

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4732029号
(P4732029)

(45) 発行日 平成23年7月27日(2011.7.27)

(24) 登録日 平成23年4月28日(2011.4.28)

(51) Int.Cl. F I
G06T 11/60 (2006.01) G O 6 T 11/60 I O O A
G06F 3/12 (2006.01) G O 6 F 3/12 C

請求項の数 12 (全 51 頁)

<p>(21) 出願番号 特願2005-189938 (P2005-189938)</p> <p>(22) 出願日 平成17年6月29日 (2005.6.29)</p> <p>(65) 公開番号 特開2007-11577 (P2007-11577A)</p> <p>(43) 公開日 平成19年1月18日 (2007.1.18)</p> <p>審査請求日 平成20年6月30日 (2008.6.30)</p> <p>前置審査</p>	<p>(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</p> <p>(74) 代理人 100126240 弁理士 阿部 琢磨</p> <p>(74) 代理人 100124442 弁理士 黒岩 創吾</p> <p>(72) 発明者 徳永 百重 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</p> <p>審査官 真木 健彦</p>
--	--

最終頁に続く

(54) 【発明の名称】 レイアウト決定方法および情報処理装置およびレイアウト決定プログラム

(57) 【特許請求の範囲】

【請求項1】

複数のオブジェクトが配置された第1ページと、前記第1ページの次のページに当たる少なくとも1つのオブジェクトが配置された第2ページを編集する情報処理装置において実行されるレイアウト決定方法であって、

オブジェクト間の距離を所定の距離まで縮小する編集処理を実行する編集対象としてページ範囲を指定する指定工程と、

前記指定工程において指定された前記ページ範囲に含まれる前記第1ページ内に配置されたオブジェクト間の距離を所定の距離まで縮小する縮小工程と、

前記指定工程において指定された前記ページ範囲に含まれる前記第2ページに配置されたオブジェクトを前記縮小工程によって前記オブジェクト間の距離を縮小することにより生じる前記第1ページの領域に移動する配置工程を備え、

前記縮小工程は、前記第1ページの複数のオブジェクト間の中から、優先度に従って縮小すべきオブジェクト間を決定することを特徴とするレイアウト決定方法。

【請求項2】

前記オブジェクトにはコンテンツデータが入力されることを特徴とする請求項1に記載のレイアウト決定方法。

【請求項3】

前記縮小工程による縮小処理を実行するための条件として、指定ページである第2ページに配置されたオブジェクトが前記指定ページを占める割合の閾値を設定する設定工程を

10

20

更に有し、

前記縮小工程は、前記指定ページに配置されたオブジェクトが前記指定ページを占める割合が前記閾値以下であると判定された場合、前記オブジェクト間の距離を縮小する処理を実行することを特徴とする請求項 1 または 2 に記載のレイアウト決定方法。

【請求項 4】

前記所定の距離はユーザの設定を受け付けることにより設定されることを特徴とする請求項 1 乃至 3 のいずれか 1 項に記載のレイアウト決定方法。

【請求項 5】

複数のオブジェクトが配置された第 1 ページと、前記第 1 ページの次のページに当たる少なくとも 1 つのオブジェクトが配置された第 2 ページを編集する情報処理装置であって

10

、
オブジェクト間の距離を所定の距離まで縮小する編集処理を実行する編集対象としてページ範囲を指定する指定手段と、

前記指定手段において指定された前記ページ範囲に含まれる前記第 1 ページ内に配置されたオブジェクト間の距離を所定の距離まで縮小する縮小手段と、

前記指定手段において指定された前記ページ範囲に含まれる前記第 2 ページに配置されたオブジェクトを前記縮小手段によって前記オブジェクト間の距離を縮小することにより生じる前記第 1 ページの領域に移動する配置手段を備え、

前記縮小手段は、前記第 1 ページの複数のオブジェクト間の中から、優先度に従って縮小すべきオブジェクト間を決定することを特徴とする情報処理装置。

20

【請求項 6】

前記オブジェクトとはコンテンツデータが入力される部分領域であることを特徴とする請求項 5 に記載の情報処理装置。

【請求項 7】

前記縮小手段による縮小処理を実行するための条件として、指定ページである第 2 ページに配置されたオブジェクトが前記指定ページを占める割合の閾値を設定する設定手段を更に有し、

前記縮小手段は、前記指定ページに配置されたオブジェクトが前記指定ページを占める割合が前記閾値以下であると判定された場合、前記オブジェクト間の距離を縮小する処理を実行することを特徴とする請求項 5 または 6 に記載の情報処理装置。

30

【請求項 8】

前記所定の距離はユーザの設定を受け付けることにより設定されることを特徴とする請求項 5 乃至 7 のいずれか 1 項に記載の情報処理装置。

【請求項 9】

複数のオブジェクトが配置された第 1 ページと、前記第 1 ページの次のページに当たる少なくとも 1 つのオブジェクトが配置された第 2 ページを編集する情報処理装置において実行されるプログラムであって、

オブジェクト間の距離を所定の距離まで縮小する編集処理を実行する編集対象としてページ範囲を指定する指定工程と、

前記指定工程において指定された前記ページ範囲に含まれる前記第 1 ページ内に配置されたオブジェクト間の距離を所定の距離まで縮小する縮小工程と、

40

前記指定工程において指定された前記ページ範囲に含まれる前記第 2 ページに配置されたオブジェクトを前記縮小工程によって前記オブジェクト間の距離を縮小することにより生じる前記第 1 ページの領域に移動する配置工程を備え、

前記縮小工程は、前記第 1 ページの複数のオブジェクト間の中から、優先度に従って縮小すべきオブジェクト間を決定することを前記情報処理装置に実行させるためのコンピュータが読み取り可能なレイアウト決定プログラム。

【請求項 10】

前記オブジェクトとはコンテンツデータが入力される部分領域であることを特徴とする請求項 9 に記載のレイアウト決定プログラム。

50

【請求項 1 1】

前記縮小工程による縮小処理を実行するための条件として、指定ページである第 2 ページに配置されたオブジェクトが前記指定ページを占める割合の閾値を設定する設定工程を更に有し、

前記縮小工程は、前記指定ページに配置されたオブジェクトが前記指定ページを占める割合が前記閾値以下であると判定された場合、前記オブジェクト間の距離を縮小する処理を

実行することを特徴とする請求項 9 または 1 0 に記載のレイアウト決定プログラム。

【請求項 1 2】

前記所定の距離はユーザの設定を受け付けることにより設定されることを特徴とする請求項 9 乃至 1 1 のいずれか 1 項に記載のレイアウト決定プログラム。

【発明の詳細な説明】**【技術分野】****【0 0 0 1】**

本発明は、自動レイアウトシステムにおける装置およびレイアウト手法に関するものである。

【背景技術】**【0 0 0 2】**

近年、商品の多品種化で商品ライフが短くなっていること、インターネット利用の普及による消費者のカスタマイズサービス指向などの要因から CRM (カスタマ・リレーションシップ・マネージメント) や、One-to-One (1 対 1) マーケティングの必要性が注目されている。これらの手法により顧客満足度を高め、顧客の開拓や囲い込みを目指すものである。

【0 0 0 3】

カスタマ・リレーションシップ・マネージメントとは、顧客のデータベースをもとに、個々のニーズに即した対応を実施して、顧客の満足度を高め、会社の収益性を向上させる仕組みである。また 1 対 1 マーケティングは、データベース・マーケティングの一種で、顧客の年齢、性別、趣味、嗜好、購買履歴等の個人属性情報をデータベース化し、その内容を分析して、顧客のニーズに合った提案を行うマーケティング手法であり、その代表的な適用例としてバリアブルプリントが挙げられる。ここ最近、DTP (デスクトップパブリッシング) 技術の進展とデジタル印刷装置の普及に伴って、文書を顧客毎にカスタマイズして出力するバリアブルプリントシステムが開発され、顧客毎に異なる量のコンテンツを最適にレイアウトした文書の作成が求められるようになった。

【0 0 0 4】

従来、バリアブルプリントシステムでは、ドキュメント上にレイアウト枠 (コンテナともいう) が作成され、各コンテナのコンテンツとしてデータベースの一定の条件を満たすレコードやフィールドが関連づけられる。こうしてデータベースとレイアウトとを関連付けることにより、バリアブルプリントは実現されていた。

【0 0 0 5】

しかし、コンテンツであるテキストおよびイメージを流し込むべきコンテナのサイズが固定であったため、データベース内のコンテンツデータがコンテナに挿入されたときに、データ量がコンテナサイズより多いとテキストのオーバーラップおよびイメージのクリッピングが発生する。またデータ量がコンテナサイズより小さいと隙間が空いてしまう。

【0 0 0 6】

それらの問題を解決するための技術として、ある文字領域のサイズが大きくなった場合、隣接した領域のサイズを小さくする技術が特許文献 1 に開示されており、文字領域に対して文字を入力するに従い該文字領域サイズを拡張していき、該文字領域と隣接する領域サイズを縮小することが記載されている。

【0 0 0 7】

また、特許文献 2 には、複数の商品情報 (商品レコード) をページ内のレイアウト領域

10

20

30

40

50

に配置する技術も記載されている。

【0008】

この技術では、レイアウト（たとえば商品名、イメージ、価格などの配置）をあらかじめ設定したマスタ部品（本件のサブテンプレートに相当）を用意しておき、商品情報ごとにどのマスタ部品を使用するかを設定する。そして、商品情報が有するデータ（商品名やイメージ、価格など）を、設定されたマスタ部品内の所定の領域に配置して、ページ内に指定されたレイアウト領域に順次配置する。

【特許文献1】特開平7-129658号

【特許文献2】特開2000-48216号

【発明の開示】

【発明が解決しようとする課題】

【0009】

しかしながら、特許文献1に記載の技術には、テキストの入力に応じて文字領域が拡大されていくことが記載されているが、文字領域のサイズが大きくなった場合、隣接した領域は間隔を保つべく縮小されてしまうため、入力されるテキスト量が増加するに従い、隣接した領域は縮小され続けてしまうといった問題点があった。

【0010】

更に特許文献1では、上述したようなデータベースと各領域を関連付けてコンテンツデータを流し込むことで、顧客毎にカスタマイズした文書を作成するバリアブルプリントシステムについては考慮されていなかった。

【0011】

また、特許文献2では、レイアウト領域に配置すべきレコード数によっては、指定されたレイアウト対象ページに各レコードのデータを流し込んだ画像部品ブロックを収めることができない場合がある。特許文献2では、レイアウト対象ページが複数ページに渡っている場合、1ページに収めることのできなかつた画像部品ブロックを次のページに自動的に配置していくことが記載されている。

【0012】

しかしながら、特許文献2に記載の上記処理を用いた場合、次のページに配置される画像部品ブロックが1つであったとしても新たなページを作成されるため、新たなページは画像部品ブロックが1つで、ページの大部分が余白といった構成になり、見栄えが悪くなるのみでなく、無駄な印刷コストが発生する恐れがある。

【0013】

本発明は上記課題を鑑みてなされたものであり、ユーザが所望とする印刷結果を容易に出力できるようにすることを目的とする。

【課題を解決するための手段】

【0014】

上述した課題を解決するために、本発明は以下の構成を備える。複数のオブジェクトが配置された第1ページと、前記第1ページの次のページに当たる少なくとも1つのオブジェクトが配置された第2ページを編集する情報処理装置において実行されるレイアウト決定方法であって、オブジェクト間の距離を所定の距離まで縮小する編集処理を実行する編集対象としてページ範囲を指定する指定工程と、前記指定工程において指定された前記ページ範囲に含まれる前記第1ページ内に配置されたオブジェクト間の距離を所定の距離まで縮小する縮小工程と、前記指定工程において指定された前記ページ範囲に含まれる前記第2ページに配置されたオブジェクトを前記縮小工程によって前記オブジェクト間の距離を縮小することにより生じる前記第1ページの領域に移動する配置工程を備え、前記縮小工程は、前記第1ページの複数のオブジェクト間の中から、優先度に従って縮小すべきオブジェクト間を決定することを特徴とする。

【発明の効果】

【0015】

本発明を適用することにより、ユーザが所望とする印刷結果を容易に出力することが可

10

20

30

40

50

能となる。

【発明を実施するための最良の形態】

【0016】

本発明の実施例を適用するのに好適である実施例について説明を行う。

【0017】

(システム構成図)

図1Aはバリアブルデータドキュメントを印刷するシステム100を図示している。この中で示された方法は図1Bで詳しく説明される汎用ホストコンピュータ101で実践される。図1Bで記述されるプロセスは、ホストコンピュータ101内で実行され、システム100上で実施可能となるレイアウト編集アプリケーションプログラム121のようにソフトウェアの全体、あるいは一部分で実行される。特にレイアウト編集や必然的に起こる印刷のステップはホストコンピュータ101によって実行されるソフトウェアの指示によって実施される。

10

【0018】

ソフトウェアは例えば以下に記述されるような記憶装置を含むコンピュータの可読媒体に格納される。ソフトウェアはコンピュータの可読媒体からコンピュータにロードされ、ホストコンピュータ101によって実行される。そのようなソフトウェアや媒体に記録されたコンピュータプログラムを持つコンピュータの可読媒体はコンピュータプログラム製品である。そのコンピュータでのコンピュータプログラム製品の使用は望ましくもドキュメントのレイアウト編集やバリアブルデータ印刷に有利な装置をもたらす。

20

【0019】

ホストコンピュータ101はキーボード132やマウス133のようなポインティングデバイスなどの入力装置をつなぎ、ディスプレイ装置144や状況に応じてはローカルプリンタ145を含む出力装置を連結する。入力/出力インターフェース138はホストコンピュータ101をネットワーク接続107から接続してシステム100の他のコンピュータ装置につなげることができる。そのネットワーク接続107の典型はローカルエリアネットワーク(LAN)、あるいはワイドエリアネットワーク(WAN)である。

【0020】

ホストコンピュータ101は典型的に少なくとも1つのプロセッサユニット135、例えば半導体のランダムアクセスメモリ(RAM)やリードオンリーメモリ(ROM)から構成されるメモリユニット136、ビデオインタフェース137を含むINPUT/OUTPUT(I/O)インターフェース、キーボード132やマウス133のためのI/Oインターフェース143を含んでいる。記憶装置139は典型的にハードディスクドライブ140やフロッピー(登録商標)ディスクドライブ141を含んでいる。図には示されていないが磁気テープドライブもまた使用される可能性がある。CD-ROMドライブ142は不揮発性のデータソースとして提供される。ホストコンピュータ101はGNU/LINUXやマイクロソフトウィンドウズ(登録商標)のようなオペレーションシステムや、典型的にはオペレーションシステムに従う形である。または関連のある技術で知られているもので形成されたコンピュータシステムの常套的なオペレーションモードによる方法によって、相互接続バス134を介して通信を行うホストコンピュータ101のコンポーネント135から143を利用する。

30

40

【0021】

なお、図1Bに記述したホストコンピュータ101の例ではIBM互換PCやSUN社のSparc Station、あるいはそれらを含んだコンピュータシステムが考えられる。

【0022】

レイアウト編集アプリケーションプログラム121は典型的にハードディスクドライブ140に常駐し、プロセッサ135により実行、読み込み、コントロールされる。レイアウト編集アプリケーションプログラム121の媒介記憶装置とネットワーク1020からフェッチされるデータはハードディスクドライブ140に呼応してメモリユニット136

50

を使用する。また他の方法としては、アプリケーションプログラム 121 が CD-ROM やフロッピー（登録商標）ディスク上でエンコードされ、対応するドライブ 142 や 141 を通じて読み込まれユーザに提供される。あるいはもう一つの方法としてアプリケーションプログラム 121 はネットワーク接続 107 からユーザによって読み込まれるかもしれない。さらにソフトウェアは磁気テープまたは ROM または集積回路、光磁気ディスクまたは無線またはホストコンピュータ 101 とその他のデバイス間の赤外線通信、PCMCIA カードのようなコンピュータ可読カード、そして Eメール通信や WEB サイト上の記録情報を持つインターネットやイントラネットを含む、他の適当な大きさのコンピュータ可読媒体からホストコンピュータ 101 内にロードされる可能性もある。前述は単に関連するコンピュータ可読メディアの模範である。他のコンピュータ可読媒体も使用される可能性はある。

10

【0023】

またレイアウト編集アプリケーション 121 はバリエブルデータ印刷 (VDP) を行うよう指示し、3つのソフトウェアコンポーネントを含んでいる。これらのうち1つめのコンポーネントはレイアウトエンジン 105 であり、これは長方形の範囲内で与えられた制限やサイズによって矩形と線の位置を計算するためのソフトウェアコンポーネントである。

【0024】

2つめのコンポーネントであるユーザインターフェース 103 はユーザにドキュメントテンプレートを作成させ、ドキュメントテンプレート内でデータソースと関連付けるメカニズムを提供する。

20

【0025】

3つめのコンポーネントである UI モデルアナライザ 104 は、ユーザインターフェース 103 が提供するコンテナや、アンカー、スライダー、リンクなどのユーザインターフェース要素をレイアウトエンジン 105 が認識できる内部形式に変換する。すなわち、UI モデルアナライザ 104 により、レイアウトエンジン 105 に変更を加えることなく、ユーザインターフェース 103 を他の表示/入力形式のユーザインターフェースに容易に取替え可能になる。

【0026】

ユーザインターフェース 103、UI モデルアナライザ 104、レイアウトエンジン 105 はコミュニケーションチャンネル 123、124 を介して通信する。ドキュメント生成のためのデータソースは一般的にデータベースアプリケーションを動かしている他のコンピュータによって構成されたデータベースサーバ 117 上にある典型的なデータベース 119 である。

30

【0027】

ホストコンピュータ 101 はネットワーク接続 107 の手段によってデータベースサーバ 117 と通信する。レイアウト編集アプリケーション 121 はホストコンピュータ 101 が一般的に他のコンピュータで構成されるファイルサーバ 115 に保存されるドキュメントテンプレートを生成する。またレイアウト編集アプリケーション 121 はデータとマージされたドキュメントテンプレートによって構成されたドキュメントを生成する。

40

【0028】

これらのドキュメントはホストコンピュータ 101 のローカルファイルシステムに保存されるか、ファイルサーバ 115 に保存されるか、あるいはプリンタ 113 に直接印刷される。プリントサーバ 109 は直接ネットワークにつながっていないプリンタにネットワーク機能を提供するコンピュータである。プリントサーバ 109 とプリンタ 113 は典型的な通信チャンネル 111 を介して接続される。

【0029】

図 2 はエンジンサーバ 227 の追加を除き、レイアウトエンジン 105 の分離バージョン 225 を含む図 1A・図 1B の類似の図である。エンジンサーバ 227 は典型的なコンピュータである。ファイルサーバ 115 に保存されたドキュメントテンプレートは印

50

刷や他の目的がある際、レイアウトエンジン 2 2 5 によってドキュメントを生成するためにデータベース 1 1 9 に保存されたデータと結合することができる。そのようなオペレーションはユーザインターフェース 1 0 3 を介して要求されるか、特定のレコードのみ印刷するように要求される。

【 0 0 3 0 】

図 4 0 は本実施形態によるバリアブルデータプリントの概略を説明する図である。レイアウト編集アプリケーション 1 2 1 のユーザインターフェースモジュール 1 0 3 (以下、ユーザインターフェース 1 0 3 と記載する)により、ユーザからの操作指示に従いページ上に複数のコンテナ 4 0 5 ~ 4 0 7 を配置し、各コンテナに位置やサイズに関する制約条件を付与することによりドキュメントテンプレート 4 0 4 が生成される。

10

【 0 0 3 1 】

また、ユーザインターフェース 1 0 3 はドキュメントテンプレート 4 0 4 とデータソース 4 0 0 (図 1 ではデータベース 1 1 9)との関連付け、更に各コンテナとデータソース 4 0 0 内の各データフィールドとの関連付けを行う。各コンテナとデータソース 4 0 0 内の各データフィールドとの関連付けた情報は、ドキュメントテンプレート内に記述され、該ドキュメントテンプレートは、HDD 1 4 0 またはメモリユニット 1 3 6 に格納される。

【 0 0 3 2 】

また、データソース 4 0 0 は、レコード単位で項目データ(バリアブルデータ)が記載されているファイルであり、HDD 1 4 0 に格納されている。レイアウトエンジン 1 0 5 は、ユーザからの印刷指示もしくはプレビュー指示に応じて、ドキュメントテンプレートの各コンテナ 4 0 5 ~ 4 0 7 に、関連付けられたバリアブルデータをデータソース 4 0 0 から読み込み、レコード単位で流し込み(例えば、データレコード 1 のデータフィールド A ~ C をコンテナ 4 0 5 ~ 4 0 7 へ流し込む)、流し込まれるべきデータのサイズに応じて各コンテナのサイズ等を調整(レイアウト調整)する。なお、レイアウト調整の方法については後述する。

20

【 0 0 3 3 】

プレビュー指示の場合は、レイアウト調整されたドキュメントイメージを生成し、ビデオディスプレイ 1 4 4 の画面上にプレビューとして表示するべく出力される。また印刷指示の場合は、レイアウトエンジン 1 0 5 もしくはプリンタドライバを用いて生成したドキュメントイメージを印刷データとしてプリントサーバ 1 0 9 へ出力する。データレコード 1, 2, 3 ... を順次処理することにより、バリアブルデータプリントが実現されることになる。

30

【 0 0 3 4 】

ドキュメント生成のためのデータソース(4 0 0)は、例えば、データベースアプリケーションを動かしている他のコンピュータによって構成されたデータベースサーバ 1 1 7 上の一般的なデータベース 1 1 9 であってもよい。この場合、ホストコンピュータ 1 0 1 はネットワーク 1 0 7 を介してデータベースサーバ 1 1 7 と通信し、データソースを取得できる。また、レイアウト編集アプリケーション 1 2 1 によって生成された、バリアブルデータプリントのためのドキュメントテンプレート(4 0 4)は、ホストコンピュータ 1 0 1 或いは他のコンピュータで構成されるファイルサーバ 1 1 5 に保存される。図 1 C で上述したように、レイアウト編集アプリケーション 1 2 1 のレイアウトエンジン 1 0 5 は、データとマージされたドキュメントテンプレートによって構成されたバリアブルデータドキュメントを生成する。これらのドキュメントは、ホストコンピュータ 1 0 1 のローカルファイルシステムに保存されるか、ファイルサーバ 1 1 5 に保存されるか、あるいはプリンタ 1 1 3 に送信されて印刷される。プリントサーバ 1 0 9 はネットワークと直接にはつながっていないプリンタにネットワーク機能を提供するためのコンピュータである。プリントサーバ 1 0 9 とプリンタ 1 1 3 は一般的な通信チャンネル 1 1 1 を介して接続される。

40

【 0 0 3 5 】

50

(アプリケーション構成図)

(メインウインドウ)

ユーザインターフェース103は、操作時に図3に示されるようなアプリケーションウインドウ301によって形成されたユーザインターフェース画面をビデオディスプレイ144に表示させる。このウインドウ301は、メニューバー302、ツールバー303、ワークエリア306とオプションのパレット311を有する。

【0036】

メニューバー302とツールバー303は非表示にすることや、スクリーン上の色々な場所に移動することが可能である。また、ワークエリア306はマウス133の操作によってその場所を移動させることが可能である。また、パレット311はオプションであり、カーソル/ポインタデバイス313はマウス133が指し示す位置を表す。

10

【0037】

メニューバー302は、周知の技術として知られているように、メニューオプションの階層の下に拡張される多くのメニューアイテム304を持つ。

【0038】

ツールバー303は、アプリケーションの特別なモードによって非表示状態にする、または表示状態にすることが可能な多くのツールボタンとウィジェット305を持つ。

【0039】

ルーラー308はオプションであり、ワークエリア内のポインタ、ページ、ライン、マージンガイド、コンテナまたはオブジェクトの位置を示すために使われる。

20

【0040】

パレット311はバリアブルデータライブラリのような追加機能にアクセスするために使われる。パレット311は移動、リサイズ、クローズをするためのウインドウコントロール312を持つ。パレット311はオプションで、ワークエリアの前面に表示される、あるいはオブジェクトの背面に隠される。パレット311はアプリケーションウインドウ301の範囲内のみに表示されることを制限される、あるいはアプリケーションウインドウ301の外側にその一部或いは全体を表示することを許される。

【0041】

ツールバー303には図4に示されるような、ユーザ選択可能な『ボタン』が配置されている。

30

(1) 選択ツールボタン403: コンテナの辺を選択、移動、サイズ変更、リサイズそしてロック/ロック解除のために使われる。コンテナの選択は、コンテナの周りに選択ボックスをドラッグすることによりなされる。また、CTRLキーを押しながら、複数のコンテナについて選択操作をすることによって、複数のコンテナを選択可能である。

(2) テキストコンテナツールボタン404: スタティックあるいはバリアブルテキストを持つコンテナを作成するために使われる。

(3) イメージコンテナツールボタン405: スタティックあるいはバリアブルイメージを持つコンテナを作成するために使われる。

(4) リンクツールボタン406: コンテナ間に関連付けを行うリンクを作成するために使われ、リンクの距離をコントロールするためにも使われる。

40

【0042】

レイアウト編集アプリケーション121の図3に示したアプリケーションウインドウ301は、ページ内に各コンテナやリンクをレイアウトすることで、基本レイアウトを決定することができる。基本レイアウトとは、バリアブルデータプリントで基本となるレイアウトのことである。基本レイアウト内の各コンテナが固定コンテナである場合は、すべてのレコードの印刷結果のレイアウトは同じになる。

【0043】

また、基本レイアウト内の少なくとも1つのコンテナが後述する可変コンテナである場合は、レコード単位に読み込まれるデータの量やサイズにより各コンテナのサイズや位置が、後述する制約の範囲内で変動することになる。よって、レイアウト編集アプリケーシ

50

ョン121で作成されるドキュメントテンプレートは、あくまで基本レイアウトを決定するものであり、可変コンテナが含まれる場合は、最終的な印刷物のレイアウトは読み込まれるデータによりレイアウト調整されることになる。

【0044】

(ドキュメントテンプレート)

図3において、ワークエリア306はドキュメントテンプレートのデザインを表示・編集するために使われる。また、ドキュメントテンプレートを単にテンプレートということもある。これはユーザがドキュメントテンプレートをデザインする過程において、印刷されるドキュメントの概観をユーザに提示することを可能とする。これにより、ユーザは、データベース119とマージされたドキュメントが、バリエブルデータの量・サイズに基づいてどのように変化するかを容易に理解できる。

10

【0045】

また、データベースがドキュメントテンプレートに関連付けられていた場合は、現在のドキュメントのプレビューができるように、対応するバリエブルテキストやイメージがレイアウトされた各コンテナに表示される。

【0046】

ドキュメントテンプレートにおけるドキュメント構造とバリエブルデータコンテナの描写をする視覚的な手がかり(コンテナの枠線、アンカー、スライダー、リンク等)は、ドキュメントテンプレート作成時には常に表示され、バリエブルデータを流し込むプレビュー時には、視覚的な手がかりは、カーソルをコンテナ上に移動させたときや、コンテナを選択したときに表示される。

20

【0047】

ワークエリア306はスクロールバー307とオプションのルーラー308とドキュメントテンプレート309を含む。ドキュメントテンプレート309はページが複数あることを示すことができる。

【0048】

与えられたドキュメントテンプレートのページサイズは、周知の技術を用いて、ユーザによって指定される。例えばメニューの「ファイル」から「ページ設定」を選択することでページサイズを設定するダイアログを表示し、そこでユーザが指定したページサイズが反映されることになる。それぞれのドキュメントでの実際のページ数は、関連付けられたデータソース内のバリエブルデータによって変化する可能性が有る。これは、ドキュメントテンプレート内に可変表のようにバリエブルデータの量により大きさが変更されるフィールドが設定されている場合、1ページ内にバリエブルデータをフィットできないバリエブルデータが読み込まれると、追加のページが自動的に作成されることも考えられるからである。

30

【0049】

それぞれのページ内に示される境界線310は、ページ上の印刷可能なオブジェクトの最大幅を示す、任意のページマージンである。

【0050】

図4は1ページのドキュメントテンプレート309上に表示され得るオブジェクトの例を示す図である。このようなオブジェクトとしては、コンテナ407、408と、任意に適用するアンカーアイコン409、固定されている辺411、414、固定されていない辺410、リンク412そしてスライダー413を持つ。

40

【0051】

アンカーアイコン409は、コンテナの矩形の角、辺、またはコンテナの中央に設定することが可能である。なお、本願で記述するオブジェクトとはドキュメントテンプレート上に表示され得るもの全てを指し、各オブジェクトを明確に区別する場合は、それぞれの名称で記述することにより明確にする。アンカーアイコン409が設定されると、設定された個所の位置が固定となる。つまり、図4の例では、アンカーアイコン409は、コンテナ407の左上の角に設定されているため、コンテナ407はバリエブルデータが流し

50

込まれ、バリエブルデータの画像サイズもしくはテキスト量が多い場合に、右方向及び下方向に拡大可能であることを示している。

【 0 0 5 2 】

例えば、アンカーアイコン 4 0 9 がいずれか 1 つの辺に設定されている場合は、その辺が固定となり、その他の 3 辺の各方向に拡大可能である。また、アンカーアイコン 4 0 9 がコンテナの中央に設定されている場合は、コンテナの中央位置が固定となり、コンテナ矩形の中央位置が変わらないように、4 方向に拡大可能である。リンク 4 1 2 についての詳細は後述するが、コンテナ 4 0 7 とコンテナ 4 0 8 が関連付けられていることを示しており、このリンクに設定されている長さ（範囲指定可能）を保ちつつ、コンテナ 4 0 8 が右方向に移動可能であることを示している（なお後述する可変サイズリンクの場合は、流し込まれるデータサイズによりリンクの長さが設定された範囲内で可変となる）。スライダー 4 1 3 は、設定されている辺と水平方向に移動可能であることを示している。

10

【 0 0 5 3 】

ここで、コンテナについて説明する。コンテナは、ドキュメントテンプレート内にデータベースから固定あるいは可変のテキスト/イメージが流し込まれ、描画されるスペース（これを部分領域またはデータ領域と呼ぶ）であり、図 4 に示されるように他のコンテナやオブジェクトと共にレイアウトされる。ユーザインターフェース画面を介して、ユーザからの操作指示により、コンテナは移動、サイズ調整、再作成される。また、コンテナに流し込まれるデータをコンテンツまたはコンテンツデータ（バリエブルデータまたはフィールドデータ）と呼ぶ。

20

【 0 0 5 4 】

より正確にはコンテナは、設定の集まり、視覚的表現、そしてインタラクションと編集動作をもっている。下記は本実施形態によるコンテナの定義である。

【 0 0 5 5 】

（ 1 ）コンテナは固定あるいは可変のコンテンツを持つ。可変コンテンツは、データベースから取得したデータがドキュメント毎、つまりレコード毎に異なる可能性があるという意味でダイナミック（動的）である。ただし、本実施形態の可変コンテンツは、アニメーション化されたもの、あるいは他の方法で時間的に変化するコンテンツは印刷には適していないため、ここでは意図していない。同様に、固定コンテンツはコンテナを使って生成される全てのドキュメントで、同じように表示される。しかしながら、固定コンテンツが可変コンテンツとリンクにより関連付けられている場合、可変コンテンツの影響を受けて、固定コンテンツはそれぞれのドキュメントで位置が異なる可能性がある。

30

【 0 0 5 6 】

（ 2 ）コンテナは、コンテンツに適用される背景色、ボーダー、フォント・スタイルのようなテキスト設定と同様の装飾機能を持っている。このような設定をコンテナ属性と呼ぶ。コンテナ属性は、コンテナごとに設定可能であるが、あるコンテナと同じコンテナ属性であるという設定を行うことも可能である。

【 0 0 5 7 】

（ 3 ）コンテナはドキュメントを生成する際にデータベースからのデータとマージされる。装飾機能は、どんな固定コンテンツでもそうであるように、印刷された出力物において可視である。可変コンテンツはデータソースからの特定のデータの表示を提供する。コンテナのこの表現は例えば印刷されるか、ビデオディスプレイ 1 4 4 のスクリーン上に表示されるか、その両方が可能である。

40

【 0 0 5 8 】

（ 4 ）コンテナは、図 4 に示されるように視覚的な手がかりとしてのユーザインターフェースを有している。例えばコンテナの編集そして表示設定のためのインタラクティブなグラフィカルユーザインターフェース（GUI）を持つ。GUIの各要素はビデオディスプレイ 1 4 4 のスクリーン上に表示されるが、ドキュメントとしては印刷されない。レイアウト編集アプリケーション 1 2 1 のユーザインターフェース 1 0 3 は、背景色やフォントのようなコンテナの装飾機能のいくつかを表示し、さらにコンテナの設定の編集や表示

50

を可能にするための機能を有している。

【 0 0 5 9 】

(リンク)

リンクは、コンテナとコンテナの関連を示している。関連とはコンテナ間の距離を示しており、リンクによって関連付けられたコンテナ同士は、互いのレイアウト変更の影響を受けてレイアウトを計算する。図 4 の 4 1 2 で示されているものがリンクであり、図 4 ではコンテナ 4 0 7 と 4 0 8 とを関連づけている。リンクの設定方法および、リンクで関連付けられたコンテナのレイアウト計算方法については、後述する。

【 0 0 6 0 】

(リンクの設定方法)

次に、コンテナ同士を関連付けるためのリンクの設定について説明する。図 5 はリンクの設定方法を示したフローチャートである。また図 6 の (A) ~ (C) はリンク設定時のユーザインターフェース (U I) の遷移例を示している。図 5、6 を用いてコンテナにリンクを設定する方法について説明する。なお、本願に記載されているフローチャートの各ステップの処理は、情報処理装置 (ホストコンピュータ) が有するプロセッサ 1 3 5 によって実行されることとなる。

【 0 0 6 1 】

まず、ステップ S 6 0 1 において、レイアウト編集アプリケーション 1 2 1 は、ユーザインターフェース画面のワークエリア 3 0 6 上に編集すべく選択されたドキュメントテンプレートを表示する。リンクを設定するためには、リンクを設定するためのコンテナ (最低 2 つ) がドキュメントテンプレート上に作成されている必要がある。図 6 の (A) ~ (C) では、ステップ S 6 0 1 で 2 つのコンテナを作成してリンクを設定する場合のユーザインターフェースの遷移例を示している。

【 0 0 6 2 】

次に、ステップ S 6 0 2 において、レイアウト編集アプリケーション 1 2 1 は、前述したリンクツールが選択状態 (図 4 のボタン 4 0 6 をクリックすることにより選択状態となる) になったかを判断する。リンクツールが選択状態でなければ、必要に応じて他の各種処理を実行し (ステップ S 6 0 9)、ステップ S 6 0 2 に戻る。

【 0 0 6 3 】

図 6 の (A) において、コンテナ 7 0 1 と 7 0 2 はすべて固定されている辺で構成されているものとする。また、7 0 3 と 7 0 4 は、図 4 の 4 0 9 と同じであり、アンカーを意味する。7 0 5 はマウスポインタを意味している。

【 0 0 6 4 】

さて、リンクツールが選択状態となっている間に、ユーザはリンクを設定する 2 つのコンテナのうち的一方 (コンテナ 7 0 1 とする) をクリックして選択する。

【 0 0 6 5 】

この操作に応じて、レイアウト編集アプリケーション 1 2 1 のユーザインターフェース 1 0 3 は第 1 のコンテナが指定されたことを認識し (ステップ S 6 0 3)、選択されたコンテナを特定する情報をメモリ 1 3 6 に保持する。また、以降のマウスカーソルの移動に応じた軌跡を画面に表示するようにする。

【 0 0 6 6 】

例えば、図 6 の (B) における線分 7 0 6 は、(A) の状態におけるクリック位置と現在のマウスカーソルの位置とを結んだ線を示しており、この U I によりどの位置にリンクが設定されるのかをユーザに明示することができる。

【 0 0 6 7 】

次にユーザは、図 6 の (B) で示されるように、もう一方のコンテナ (コンテナ 7 0 2) までマウスポインタを移動してクリックする。この操作に応じて、ユーザインターフェース 1 0 3 は第 2 のコンテナが指定されたことを認識し (ステップ S 6 0 4)、レイアウト編集アプリケーション 1 2 1 は、ステップ S 6 0 3 で保持した第 1 のコンテナと、ステップ S 6 0 4 で指定が認識された第 2 のコンテナとの間にリンクを設定する。

10

20

30

40

50

【0068】

こうして、ユーザにより選択された2つのコンテナ701, 702の間にリンクが設定されると、リンクUI707が表示される(ステップS605)。更に、このリンク設定を受けて、コンテナの表示状態は図6の(C)の状態になる(ステップS606)。

【0069】

すなわち、リンクが設定されたことにより、コンテナのUIが自動的に変更される。ここでは、リンクによって関連付けられた辺が可変となり、点線で示される。図6の(C)において、708は点線で示されている辺であり、前述した通り可変の辺を示すものである。

【0070】

なお、図6の(C)のようなコンテナの辺の状態の変化は、リンクを設定したことによりコンテナの辺を可変にする必要が生じたことにより自動的に実行されたものであり、リンクを設定して、レイアウトの調整を行おうとしているにもかかわらず、全ての辺が固定のためレイアウト調整が行えないという矛盾の発生を防ぐことを目的としている。インジケータ709は、リンクを設定したことにより、コンテナが変化できる方向をユーザに視覚的に示したマークである。

【0071】

また、図6の(C)の例では、左のコンテナの右辺と右のコンテナの左辺が可変な状態へ変化したが、これは一例であり、たとえば、右コンテナが図4の413で示したスライダーを持つ設定に変化してもかまわない。

【0072】

(レイアウトエンジンによるレイアウト計算処理)

(レイアウト計算方法(全体フロー))

本実施形態のレイアウト編集アプリケーションは、ユーザインターフェース103を用いてコンテナを作成し、そのコンテナ間に関連付け(リンク設定)を行ってレイアウトを作成するレイアウトモードと、レイアウトエンジン105により、作成したレイアウトにデータソースの各レコードを挿入して、実際にレコードが挿入された後のレイアウト結果をプレビューするプレビューモードに分けられる。このプレビューモードにおいて、実際のレコードが有するコンテンツデータが挿入され、レイアウトを計算する。ただし、プレビューモードは、表示上でのレイアウト計算である。実際に印刷する場合においても、レイアウトエンジン105が各コンテナにデータを挿入してレイアウトを計算するが、その際の計算方法はプレビューモードと同じである。なお、コンテンツデータを流し込むべきコンテナに優先順位が設定されている場合は、該優先順位に従ってレイアウト計算を行うこととなる。優先順位に従うレイアウト計算の一例として、優先順位の高いコンテナに流し込まれるコンテンツデータは、できる限りオリジナルサイズ(元のデータサイズ)で表示するように制御される。

【0073】

図7はレイアウト計算のフローを示している。まず、プレビューモードが選択される(ステップS801)。プレビューモードになったら、レイアウト編集アプリケーション121は、ユーザにプレビューするレコードをデータソースより選択させ、選択されたレコードの各フィールドデータを各コンテナに挿入するよう決定する(ステップS802)。

【0074】

各コンテナへのフィールドデータの挿入が決定されると、レイアウト編集アプリケーション121は、そのレコードをレイアウトするための計算を行い、必要に応じてレイアウト調整を行う(ステップS803)。ステップS803におけるレイアウト計算の詳細については後述する。そして、レイアウト編集アプリケーション121は、ステップS803で計算されたレイアウトを表示する(ステップS804)。

【0075】

レイアウト編集アプリケーション121は、他のレコードについてもプレビューを行うかどうかをユーザの指示により判断する(ステップS805)。ステップS805で、他

10

20

30

40

50

のレコードについてプレビューを行う必要がないと判断した場合は、プレビューモードを終了する（ステップS 8 0 7）。他のレコードについてプレビューを行うのであれば、レイアウト編集アプリケーション1 2 1は、他のレコードを選択して再度レイアウト計算を行い、プレビューを行う（ステップS 8 0 6）。

【 0 0 7 6 】

なお、プレビューモードでなく印刷時においては、印刷するレコード全てについて順にレイアウトの計算を行う。したがって、ステップS 8 0 4は存在せず、ステップS 8 0 5は印刷するレコードを全て処理したかの判断を行う。ステップS 8 0 3でレイアウト計算された結果を、描画出力して出力し、プリンタドライバを用いて印刷データとして生成し、プリンタに印刷データが出力される。この場合、全てのレコード（印刷すべく指定された全レコード）について印刷データの出力が終了した時点で本処理を終了することになる。

10

【 0 0 7 7 】

（レイアウト計算方法（詳細））

次に、上記ステップS 8 0 3によるレイアウト計算の詳細について説明する。なお、本フローチャートによる処理を用いることによりコンテンツデータのサイズに基づいて、コンテンツデータが流し込まれた領域（コンテナまたは後述するサブテンプレート）の配置位置や領域サイズを決定することが可能となる。

【 0 0 7 8 】

図8は本実施形態によるレイアウトの優先順位を設定しない場合のレイアウト計算の方法を示したフローチャートである。また、図9はそのときのUI表示例を示した図である。本図はレイアウト計算の処理方法についてのみ説明するためのフローチャートであるため、バリアブルデータプリントの1レコードの印刷/プレビュー時のレイアウト計算方法に相当する。複数レコードの場合は、下記の処理が繰り返されることになる。

20

【 0 0 7 9 】

まず、レイアウト編集アプリケーション1 2 1は、レイアウトを計算するコンテナの集合を求める（ステップS 9 0 1）。レイアウト計算は、関連付けられたコンテナを一つの集合として計算を行う。例えば図10を参照すると、ページ上に4つのコンテナがレイアウトされており、各コンテナに関連付けが設定されている。この場合、コンテナAとコンテナB、そしてコンテナCとコンテナDがリンクによって関連付けされている。したがって、コンテナA、Bが集合1、コンテナC、Dが集合2となる。すなわち、リンクによって接続されたコンテナ群を一つの集合として特定する。

30

【 0 0 8 0 】

前述したように、1 1 0 1はアンカー、1 1 0 2は固定された辺、1 1 0 3はコントローラー、1 1 0 4は可変の辺の変化方向を示している矢印、1 1 0 5は可変の辺、1 1 0 6はリンク、そして1 1 0 7はスライダーを示している。

【 0 0 8 1 】

次に、レイアウト編集アプリケーション1 2 1は、ステップS 9 0 1で求めたコンテナの集合から、レイアウトを計算するために一つを選択する（ステップS 9 0 2）。そして、選択したコンテナの集合について、レイアウトの計算を行う。まず、選択したコンテナの集合に含まれる可変要素である2つのコンテナ（A、B）について、流し込まれるデータの画像サイズもしくはテキスト量から各コンテナがなにも制約を受けない場合の大きさを計算する。具体的には、レイアウト編集アプリケーション1 2 1は、コンテナAが画像データ用コンテナであるか、テキスト用コンテナであるかを判断する。この判断は、前述したように、コンテナに対して設定されている属性により判断できる。

40

【 0 0 8 2 】

次に、レイアウト編集アプリケーション1 2 1は、コンテナAに流し込まれるデータを読み込み、コンテナAが画像データ用コンテナである場合は、その画像データのサイズ（幅、高さのピクセル数、および解像度）がコンテナAの制約を受けない場合の大きさになる。また、コンテナAがテキスト用コンテナである場合は、そのテキストデータも文字数

50

と、コンテナ A のコンテナ属性で指定されているフォントタイプ、フォントサイズ、文字ピッチ、行ピッチなどの文字属性に基づいて、コンテナ A に流し込まれるべきデータ量が計算できる。ここで、テキスト用コンテナの場合は、コンテナ A の縦横比が制約を考えないと決定できないため、制約を当てはめる。

【 0 0 8 3 】

図 10 の例では、コンテナ A は、左上および左下の角にアンカーが設定されているため、高さ（縦方向）が固定となる。よって、レイアウト編集アプリケーション 1 2 1 は、コンテナ A の基本パターンとして設定されている幅（横方向）のコンテナ A に、計算したデータ量（テキスト量）の文字を流し込めるか否かを判断する。すべて流し込めると判断された場合は、コンテナ A は、基本パターンで設定されているサイズ（幅、高さ）に変更はない。

10

【 0 0 8 4 】

また、すべて流し込めないと判断された場合は、コンテナ A は、アンカー設定により高さが固定であるため、横方向に伸びることになる。ここで、レイアウト編集アプリケーション 1 2 1 は、コンテナ A の幅がどれだけになると、計算したデータ量の文字を流し込めるかを計算し、コンテナ A のサイズを算出する。

【 0 0 8 5 】

次に、レイアウト編集アプリケーション 1 2 1 は、レイアウトされるコンテナのサイズが、実際のコンテンツのサイズとできる限り差が少なくなるように、レイアウトの最適化を行う（ステップ S 9 0 3）。レイアウトの最適化は、動的にサイズを変化することが可能なように関連付けられたコンテナにおいて、それぞれに挿入されるコンテンツのサイズとレイアウトされるサイズとの差が、できる限り同じになるように行われる。

20

【 0 0 8 6 】

レイアウト編集アプリケーション 1 2 1 は、ステップ S 9 0 2 で算出したコンテナの集合のサイズ、つまりコンテナ A とコンテナ B とリンク 1 1 0 6（ここでは固定リンク）の合計サイズを求め、この合計サイズと、基本レイアウトにおける当該コンテナの集合のサイズ（図 10 の例ではコンテナ A とコンテナ B のそれぞれのアンカーアイコンの距離に相当する）との差を求める。コンテナ A やコンテナ B の幅が大きくなると前ステップで計算されている場合は、差分値が発生する。レイアウト編集アプリケーション 1 2 1 は、この差分値をコンテナの集合の各要素に均等に分配することでレイアウト調整を行う。以上の処理により、流し込むべきコンテンツデータのコンテンツサイズとコンテンツデータが流し込まれるべきコンテナ（データ領域）とのサイズの差異が小さくなるようにコンテナの配置位置およびサイズを決定することが可能となる。

30

【 0 0 8 7 】

レイアウト編集アプリケーション 1 2 1 は、レイアウトの最適化を行い、ルールに違反していた場合は、再度ルールを違反しないように計算をする（ステップ S 9 0 4）。ここで記述したルールとは、レイアウト作成時にユーザによって設定される制限であり、コンテナのサイズの可変範囲や位置の制限、可変リンクの場合はリンクの長さの変化の制限などである。

【 0 0 8 8 】

ルールを違反しないようにレイアウトが計算されたら、その集合のレイアウトは完成される。そして、ステップ S 9 0 2 ~ S 9 0 4 の処理をページ上のすべての集合について施し、レイアウト編集アプリケーション 1 2 1 は、ページ全体のレイアウトを計算する（ステップ S 9 0 5）。

40

【 0 0 8 9 】

図 9 の（A）～（C）は優先順位の設定なしでのレイアウト時の UI 例である。

【 0 0 9 0 】

図 9 の（A）は、あるレコードが挿入されレイアウトが決定されている状態を表している。1 0 0 1 と 1 0 0 2 はアンカー、1 0 0 3 と 1 0 0 4 は固定された辺、1 0 0 5 は可変の辺、1 0 0 6 は可変の辺の変化方向を示している矢印、1 0 0 8 はリンクをそれぞれ

50

示している。この状態において、レコードを変更し、異なったサイズのコンテンツを挿入する。

【 0 0 9 1 】

図 9 の (B) は (A) の状態に新しいコンテンツのサイズを重ねて示している。1 0 0 9 はそれぞれのコンテナに挿入されるコンテンツサイズを表している。

【 0 0 9 2 】

図 9 の (C) はレイアウト計算された結果を示している。計算後の各コンテナのサイズは、実際挿入されるコンテンツのサイズと同等に差異があるように計算され、且つ前述したルールを違反しないように計算される。図 9 の (C) で示されるように、(B) で示した挿入されるコンテンツサイズ (1 0 0 9) と計算後のコンテンツサイズ (1 0 1 0) は、双方において同等な差異がある。

10

【 0 0 9 3 】

しかし、単純に同等の差異をもってコンテナサイズを変更するだけでは、図 1 0 のような横方向のみのレイアウト計算の場合は問題ないが、図 1 1 のように縦方向および横方向を考慮したレイアウト計算を行わなければならない場合、対応できなくなる可能性がある。

【 0 0 9 4 】

横方向と縦方向の 2 次元的なレイアウトを扱う図 1 1 の場合、図 8 のレイアウト計算方法を用いて、まず横方向に調整を行い、後に縦方向に調整を行う。すなわち、コンテナ 3 2 1 と 3 2 2 の間で、上述した手順で設定値 D に収まるようレイアウト調整を行い、その後、コンテナ 3 2 2 と 3 2 3 の間で設定 G に収まるようレイアウト調整を行えばよい。

20

【 0 0 9 5 】

しかし、横方向に調整を行い、後に縦方向に調整を行うという調整方法では、ユーザが所望とする基本レイアウトからずれたレイアウト結果になることが想定される。

【 0 0 9 6 】

そのため、各コンテナについて、基本パターン (縦横比) からなるべくずれないように、例えば、基本パターンの幅とレイアウト調整後の幅の差の 2 乗と、基本パターンの高さ
とレイアウト調整後の高さの差の 2 乗との和を求め、これをすべてのコンテナについて算出する方法が考えられる。

【 0 0 9 7 】

そして、レイアウト編集アプリケーション 1 2 1 は、各コンテナの 2 乗の和の合計値が最小となるように、それぞれのコンテナの幅と高さを算出してレイアウト調整を行っても良い。この最小 2 乗法を用いてレイアウト調整を行うことにより、レイアウト編集アプリケーション 1 2 1 は、コンテンツデータのコンテンツサイズとコンテンツデータが流し込まれるデータ領域のサイズとの差異が小さくなるようにレイアウト調整を行い、全体としてバランスのとれた最終レイアウトを決定することができる。

30

【 0 0 9 8 】

なお、コンテナ間には後述する可変リンクが設定されているため、流し込まれるコンテンツデータのサイズに応じて後述するようにリンクサイズは変更する。本願記載のレイアウト編集アプリケーションはこのようなレイアウト調整ユニットを備えることにより、割り当てられたデータサイズに応じて複数の部分表示領域 (コンテナ) の配置位置を調整することが可能となる。このようなレイアウト調整方法を用いることにより、タが入れのレイアウトサイズを調整することが可能となる。

40

【 0 0 9 9 】

(長さが可変のリンクの設定)

図 1 2 では可変リンクの設定のためのユーザインターフェースを表している。図 4 と同様にアプリケーションウィンドウ 3 0 1 とツールバー 3 0 3 がある。

【 0 1 0 0 】

図 1 2 の状態では、ドキュメントテンプレート 3 0 9 上にコンテナ 1 2 0 3 とコンテナ 1 2 0 4 が存在する。それぞれのコンテナはアンカーアイコン 1 2 0 1、アンカーアイコ

50

ン 1 2 0 2 と固定された辺 1 2 0 5、辺 1 2 0 6 を含んで構成されている。

【 0 1 0 1 】

コンテナ 1 2 0 3 と 1 2 0 4 の間には可変サイズのリンク 1 2 0 9 があり、コンテナ 1 2 0 3 とコンテナ 1 2 0 4 を結んでいる。コンテナ 1 2 0 3 とコンテナ 1 2 0 4 の間にはリンクが設定されているのでそれぞれの右辺 1 2 0 7 と左辺 1 2 0 8 は点線で表現されている。このため各コンテナにインジケータ 1 2 1 0、インジケータ 1 2 1 1 が表示され、それぞれ辺 1 2 0 7 と辺 1 2 0 8 が可変であることを示している。

【 0 1 0 2 】

また、図 1 4 は、リンク設定手段におけるユーザインタフェース画面であり、リンク 1 2 0 9 の情報をセットするためのダイアログウインドウ 1 4 0 1 の例である。このダイアログは、タイトルバー 1 4 0 2、ツールボタン 1 4 0 3、ダイアログウインドウの開閉を行うボタン 1 4 0 4、各種の情報をセットするエリア 1 4 0 9 で構成されている。このダイアログウインドウではリンクタイプが可変長 (1 4 0 7) のリンクであるか、あるいは固定長 (1 4 0 6) のリンクであるかの択一的な選択を行える。リンクタイプが可変の場合にはリンクの長さの最小値 (Min . D i s t a n c e 1 4 1 0)、最大値 (Max . D i s t a n c e 1 4 1 2)、ならびに基準値 (D i s t a n c e 1 4 1 1) が設定できる。図 1 4 のダイアログ 1 4 0 1 は、たとえば、図 6、図 7 で説明したリンクの設定操作によって 2 つのコンテナ間にリンクを設定した後に、この設定されたリンクをクリック等の操作によって選択したときに表示される。あるいは、リンクを設定した直後に、当該リンクに関するダイアログウインドウ 1 4 0 1 が自動的に表示されるようにしてもよい。ここで各コンテナ間の距離の基準値 1 4 1 1 は、データを流し込んだ際に各コンテナのサイズが変更されない場合に用いられるリンクの長さである。

【 0 1 0 3 】

図 1 3 は、本実施形態のユーザインターフェース 1 0 3 による可変リンクの設定手順を説明するフローチャートである。例えば図 1 1 のコンテナ A とコンテナ B の間に図 6、図 7 で説明した手順でリンクを張ると、まず固定サイズのリンクが張られる。そして、このリンクを選択して図 1 3 に示す処理を実行することにより、当該リンクを固定サイズのリンク 1 1 0 6 (図 1 1) の状態から可変サイズのリンク 1 2 0 9 (図 1 2) へと遷移させることができる。

【 0 1 0 4 】

まず、マウスにより所望のリンク (たとえばリンク 1 1 0 6) を選択状態とし、リンクプロパティを表示させるための所定の操作が行われると、レイアウト編集アプリケーション 1 2 1 のユーザインターフェース 1 0 3 は、リンクプロパティの表示指示の入力として認識する。リンクプロパティの表示指示を認識すると、選択状態のリンク (以下、対象リンクという) に対応したプロパティダイアログウインドウ 1 4 0 1 (図 1 4) が表示される。次に、ユーザインターフェース 1 0 3 は、リンクプロパティを表示する (ステップ S 1 3 0 3)。なお、リンクの選択操作としては、コンテナの基本パターンの設定時と同様に、マウスの右クリックあるいはキーボードの特定のキーの操作等、いかなるものであってもよい。

【 0 1 0 5 】

ステップ S 1 3 0 3 で表示されるダイアログウインドウ 1 4 0 1 には選択されたリンクの現在の状態が示される。本例では、リンク 1 1 0 6 が選択されたので、この段階ではリンクサイズは固定であり、Link Type 1 4 0 5 においては、固定長を示す Fixed Length 1 4 0 6 が選択されている。

【 0 1 0 6 】

このダイアログウインドウ 1 4 0 1 においてリンクを固定サイズから可変サイズに変更するために、Link Type 1 4 0 5 においてリンクサイズを可変に設定するための Flexible Length 1 4 0 7 を選択する。これにより Link Distance 1 4 0 8 内に配置されている Max . D i s t a n c e 1 4 1 2、Min . D i s t a n c e 1 4 1 0、D i s t a n c e 1 4 1 1 が有効になり、数値の設定が可能となる

10

20

30

40

50

。ユーザは、リンクの可変サイズを設定するために、そのリンクの長さの最大値を `Max . Distance 1412` に、最小値を `Min . Distance 1410` に、現在の値を `Distance 1411` に設定することになる。

【0107】

設定を終えると、ユーザは一般的なダイアログウインドウ開閉ボタン `1404` によって当該設定の適用を指示する。ユーザインターフェース `103` はこの指示を検出すると、ステップ `S 1303` からステップ `S 1304` 以降へ処理を進め、当該対象リンクに上記設定状態を反映させる。

【0108】

すなわち、まず、ステップ `S 1304` において、レイアウト編集アプリケーション `121` は、リンクのタイプとして可変リンクが選択されたことを認識する (`S 1304`)。 10

【0109】

ステップ `S 1304` において、可変サイズが指定されていれば、ステップ `S 1305` にて、上記ダイアログウインドウ `1401` によって設定された対象リンクの現在値 (基準値)、最大値、最小値を登録する。その後、ステップ `S 1306` で対象リンクの表示状態を「可変リンク」を表す「点線」とする。この結果、図 `12` のリンク `1209` に示すような状態にリンクのUI表示が変化する。以上のダイアログウインドウ `701` の設定情報はメモリに格納される。

【0110】

なお、`Distance 1411` に設定される現在の値には、現在のレイアウトとして配置されているコンテナの間の距離がデフォルト値として自動的に入力されるようにしてもよい。 20

【0111】

図 `15` は固定サイズのリンクを使用した場合であり、図 `12` に記載されている基本レイアウトにコンテンツを流し込んだレイアウト結果を示している。なお、図 `12` のリンク `1209` は可変リンクとなっているが、固定リンクが設定されているものと想定する。レイアウト計算方法は前述したとおりに従って行われる。

【0112】

例えば図 `12` においてコンテナ `1203` とコンテナ `1204` にそれぞれ違ったサイズのイメージデータ (`1504` および `1505`) が挿入された場合を考える。 30

【0113】

この場合、それぞれのコンテナはデータの大きさを最適とみなし、コンテナ `1203` は挿入されたイメージサイズになる枠 `1504` (最適コンテナサイズ) に近づこうと右方向へ、同様にコンテナ `1204` は挿入されたイメージサイズになる枠 `1505` (最適コンテナサイズ) に近づこうと左方向へサイズを変更しようとする。

【0114】

しかしコンテナ `1203` とコンテナ `1204` はアンカー `1201` とアンカー `1202` によってそれぞれ左辺 `1212` と右辺 `1213` の移動ができず、上記のようにサイズを変更しようとする両者の間隔を狭めるしかない。しかしながら、コンテナ間には固定サイズのリンク `1503` が設定されており、レイアウト計算時にその長さ維持されるため、コンテナ `1203` とコンテナ `1204` のサイズが変更されることになる。 40

【0115】

その結果、コンテナ `1203` とコンテナ `1204` はデータサイズ (`1504` および `1505`) に合わせた最適なサイズを確保することが出来ず、最終的に図 `15` に示すように、最適なサイズ (枠 `1504`、枠 `1505`) よりも小さくなってしまふ (コンテナ `1501` および `1502`)。

【0116】

すなわちリンク `1503` のサイズが固定であるためコンテナ `1501` とコンテナ `1502` は最適サイズを達成できない (図 `15` において、各コンテナ内の一点差線で示した範囲がデータの持つサイズである)。 50

【 0 1 1 7 】

一方、図 1 6 は図 1 5 と同様の状態でリンクを可変サイズにした場合を示している。この場合、上記の例でコンテナ 1 2 0 3 とコンテナ 1 2 0 4 の間には図示の通り可変サイズ 1 2 0 9 のリンクが設定されている。

【 0 1 1 8 】

したがって、コンテナ 1 2 0 3 とコンテナ 1 2 0 4 のサイズが変更される際には、リンクサイズが縮まることでコンテナ 1 2 0 3 とコンテナ 1 2 0 4 のサイズを図 1 5 の例より大きくすることができる。

【 0 1 1 9 】

この結果、挿入されるデータサイズに合わせた最適なサイズを達成できる、あるいはより挿入データサイズ（最適サイズ）に近いコンテナの枠を設定することが出来る。

10

【 0 1 2 0 】

図 1 6 はこの結果を示しており、可変リンク 1 2 0 9 はレイアウト計算の結果、可変リンク 1 6 0 3 に示されるようなサイズ状態となる。なお、この場合コンテナ 1 2 0 3 とコンテナ 1 2 0 4 は、データサイズに合った大きさである最適なサイズ（コンテナ 1 6 0 1 および 1 6 0 2 ）になっている。

【 0 1 2 1 】

（マルチレコード（マルチデータ）機能）

マルチレコードについて概要を説明する。なお、本願では後述するサブテンプレートに従って 1 レコードの情報を配置し、複数のサブテンプレートをフローエリアに順次配置する機能をマルチレコード機能と呼ぶ。また、その動作は、「サブテンプレートをフローエリアに流しこむ」として表現する。

20

【 0 1 2 2 】

通常のドキュメントテンプレートでは、1 レコードが 1 ドキュメントに対応しているが、サブテンプレートに対応するマルチレコードは、基準となる項目が変化するブレイク（区切り）が発生するまでのレコードが流し込まれるため、1 ドキュメント内に複数のレコードが入る。よって、通常のドキュメントテンプレートに対応するバリアブルデータのデータベースと、サブテンプレートに対応するバリアブルデータのマルチレコードのデータベースとは異なるデータベースとすることが望ましい。なお、サブテンプレートの説明については後述する。

30

【 0 1 2 3 】

図 1 7 は、マルチレコードとそのレイアウト方法の概要の一例を示した図である。1 7 0 1 はドキュメント、1 7 0 2 および 1 7 0 5 はページ、1 7 0 3 はサブテンプレート、1 7 0 4 はデータベースを示している。

【 0 1 2 4 】

1 7 0 4 で示すデータベースを見てみると、No 1 ~ 7 のレコードが表示されている。1 レコード 1 ドキュメントのバリアブルプリントでは、7 つレコードがあるので、この場合 7 ドキュメントが作成される。しかし、多レコード 1 ドキュメントのマルチレコードをサポートするバリアブルプリントでは、次のような処理を行う。

【 0 1 2 5 】

まずマルチレコードを指定するカラムをユーザが任意に指定する。指定はたとえばテンプレート作成時になされる。この指定されたフィールドを特定するための情報は、マルチレコード情報としてメモリ 1 3 6 に保持されている。

40

【 0 1 2 6 】

図 1 7 の例ではフィールド名：Name が指定されたものとする。すると、その指定されたフィールドの値が同じレコードは、1 ドキュメントにレイアウトされるレコードであると判定されて、指定されたサブテンプレートを用いてレイアウトされ、1 つのフローエリアに流しこまれる。

【 0 1 2 7 】

図 1 7 の例では、No 1 ~ 4 のレコードはフィールド Name の値が「Tom」である

50

ため、ひとつのフローエリアに流しこまれ、ひとつのドキュメント 1702 上にレイアウトされる。No5～7のレコードは、フィールドNameの値が「Nancy」であるため、「Tom」と同じテンプレートを用いて、しかし異なるドキュメント上にレイアウトされる。すなわち、いったん「Tom」のドキュメントを完結させ、「Nancy」のドキュメント 1705 を作成する。

【0128】

具体的には、たとえばテンプレートデータにおける設定に従って「Tom」のドキュメントにフッタ等を付加（不図示）してから改ページする。これにより「Tom」のドキュメントは完結する。

【0129】

次にそれまでと同じテンプレートデータを用いて、「Nancy」のドキュメントのヘッダ等を新たなページに付加する（不図示）。そしてテンプレートに定義されたフローエリアに、これもサブテンプレート情報として設定されたサブテンプレートを用いて、NAMEフィールドの値が「Tom」のレコードについてレイアウトしてフローエリアに流しこむ。これが通常のマルチレコードである。

【0130】

ただし本実施形態では、NancyとTomとでは、同じテンプレートデータを用いるが、個々のレコードについて同じサブテンプレートを用いるとは限らない。別途指定した条件に応じたサブテンプレートを用いることができるためである。

【0131】

図18は本発明に係るレイアウト編集システムによるテンプレート作成モードの説明UIを示している。

【0132】

レイアウト編集アプリケーション121が実行され、新たなテンプレートの作成の指示をユーザが入力することで、図18の画面（ただし矩形1804、1805、1806は除く。）が表示される。

【0133】

図18のアイコン1803が押下されたことを認識することにより、フローエリア作成モードへと遷移し、ページマージンのエリア1802において、ユーザのマウスのドラッグによりフローエリアが描画される。なお描画されたフローエリアの位置およびサイズは、フローエリア情報としてメモリユニット136に保存される。

【0134】

なお、矩形の1804や1805は、フローエリア1806外に定義されるテキストコンテナやイメージコンテナである。これらのコンテナは、フローエリア内にコンテンツを流しこむ処理とは関係がないが、たとえばヘッダ、フッタとして設けておくことができる。これらコンテナもまた、入力されたその位置及びサイズがメモリユニット136に保存される。

【0135】

フローエリア1806は、それらのコンテナとは色や線種などの矩形UIを異ならせることにより、ユーザに視覚的に認識しやすいようにする。

【0136】

また、作成されたフローエリアはデータベースとの関連付けが行われる。これは、たとえばフローエリアに関連づけられるデータベース名および各レコードにおけるフィールド名等の識別子を入力する処理である。操作上は、たとえばフローエリアをクリックするなどして対象のフローエリアを特定し、そこで一定の操作の流れに従ってこれら必要な情報を入力させる。その情報についてもメモリユニット136に保存される。

【0137】

（フローエリアのプロパティ設定）

図19はフローエリアの各種設定（プロパティ設定）を行う設定ダイアログの例を示している。なお、文中のサブテンプレートおよび条件式については後で説明する。

10

20

30

40

50

【 0 1 3 8 】

まず、ユーザがラジオボタン 1 9 0 2 によってまずフローエリアに関連付けるものがサブテンプレートであるか、それとも条件式であるかを選択する。その後、サブテンプレートの関連づけが選択されている場合には、リストボックス 1 9 0 3 からフローエリアで使用するサブテンプレートをユーザが指定する。あるいは条件式の関連づけが選択されている場合には、リストボックス 1 9 0 4 からフローエリアで使用する条件式をユーザが指定する。

【 0 1 3 9 】

サブテンプレートおよび条件式は、編集集中のドキュメントテンプレート内に、あるいはテンプレート外であっても所定の場所に、それぞれの特有な UI 手段によって別途作成できる。また、条件式は後述する図 2 1 の UI により作成することができる。しかし他の手法としては別のファイルに保存され、この設定ダイアログからはファイルを指定することによって選択指定できるものとしてもかまわない。

10

【 0 1 4 0 】

本実施形態では、ドキュメントテンプレート内にそれぞれ複数のサブテンプレートおよび条件式を作成できるが、リストボックスから選択するという特性上からも、それぞれが固有の名前を持ち識別できるように設計されるべきである。その他に、本設定ダイアログからは、サブテンプレートをフローエリアに流しこむ際のフローの方向（すなわちコンテンツを並べる方向）を選択欄 1 9 0 5 から選択する。コンテンツの間隔は、縦は設定欄 1 9 0 6、横は設定欄 1 9 0 7 で設定することが可能である。

20

【 0 1 4 1 】

以上の設定を確定するには、OK ボタン 1 9 0 8 を押下する。OK ボタンが押されると、そのときの設定がフローエリア情報としてメモリユニット 1 3 6 に保存される。

【 0 1 4 2 】

例えば、対象としているフローエリアがサブテンプレートと関連づけられたのかそれとも条件式と関連づけられたかを示すフラグ情報、そのフラグ情報に応じてサブテンプレート名あるいは条件式名（条件式ファイルの名称）、フロー方向、間隔という情報が保存される。また、キャンセルボタン 1 9 0 9 を押下すると指定された各種設定はすべてキャンセルされ、保存されない。

【 0 1 4 3 】

（サブテンプレートのレイアウト例）

前述したサブテンプレートについて、概要を説明する。図 2 0 はサブテンプレートのレイアウト例（つまりサブテンプレートを用いてデータをレイアウトした例という意味。以下同じ。）を示している。

30

【 0 1 4 4 】

サブテンプレートは、任意数のイメージコンテナ、任意数のテキストコンテナが配置されている領域である。そして、サブテンプレート内に配置されているコンテナは、レコードのどのデータ（フィールド）を割り当てることが設定された状態で、テンプレートとして保存してあるデータを指す。同じレイアウトを一つのドキュメントで数多く使う場合に便利である。

40

【 0 1 4 5 】

サブテンプレート内に配置されるコンテナは、前記イメージコンテナ、テキストコンテナと同様であり、各コンテナ間にはリンクを設定することも可能である。したがって、サブテンプレート内に配置されたコンテナにレイアウトするためのデータを流し込み、図 8 に記載のレイアウト計算方法により、レイアウトサイズを最適に変化させることが可能である。

【 0 1 4 6 】

図 2 0 はサブテンプレートのレイアウト例を示している。2 0 0 2 と 2 0 0 5 はイメージコンテナ、2 0 0 3 と 2 0 0 6 はテキストコンテナ、2 0 0 7 はリンクを示している。

【 0 1 4 7 】

50

図20では、サブテンプレート2001と2002に配置された各コンテナに流し込まれるイメージ、テキストのコンテンツサイズが異なるため、レコードごとにそれぞれ最適なレイアウトを計算してレイアウトを変更している。そのため、同じサブテンプレートを用いているにもかかわらず、サブテンプレート2001と2002では異なるレイアウトとなる。なお、サブテンプレート内に配置されているコンテナのサイズ変更やレイアウト決定に応じて、サブテンプレートのサイズが変更するようにしても構わない。

【0148】

(サブテンプレート条件式)

サブテンプレート条件式は、ひとつのフローエリアに対して複数のサブテンプレートを選択的に割り当てるために用いられる条件を示す。フローエリアには、サブテンプレートあるいは条件式のいずれかひとつを関連付けることができる。

10

【0149】

もし、フローエリアに対してサブテンプレートを関連付けた場合、取得されたレコードは関連付けられたサブテンプレート内に流し込まれることとなる。なお、サブテンプレートを関連付ける際に条件式を設定し、該条件式に適合するレコードのみをフローエリアに関連付けられたサブテンプレートに流し込むようにしても構わない。

【0150】

また、条件式には条件適合時に適用されるサブテンプレートの識別情報を含む。図21は条件式を設定するUI例であり、フローエリアに設定された条件式に適合するレコードのコンテンツデータが、サブテンプレートのコンテナに流し込まれることとなる。そして、レコードのコンテンツデータが流し込まれたサブテンプレートがフローエリアに配置される。

20

【0151】

この入力領域は、条件式に名前を付けるテキストボックス2102と、もっとも基本的な条件判定を構成する要素であるIF文2103とELSE文2104のペアを最低限有している。IF文の入力領域においては、条件の判定対象となるフィールドの選択を行うリストボックス2105と、判定の論理式を設定するリストボックス2106、判定に用いられる値を入力するテキストボックス2107、条件判定が正だったときに動作を決定する、該レコードを挿入するためのサブテンプレートを決定するTHEN__USE処理文2108、また、同じく条件判定結果が不正だったときに該レコードを挿入するためのサブテンプレートを決定するELSE__USE処理文2104からなっている。判定の論理式2106には、EQUAL(等しい)、NOT__EQUAL(等しくない)、LESS__THAN(小さい)、LARGER__THAN(大きい)、START__WITH(で始まる)、END__WITH(で終わる)などさまざまな条件が用意される。

30

【0152】

さらに、複雑な条件式を設定するためには、IF文中に複合条件を設定するためにAND/OR接続文コンボボックス2109を有する。このコンボボックスでANDまたはORの複合条件を選択すると、IF文中に条件行が1行追加される。ANDの場合、複数の条件がすべて正であるときのみ、ORの場合いずれかの条件が正であるときにIF文判定が真であると判定される。

40

【0153】

また、さらに複雑な条件分岐を設定するためには、ELSE__IF文追加ボタン2110を有する。ボタン2110を押下すると、前述IF文とELSE文の間に、あらたにELSE__IF文が追加挿入される。ELSE__IF文の内容と動作は、IF文における内容と同じである。

【0154】

IF文が不正のときにひとつ下のELSE__IF文を判定し、さらに不正のときにもうひとつ下のELSE__IF文を判定し、これを続けて最後にELSE文を判定する、という順に処理を行う。2111は選択されているELSE__IF文を削除する削除ボタンである。ELSE__IF文が多くなった場合、スクロールバー2112を使って条件のリス

50

トを表示することが可能である。いずれかの I F 文あるいは E L S E _ I F 文で真と判定された場合、その U S E 処理文が実行される。U S E 処理文にはサブテンプレート名が入力（選択）される。

【 0 1 5 5 】

このように、条件式を通して、さまざまな値を取るレコードのそれぞれに、条件に合ったサブテンプレートを自動的に選択することができ、ひとつの条件式においてさまざまなレイアウトを実現することが可能になる。この条件式をフローエリアに関連付けることによって、ひとつのフローエリアにおいて、さまざまなレイアウトを持った複数のサブテンプレートをフローエリア内に配置することが可能になる。

【 0 1 5 6 】

ユーザは条件を入力すると、最後に O K ボタン 2 1 1 4 を押す。O K ボタン 2 1 1 4 の押下により、U I で設定されたサブテンプレート条件を示す情報がメモリユニット 1 3 6 に保存される。メモリユニット 1 3 6 に保持されている条件式からコンテンツデータを流し込むべきサブテンプレートを決定することができる。なお、サブテンプレート内部のコンテナサイズの変更に応じて、サブテンプレートの外枠サイズが変更されても、常にサブテンプレートの外枠サイズは固定であっても構わない。

【 0 1 5 7 】

図 2 2 は本実施例で使用される図の概要を示している。

【 0 1 5 8 】

2 2 0 1 はドキュメント、2 2 0 2 はページ、2 2 0 3 はマルチレコードのフローエリア、2 2 0 4 はコンテナ、2 2 0 5 はサブテンプレート示している。

【 0 1 5 9 】

本発明では、2 2 0 5 の各サブテンプレート間のギャップ（距離）を所定のページに配置されたサブテンプレートの配置位置や所定のページを占めるサブテンプレートの割合が、所定のページに設定されている条件を満たすか否かの判断結果に基づいて変更する。また、本願では、通常の印刷設定の場合と製本印刷設定の場合での再レイアウト方法を定義している。さらに、本願では、サブテンプレート間に生じる距離をギャップとして表現している。

【 0 1 6 0 】

図 2 3 は本発明における通常印刷設定時の処理の流れをフローチャートで示している。

【 0 1 6 1 】

まず、レイアウト編集アプリケーション 1 2 1 は、所定の操作により再レイアウト処理の実行内容を設定するため、図 2 4 に記載のダイアログ 2 4 0 1 を表示する。

【 0 1 6 2 】

ここで図 2 4 のギャップ設定ダイアログ 2 4 0 1 について説明する。ギャップ設定ダイアログ 2 4 0 1 には、印刷時にサブテンプレート間にどの程度のギャップ（距離）を保持して印刷を実行するかを設定するための領域 2 4 0 2 がある。図 2 4 では「1 0 m m」と設定されており、サブテンプレートは 1 0 m m のギャップ（距離）を保持してフローエリアに配置される。つまり、レイアウト編集アプリケーション 1 2 1 は、図 2 4 の例では各サブテンプレート間の距離を 1 0 m m として各サブテンプレートの配置位置を決定する。

【 0 1 6 3 】

また、本願ではサブテンプレート間のギャップ（距離）を変更することによる再レイアウト処理を行っているため、ユーザは、ダイアログ 2 4 0 1 を用いて再レイアウト処理を実行するか否かを選択する。具体的には、「固定 2 4 0 3」が選択された場合は、ギャップを変更することによる再レイアウトは行わず、「微調整可 2 4 0 4」が選択された場合、サブテンプレート間のギャップ（距離）を変更することによる再レイアウト処理を実行する。なお、再レイアウト時にサブテンプレート間の距離を変更するが、段階的に変更処理を行うため調整幅 2 4 0 5 を設定する。

【 0 1 6 4 】

また、各サブテンプレート間が最低限の間隔を保てるように最小値（G m i n）を設定

10

20

30

40

50

する。その最小値の設定領域が「最小値 2 4 0 6」である。

【 0 1 6 5 】

ここで具体例を挙げて説明をすると、図 2 4 においては「調整幅 = 1 mm」「最小値 = 5 mm」と設定されているため、サブテンプレート間の距離は最小値 5 mm まで 1 mm 単位で変更する再レイアウト処理が実行される。そして、レイアウト編集アプリケーション 1 2 1 は、ダイアログ 2 4 0 1 に設定された内容を認識して、メモリユニット 1 3 6 に保持する (S 2 3 0 1)。

【 0 1 6 6 】

続いて、レイアウト編集アプリケーション 1 2 1 は、S 2 3 0 1 により再レイアウトの指示が設定されているか否かを判定する (S 2 3 0 2)。詳細には、図 2 4 を用いて設定された内容はメモリユニット 1 3 6 に保持されているため、レイアウト編集アプリケーション 1 2 1 が、メモリユニット 1 3 6 を参照することにより「微調整可 2 4 0 4」が選択されているか否かを判定する。

【 0 1 6 7 】

S 2 3 0 2 により、再レイアウト指示が設定されていると判定した場合、レイアウト編集アプリケーション 1 2 1 は再レイアウトを実行する条件である閾値を設定させるため、図 2 5 (A) に示すダイアログ 2 5 0 1 を表示して、閾値を設定する (S 2 3 0 3)。つまり、後述する S 2 3 0 5 の再レイアウトステップにより領域間の距離の変更処理を実行するための条件となる閾値を設定する。

【 0 1 6 8 】

図 2 5 (A) の設定ダイアログ 2 5 0 1 では、まず対象ページを指定するために 2 5 0 2 および 2 5 0 3 がある。ユーザにより 2 5 0 2 が選択された場合、レイアウト編集アプリケーション 1 2 1 が、最終ページを占めるサブテンプレートの割合が設定された閾値以下であると判定することによりサブテンプレート間の距離を変更する再レイアウト処理を実行する。

【 0 1 6 9 】

また、2 5 0 3 が選択された場合、ページ設定領域 2 5 0 4 が有効となる。レイアウト編集アプリケーション 1 2 1 は、ページ設定領域 2 5 0 4 に入力されたページを占めるサブテンプレートの割合が設定された閾値以下であると判断することによってサブテンプレート間の距離を変更する再レイアウト処理を実行する。なお、図 2 5 では最終ページが選択されているため、ページ設定領域 2 5 0 4 はページ指定ができないように表示される。また、図 2 5 A では、サブテンプレートが占める割合で判定するとの記述をしたが、これに限ることは無く、例えばサブテンプレートの配置位置で判定しても良い。この場合、例えば、閾値として 5 0 % と指定された場合、サブテンプレートの配置位置が指定されたページの 5 0 % の位置 (つまりページを垂直方向に 2 等分した位置) よりも上にあるか下にあるかで判定する。サブテンプレートが上にある場合は、閾値まで至っていないため、レイアウト編集アプリケーション 1 2 1 は、サブテンプレート間の距離を変更する再レイアウト処理を実行すべきと判断する。なお、このようなサブテンプレートが配置された各ページにおいてサブテンプレート領域が占める割合情報やサブテンプレート領域の配置位置情報を配置情報とする。なお、本願では割合情報を用いて再レイアウト処理の実行を判定するものとする。

【 0 1 7 0 】

ダイアログ 2 5 0 1 を用いてページを指定した後、閾値設定領域 2 5 0 5 に対して閾値を設定する。例えば、ダイアログ 2 5 0 1 では「最終ページ」および「1 0 %」が設定されている。そのため、レイアウト編集アプリケーション 1 2 1 は、図 8 によるレイアウト処理を実行した結果、最終ページに占めるサブテンプレートの割合が最終ページの 1 0 % 以下であれば、サブテンプレート間の距離を変更する再レイアウト処理を実行する。

【 0 1 7 1 】

また、図 2 5 (B) のようにユーザインタフェース (U I) 2 5 1 1 を用いて閾値を設定することもできる。U I 2 5 1 1 を用いて閾値を設定するためには矢印 2 5 1 4 をマウ

10

20

30

40

50

ス等により操作して、領域 2 5 1 2 を拡大または縮小することにより閾値を設定するようにしても構わない。なお、図 2 5 (B) に示した U I 2 5 1 1 では、矢印 2 5 1 4 を操作することによる領域 2 5 1 2 のサイズで閾値を設定しており、該矢印の操作に応じて、その時の閾値が領域 2 5 1 3 に表示される。図 2 5 を用いて設定された内容は、メモリユニット 1 3 6 に保持される。

【 0 1 7 2 】

S 2 3 0 3 までの処理により閾値及び再レイアウト方法の設定が終了する。続いてレイアウト編集アプリケーション 1 2 1 は、コンテンツデータをサブプレート内のコンテンツに流し込み、図 8 の調整方法を用いてコンテンツデータのサイズおよび指定された領域間の距離に基づいて、コンテンツデータを流し込むべき領域（例えばサブプレート領域）の配置位置を決定する。図 8 の調整方法によりレイアウトが決定されたサブプレートをドキュメントプレート内のフローエリアに配置することに応じて、図 2 5 で指定されたページを占めるサブプレートの割合を算出し、S 2 3 0 3 で設定した閾値と比較する（S 2 3 0 4）。つまり、レイアウト編集アプリケーション 1 2 1 は、図 8 の調整方法により決定されたサブプレートの配置位置に基づいて各サブプレートが配置されたページ位置を認識する。そして、レイアウト編集アプリケーション 1 2 1 が認識したサブプレートから図 2 5 で指定したページに配置されたサブプレートを特定し、指定ページに配置されたサブプレートの配置情報が図 2 5 で設定した条件である閾値を満たすか否かを判定する。詳細には、レイアウト編集アプリケーション 1 2 1 は、図 2 5 にて指定したページに配置された領域の配置位置または図 2 5 にて指定したページに配置された領域（サブプレート）が占める指定ページに対するサブプレートの割合が、図 2 5 を用いて設定した閾値情報を越えているか否かを判定する。各サブプレートの配置情報はレイアウト編集アプリケーション 1 2 1 を構成するモジュールの 1 つであるレイアウトエンジンを用いて計算することができる。例えばレイアウトエンジンが、各サブプレートの配置位置を座標値等を用いて認識することにより、S 2 3 0 3 にて指定されたページを占めるサブプレート領域の割合を算出することが可能である。算出された指定ページを占めるサブプレートの割合と S 2 3 0 3 で設定した閾値とを比較することにより S 2 3 0 4 が実行される。すなわち、レイアウト編集アプリケーション 1 2 1 は、図 8 のレイアウト処理により決定された各サブプレートの配置位置を認識し、図 2 5 A にて指定されたページに配置されたサブプレートの配置情報を解釈する。その結果、図 2 5 A にて指定されたページに配置されたサブプレートの配置情報が、図 2 4 を用いて設定された条件を満たすか否かを判定することが可能となる。

【 0 1 7 3 】

S 2 3 0 4 の比較処理により、レイアウト編集アプリケーション 1 2 1 が、図 2 5 の設定ダイアログを用いて指定された所定のページを占めるサブプレートの割合が図 2 5 で指定した条件を満たさない（閾値以下である）と判定した場合、S 2 3 0 1 で設定された値を用いてサブプレート間の距離を変更し、再レイアウト処理を行う（S 2 3 0 5）。再レイアウト処理は、図 2 4 のダイアログ 2 4 0 1 にて設定された値を用いて実行される。例えば、最終ページに占めるサブプレートの割合が閾値以下である場合、最終ページには、多くの余白が存在すると考えられる。その際、最終ページの前にあるページ（例えば最終ページの 1 つ前のページ）のサブプレート間の距離を図 2 4 にて設定された調整幅を用いて変更（縮小）する。この際、再レイアウト処理の対象となるページを指定しなければならない。そこで、S 2 3 0 5 による再レイアウト処理が始まる前に、レイアウト編集アプリケーション 1 2 1 は、図 2 6 に記載のギャップ調整範囲指定ダイアログ 2 6 0 1 を表示する。ダイアログ 2 6 0 1 では、再レイアウト処理を実行すべき対象ページを選択するための選択ボタン 2 6 0 2 ~ 2 6 0 4 が表示されている。なお、図 2 6 を用いて実行する設定処理は、S 2 3 0 5 による再レイアウト調整を実行する前に行えば良く、例えば S 2 3 0 1 の段階で設定するようにしても構わない。

【 0 1 7 4 】

本願では再レイアウト方法の一例として以下の 3 つの方法を定義している。1 つ目の再

10

20

30

40

50

レイアウト方法は全ページに存在するサブテンプレート間のギャップを変更する「全ページ」(2602)。2つ目の再レイアウト方法は、ユーザの指定したページ内に存在するサブテンプレート間のギャップを変更する「指定範囲ページ」(2603)。3つ目の再レイアウト方法は、最終ページの1つ前ページ内に存在するサブテンプレート間のギャップを変更する「該当ページ」(2604)である。なお、いずれの方法を用いても、図25Aにて指定したページとは異なるページに属する領域間の距離が変更対象となる。そして、レイアウト編集アプリケーション121は、変更された領域間(サブテンプレート間)の距離を用いてコンテンツデータが流し込まれた領域の配置位置を再決定する。

【0175】

例えば、図26に示す通り「該当ページ」2604が選択された場合、S2305に記載の再レイアウト処理は、最終ページの1つ前のページ内に配置されたサブテンプレート間の距離を図24にて設定された調整幅を用いて変更する。つまり、レイアウト編集アプリケーションは、S2304の判定ステップにより図25にて指定したページに配置された領域の配置情報が図25にて指定した条件である閾値を満たさないと判定された場合(閾値以下であると判定された場合)、図25で指定したページとは異なるページ(図26にて指定したページ)に配置された領域間の距離を変更する。なお、レイアウト編集アプリケーション121は、図24にて設定した調整幅2405の値を用いて領域間の距離を変更し、変更された領域間の距離を用いて再レイアウト処理を行い、サブテンプレートの配置位置を再決定する。

【0176】

S2305の再レイアウト処理の結果、レイアウト編集アプリケーション121は、図26により再レイアウト処理の対象として指定されたページのサブテンプレート間の距離が最小ギャップサイズGmin以上か否かを判定する(S2306)。詳細には、図24を用いて設定された最小ギャップサイズはメモリユニット136に保持されている。そのため、レイアウト編集アプリケーション121が、メモリユニット136を参照することにより最小ギャップサイズを認識することは可能である。また、サブテンプレート間の距離は、レイアウト編集アプリケーション121を構成するレイアウトエンジンが各サブテンプレートの配置位置を認識することにより把握可能である。そして、両者を比較することによりS2306の処理が可能となる。

【0177】

S2306の判定処理によりレイアウト編集アプリケーション121が、図26にて設定された対象ページのサブテンプレート間の距離が最小ギャップサイズ以下であると判定されなかった場合(S2306-No)、S2304に戻る。レイアウト編集アプリケーション121は、S2304にて指定ページ内においてサブテンプレートが占める割合が図25のダイアログにて設定した閾値以上か否かを再度判定する。つまり、レイアウト編集アプリケーション121は、S2305により実行した再レイアウト処理(サブテンプレートの配置位置の最終決定処理)に基づいて、各サブテンプレートが配置されたページ位置を認識する。そして、図25にて指定したページに配置されたサブテンプレートの配置情報が図25にて指定した条件を満たしているか(閾値以上であるか)を再判定する。ここで、指定ページ内を占めるサブテンプレートの割合が閾値以上であると判定された場合

【0178】

しかし、S2304により再度、閾値以下であると判定された場合(S2304-Yes)、S2305へと進み、レイアウト編集アプリケーション121は、図24にて設定された調整幅を用いて図26のダイアログにて設定された対象ページのサブテンプレート間の距離を変更(縮小)する。よって、本フローチャートは、再判定によって図25にて指定した条件を満たすと判定されるか、または図25で指定したページとは異なるページに配置される領域間のサイズが図24にて指定した最小値となるまで該異なるページに配置される領域間のサイズを変更する。つまり、S2305による再レイアウト処理を実行した後に、S2304による再判定において指定されたページに配置されるサブテンプレ

10

20

30

40

50

ートが閾値以上となるか、再レイアウト処理の対象ページのサブテンプレート間の距離が図24にて設定した最小値未満となるまで実行される。

【0179】

以上の処理により、コンテンツデータが流し込まれた領域間の距離を変更することにより、印刷結果の見栄えを改善すると共に、印刷時に無駄なコストが発生することを防止することが可能となる。なお、変更対象となる領域間における領域とは、上述したコンテナ（データ領域）であっても、サブテンプレート領域であっても良い。

【0180】

図27は本発明における製本印刷設定時の処理の流れをフローチャートで示している。まず、図27のフローチャートを説明する前に、製本印刷設定時に発生する再レイアウト処理が必要となる場合の例について図41および42を用いて説明する。まず、図41ではドキュメント内に上述したフローエリアが設定されている場合を想定している。図41の4109から4114のようにフローエリアを有することにより、フローエリアに設定される条件によっては、図41に示すように見開きページにおいて一方のページにサブテンプレートが偏る場合が考えられる。例えば、ページ4104および4105は見開きページとなるが、各ページに設定されているフローエリア4111および4112に設定されている条件が異なることにより、配置されるサブテンプレート数に差異が生じる。図27および図29のフローチャートは、この見開きページにおいて生じるサブテンプレート数の差異を調整するために実行される。

【0181】

また、製本印刷では1枚の記録媒体に対して4ページ分のページ領域が存在する。その結果、製本印刷を指定した場合、必ず4の倍数分のページ領域が存在する。例えば6ページ分の印刷データを製本印刷する場合、2枚の記録媒体上に8ページ分のページ領域が存在するため、ページ4206から4208は図42に示すとおり空白部分が多くなる。

【0182】

図42に示すようにページ4201～4205については1ページ分を埋めるだけのコンテンツが存在し、ページ4206はページの一部にしかコンテンツが存在せず、ページ4207およびページ4208は配置すべきコンテンツがないため空白となっている。この場合、図45のフローチャートによる再レイアウト処理を実行することで、ページ4201～4206の6ページに配置されるべきコンテンツをページ4201～4208において指定された範囲内のページに対して均等に配置することが可能となる。その結果、コンテンツの平準化を図ることが可能となり、図42のように余白ページが大部分を占めるページ4206や空白ページ4207およびページ4208を有効的に活用することが可能となる。

【0183】

図27のフローチャートについて説明する。まず、レイアウト編集アプリケーション121は、現在処理対象としているドキュメントの印刷設定（またはプレビュー設定）に対して製本印刷が設定されているか否かを判定する（S2701）。詳細には、ドキュメント毎に設定されている印刷設定は、メモリユニット136に保持されている。そのため、レイアウト編集アプリケーション121が、メモリユニット136を参照することによりS2701の処理が可能となる。

【0184】

S2701の判定の結果、製本印刷が設定されていると判定された場合（S2701-Yes）、レイアウト編集アプリケーション121は、対象としているドキュメントにフローエリアが指定されているか否かを判定する（S2702）。なお、レイアウト編集アプリケーション121は、ユーザにより指定されたフローエリアを認識して、メモリユニット136に保持する。そのため、レイアウト編集アプリケーション121が、メモリユニット136を参照することによりS2702の判定処理が可能となる。

【0185】

S2702の判定処理により、フローエリアが設定されている判定された場合（S27

10

20

30

40

50

02 - Yes)、S2703へと進み、フローエリアが設定されていないと判定された場合(S2702 - No)、図45に記載のフローチャートへと処理を進める。なお、図45のフローチャートについては後述する。

【0186】

S2702の判定処理により、フローエリアが設定されていると判定された場合(S2702 - Yes)、レイアウト編集アプリケーション121は、所定の操作を認識して図24のダイアログ2401を表示する(S2703)。なお図24については上述したものと同一であるため詳細な説明については省略する。レイアウト編集アプリケーション121は、ダイアログウィンドウ2401に設定された内容を認識して、メモリユニット136に保持する。

10

【0187】

続いて、レイアウト編集アプリケーション121は、S2703により表示した図24のダイアログ2401により再レイアウトの指示がなされたか否かを判定する(S2704)。詳細については、図23のS2702と同等であるため省略する。

【0188】

S2704により再レイアウトが指定されたと判定した場合は、S2705に進み、再レイアウトが指定されていない場合は本フローチャートの処理を終了する。

【0189】

S2704により、レイアウト編集アプリケーション121が、再レイアウトが設定されていると判定した場合、再レイアウトを実行する条件である閾値を設定させるため、閾値設定ダイアログを表示して、閾値を設定する(S2705)。なお、S2705により表示される閾値設定ダイアログは、基本的には図25に記載のダイアログと同様であるが、本フローでは製本印刷すべきページ群が判定対象となるため、図25に入力された閾値情報はページ群に対して設定される。

20

【0190】

続いて、レイアウト編集アプリケーション121は、コンテンツデータをサブテンプレート内のコンテナに流し込み、図8の調整方法を用いてサブテンプレートの配置位置を決定する。配置位置決定処理が終了したサブテンプレートをドキュメントテンプレート上のフローエリアに配置することに応じて、各ページ内にレイアウトされたサブテンプレートの占める割合を算出し、S2705で設定した閾値と比較する(つまり、見開きページを構成する2つのページの少なくともいずれか一方のページにおいて、製本印刷すべきページ群に設定されている条件を満たさないページがあるか否かを判定する)。製本印刷の場合、1ページ目が見開きページとなることはないので、本フローチャートにおける各処理の対象となるのは2ページ目以降となる。

30

【0191】

見開きページの各ページにおいてサブテンプレートが占める割合を算出し、S2705により設定された閾値と各ページに占めるサブテンプレートの割合を比較する。比較処理の結果、ページ内を占めるサブテンプレートの割合が閾値以下であると判定されるページがある場合(つまり、製本印刷すべきページ群に設定された条件(閾値)を満たさないページがあると判定された場合)、図28に示すギャップ調整確認ダイアログが表示される。

40

【0192】

ダイアログ2801には再レイアウト処理を実行するか否かを選択するための選択ボタン2802および2803がある。ここで、選択ボタン2802が選択され、OKボタン2804が押された場合、レイアウト編集アプリケーション121は、見開きページに配置されるべきサブテンプレートの情報量を均等にすべく、見開きページに配置されるサブテンプレート間の距離を変更する再レイアウト処理を実行する(S2706)。なお、S2706の詳細については、図29に記載する。

【0193】

レイアウト編集アプリケーション121は、S2706により計算されたサブテンプレ

50

ート間の距離を用いて、再レイアウト処理を実行する (S 2 7 0 7)。

【 0 1 9 4 】

図 2 9 は本発明における製本印刷設定時に見開きページのサブテンプレート量 (情報量) を均等に配置するために変更すべきサブテンプレート間の距離を算出する処理の流れをフローチャートで示している。

【 0 1 9 5 】

レイアウト編集アプリケーション 1 2 1 は、フローエリアを有する全てのページ z ($1 < z < n$) の処理が終了するまで以下の処理を繰り返す (S 2 9 0 1)。なお、 z は任意のページ数を指し、 n はフローエリアを有するページの最終ページを指す。

【 0 1 9 6 】

まず、レイアウト編集アプリケーション 1 2 1 は、任意のページ z が奇数ページであるか否かを判断する (S 2 9 0 2)。詳細には、各ページのページ属性情報をメモリユニット 1 3 6 に保持しているため、レイアウト編集アプリケーション 1 2 1 がページ属性情報を参照することにより、現在対象としている z ページ目がどのページであるかを認識可能であるため、S 2 9 0 2 の判断処理が可能となる。

【 0 1 9 7 】

S 2 9 0 2 の判断処理により、 z ページ目が奇数ページであると判断された場合、レイアウト編集アプリケーション 1 2 1 は、 $z - 1$ または z ページ目に占めるサブテンプレートの割合が、上述した閾値以下であるか否かを判定する (S 2 9 0 3)。詳細には、図 2 3 の S 2 3 0 4 に記載の処理と同等であるため、詳細な説明については省略する。

【 0 1 9 8 】

S 2 9 0 3 により、 $z - 1$ または z ページ目に占めるサブテンプレートの割合が、設定された閾値以下であると判定された場合、レイアウト編集アプリケーション 1 2 1 は、 z ページ目および $z - 1$ ページ内のサブテンプレートの個数 $C n t$ を取得する (S 2 9 0 4)。詳細には、レイアウト後のレイアウト結果はメモリユニット 1 3 6 に保持されており、レイアウト編集アプリケーション 1 2 1 は各サブテンプレートの位置情報を把握することで、S 2 9 0 4 の処理が可能となる。

【 0 1 9 9 】

続いてレイアウト編集アプリケーション 1 2 1 は、ページ z および $z - 1$ に配置されている各サブテンプレートのサイズ $S j$ ($1 \leq j \leq C n t$) を取得する (S 2 9 0 5)。各サブテンプレートのサイズについては上述した図 8 のレイアウト調整により算出したレイアウト結果をメモリユニット 1 3 6 に保持している。そのため、レイアウト編集アプリケーション 1 2 1 は各サブテンプレートサイズを取得することが可能となる。

【 0 2 0 0 】

レイアウト編集アプリケーション 1 2 1 は、S 2 9 0 5 により取得したページ z および $z - 1$ のページ内における各サブテンプレートサイズの合計値 $S u m_a l l$ を算出する (S 2 9 0 6)。なお、S 2 9 0 6 により算出した各サブテンプレートサイズの合計値はページごとにメモリユニット 1 3 6 に保存される。

【 0 2 0 1 】

処理 S 2 9 0 7 から S 2 9 1 4 は、見開きページである z ページと $z - 1$ ページ内のサブテンプレート量 (情報量) をほぼ等しくするために、基準となる中間サブテンプレートを決定する処理である。つまり、 $z - 1$ ページに配置すべきサブテンプレートと z ページに配置すべきサブテンプレートの境目となるサブテンプレートを決定する。

【 0 2 0 2 】

レイアウト編集アプリケーション 1 2 1 は、サブテンプレートの合計を格納するための変数 $S u m$ を 0 で初期化する (S 2 9 0 7)。

【 0 2 0 3 】

レイアウト編集アプリケーション 1 2 1 は、 z ページおよび $z - 1$ ページに配置されたサブテンプレートにおいて中間サブテンプレートとなる $S p l i t$ が決定するまで S 2 9 0 8 ~ S 2 9 1 3 の処理を繰り返す (S 2 9 0 8)。

10

20

30

40

50

【0204】

レイアウト編集アプリケーション121は、現在注目しているサブテンプレートjの1つ前までのサブテンプレートサイズの合計を格納しておくための変数Tempを定義し、現在注目しているサブテンプレートjの1つ前のサブテンプレートj-1までのサブテンプレートサイズの合計を格納する。次にレイアウト編集アプリケーション121は、Sumにサブテンプレートjのサイズをプラスする(S2909)。なお、S2909で計算された値はメモリユニット136に保持される。

【0205】

続いて、レイアウト編集アプリケーション121は、サブテンプレートjのサイズをSumにプラスした後(S2909の処理後)、S2906で算出したzページ及びz-1ページに配置される全てのサブテンプレートのサイズの合計値Sum_allを2で割った値とS2909により算出したSumの値が等しいか否かを判定する(S2910)。詳細には、S2906により算出したzページおよびz-1ページ内のサブテンプレートの合計サイズ(Sum_all)はメモリユニット136に保持されている。そのため、レイアウト編集アプリケーションは、Sum_allを2で割った値とS2909により算出したSumの値を比較することは可能となる。

【0206】

S2910による処理の結果、S2909により算出したSumとS2906の算出結果であるSum_allを2で割った値が等しいと判定された場合、レイアウト編集アプリケーション121は、サブテンプレートjを中間サブテンプレートSplitとしてメモリユニット136に格納する(S2911)。つまり、現在着目しているサブテンプレートjがzページ及びz-1ページにおける全てのサブテンプレートの中間に該当するサブテンプレートとなる。なお、S2911の処理の結果、中間サブテンプレートSplitが決定された場合、その旨をしめすフラグ等を立てておく。これにより、レイアウト編集アプリケーションは、S2912およびS2913の処理を省略して、S2915の処理へと進むこととなる。

【0207】

また、S2910の判定の結果、SumとSum_all/2の値が等しくないと判定された場合、レイアウト編集アプリケーション121はS2909で算出したSumがSum_all/2よりも大きいかどうか判断する(S2912)。S2912により、SumがSum_all/2よりも小さい場合、現在注目しているサブテンプレートjは、中間となるサブテンプレートの手前のサブテンプレートに該当すると判断される。そこで、次のサブテンプレートj+1において同様の処理を繰り返すため、S2908へ戻る。また、S2912において、SumがSum_all/2よりも大きい場合、レイアウト編集アプリケーション121は、サブテンプレートj-1までのサイズの合計とサブテンプレートjまでのサイズの合計を比較する。この比較処理により、Sum_all/2に値が近くなるような中間サブテンプレートSplitを決定する(S2913)。つまり、S2912により条件を満たすと判定された場合、現在注目しているサブテンプレートjは中間となるサブテンプレートを越えていると判断され、S2913の処理が実行される。詳細には、Sum_all/2とサブテンプレートjまでのサイズの合計Sumとの差の絶対値とSum_all/2とサブテンプレートj-1までのサイズの合計Tempとの差の絶対値の小さい方を判定し、中間サブテンプレートSplitとして、メモリユニット136に格納する。

【0208】

この処理により、S2912の判断条件を満たす場合、現在着目しているサブテンプレートを中間サブテンプレートとするのか、それとも現在着目しているサブテンプレートの1つ前にあるサブテンプレートを中間サブテンプレートとするのかを決定する。よって、S2911またはS2913により中間サブテンプレート(Split)が決定された場合、レイアウト編集アプリケーション121は、S2908~S2914の処理を終了して、S2915へと進む。

10

20

30

40

50

【0209】

レイアウト編集アプリケーション121は、S2908～S2914の処理により中間サブテンプレートSplitを決定した場合、zページ及びz-1ページにおけるサブテンプレート間の距離を決定する。詳細には、レイアウト編集アプリケーション121は、現在対象としているフローエリアのサイズFsizeを取得する。フローエリアのサイズ(Fsize)は、フローエリア設定時にメモリユニット136に保存されるため、メモリユニット136を参照することにより取得可能である。続いて、レイアウト編集アプリケーション121は、z-1ページ内のサブテンプレートの合計値Sum1(中間サブテンプレートまでのサブテンプレートサイズの合計値)及びzページ内のサブテンプレートサイズの合計値Sum2(中間サブテンプレートの次のサブテンプレートからzページの最も下の位置に配置されるサブテンプレートまでのサイズ合計値)を算出する(S2915)。S2915の処理は、S2905の処理を実行するために用いた方法を適用することに実現可能である。

10

【0210】

次に、レイアウト編集アプリケーション121は、各ページにおけるサブテンプレート間の距離を設定するべく、算出処理を実行する(S2916)。本発明では変更すべきサブテンプレート間の距離を算出する一例として{Fsize(z-1またはzページ内のサブテンプレートの合計サイズSum1、Sum2)/ギャップ個数}という算出式を用いている。つまり、「Fsize(z-1またはzページ内のサブテンプレートの合計サイズSum1、Sum2)」により、フローエリアにおいてサブテンプレート間の距離の合計値を算出する。そして、該算出された合計値を各ページが有するサブテンプレート間の個数で均等に割り振ることにより、サブテンプレート間の距離を算出することが可能となる。なお、サブテンプレート間の個数は各ページに配置されるサブテンプレートの個数+1の値であるため算出可能である。

20

【0211】

レイアウト編集アプリケーション121は、マルチレコードのフローエリア内における全てのページに対し上記処理が終了したと判定した場合(S2917)、図29におけるフローチャートの処理を終了する。

【0212】

上記図29のフローチャートを用いることにより、S2903により見開きページのいずれか一方でも、製本印刷を実行すべきページ群に設定された条件を満たさないページがあると判定された場合、製本印刷における見開きページ単位に情報量が均等になるようにコンテンツデータが流し込まれる領域(サブテンプレート)間の距離を変更することが可能となる。よって、見開きページ間で見栄えの良い製本印刷物を得ることが可能となる。

30

【0213】

続いて図27のS2702にてフローエリアなしと判定された場合(S2702-No)に実行すべき図45のフローチャートについて説明する。レイアウト編集アプリケーション121は、再レイアウト処理の対象となるページ群を指定する(S4501)。なお、S4501により指定された指定範囲内のページがメモリユニット136に保持される。レイアウト編集アプリケーション121は、再レイアウト処理を実行する条件である閾値を設定させるためのダイアログを表示して、指定範囲内のページ群に対して閾値を設定する(S4502)。なお、S4502により設定される閾値は、S4501にて指定された指定範囲内のページ群に対して設定される。閾値設定ダイアログは、基本的には図25に記載のダイアログと同様である。なお、S4502にて設定された閾値はメモリユニット136に保持される。

40

【0214】

続いてレイアウト編集アプリケーション121は、S4502において設定された条件である閾値を満たさないページが指定範囲内にあるか否かを判定する(S4503)。詳細には、レイアウト編集アプリケーション121は、コンテナに対してコンテンツデータを流し込み、図8の調整方法を用いてレイアウトを決定する。該レイアウト調整が終了し

50

たコンテナをドキュメントテンプレート上に配置することに応じて、指定された範囲内において、条件を満たさないページがあるか否かを判定する。つまり、レイアウト編集アプリケーション 121 は、S 4 5 0 2 にて設定した閾値以下のサブテンプレートしかレイアウトされていないページがあるか否かを判定する。S 4 5 0 3 の処理により、製本印刷が設定されていると判定された場合、製本印刷すべきページであり、指定範囲内のページ群に配置されたサブテンプレート領域の配置情報が指定範囲内のページ群に設定されている条件を満たすか否かをページ毎に判定することが可能となる。

【 0 2 1 5 】

S 4 5 0 3 により条件を満たさないページがないと判定された場合 (S 4 5 0 3 - N o)、レイアウト編集アプリケーション 121 は再レイアウト処理を実行する必要がないと判定して図 4 5 のフローチャートによる処理を終了する。

10

【 0 2 1 6 】

また、S 4 5 0 3 により条件を満たさないページがあると判定された場合 (S 4 5 0 3 - Y e s)、レイアウト編集アプリケーション 121 は、S 4 5 0 1 にて指定されたページ内のサブテンプレート数を計算する (S 4 5 0 4)。詳細には、レイアウト編集アプリケーション 121 は図 8 の調整方法によってレイアウトされたレイアウト結果を保存しており、各コンテンツが配置されたページ領域を有するサブテンプレート情報を把握することが可能である。よって、レイアウト編集アプリケーション 121 は、S 4 5 0 2 にて指定されたページおよび該指定されたページ内に配置されるサブテンプレート数をメモリユニット 136 を参照することにより認識する。

20

【 0 2 1 7 】

レイアウト編集アプリケーション 121 は、S 4 5 0 4 により計算した指定されたページ内にレイアウトされたサブテンプレート数を S 4 5 0 1 にて指定されたページ数を用いて割り算処理を実行する (S 4 5 0 5)。例えば、図 4 2 にてページ 4 2 0 2 ~ 4 2 0 7 が指定された場合、該指定ページ内にレイアウトされているサブテンプレート数は 13 個である。よって、 13 (コンテナ数) \div 6 (指定されたページ数) の計算処理を実行する。

【 0 2 1 8 】

レイアウト編集アプリケーション 121 が、S 4 5 0 5 による計算の結果、割り切れたと判定した場合 (S 4 5 0 6 - Y e s)、レイアウト編集アプリケーション 121 は、S 4 5 0 6 による計算結果により導かれた数のコンテナを指定されたページ内の各ページに配置する (S 4 5 0 8)。

30

【 0 2 1 9 】

また、S 4 5 0 6 による計算の結果、割り切れないと判定した場合 (S 4 5 0 6 - N o)、レイアウト編集アプリケーション 121 は、どのページに対して優先的に余りのコンテナをレイアウトすべきかを選択するための UI を表示する。レイアウト編集アプリケーション 121 は、該 UI により選択されたページから優先的に余りのサブテンプレートを配置する (S 4 5 0 7)。例えば、前から優先的にレイアウトすべき指定がされた場合、指定ページの前半ページから優先的に余ったサブテンプレートを配置する。図 4 2 では、 13 (サブテンプレート数) \div 6 (指定されたページ数) の計算により、各ページには 2 つのサブテンプレートが均等にレイアウトされると共に、余った 1 つのコンテナは選択された前半ページからレイアウトされる。その結果が、図 4 4 に示す図である。なお、S 4 5 0 8 および S 4 5 0 7 の処理により決定された各ページ領域に配置すべきサブテンプレート数は、当初各ページに配置されていたサブテンプレート数とは変更されている。例えば、当初は図 4 2 のような形式でページ 4 2 0 2 から 4 2 0 5 に 3 つのサブテンプレートが配置されており、ページ 4 2 0 6 には 1 つのサブテンプレートが配置されていた。これに対し、図 4 5 のフローチャートによる処理を実行することで、図 4 4 のように指定範囲内の各ページに配置されるサブテンプレート数に変更されている。この変更に応じて、指定範囲内のページ群に配置されている複数のサブテンプレートを指定範囲内のページ群における各ページに対して均等に割り付けるべく、サブテンプレート間の距離を変更して、

40

50

配置位置を決定する。

【0220】

製本印刷を実行する場合、上述したように、1枚の記録媒体に対して4ページ分のページ領域が存在するため、必ず4の倍数分のページ領域が存在する。これに対して製本印刷時に生成される印刷データを各ページに配置した結果、必ずしも4の倍数分のページ領域に配置されず空白ページ領域が生じることにより、見栄えが悪くなる可能性がある。このような課題に対して図45の処理を用いることで、製本印刷時の所定の範囲のページに対して均等にコンテンツデータを流し込むべき領域（例えばサブテンプレート領域）を配置することができ、所定の範囲のページを通して見栄えの良い製本印刷物を得ることが可能となる。

10

【0221】

図30は本発明におけるサブテンプレート間に優先順位が設定されている場合の処理の流れをフローチャートで示している。ユーザによっては、サブテンプレート間の距離を変更しても問題ないと考えているサブテンプレート間と、可能であれば変更して欲しくないとするサブテンプレート間がある。よって、図30では、サブテンプレート間に優先順位を設定し、該優先順位が高く設定されたサブテンプレート間はできる限りサイズ変更しないように処理を進める。

【0222】

まず、レイアウト編集アプリケーション121は、所定の操作により図24のダイアログ2401を表示する。なお図24については上述したものと同等であるため詳細な説明については省略する。レイアウト編集アプリケーションは、ダイアログウィンドウ2401に設定された内容を認識して、メモリユニット136に保持する(S3001)。

20

【0223】

続いて、レイアウト編集アプリケーション121は、S3001により再レイアウトの指示がなされたか否かを判定する(S3002)。なお、S3002の処理はS2302と同等であるため、詳細な説明については省略する。

【0224】

S3002により、再レイアウト指示がなされたと判定した場合、ダイアログを用いてサブテンプレート間に優先度を設定させるため、レイアウト編集アプリケーション121は、図31Aに示すダイアログ3101を表示して、優先度を設定する(S3003)。図30では優先度を $P_i (1 \leq i \leq n)$ で示しており、 i の値が大きいほど優先度が高いことを示している。なお、各サブテンプレート間と対応付けて設定された優先度はメモリユニット136に保存される。つまり、レイアウト編集アプリケーション121は、優先的に領域間の距離を変更すべき領域間に対して低優先度を設定する。

30

【0225】

続いて、ユーザインターフェースを用いて再レイアウトを実行する条件である閾値を設定させるため、レイアウト編集アプリケーション121は図25に示すダイアログ2501を表示して、閾値を設定する(S3004)。なお、図25については上述したものと同等であるため詳細な説明については省略する。

【0226】

続いてレイアウト編集アプリケーション121は、各サブテンプレート間を優先度の低い順に調整するために、変数 i の値を0で初期化する(S3005)。

40

【0227】

S3005までの処理により閾値及び再レイアウト方法の設定が終了する。レイアウト編集アプリケーション121は、コンテンツデータをサブテンプレート内のコンテナに流し込み、図8の調整方法を用いてサブテンプレート内においてレイアウト調整を行う。レイアウト編集アプリケーション121は、該レイアウト調整が終了したサブテンプレートをドキュメントテンプレート上に配置することに応じて、図25で指定されたページ内にレイアウトされたサブテンプレートの占める割合を算出し、S3004で設定した閾値と比較する(S3006)。S3006の処理は、図23のS2304と同等であるため詳

50

細な説明は省略する。

【0228】

S3006において比較した結果、特定ページ内におけるサブテンプレートの占める割合が閾値以下の場合、レイアウト編集アプリケーション121は、各サブテンプレート間の距離を優先度順に変更するために、変数*i*の値に対してインクリメント処理を施す(S3007)。レイアウト編集アプリケーション121は、優先度の低い順に各ギャップサイズを再レイアウト対象に設定する(S3008)。具体的には、初めてS3008の処理を実行する場合には、「*i* = 1」の場合であるため、優先度 $P_i = 1$ と設定されたサブテンプレート間が再レイアウト処理の調整対象として認識される。各サブテンプレート間に設定されている優先度はメモリユニット136に保持されているため、レイアウト編集アプリケーション121が参照することにより、各サブテンプレート間の優先度を認識することは可能である。なお、S3008により認識されたサブテンプレート間は、一時的にメモリユニット136に保持される。

10

【0229】

再レイアウト対象のサブテンプレート間を決定後、レイアウト編集アプリケーション121は、該サブテンプレート間の距離を図24で指定した調整幅を用いて再レイアウト処理する(S3009)。詳細には、再レイアウトの対象となるサブテンプレート間および、図24を用いて設定された再レイアウト処理の処理内容はメモリユニット136に保持されている。そのため、レイアウト編集アプリケーション121がメモリユニット136を参照することによりS3009は実行可能となる。

20

【0230】

レイアウト編集アプリケーションは、S3009による再レイアウト結果を認識して、図24を用いて設定した最小値(G_{min})以下であるか否かを判断する(S3010)。サブテンプレート間の距離が最小ギャップサイズ G_{min} 以上の場合は、処理S3006に戻り、特定ページ内におけるサブテンプレートの占める割合が閾値以上であるか否かを再度判定する。ここで、特定ページ内におけるサブテンプレートの占める割合が閾値以上であると判定された場合、処理は正常に終了する。しかし、S3006により再度閾値以下であると判定された場合、S3007へと進み、次に優先度の高いサブテンプレート間対象として、再レイアウト処理が実行される。以上の処理を、図25で指定したページを占めるサブテンプレートの割合が閾値以上になるか、またはS3001で設定した最小値未満になるまで繰り返す。

30

【0231】

図30に記載の処理を用いることにより、優先度の低いサブテンプレート間から再レイアウト処理の編集対象となるため、ユーザの意図を反映したレイアウトを実現することが可能となる。また、図30の処理では優先度の低いサブテンプレートから再レイアウト処理の対象となとしたが、それに限ることは無く、サブテンプレート間の優先度を比較して、優先度が低いと判定されたサブテンプレート間の距離を変更するようにしても構わない。

【0232】

図31A～Bはサブテンプレート間の優先度設定ダイアログを示している。

40

【0233】

図31Aにおける3104は優先度ON設定の場合の優先度レベルを設定するためのドロップダウンメニューを示している。3104を指示することにより3105に優先度を示す値が表示される。なお、優先度の設定のON/OFFはフローエリア内に適用されるものであり、優先度設定をONにした場合はすべてのサブテンプレートで優先度設定がONとなり、優先度 P_1 で初期化される。各サブテンプレート間の優先度を変更する場合は、優先度設定ウィンドウ3101で優先度を変更する必要がある。

【0234】

図31Aを用いて優先度を設定する例を図31Bに示す。

【0235】

50

サブテンプレートが配置されたドキュメントテンプレートが図3 1 Bのように表示されているため、ユーザは優先度を設定したい所望のサブテンプレート間をマウス等を用いて指定する。例えば図3 1 Bの例でいえば、ページ3 1 1 1～ページ3 1 1 3で1つのドキュメントが構成されているとする。ユーザはページ3 1 1 2のサブテンプレート3 1 1 4および3 1 1 5の間を指定した状態で所定の操作をすることにより優先度設定ダイアログ3 1 0 1が表示される。

【0 2 3 6】

図3 1 Aにおける優先度を設定することを示すボックス3 1 0 2をチェックし、3 1 0 4を用いて優先度を設定することにより、サブテンプレート3 1 1 4および3 1 1 5の間に対して優先度が設定される。なお、優先度は重複しないようにすることも可能であるし、同じ優先度を設定できるようにしても構わない。なお、同じ優先度が設定されたサブテンプレート間は、同時に再レイアウト処理の対象となる。また、ここで設定された優先度はメモリユニット1 3 6に保持される。

10

【0 2 3 7】

図3 2は、図2 6の2 6 0 4「全ページ」が選択された場合の再レイアウトの処理の流れをフローチャートで示している。

【0 2 3 8】

レイアウト編集アプリケーション1 2 1は、図2 4により設定された調整幅を用いて再レイアウト処理を実行する。すなわち、マルチレコードのフローエリア内における全てのサブテンプレート間の処理が終了するまで(S 3 2 0 1)、サブテンプレート間の距離を図2 4で指定された調整幅ずつ変更して(S 3 2 0 2)。詳細には、レイアウト編集アプリケーション1 2 1を構成するレイアウトエンジンが再レイアウト処理の対象となっているサブテンプレート間を生成している2つのサブテンプレートのいずれか一方を調整幅分だけ移動することにより実現される。なお、サブテンプレート間を生成する両方のサブテンプレートを調整幅の半分ずつ移動することにより、S 3 2 0 3の処理を実現しても構わない。

20

【0 2 3 9】

図3 3は、図2 6の2 6 0 5「指定範囲ページ」が選択された場合の再レイアウトの処理の流れをフローチャートで示している。

【0 2 4 0】

レイアウト編集アプリケーション1 2 1は、ドキュメントテンプレート上のフローエリア内における全てのサブテンプレート間への処理が終了するまで、以下の処理を続ける(S 3 3 0 1)。

30

【0 2 4 1】

レイアウト編集アプリケーション1 2 1は、現在対象としているサブテンプレート間が図2 6のダイアログを用いて指定したページ内に存在するサブテンプレート間か否かを判断する(S 3 3 0 2)。詳細には、サブテンプレート間を生成しているサブテンプレートの位置を把握することにより、現在対象としているサブテンプレート間がどのサブテンプレートにより生成されているギャップかを判定する。その際に、メモリユニット1 3 6に保持されているサブテンプレートの位置情報を用いてサブテンプレート間の位置を特定することができる。この特定処理によりS 3 3 0 2を実現することが可能となる。

40

【0 2 4 2】

S 3 3 0 2の判定の結果、指定されたページ内のサブテンプレート間であると判定された場合、レイアウト編集アプリケーション1 2 1は、図2 4において指定された調整幅を用いてサブテンプレート間の距離を変更する(S 3 3 0 3)。詳細についてはS 3 3 0 2と同様であるため省略する。サブテンプレートが指定ページ内に存在するサブテンプレートでない場合は、そのまま次のサブテンプレート間の処理へ進む。全てのサブテンプレート間について処理終了後、図3 3におけるフローチャートの処理を終了する。図3 3の処理により、指定されたページに配置されている領域が生成する領域間の距離を変更することにより、ユーザの意図した再レイアウト処理を実行することが可能となる。

50

【 0 2 4 3 】

図 3 4 は、図 2 6 の 2 6 0 6 「該当ページ（最終ページの 1 つ前のページ）」が選択された場合の再レイアウトの処理の流れをフローチャートで示している。

【 0 2 4 4 】

レイアウト編集アプリケーション 1 2 1 は、各コンテンツデータが流し込まれレイアウト済みのサブテンプレートフローエリアに配置することにより得られるレイアウト後のページ数 n を取得する (S 3 4 0 1)。レイアウト後のレイアウト情報はメモリユニット 1 3 6 に保持されているため、レイアウト編集アプリケーション 1 2 1 がメモリユニット 1 3 6 を参照することにレイアウト後のページ数を取得することが可能となる。

【 0 2 4 5 】

続いてレイアウト編集アプリケーション 1 2 1 は、S 3 4 0 1 によりページ数取得後、最終ページの 1 つ前のページ $n - 1$ 内における全てのサブテンプレート間に対する処理が終了するまで (S 3 4 0 2)、 $n - 1$ ページ内のサブテンプレート間の距離を図 2 4 で設定された調整幅を用いてサブテンプレート間の距離を変更する (S 3 4 0 3)。なお S 3 4 0 3 の処理は、図 3 2 の S 3 2 0 2 と同様であるため詳細な説明は省略する。

【 0 2 4 6 】

レイアウト編集アプリケーション 1 2 1 は、最終ページの 1 つ前のページ ($n - 1$ ページ) 内における全てのサブテンプレート間について S 3 4 0 3 の処理を実行することにより図 3 4 のフローチャートの処理を終了する。

【 0 2 4 7 】

図 3 5 は本発明の通常印刷設定時における図 8 のフローチャートを用いて説明した最初のレイアウト調整後のイメージを示している。

【 0 2 4 8 】

3 7 0 1 は、ドキュメントテンプレート上に定義されたコンテナに対してコンテンツデータを挿入し、レイアウトした結果のイメージを示している。太枠 3 7 0 3 は、サブテンプレートが配置されるフローエリアである。3 7 0 4 はコンテンツデータを実際に流し込むためのコンテナ、3 7 0 5 ~ 3 7 0 9 はサブテンプレート示している。なお、図 3 5 では省略しているが 3 7 0 5 ~ 3 7 0 9 のサブテンプレート内にもコンテンツデータを挿入するためのコンテナが配置されている。3 7 1 0 はコンテンツデータの挿入によりレイアウトした結果、点線の分だけサブテンプレート 3 7 0 8 が拡大したことを示している。その結果、拡大したサブテンプレート 3 7 0 8 の次のサブテンプレート 3 7 0 9 がページ 3 7 1 3 にレイアウトされていることを示している。また、図 3 7 A では 3 つのページに渡ってサブテンプレートが配置されているため、1 ページ目を 3 7 1 1、2 ページ目を 3 7 1 2、3 ページ目を 3 7 1 3 とする。なお、図 3 7 A では図 2 5 の閾値設定ダイアログ 2 5 0 1 を用いて、最終ページに配置されるサブテンプレートが最終ページの 5 0 % 以下であれば再レイアウト処理を実行すると設定されているとする。また、最終ページである 3 7 1 3 に配置されたサブテンプレート 3 7 0 9 は閾値以下であるとする。

【 0 2 4 9 】

図 3 6 ~ 3 8 は、図 3 5 に対して本発明における通常印刷設定時のサブテンプレート間の距離を変更 (縮小) することによる再レイアウト処理を適用した後のイメージを示している。

【 0 2 5 0 】

なお、図 3 6 ~ 3 8 において、図 3 5 と同等の番号を示すものは同じオブジェクトであるため、説明は省略する。

【 0 2 5 1 】

図 3 6 は本発明における「全ページ」のサブテンプレート間の距離を縮小した結果のイメージを示している。3 7 0 5 から 3 7 0 8 の全てのサブテンプレート間の距離を図 2 4 で指定した縮小幅を用いて均等に縮小した再レイアウト処理により、ページ 3 7 1 3 にレイアウトされたサブテンプレート 3 7 0 9 が 2 ページ目にあたるページ 3 7 1 2 にレイアウトされていることを示している。

10

20

30

40

50

【0252】

この処理を用いることにより、全ページにおける全サブテンプレート間の距離を均等に
変更して、レイアウト当初のバランスを保持したまま再レイアウト処理を実行することが
可能となる。

【0253】

図37は本発明における「指定範囲ページ」のサブテンプレート間の距離を縮小した結
果のイメージを示している。図37の場合、2ページ目にあたるページ3712がユーザ
指定のページを示している。再レイアウト前にページ3712内に存在していたサブテン
プレート3707、3708間の距離が縮小され、サブテンプレート3709がページ3
712内にレイアウトされていることを示している。

10

【0254】

この処理を用いることにより、ユーザが指定したページにおけるサブテンプレート間の
距離のみを再レイアウト対象とするため、ユーザの意図を反映した再レイアウト処理を実
行することが可能となる。

【0255】

図38は本発明における「該当ページ」のサブテンプレート間の距離を縮小した結果の
イメージを示している。再レイアウト前に最終ページである3ページ目に該当するページ
3713の1つ前のページであるページ3712内に存在していたサブテンプレート37
07、3708間の距離が縮小され、サブテンプレート3709がページ3712にレイ
アウトされていることを示している。

20

【0256】

最終ページに閾値以下のサブテンプレートしかレイアウトされなかった場合に、最終ペ
ージの1つ前のページのサブテンプレート間の距離を変更することにより、最終ページ
のサブテンプレートを1つ前のページにレイアウトすることが可能となるため、ある一部
分のサブテンプレートのために生じるページを削減することが可能となる。

【0257】

(実施例2)

本発明の実施例2について説明する。本発明の実施例2は、実施例1のように自動で再
レイアウトを行うのではなく、ユーザがマウス等のユーザインターフェースを利用して手
動でギャップ調整を行う箇所を指定し、再レイアウトを行う自動レイアウトシステムにつ
いてである。

30

【0258】

本発明における手動でのギャップ調整は印刷設定によらず有効であるが、本実施例では
通常印刷設定を例に説明する。

【0259】

図39A～Cは本実施例で使用される図の概要を示している。

【0260】

図39Aは、ドキュメントテンプレートに定義されたコンテナに対してコンテンツを挿
入し、図8で説明したレイアウト処理を実行した結果のイメージを示している。図39A
における3901はドキュメント、3912～3914はページ、3903はマルチレ
コードのフローエリア、3904はコンテナ、3905～3909はサブテンプレートを示
している。なお、図39A～Cにおいて同じ数値を示すものは同様のオブジェクトである
。

40

【0261】

図39Bでは、本発明における手動でのサブテンプレート間の距離を変更する場合のマ
ウス操作の様子を示している。マウスポインタ3911は、サブテンプレート3908を
上方向に移動するように操作している様子を表し、その結果マウス操作に合わせてサブ
テンプレート3908が上方向に移動される様子を示している。

【0262】

図39Cでは本発明における手動でのサブテンプレート間の距離を変更した結果を示し

50

ており、図39Bでサブテンプレート3908を上方向に操作した結果、サブテンプレート3907および3908の距離が縮小されることにより、3ページ目に該当するページ3914にレイアウトされていたサブテンプレート3909が2ページ目に該当するページ3913にレイアウトされる。

【0263】

実施例2によれば、ユーザが所望とするサブテンプレートを視覚的に確認しながら再レイアウト処理を実行することができるため、ユーザが意図した再レイアウト処理を実行することが可能となる。

【0264】

なお、本実施例では説明のためにサブテンプレート間の距離を変更することにより再レイアウト処理を実行すると記載したがこれに限られることはなく、サブテンプレートをコンテンツデータが流し込まれるコンテナ(データ領域)としても本発明は実現可能であり課題を解決することができる。つまり、本発明の情報処理装置(レイアウト決定方法)では、コンテナ間の距離を同様に変更しても良い。よって、本願発明では、コンテンツデータを流し込む領域とは、コンテンツデータを流し込むデータ領域またはデータ領域が配置されているサブテンプレート領域であり、領域間の距離を変更して、再レイアウト処理を実行するものである。また、本願発明の再レイアウト処理を実行する場合、図25にて指定したページとは異なるページのフローエリア内に配置されたサブテンプレート領域間の距離を変更し、フローエリア外の領域間の距離を変更しないように制御しても良い。

【0265】

なお、本発明は、上記形態の機能を実現するソフトウェアのプログラムコードを記憶した記憶媒体(又は記録媒体)を、システム又は装置に供給し、そのシステム又は装置のコンピュータ(又はCPUやMPU)が記憶媒体に格納されたプログラムコードを読み出して実行することによっても達成されることは云うまでもない。

【0266】

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上記実施の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。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、上記実施の形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム(OS)などが実際の処理の一部又は全部を行い、その処理によって上記実施の形態の機能が実現される場合も含まれることは云うまでもない。

【0267】

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記憶媒体に書込まれた後、その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等が実際の処理の一部又は全部を行い、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云うまでもない。

【0268】

また、本発明は、上記実施の形態の機能を実現するソフトウェアのプログラムコードがネットワークを介して配信されることにより、システム又は装置のハードディスクやメモリ等の記憶手段又はCD-RW、CD-R等の記憶媒体に格納され、そのシステム又は装置のコンピュータ(又はCPUやMPU)が当該記憶手段や当該記憶媒体に格納されたプログラムコードを読み出して実行することによっても、達成されることは云うまでもない。

【図面の簡単な説明】

【0269】

【図1】Aは本発明のコンピュータシステム構成の一例を示す図。Bは本発明の図1Aにおけるコンピュータモジュールの概略の一例を示す図

【図2】図1Aのコンピュータシステムにエンジンサーバーを追加した構成の一例を示す

10

20

30

40

50

図

- 【図 3】本発明のメインウィンドウの一例を示す図
- 【図 4】本発明のコンテナ間のリンク、アンカーやスライダーを持つ典型的なコンテナの一面をスクリーン、ツール、アイコンの一例を示す図
- 【図 5】本発明のリンクの設定方法を示したフローチャートの一例を示す
- 【図 6】リンク作成時のユーザインタフェースの遷移例の一例を示す図
- 【図 7】レイアウトエンジンによるレイアウト計算処理の一例を示すフローチャート
- 【図 8】レイアウトエンジンによるレイアウト計算処理の一例を示すフローチャート
- 【図 9】実施形態によるレイアウト計算処理の説明の一例を示す図
- 【図 10】コンテナの集合についての説明の一例を示す図 10
- 【図 11】実施形態によるレイアウト計算処理の説明の一例を示す図
- 【図 12】可変リンクによってコンテナを配置した時のユーザインタフェースの表示例
- 【図 13】可変リンクを設定の一例を示すフローチャート
- 【図 14】リンクの設定ダイアログウィンドウの一例を示す図
- 【図 15】固定リンクによるレイアウト結果の一例を示す図
- 【図 16】可変リンクによるレイアウト結果の一例を示す図
- 【図 17】マルチレコードの概要の一例を示す図
- 【図 18】フローエリアの設定のユーザインタフェースの一例を示す図
- 【図 19】フローエリアの設定ダイアログのUIの一例を示す図
- 【図 20】サブテンプレートのレイアウトの一例を示す図 20
- 【図 21】条件式の設定UIの一例を示す図
- 【図 22】本発明における実施例 1 の概略の一例を示す図
- 【図 23】本発明における通常印刷設定時の処理の一例を示すフローチャート
- 【図 24】再レイアウト処理の実行内容を設定するダイアログウィンドウの一例を示す図
- 【図 25】閾値設定ダイアログウィンドウの一例を示す図
- 【図 26】再レイアウト処理の対象ページ設定ダイアログウィンドウの一例を示す図
- 【図 27】製本印刷設定時の処理の一例を示すフローチャート
- 【図 28】製本印刷設定時のギャップ調整確認ダイアログウィンドウの一例を示す図
- 【図 29】製本印刷設定時のギャップサイズを算出する一例を示すフローチャート
- 【図 30】サブテンプレート間のギャップに優先順位が設定されている場合の一例を示すフローチャート 30
- 【図 31】A は優先度設定ダイアログウィンドウの一例を示す図。B は優先度設定時の一例を示す図
- 【図 32】通常印刷設定時の全ページのギャップを調整して再レイアウトする一例を示すフローチャート
- 【図 33】通常印刷設定時の指定範囲ページのギャップを調整して再レイアウトの一例を示すフローチャート
- 【図 34】通常印刷設定時の該当ページのギャップを調整して再レイアウトの一例を示すフローチャート
- 【図 35】通常印刷設定時における最初のレイアウト後の一例を示す図 40
- 【図 36】本発明における「全ページ」のサブテンプレート間のギャップを縮小した結果の一例を示す図
- 【図 37】本発明における「指定範囲ページ」のサブテンプレート間のギャップを縮小した結果の一例を示す図
- 【図 38】本発明における「該当ページ」のサブテンプレート間のギャップを縮小した結果の一例を示す図
- 【図 39】A は本発明におけるレイアウト後の結果の一例を示す図。B は本発明におけるサブテンプレートをマウスにて移動することの一例を示す図。C は本発明における再レイアウト後の結果の一例を示す図
- 【図 40】バリアブルデータプリントの概要を説明の一例を示す図 50

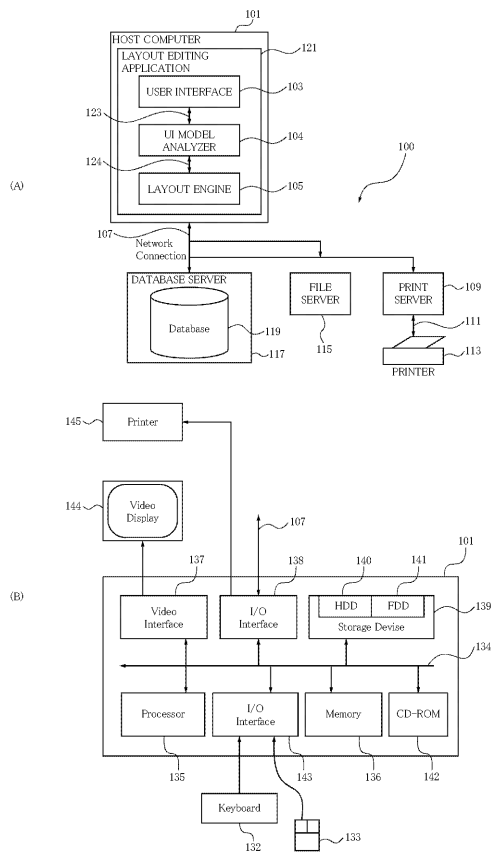
- 【図41】製本印刷設定時に発生する再レイアウト処理が必要となる場合の一例を示す図
- 【図42】製本印刷設定時に発生する再レイアウト処理が必要となる場合の一例を示す図
- 【図43】製本印刷設定時に発生する再レイアウト処理の実行結果の一例を示す図
- 【図44】製本印刷設定時に発生する再レイアウト処理の実行結果の一例を示す図
- 【図45】製本印刷時に発生する再レイアウト処理を説明の一例を示すフローチャート

【符号の説明】

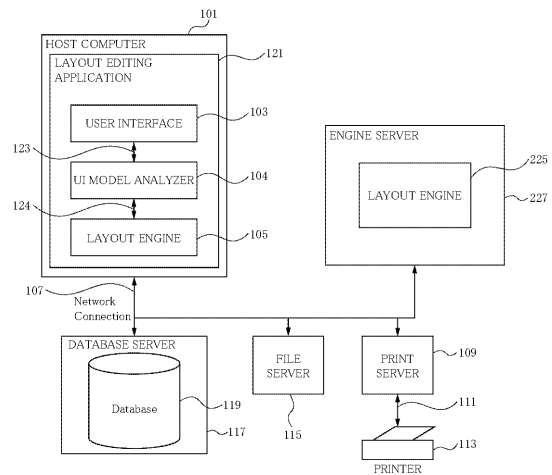
【0270】

- 101 ホストコンピュータ
- 103 ユーザインターフェース
- 105 レイアウトエンジン
- 119 データベース
- 121 レイアウト編集アプリケーション

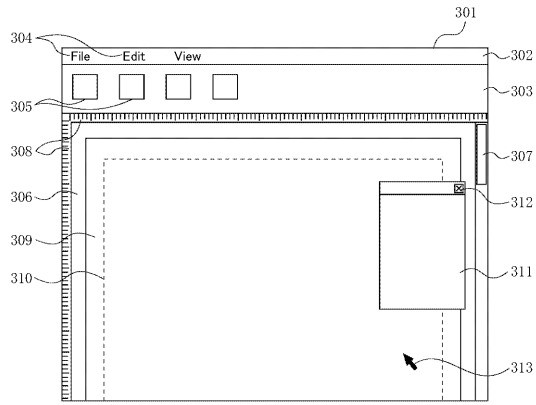
【図1】



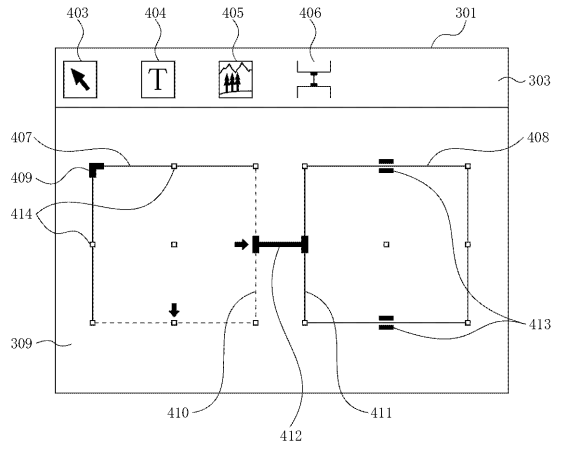
【図2】



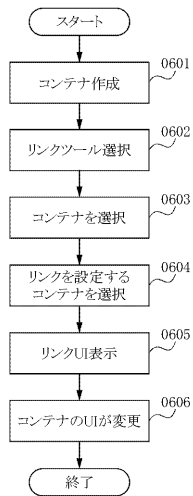
【図3】



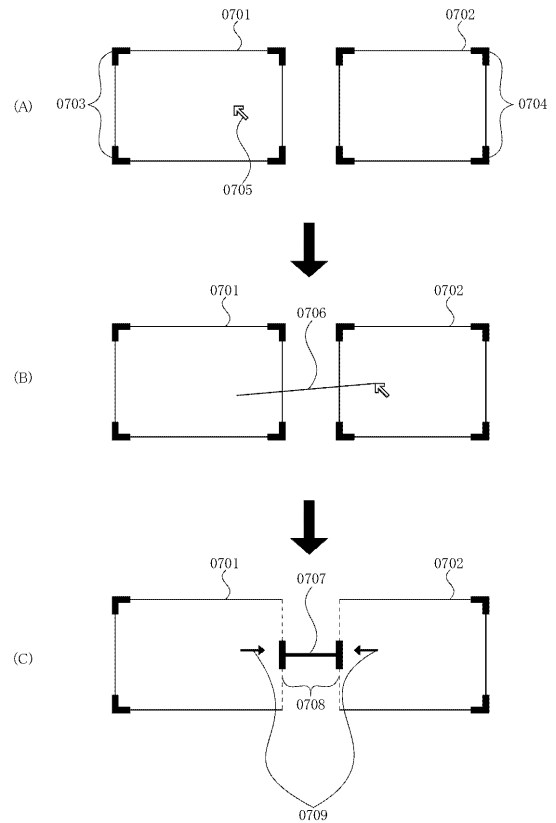
【図4】



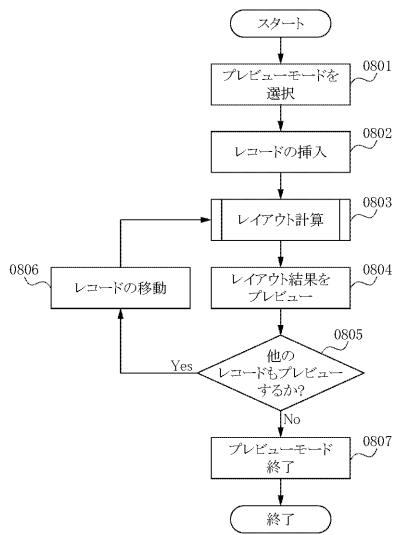
【図5】



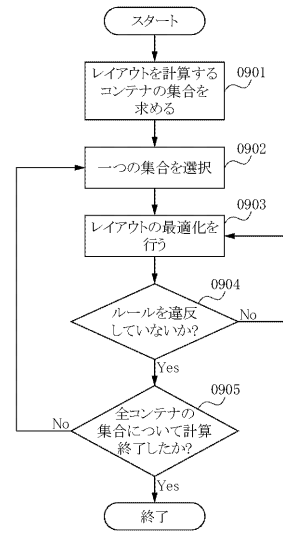
【図6】



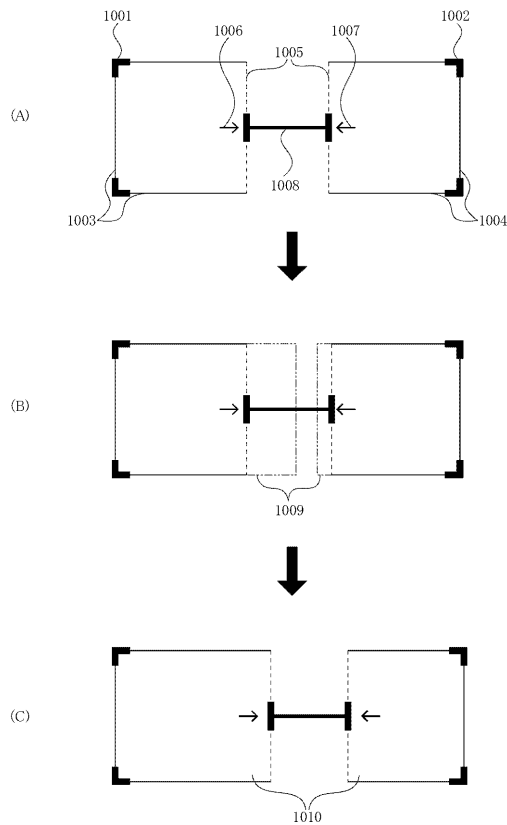
【図7】



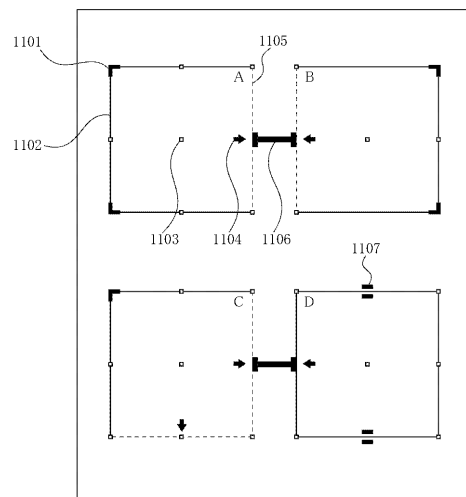
【図8】



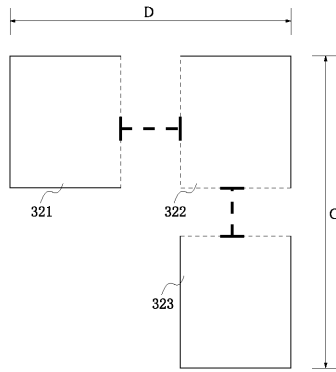
【図9】



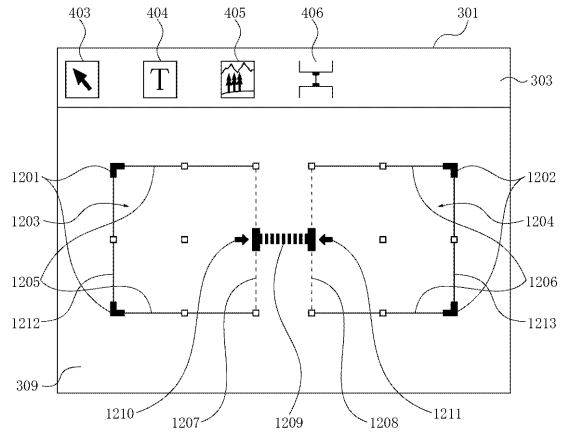
【図10】



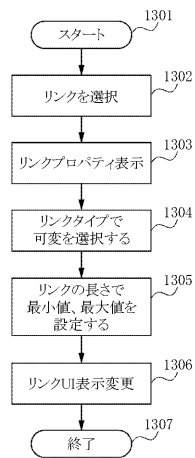
【図 1 1】



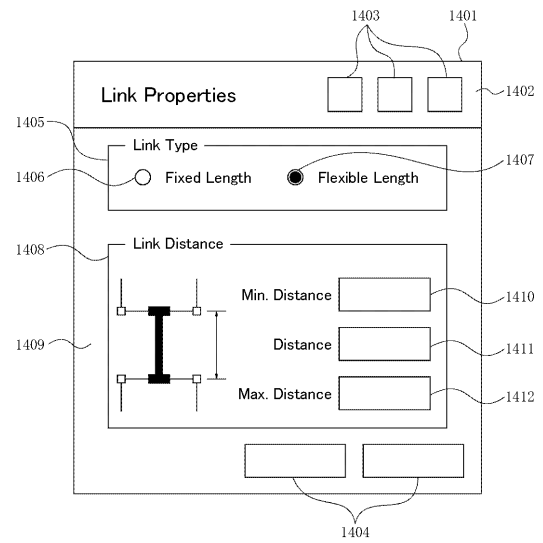
【図 1 2】



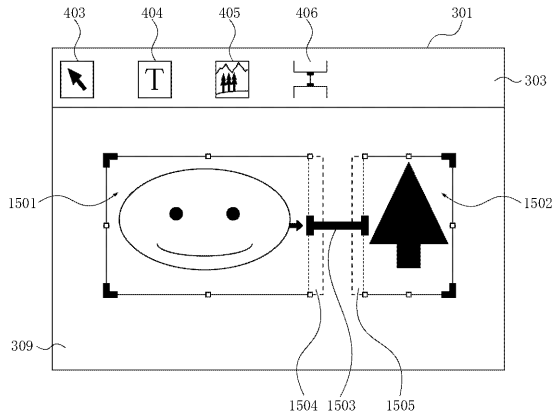
【図 1 3】



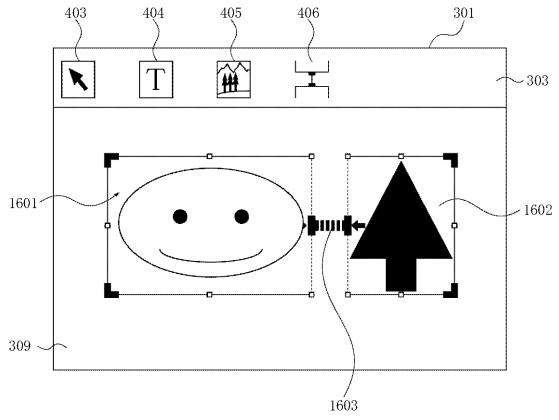
【図 1 4】



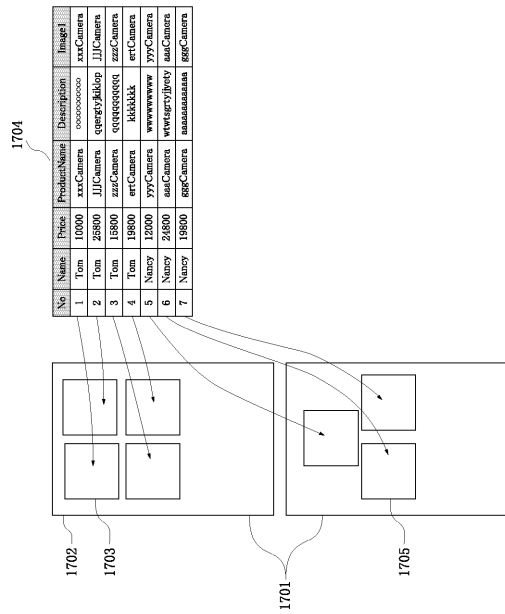
【 図 15 】



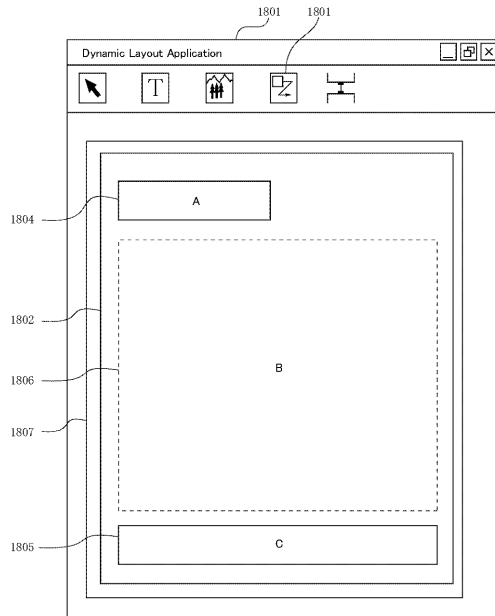
【 図 16 】



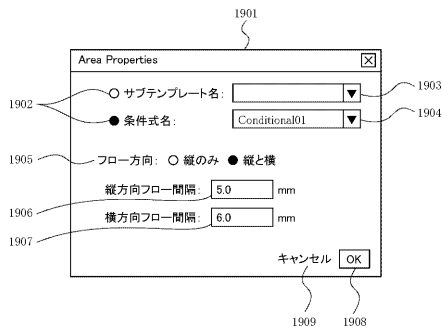
【 図 17 】



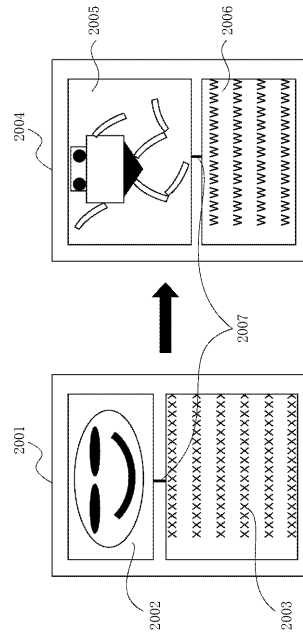
【 図 18 】



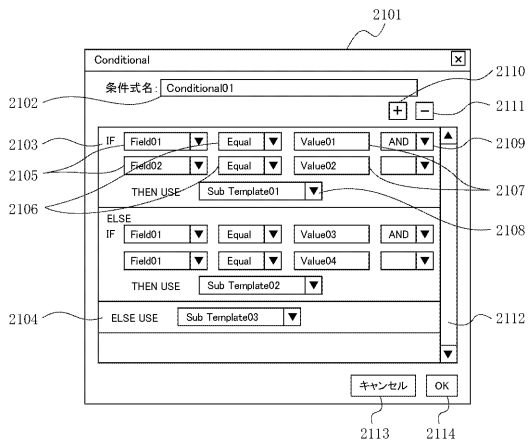
【図19】



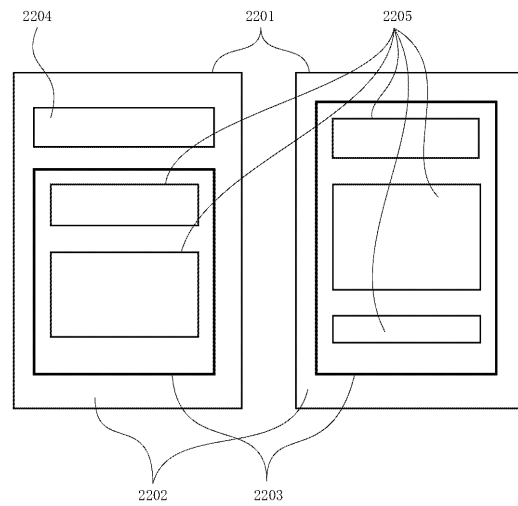
【図20】



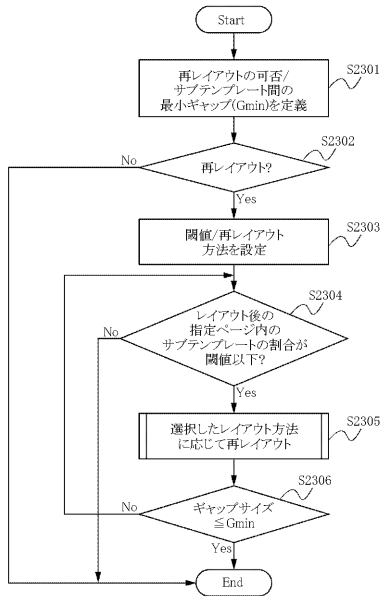
【図21】



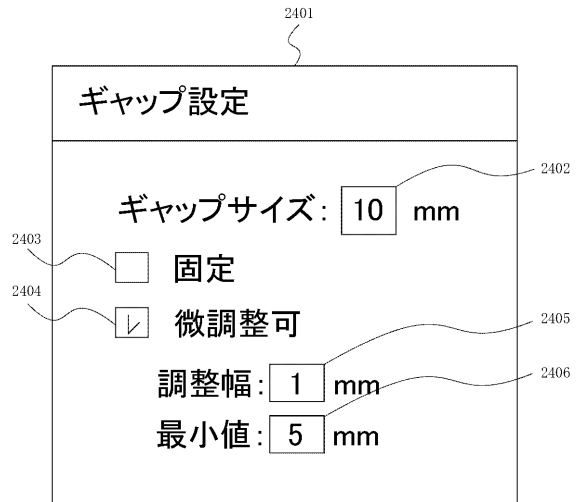
【図22】



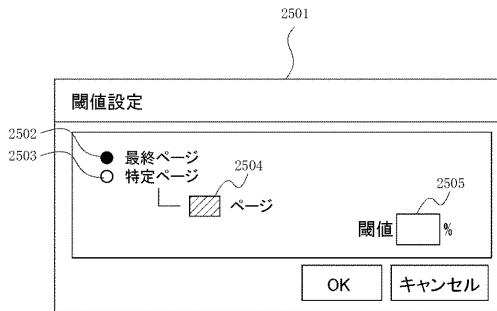
【図23】



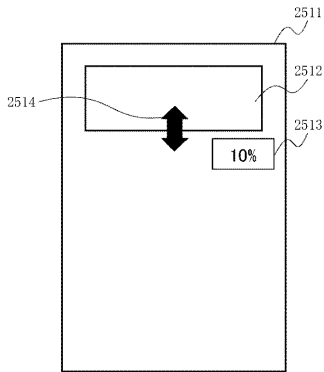
【図24】



【図25】

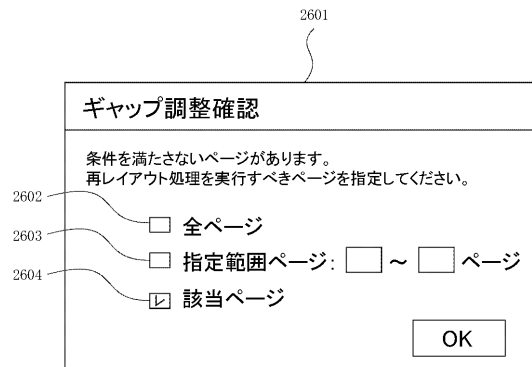


(A)

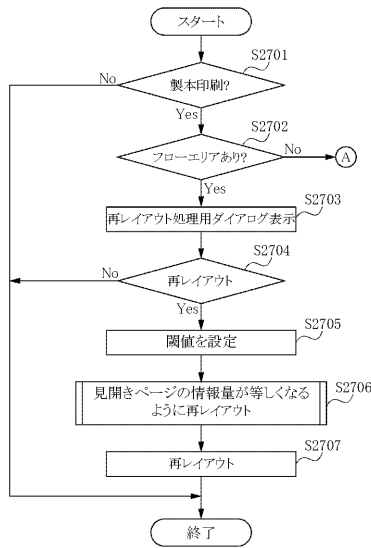


(B)

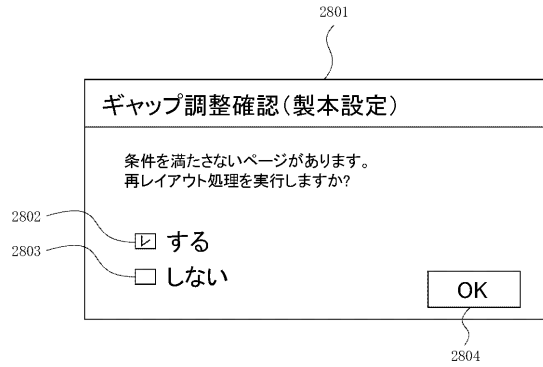
【図26】



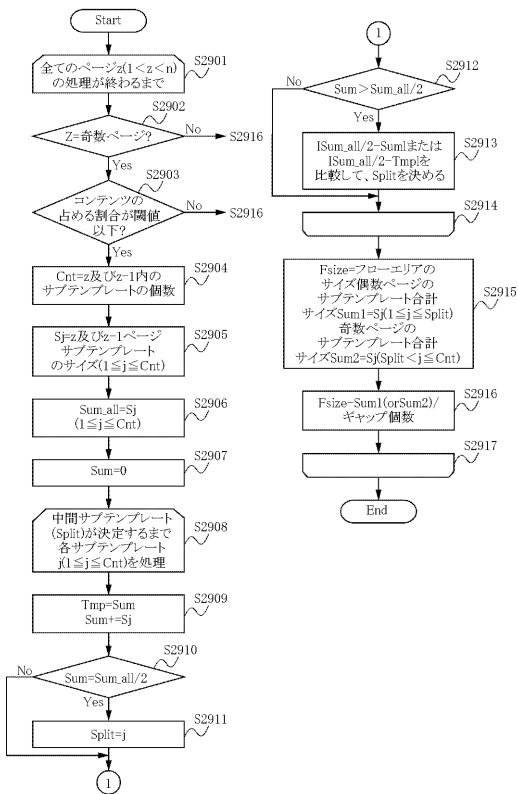
【図27】



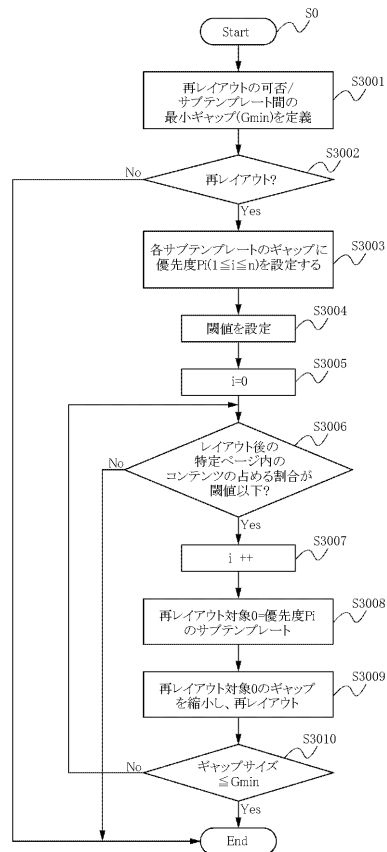
【図28】



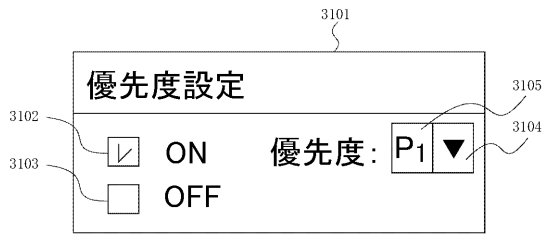
【図29】



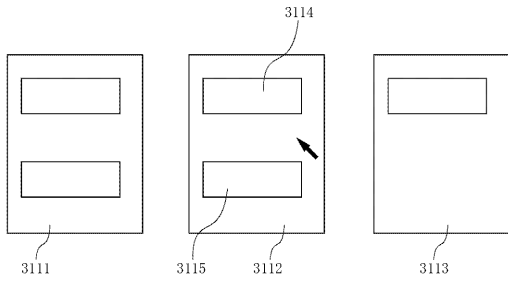
【図30】



【図31】

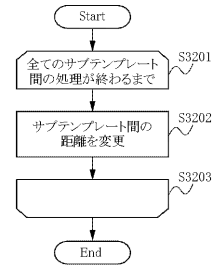


(A)

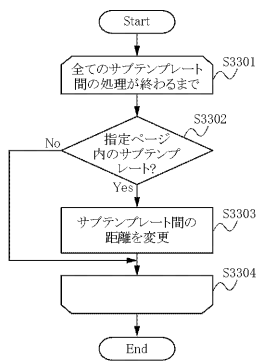


(B)

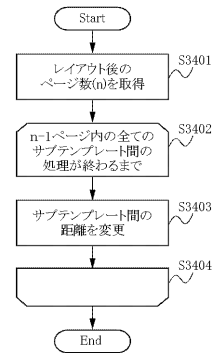
【図32】



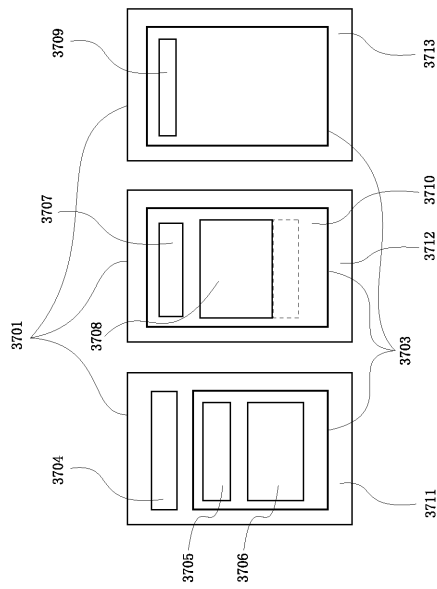
【図33】



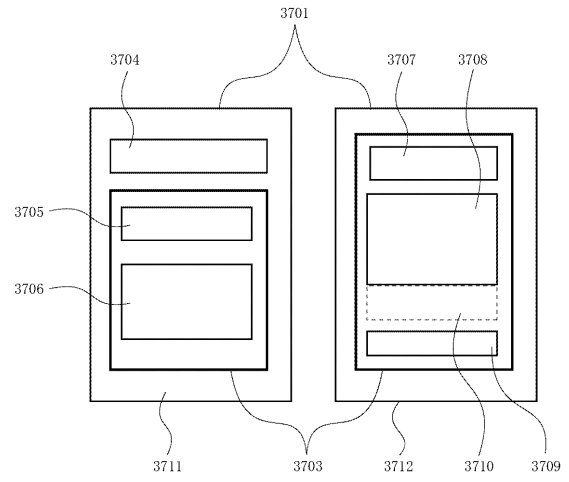
【図34】



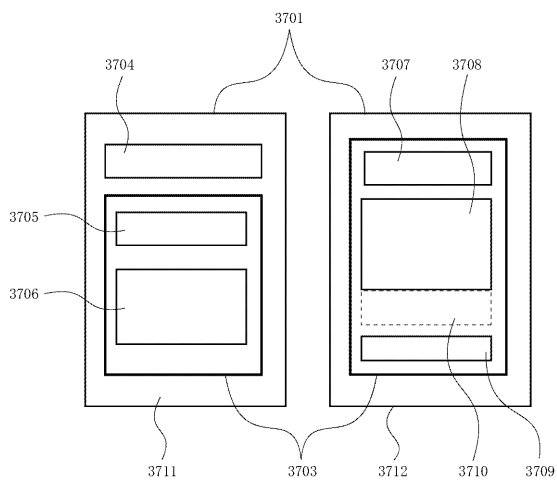
【図 35】



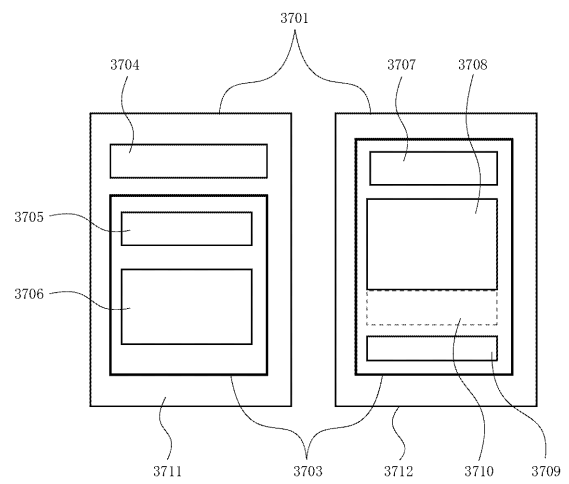
【図 36】



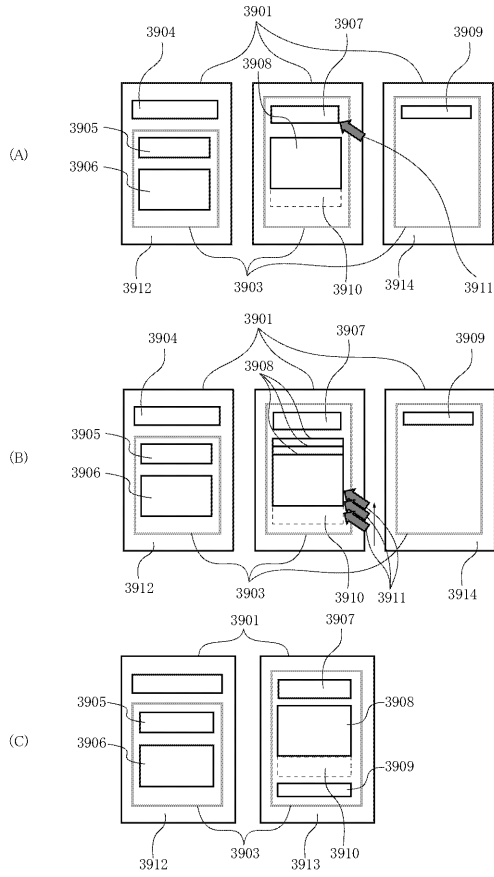
【図 37】



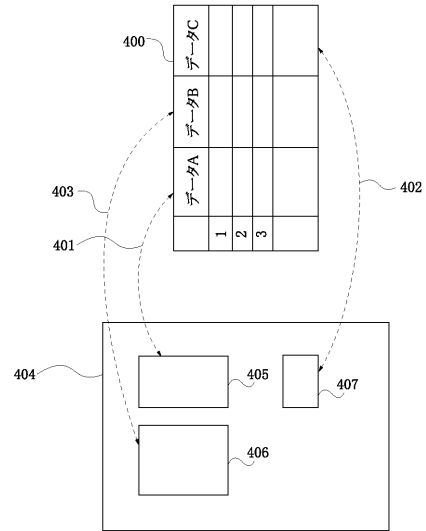
【図 38】



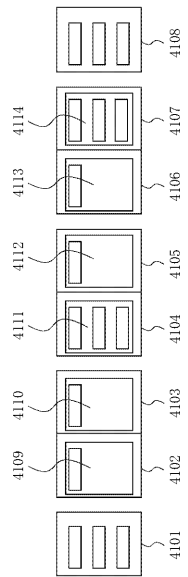
【図39】



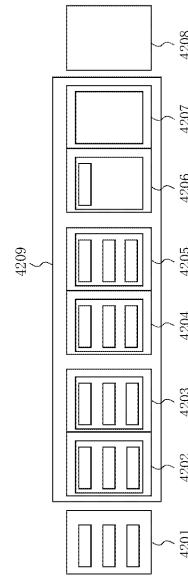
【図40】



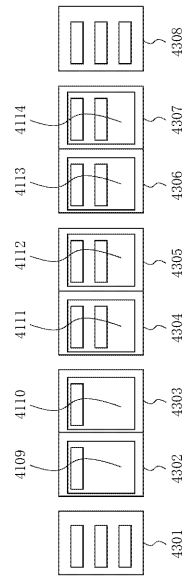
【図41】



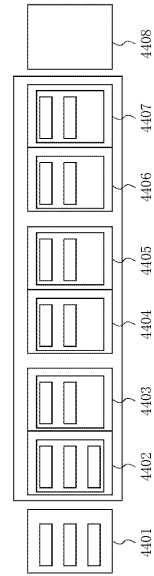
【図42】



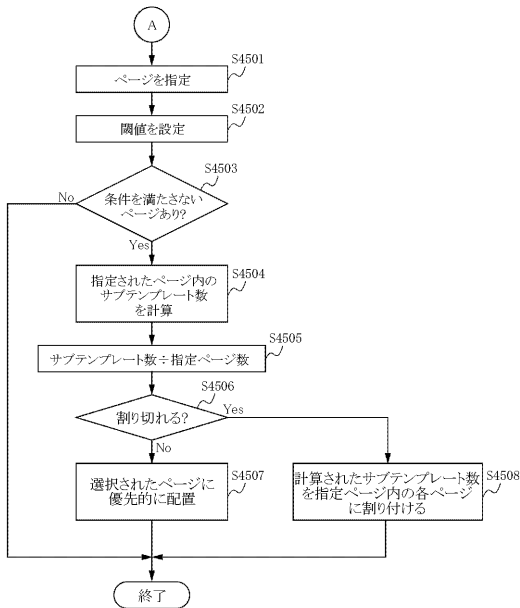
【図43】



【図44】



【図45】



フロントページの続き

(56)参考文献 特開2003-276271(JP,A)
特開平11-250271(JP,A)
特開2005-122727(JP,A)

(58)調査した分野(Int.Cl., DB名)
G06T 11/60