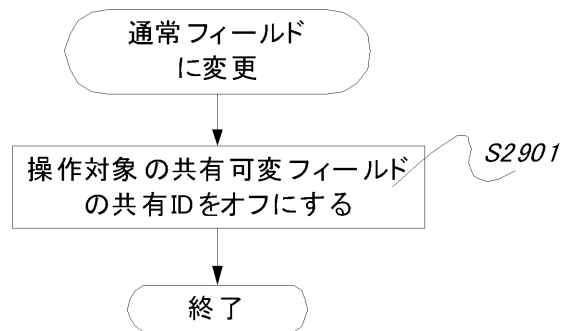
【図 27】



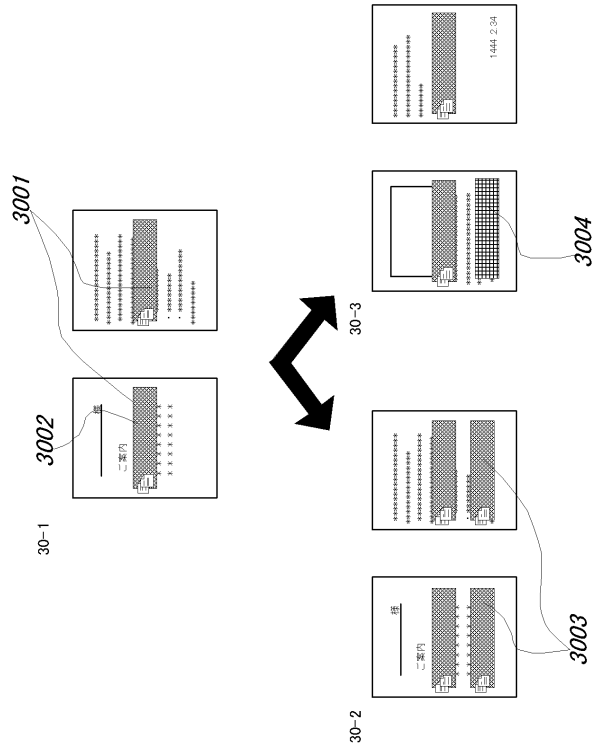
【図 28】



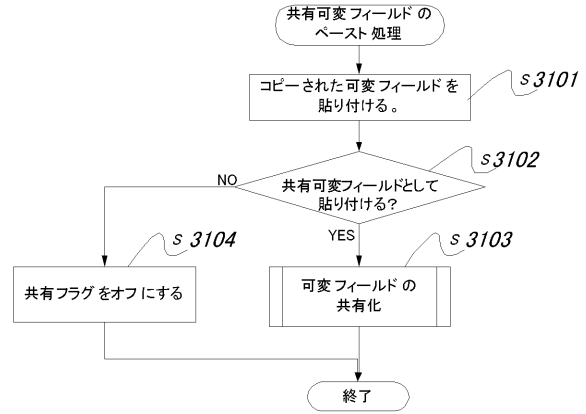
【図 29】



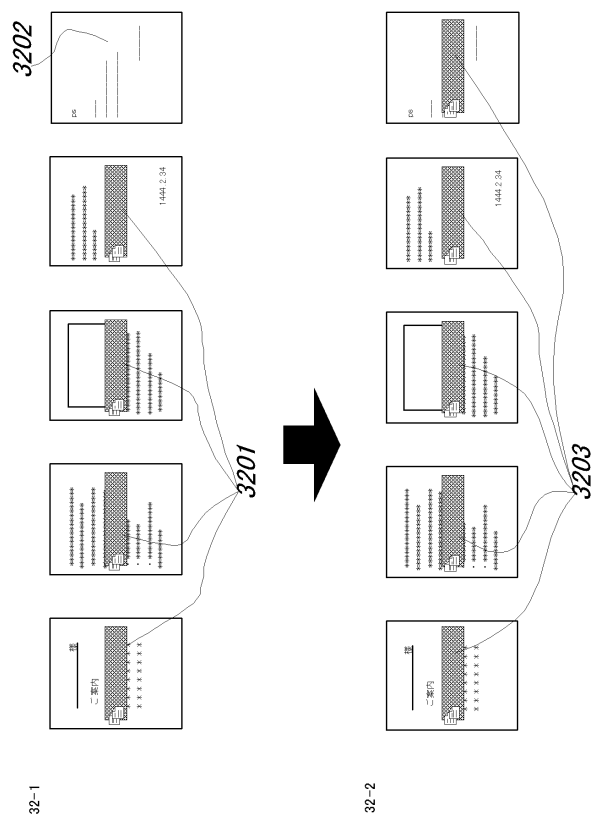
【図 30】



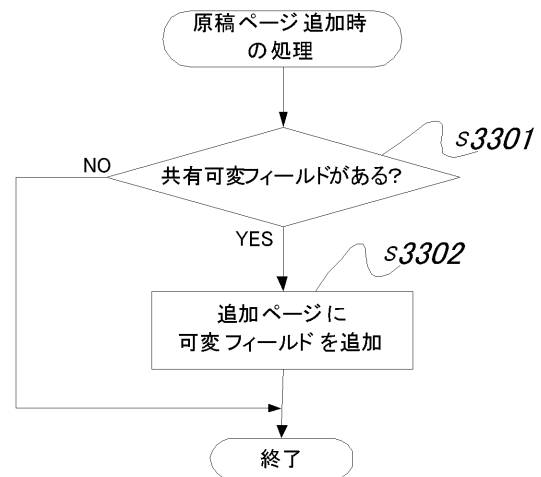
【図 31】



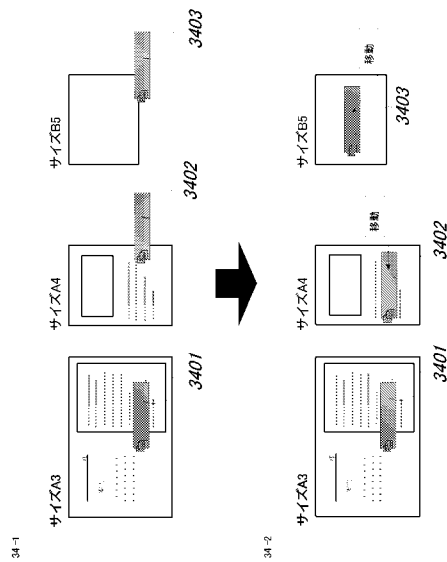
【図 32】



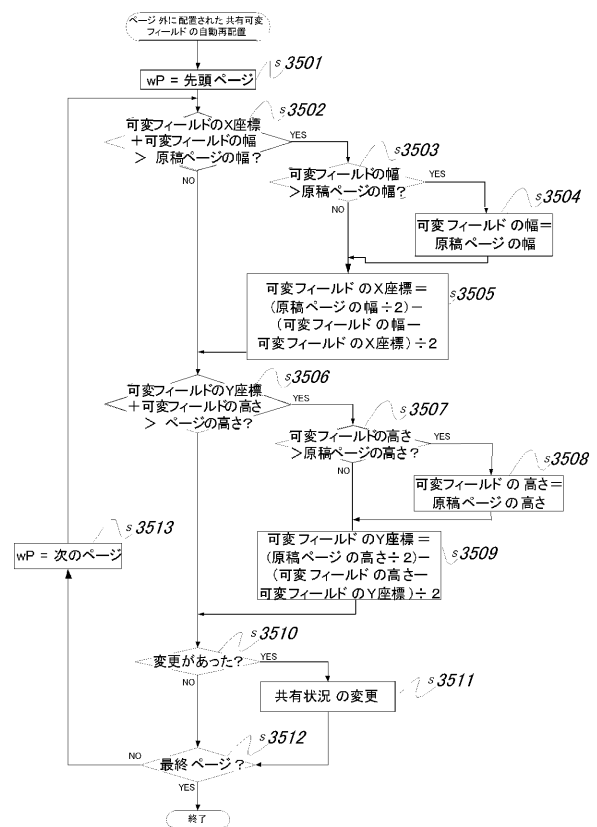
【図 33】



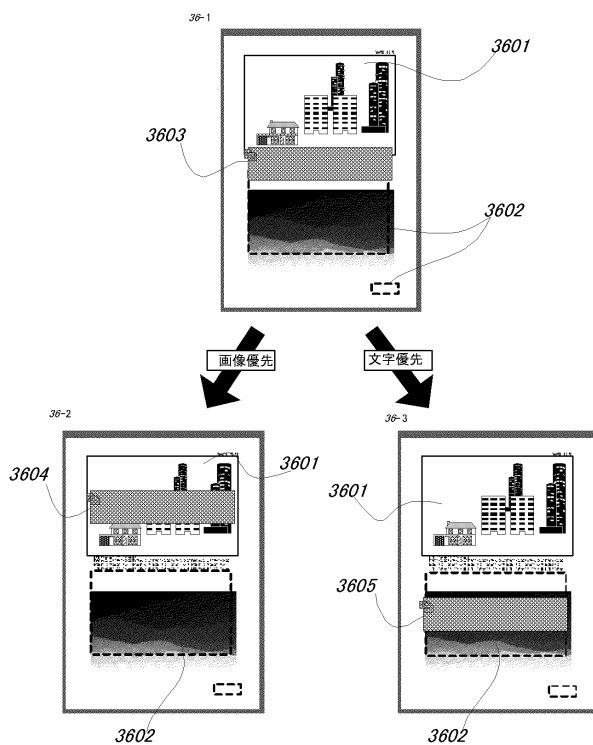
【図 34】



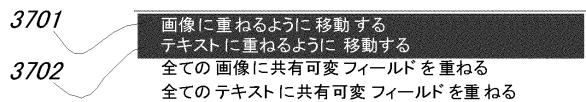
【図 35】



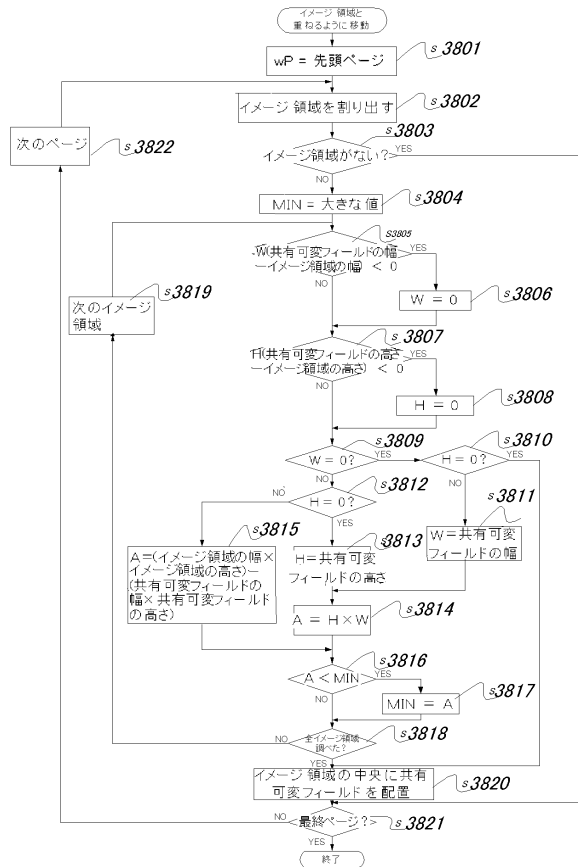
【図 36】



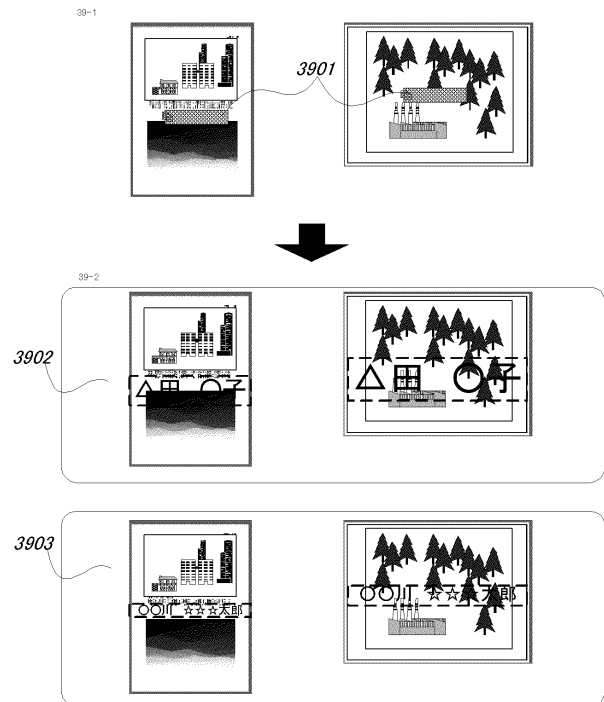
【図 37】



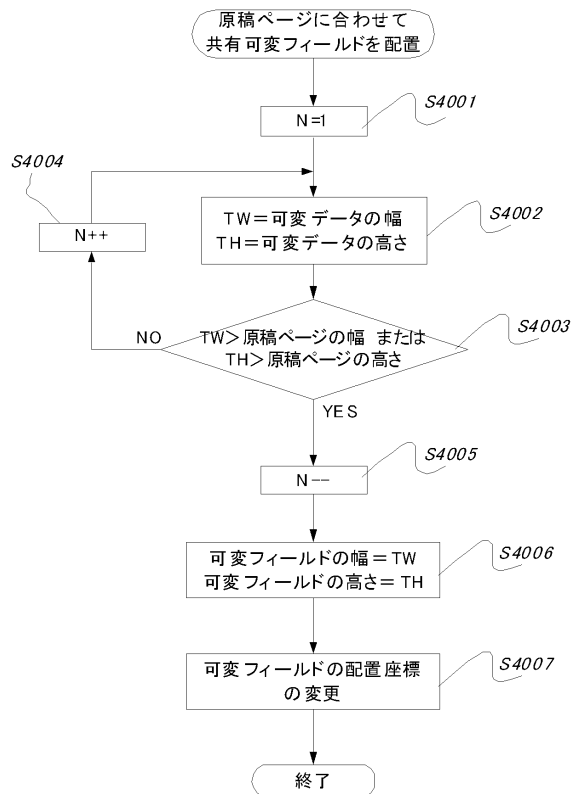
【図 38】



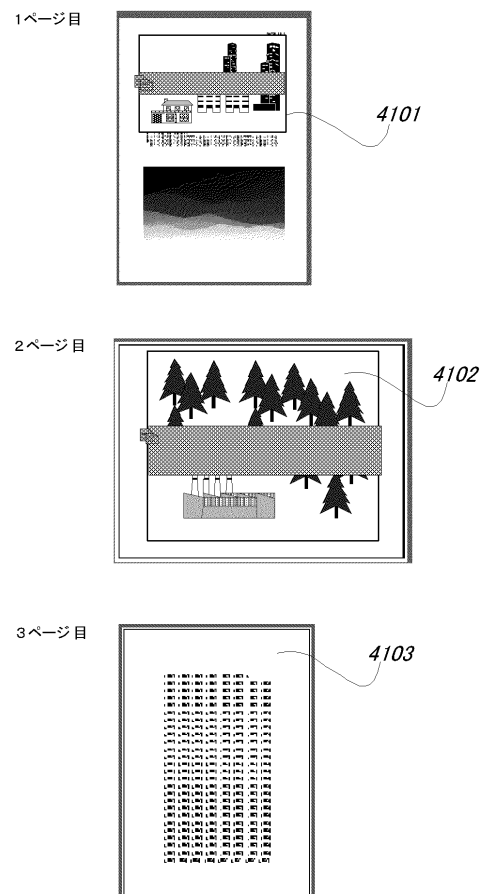
【図 39】



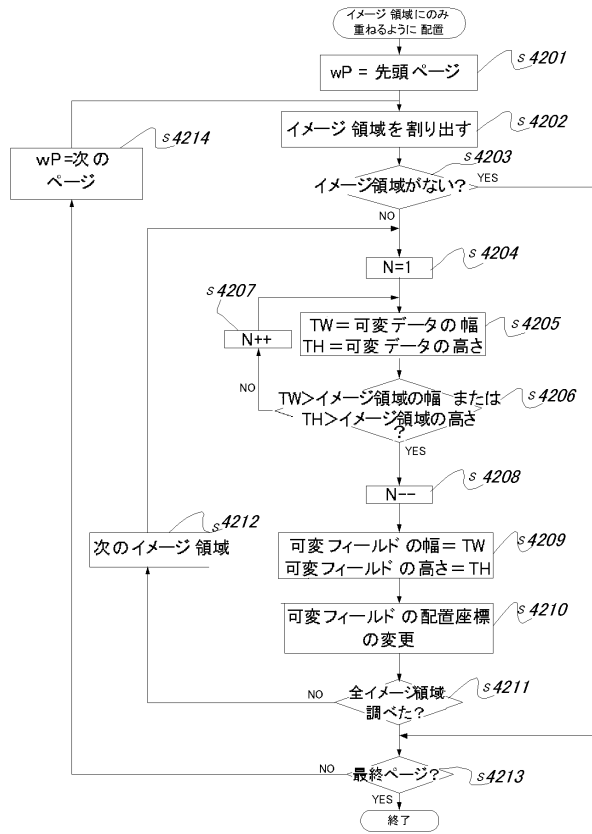
【図 40】



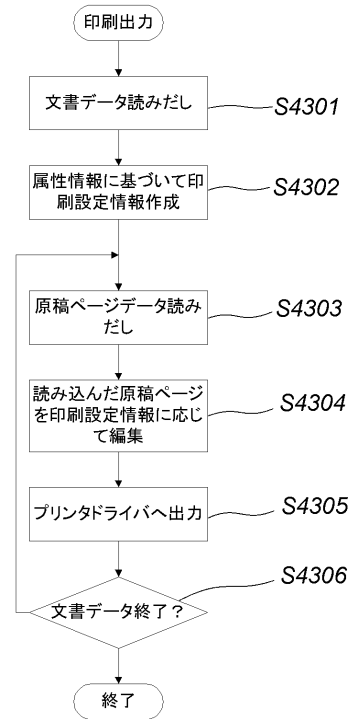
【図 41】



【図 4 2】



【図 4 3】



フロントページの続き

(51)Int.Cl.

F I

H 0 4 N 1/387

審査官 衣川 裕史

(56)参考文献 特開 2 0 0 1 - 3 2 8 2 3 5 (J P , A)

特開 2 0 0 4 - 0 0 5 5 7 6 (J P , A)

(58)調査した分野(Int.Cl. , D B 名)

G 0 6 F 3 / 1 2

G 0 6 F 1 7 / 2 1

G 0 6 T 1 1 / 6 0

H 0 4 N 1 / 3 8 7