

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4756907号  
(P4756907)

(45) 発行日 平成23年8月24日 (2011. 8. 24)

(24) 登録日 平成23年6月10日 (2011. 6. 10)

(51) Int. Cl.

F I

**B 4 1 J 21/00 (2006. 01)**  
**G 0 6 F 3/12 (2006. 01)**B 4 1 J 21/00 Z  
G 0 6 F 3/12 M

請求項の数 9 (全 22 頁)

(21) 出願番号 特願2005-141183 (P2005-141183)  
 (22) 出願日 平成17年5月13日 (2005. 5. 13)  
 (65) 公開番号 特開2006-15734 (P2006-15734A)  
 (43) 公開日 平成18年1月19日 (2006. 1. 19)  
 審査請求日 平成20年5月13日 (2008. 5. 13)  
 (31) 優先権主張番号 特願2004-163561 (P2004-163561)  
 (32) 優先日 平成16年6月1日 (2004. 6. 1)  
 (33) 優先権主張国 日本国 (JP)

(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 100090273  
 弁理士 國分 孝悦  
 (72) 発明者 鈴木 明  
 東京都大田区下丸子3丁目30番2号 キ  
 ヤノン株式会社内

審査官 名取 乾治

最終頁に続く

(54) 【発明の名称】 画像形成装置、画像形成装置の処理方法、コンピュータプログラム及びコンピュータ読み取り可能な記録媒体

(57) 【特許請求の範囲】

【請求項 1】

構造化文書を解析して得られる複数の要素ブロックからなるツリーを辿り、出現順に各要素ブロックを複数のページに配置する画像形成装置であって、

第一のページに配置処理されている第一の要素ブロックを参照するための第一の参照ポイントと、前記第一の要素ブロックの次に出現する第二の要素ブロックを参照するための第二の参照ポイントとを設定する設定手段と、

前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであるか否かを判断する判断手段と、

前記判断手段により、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであると判断された場合、前記第一の参照ポイントを用いることにより第一の要素を第二のページに移動してから配置処理し、かつ、前記第二の参照ポイントを用いることにより第二のページにおいて第一の要素に続き第二の要素を配置処理するとともに、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルでないと判断された場合、前記第一の参照ポイントを用いず前記第二の参照ポイントのみを用いて第二のページに第二の要素を配置処理する配置手段と、

前記配置手段による配置処理の後、前記第二の要素ブロックを参照するように前記設定

手段により設定されている第一の参照ポイントを更新し、前記第二の要素ブロックの次に出現する第三の要素ブロックを参照するように前記設定手段により設定されている第二の参照ポイントを更新する更新手段と、  
を有することを特徴とする画像形成装置。

【請求項 2】

前記配置手段は、前記判断手段により、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであると判断された場合、前記ツリーを辿ることなく前記第一の参照ポイントを用いることにより第一の要素を第二のページに移動してから配置処理し、かつ、前記第二の参照ポイントを用いることにより第二のページにおいて第一の要素に続き第二の要素を配置処理するとともに、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルでないと判断された場合、前記第一の参照ポイントを用いず前記第二の参照ポイントのみを用いて第二のページに第二の要素を配置処理することを特徴とする請求項 1 に記載の画像形成装置。

10

【請求項 3】

前記第二の要素ブロックが前記第一のページ内に収まるか否かを判定する判定手段を更に有し、

前記判定手段により、前記第二の要素ブロックが前記第一のページに収まらないと判定された場合に、前記判断手段による判断処理と前記配置手段による配置処理とを前記第二の要素ブロックに対して行うことを特徴とする請求項 1 又は 2 に記載の画像形成装置。

20

【請求項 4】

前記配置手段は、前記第二の要素ブロック内のコンテンツが画像データの場合には、前記第二の要素ブロックを次のページの先頭に移動させることを特徴とする請求項 3 に記載の画像形成装置。

【請求項 5】

前記配置手段は、前記第二の要素ブロックに背景の指定がある場合、前記背景の全体を次ページの先頭に移動させることを特徴とする請求項 3 に記載の画像形成装置。

【請求項 6】

前記配置手段は、前記第二の要素ブロック内のコンテンツがテキストデータの場合には、前記第二の要素ブロックを行ごとに分割して改ページ処理することを特徴とする請求項 3 に記載の画像形成装置。

30

【請求項 7】

構造化文書を解析して得られる複数の要素ブロックからなるツリーを辿り、出現順に各要素ブロックを複数のページに配置する画像形成装置の処理方法であって、

第一のページに配置処理されている第一の要素ブロックを参照するための第一の参照ポイントと、前記第一の要素ブロックの次に出現する第二の要素ブロックを参照するための第二の参照ポイントとを設定する設定ステップと、

前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであるか否かを判断する判断ステップと、

40

前記判断ステップにおいて、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであると判断された場合、前記第一の参照ポイントを用いることにより第一の要素を第二のページに移動してから配置処理し、かつ、前記第二の参照ポイントを用いることにより第二のページにおいて第一の要素に続き第二の要素を配置処理するとともに、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルでないと判断された場合、前記第一の参照ポイントを用いず前記第二の参照ポイントのみを用いて第二のページに第二の要素を配置処理する配置ステップと、

50

前記配置ステップにおける配置処理の後、前記第二の要素ブロックを参照するように前記設定ステップにおいて設定されている第一の参照ポイントを更新し、前記第二の要素ブロックの次に出現する第三の要素ブロックを参照するように前記設定ステップにおいて設定されている第二の参照ポイントを更新する更新ステップと、  
を有することを特徴とする画像形成装置の処理方法。

【請求項 8】

構造化文書を解析して得られる複数の要素ブロックからなるツリーを辿り、出現順に各要素ブロックを複数のページに配置する画像形成装置の制御方法の各ステップをコンピュータに実行させるコンピュータプログラムであって、

第一のページに配置処理されている第一の要素ブロックを参照するための第一の参照ポイントと、前記第一の要素ブロックの次に出現する第二の要素ブロックを参照するための第二の参照ポイントとを設定する設定ステップと、

前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであるか否かを判断する判断ステップと、

前記判断ステップにおいて、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであると判断された場合、前記第一の参照ポイントを用いることにより第一の要素を第二のページに移動してから配置処理し、かつ、前記第二の参照ポイントを用いることにより第二のページにおいて第一の要素に続き第二の要素を配置処理するとともに、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルでないと判断された場合、前記第一の参照ポイントを用いず前記第二の参照ポイントのみを用いて第二のページに第二の要素を配置処理する配置ステップと、

前記配置ステップにおける配置処理の後、前記第二の要素ブロックを参照するように前記設定ステップにおいて設定されている第一の参照ポイントを更新し、前記第二の要素ブロックの次に出現する第三の要素ブロックを参照するように前記設定ステップにおいて設定されている第二の参照ポイントを更新する更新ステップと、  
をコンピュータに実行させることを特徴とするコンピュータプログラム。

【請求項 9】

請求項 8 に記載のコンピュータプログラムを記録したことを特徴とするコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は画像形成装置、画像形成装置の処理方法、コンピュータプログラム及びコンピュータ読み取り可能な記録媒体に関し、特に、デジタルデータ放送や、インターネットを始めとするデータ通信を用いて配信するコンテンツを印刷するために用いて好適な技術に関するものである。

【背景技術】

【0002】

従来では、HTMLファイルのコンテンツを出力する際、出力画像の横幅が出力媒体の用紙サイズに収容できない大きさであれば、サイズを縮小したり、あるいは画像方向を変換（回転）したり、または改行禁止指示や横幅指定を無視したりする処理を実行する印刷装置が提案されている（例えば、特許文献 1、特許文献 2 を参照）。

【0003】

そして、この従来の印刷装置においては、改ページ部分にHTMLファイルに記述された文字が含まれてしまう場合、次ページのレイアウトとして形成される画像情報にその文字を含む行を挿入するようにしている。

【0004】

さて、HTMLファイルを解析することによって得られた要素ツリーにおいて、印刷処理対象である要素の配置に関連して、先行して配置した要素ブロックと干渉するケースがある。例えば、改ページ処理では、先行する要素を再配置するため再帰的に配置済みブロックを参照する必要が発生する。

【0005】

【特許文献1】特開平11-15749号公報

【特許文献2】特開2000-330908号公報

【発明の開示】

【発明が解決しようとする課題】

【0006】

従来このようなケースにおいて、配置済みブロックを参照するためには、要素ツリーを毎回辿る必要があった。すなわち、要素ツリーを一旦上層の要素に戻り、しかる後に下層に下ってリンクを辿って目的の要素ブロックのポインタを求めるという処理を行わなければならない。

【0007】

ところで、デジタルテレビはもとより、最近では携帯電話やデジタルカメラ等の携帯機器においても、マークアップ言語で記述したコンテンツを取り扱うことが多くなってきている。ところが、このような機器では資源が限られているため処理負荷をできるだけ小さくすることが望まれている。例えば、コンテンツを印刷データに変換するレイアウト処理は、要素ブロックを保持するためにメモリ資源を特に多く消費するので、負荷の大きい処理となっていた。

【0008】

また、ディスプレイなどにHTML文書から得られたコンテンツを表示する際、コンテンツが表示可能領域を越えてしまう場合には、スクロールバーが現れて操作者にコンテンツの表示領域内での移動操作をできるようにしている。一方、印刷系では印刷媒体の大きさが限られているので、視覚的に違和感がないようにHTML文書から得られたコンテンツのページ割付方法が求められていた。

【0009】

本発明は前述の問題点にかんがみ、レイアウトが厳密には規定されない言語で記述されたHTML文書のコンテンツから印刷データを作成するにあたり、印刷装置側の処理負担を軽減できるようにするとともに、印刷データを効率的に処理することが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】

【0010】

本発明の画像形成装置は、構造化文書を解析して得られる複数の要素ブロックからなるツリーを辿り、出現順に各要素ブロックを複数のページに配置する画像形成装置であって、第一のページに配置処理されている第一の要素ブロックを参照するための第一の参照ポインタと、前記第一の要素ブロックの次に出現する第二の要素ブロックを参照するための第二の参照ポインタとを設定する設定手段と、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであるか否かを判断する判断手段と、前記判断手段により、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであると判断された場合、前記第一の参照ポインタを用いることにより第一の要素を第二のページに移動してから配置処理し、かつ、前記第二の参照ポインタを用いることにより第二のページにおいて第一の要素に続き第二の要素を配置処理するとともに、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルでないと判断された場合、前記第一の参照ポインタを用いず前記第二の参照ポインタのみを用いて第二のページに第二の要素を配置処理する配置手段と、前記配置手段による配置

10

20

30

40

50

処理の後、前記第二の要素ブロックを参照するように前記設定手段により設定されている第一の参照ポイントを更新し、前記第二の要素ブロックの次に出現する第三の要素ブロックを参照するように前記設定手段により設定されている第二の参照ポイントを更新する更新手段とを有することを特徴とする。

【0011】

本発明の画像形成装置の処理方法は、構造化文書を解析して得られる複数の要素ブロックからなるツリーを辿り、出現順に各要素ブロックを複数のページに配置する画像形成装置の処理方法であって、第一のページに配置処理されている第一の要素ブロックを参照するための第一の参照ポイントと、前記第一の要素ブロックの次に出現する第二の要素ブロックを参照するための第二の参照ポイントとを設定する設定ステップと、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであるか否かを判断する判断ステップと、前記判断ステップにおいて、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであると判断された場合、前記第一の参照ポイントを用いることにより第一の要素を第二のページに移動してから配置処理し、かつ、前記第二の参照ポイントを用いることにより第二のページにおいて第一の要素に続き第二の要素を配置処理するとともに、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルでないと判断された場合、前記第一の参照ポイントを用いず前記第二の参照ポイントのみを用いて第二のページに第二の要素を配置処理する配置ステップと、前記配置ステップにおける配置処理の後、前記第二の要素ブロックを参照するように前記設定ステップにおいて設定されている第一の参照ポイントを更新し、前記第二の要素ブロックの次に出現する第三の要素ブロックを参照するように前記設定ステップにおいて設定されている第二の参照ポイントを更新する更新ステップとを有することを特徴とする。

【0012】

本発明のコンピュータプログラムは、構造化文書を解析して得られる複数の要素ブロックからなるツリーを辿り、出現順に各要素ブロックを複数のページに配置する画像形成装置の制御方法の各ステップをコンピュータに実行させるコンピュータプログラムであって、第一のページに配置処理されている第一の要素ブロックを参照するための第一の参照ポイントと、前記第一の要素ブロックの次に出現する第二の要素ブロックを参照するための第二の参照ポイントとを設定する設定ステップと、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであるか否かを判断する判断ステップと、前記判断ステップにおいて、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルであると判断された場合、前記第一の参照ポイントを用いることにより第一の要素を第二のページに移動してから配置処理し、かつ、前記第二の参照ポイントを用いることにより第二のページにおいて第一の要素に続き第二の要素を配置処理するとともに、前記第二の要素ブロックの改ページ処理のスタイルが、前記第一の要素ブロックを前記第二の要素ブロックが配置処理される第二のページに移動してから配置処理するスタイルでないと判断された場合、前記第一の参照ポイントを用いず前記第二の参照ポイントのみを用いて第二のページに第二の要素を配置処理する配置ステップと、前記配置ステップにおける配置処理の後、前記第二の要素ブロックを参照するように前記設定ステップにおいて設定されている第一の参照ポイントを更新し、前記第二の要素ブロックの次に出現する第三の要素ブロックを参照するように前記設定ステップにおいて設定されている第二の参照ポイントを更新する更新ステップとをコンピュータに実行させることを特徴とする。

【0013】

本発明の記録媒体は、前記に記載のコンピュータプログラムを記録したことを特徴とする。

【発明の効果】

【0014】

本発明によれば、レイアウトが厳密には規定されない言語で記述したコンテンツから印刷するための画像データを形成するにあたり、画像形成装置側の処理負担を軽減できるようにするとともに、印刷データを効率的かつ的確に処理することができる。

【発明を実施するための最良の形態】

【0015】

(第1の実施の形態)

以下、添付の図面に沿って本発明の実施の形態を説明する。

以下、図面を参照しながら本実施の形態の画像形成装置の一例としての印刷装置を説明する。なお、本実施の形態は、W3C(World Wide Web Consortium)が定めているXHTML1.1TM(extensible Hyper Text Markup Language)及びCSS2.0TM(Cascading Style Sheet 2.0)で定める仕様書に基づいて記載している。

【0016】

また、本実施の形態で言うところの「ノード・リンク構造」とは、ノードとリンクに区別できる項目を含み、各リンクが2つ以上のノードに関係している構造を言う。さらに、ノード・リンク関係とは、ノードのパス及び要素間のリンクに基づく要素間の関係である。

【0017】

「システムの概要」

図1は、本実施の形態の印刷システム全体の概要構成を示す図である。

図1に示したように、画像形成装置である本実施の形態の印刷装置30は、コントローラ31、プリントエンジン32を有する。コントローラ31は、PC(パーソナルコンピュータ)11、PDA(personal digital assistant:個人向け携帯情報機器の総称)12、デジタル放送機器13などのホスト(コンピュータ)10より、有線または無線の接続形態で入力されるコンテンツ(HTMLなどの構造化文書)20を解釈し、印刷データ33に変換する。プリントエンジン32は、印刷データ33を紙などの記録メディア40上に形成するために、矢印34で示す印刷処理を行う。なお、コントローラの31の実態は、CPU、記憶装置及びインターフェースを含むシングルボードコンピュータによって構成され、記憶装置に記憶されたプログラムに基づいてCPUが印刷装置を制御している。

【0018】

「要素ツリーの構築」

図2は、構造化文書をコントローラ31により字句解析、構文解析により得た全ての要素ブロックを含む全要素ツリー201(多方向分岐リスト)を示したものである。この要素ツリーの実体は、ノード・リンクのグラフである。全要素ツリー201の各ノードは、要素ブロックとして、3つのデータ構造体から構成される。すなわち、核となる要素の実体データ(コンテンツデータ)である「要素構造体202」、スタイルシートを参照して、要素ブロック内のスタイル表記言語で記述された字句を解析して得られたスタイルに関する「スタイル構造体203」、及び全要素ツリー201のルート要素から順にツリーを辿り、要素ブロックに対応するオブジェクトの配置位置を計算することによって得られるレイアウト情報を格納する「レイアウト構造体204」である。

【0019】

「レイアウトの手順」

要素ブロックを記録メディア(印刷用紙)40に効率的に配置する手順を以下に説明する。

まず、コントローラ31は、要素ツリー上のルート(構造化文書内に記述された<html>の記述)から順に辿り、要素ブロック毎に構成された「スタイル構造体203」に基づいて、BOXモデル、文字、画像などのページ内レイアウト処理(対象要素ブロックの配置処

10

20

30

40

50

理) 4 1を行う。

#### 【 0 0 2 0 】

要素ブロックをレイアウトした結果は、配置・視覚情報を保持する「レイアウト構造体 2 0 4」へ格納される。このとき、対象要素のブロックに含まれる子要素ブロックの配置可能範囲 4 2も「レイアウト構造体 2 0 4」へ格納される。配置・視覚情報が格納された「レイアウト構造体 2 0 4」は子要素ブロックに引き渡される。すなわち、コントローラ 3 1は、親要素ブロックで指定された子要素配置可能範囲 4 2内に子要素ブロックのレイアウトを行う。以下同様に、「レイアウト構造体 2 0 4」へ結果が格納され、さらに孫要素への引き渡し、処理の遷移が進行する。

#### 【 0 0 2 1 】

「ページ情報の管理」

図 3 は、ページ情報リストの構築を説明する図である。

まず、スタイルシート 3 0 1 に記述された、改ページに関する処理を指定するための規則"@page"で表現したページ情報に基づいて、複数の「ページ情報構造体」3 0 2 のリストが作成される。「ページ情報構造体」は、コンテンツを印刷出力する際のページの設定が記述されたものである。

#### 【 0 0 2 2 】

「ページ情報構造体」は、具体的には、@page: xxxx[]のフォーマットによって記述される。"xxxx"には、先頭ページを示す"first"、見開き左側ページを示す"left"、見開き右側ページを示す"right"、名前付きページのページ識別子を示す"name1"、"name2"などのページ識別子が記述される。また、[]内には、"xxxx"において指定されたページのsize(用紙サイズと向き)、margin(余白)が記述される。なお、"@page []"のように識別子"xxxx"の指定がない場合は、全ページ共通の「ページ情報構造体」として作成する。

#### 【 0 0 2 3 】

コントローラ 3 1 は、「ページ情報構造体」の作成の際、スタイル初期値及び@media printのデフォルトスタイルシートにおいて指定されたページスタイルを「ページ情報構造体」3 0 2 に展開する。図 3 には、一例として、all(1)、first(2)、left(3)、right(4)、name(5~) からなる「ページ情報構造体」3 0 2 リストがRAMに格納されることが示されている。

#### 【 0 0 2 4 】

なお、name="all"の「ページ情報構造体」3 0 2 は、@pageにより識別子指定が無いものであり、全ページ共通の設定が記述されているページ構造体であり、リストの 1 番目に登録される。

#### 【 0 0 2 5 】

name=firstの「ページ情報構造体」3 0 2 は、@page: firstによる先頭ページの設定が記述されているものであり、2 番目に登録される。leftは、@page:leftで見開き左側ページの設定が記述されているものであり、3 番目に登録される。rightは@page:rightで見開き右側ページの設定が記述されているものであり、4 番目に登録される。そして、5 番目以降は@page:xxxxで名前付きページとして設定されたページ構造体が個別に登録される。

#### 【 0 0 2 6 】

コントローラ 3 1 は、ページレイアウトの処理において、現在処理しているページに対応したページ情報構造体を参照し、ページ構造体の実体を作成する。例えば、1 ページ目は、all first leftの順に参照し、名前付きページ(name1)の 2 ページ目では、all right name1の順に参照し、「ページ構造体」を実体化する。そして、実体化した「ページ構造体」にはページ内に配置した要素、すなわち要素ブロックを多分岐リストとして登録する「ページ内要素ツリー」page-treeが含まれる。

#### 【 0 0 2 7 】

「要素をページに配分」

図 4、図 5 は、ページ内レイアウトにおけるページ構造体を用いた処理を説明する図である。図 4 では、1 ページ目の処理を示している。図 5 では、2 ページ目の処理を示して

10

20

30

40

50

いる。

【 0 0 2 8 】

前述の通り、コントローラ 3 1 は、現在のページに該当するページ情報として「ページ情報構造体」のリストの中から選択・参照し、現ページ用に「ページ構造体」の実体を作成する。

【 0 0 2 9 】

「ページ構造体」の実体は、@pageで記述されたsize (width、 height、 portrait/landscape) , margin、レクタングル情報を含む要素群を多分岐リストとして管理するためのpage-tree「ページ内要素ツリー」である。

【 0 0 3 0 】

以下、「ページ内要素ツリー」への登録方法を説明する。

まず、コントローラ 3 1 は、最初のページの「ページ構造体」の実体を作成する。そして、レイアウト処理において、<html>をルートとする「要素ツリー」を順に辿り、「要素構造体 2 0 2 + スタイル構造体 2 0 3 + レイアウト構造体 2 0 4」の情報に基づき、<body>以下のタグをページ内に順に配置する。配置確定情報はレイアウト構造体 2 0 4 に格納される。

【 0 0 3 1 】

ページ内に配置した要素ブロックは「ページ構造体」の「ページ内要素ツリー」へ登録される。このとき配置された要素ブロックがページ内に収まりきれなかった場合やpage-breakスタイルによって改ページの指示があれば、改ページ処理が行われる。

【 0 0 3 2 】

コントローラ 3 1 は、改ページ処理において、次ページ用の「ページ内要素ツリー」を含む「ページ構造体」の実体を新たに作成する。そして、次ページの「ページ構造体」の領域に配置する為の再計算を行う。具体的には、現在の「ページ内要素ツリー」で配置が完結しなかった要素ブロックを次ページの「ページ内要素ツリー」にコピーし、改ページした用紙縦幅の分だけ配置情報を前方へ移動させる。

【 0 0 3 3 】

すなわち、本実施の形態では、要素ブロックの実体（要素構造体 2 0 2）だけではなく、スタイルに関する情報（スタイル構造体 2 0 3）も移動することになる。次ページ以降のレイアウト処理では、出力未完部分の要素をコピーして次ページの「ページ内要素ツリー」へ登録するとともに現ページに配置されなかった要素ブロックを次ページに配置する処理が行われる。

【 0 0 3 4 】

ところで、全ての要素ブロックを包含するHTML文書内の<body>の記述は、最終ページの出力まで出現しない。<body>における全ての子要素の出力が終了した時点で最終ページが完了しレイアウト処理が終了することになる。

【 0 0 3 5 】

「スタイルの指定」

以下、図 6、図 7 を用いてスタイル指定による改ページ処理について説明する。

図 6 は、従来から行われているページ区切り (page-break) の処理について説明する図である。図 6 では、ターゲットとなる要素ブロックを配置する過程で、改ページ処理によって要素ブロックのはみ出しが発生した際の処理を説明している。図 6 では、要素ブロックが 2 分割され、要素ブロックの前半を前ページに配置して、要素ブロックの後半を前ページに続く後ページに配置している。

【 0 0 3 6 】

図 7 - a に示すpage-break-before:alwaysのスタイルの指定は、指定されたタグ内のコンテンツのレイアウト処理を始める前に強制的に改ページ処理を行うための指定である。

図 7 - b に示すpage-break-before: leftまたはpage-break-before: rightのスタイルの指定は、指定されたタグのサイドのページになるまで強制改ページをさらに行う指定である。

10

20

30

40

50



## 【 0 0 3 7 】

図 7 - c に示すpage-break-before:avoidのスタイルの指定は、指定されたタグの前で改ページが行われるが、この場合、前ページの「ページ要素内ツリー」から「直前ブロック要素」以下のサブツリーを現在の「ページ要素内ツリー」へ移動してからレイアウト処理を行う指定である。

## 【 0 0 3 8 】

図 7 - d に示すpage-break-inside:avoidのスタイルの指定は、指定タグがページ内に収まりきらなかった場合は「現在ブロック要素」以下のツリーを次ページの「ページ内要素ツリー」へ移動するスタイルの指定である。

## 【 0 0 3 9 】

図 7 - e に示すpage-break-after:always left、rightのスタイルの指定は、指定されたタグのレイアウト処理の最後で強制的に改ページ処理を行うスタイルの指定である。page-break-after:leftまたはpage-break-after:rightの指定は、さらに指定サイドのページになるまで強制改ページ処理を行う指定である。

## 【 0 0 4 0 】

図 7 - f に示すpage-break-after:avoidのスタイルの指定は、次の要素に対してafter-avoid処理中のフラグを引き渡す。after-avoid処理中とされた次の要素ではpage-break-before: avoidと指定されている場合と同じ処理を行う。なお、page-break-before/inside /afterで前記に該当しない場合は改ページ処理を行わない。

## 【 0 0 4 1 】

## 「原点制御」

図 8 は、改ページ処理における原点位置について説明する図である。

ページ毎の配置開始位置である原点は、前述した「ページ構造体」で示される用紙サイズ、用紙の向き、マージンの値によって決定する「Page BOX」の左上に設定される。「Page BOX」とは、要素の実体を配置するためのエリアを示し、ページ毎に異なる設定が可能である。例えば、スタイルの指定として、position:absolute、すなわち絶対位置では、PageBOXの原点に<body>以下のタグを配置していく。また、スタイルの指定として、position:staticすなわち相対位置では、先頭ページの開始位置を原点として、改ページ後は前ページからの続きの位置に開始位置を設定しタグを継続して配置する。

## 【 0 0 4 2 】

## 「要素のハンドル」

以下は、改ページ処理におけるブロック要素の参照についての説明である。

図 9 は全要素ツリーを表す図であり、図 1 0 は全要素ツリーから順次コピーして成るページ内要素ツリーを表す図である。

## 【 0 0 4 3 】

レイアウト処理では、「ページ内要素ツリー」を参照し、要素ブロックの出現ごとに、直前に配置した要素ブロックを「直前要素ブロック」、現在処理中の要素ブロックを「現在要素ブロック」として識別する参照ポイントをRAMに記憶更新する。すなわち、コントローラ 3 1 は、要素ブロックが出現する毎に参照ポイントが更新され、現在処理中の要素ブロックに対する直前及び現在の要素ブロックを管理する。

## 【 0 0 4 4 】

図 2 2 は、要素ブロックを配置する際にコントローラ 3 1 によって実行される改ページ処理の手順を説明するフローチャートである。

まず、ステップ S 1 0 において、現在のページに要素ブロックの配置を開始する。そして、ステップ S 1 1 において、直前に配置した要素ブロックを「直前要素ブロック」、その次に配置処理する対象の要素ブロックを「現在要素ブロック」として識別する参照ポイントをRAMに記憶・更新する。

## 【 0 0 4 5 】

ステップ S 1 2 において、現在配置処理する対象の要素ブロックが現在のページ内に収まるか否か判断する。現在のページ内に収まると判断した場合、ステップ S 1 0 に戻り、

10

20

30

40

50

次の要素ブロックの配置を行う。

【 0 0 4 6 】

一方、現在のページ内に要素ブロックが収まらないと判断した場合、ステップ S 1 3 に進む。ステップ S 1 3 において、要素ブロックのスタイル構造体に「page-break-before: avoid」のスタイルの指定があるか否かを判断する。

【 0 0 4 7 】

この判断の結果、「page-break-before:avoid」のスタイルの指定があると判断された場合、ステップ S 1 4 に進み、「直前要素ブロック」の参照ポイントを参照し、図 7 - c に示したように、「直前要素ブロック」以下のサブツリーを現在の「ページ要素内ツリー」へ移動してからレイアウト処理を行う。

10

【 0 0 4 8 】

一方、ステップ S 1 3 において、「page-break-before:avoid」のスタイルの指定がないと判断された場合、ステップ S 1 5 に進み、「現在要素ブロック」の参照ポイントを参照し、その現在ブロック要素のレイアウト指定に対応する改ページ処理が行われる。

ステップ S 1 6 では、全ての要素ブロックの配置が終了したか否かを判断し、全ての処理が終了するまで前述の処理を繰り返し行う。また、全ての要素ブロックの配置が終了したと判断したら処理を終了する。

【 0 0 4 9 】

なお、図 2 2 のフローチャートでは、「直前要素ブロック」を参照する一例として「page-break-before:avoid」のスタイルの指定を示したが、その他「直前要素ブロック」を用いるスタイルの指定にもフローチャートを適用することができる。

20

【 0 0 5 0 】

本実施の形態では、以下に示すとおり、要素ブロックの実体として配置されるデータの種類（テキストデータ、画像データ）に対応して改ページ処理が行われる。コントローラ 3 1 内の CPU は、実体（コンテンツ）の種類に関する情報を構造化文書内のタグを参照することによって得ることができる。そして、コントローラ 3 1 内の CPU は、実体（コンテンツ）の種類に関する情報を要素構造体 2 0 2 に格納し、以下の要素ブロックの配置方法のいずれかをを用いるかを格納された実体（コンテンツ）の種類に関する情報を参照することにより決定する。

【 0 0 5 1 】

30

「テキストの配置」

図 2 0 は、テキストデータの要素ブロックがページ領域の下端を超えた場合、ボックス内のテキストが配置される行を区切りの単位（行BOX）として分割配置する様子を示したものである。

【 0 0 5 2 】

図 1 1 は、テキストデータの要素ブロックのページ端末配置をさらに詳細に説明する図である。

要素ブロック内のテキストデータは、行BOX毎に分割される。分割された行BOXには、配置完了 / 未完了の識別マークがそれぞれ付加される。未完了の行BOXを含むテキストデータの要素ブロックは、現在のページ内要素ツリーに残されるとともに、次のページ内要素ツリーへコピーされる。コントローラ 3 1 は、次ページのレイアウト処理において、配置完了 / 未完了の識別マークを参照することにより、前ページにおいて配置完了した行BOXを使用せずに未完了の行BOXから配置処理を行う。すなわち、未完了の行BOXの開始位置をPageBOXの先頭に設定し、配置を継続する。

40

【 0 0 5 3 】

図 2 3 は、要素ブロックとしてテキストデータが配置される際にコントローラ 3 1 によって実行される改ページ処理の手順を説明するフローチャートである。

処理が開始されると、ステップ S 2 1 において、オブジェクトがテキストデータである要素ブロック内を行ごとに分割する。分割されたブロックをそれぞれ行BOXと呼ぶ。そして、ステップ S 2 2 に進み、行BOXの配置の際、ページ内に行BOXが収まるか否か

50

判断する。

【 0 0 5 4 】

この判断の結果、ページ内に行 B O X が収まると判断した場合、ステップ S 2 3 に進み、その対象の行 B O X の配置処理を行うとともに、配置完了の識別子を付加する。一方、ステップ S 2 2 の判断の結果、ページ内に行 B O X が収まらないと判断された場合、ステップ S 2 4 に進み、次ページのページ内要素ツリーにオブジェクトがテキストデータである要素ブロックを複写する。ステップ S 2 5 では、全ての要素ブロックの配置が終了したか否かを判断し、全ての処理が終了するまで前述の処理を繰り返し行う。

【 0 0 5 5 】

「画像の配置」

10

図 1 2 は、従来行われている画像の要素ブロックの改ページ処理の例を示す図である。

従来の処理では、要素ブロック内の画像がページ領域に収まらない場合は、親の要素ブロックに指定されたスタイル、例えば、Boxモデルの背景等は考慮せずに画像のみ改ページ処理が行われていた。

【 0 0 5 6 】

ここで、画像は現在のページには配置せず、配置未完了の識別マークを付して次のページ処理に委ねる。次ページでのレイアウトでは画像の開始位置をPageBOX先頭に設定してから、配置処理を継続する。

【 0 0 5 7 】

図 1 3 は、画像ブロックのページ端末置処理の実施の形態を説明する図である。図 1 3 においては、画像とそれに付随するスタイルを改ページにおいてどのように扱うかについて提案している。すなわち、画像に背景指定があるとき、画像があるページの余白域に収まらず、次のページに送られる場合は、背景も同時に送る処理をする、ページ割付方法を提案している。これは、要素ブロック内に背景情報を有するレイアウト構造体を含めたことによって達成される。

20

【 0 0 5 8 】

「大きな画像の配置」

図 1 4 は、改ページ処理を行っても、なお画像がページ領域内に収まらない場合の処理について説明している。即ち、アスペクト比を保持しつつ横方向が収容できる倍率に縮小してもなお、縦方向がページに収まらない場合は、縦方向に分割する。例えば「page-brake-inside:split」なるレイアウト処理のスタイルを新規提案している。

30

【 0 0 5 9 】

具体的には、コントローラ 3 1 は、「page-brake-inside:split」の指定に対して、対象の画像が現ページ領域に収まらずと判断した場合、まず改ページ処理を行う。次に、画像の横方向がページ領域に収まるまでアスペクト比を保持しつつ画像を縮小する処理を行う。それでもなお画像の縦方向がページからはみ出すと判断した場合は、画像を複数に分割して、それぞれを各ページに順次配置する処理を行う。

【 0 0 6 0 】

図 1 5 は、サイズが大きい画像に対して、従来から用いられているスタイル「page-brake-inside:avoid」が指定されたときの本実施の形態のレイアウト処理を示している。具体的には、コントローラ 3 1 は、「page-brake-inside:avoid」の指定に対して、対象の画像が現ページ領域に収まらずと判断した場合、まず改ページ処理を行う。そして、改ページ処理を行ってもなお画像が新しいページ領域内に収まらずと判断した場合に、画像の分割は行わず画像のアスペクト比を保持した状態で縮小し、画像がページ内に収まるようにする処理が実行される。

40

【 0 0 6 1 】

図 2 1 は、改ページしてもなお画像がページ領域内に収まらない場合の別の処理を説明している。図 2 1 で示す処理では、画像の縮小や分割を行わず、ページ内に収まった範囲を印刷処理する形態である。

【 0 0 6 2 】

50

図 2 1 では、「page-brake-inside:clip」なるスタイルを新規提案している。具体的には、コントローラ 3 1 は、「page-brake-inside:clip」の指定に対して、対象の画像が現ページ領域に収まらないと判断した場合、まず改ページ処理を行う。次に、改ページ処理された画像のうち、ページ内の画像領域に対して印刷処理が行われ、ページからはみ出た領域は印刷しない処理が行われる。

【 0 0 6 3 】

すなわち、図 2 1 に示す処理では、画像の縮小も分割もせず、配置する領域のサイズに合わせて画像をクリッピングする処理が行われる。このクリッピング処理は、例えば、画像の左上を有効領域の原点にあわせて、はみ出た部分を除くようにクリップする。

【 0 0 6 4 】

図 1 6 は、コントローラ 3 1 によって実行される要素ブロックが画像データである場合の改ページ処理のフローチャートである。

最初のステップ S 1 では、配置する要素ブロックがページ内の未配置領域に収まる大きさか否かを判定する。この判定の結果、配置可能であると判定した場合は配置する。

【 0 0 6 5 】

ステップ S 1 において要素ブロックが配置することができないと判断された場合、ステップ S 2 に進み、改ページにより、次ページに新たな 1 ページ領域を確保する処理を行う。

次に、ステップ S 3 では、配置指定を調べてそれぞれに対応するルーチン、"avoid" , "clip" , "split" に遷移する。

【 0 0 6 6 】

図 1 7 は、コントローラ 3 1 によって実行される画像データである要素ブロックの改ページ処理のフローチャートで、"avoid" が指定された場合実行されるものである。この場合、ステップ S 4 において、画像ブロックが 1 ページに収まるように縮小してから配置する処理が行われる。

【 0 0 6 7 】

図 1 8 は、コントローラ 3 1 によって実行される画像データである要素ブロックの改ページ処理のフローチャートで、"clip" が指定された場合実行されるものである。この場合、ステップ S 5 において、画像ブロックをそのまま配置する。ただし、はみ出た領域は印刷しない処理が行われる。

【 0 0 6 8 】

図 1 9 は、コントローラ 3 1 によって実行される画像ブロックの改ページ処理のフローチャートで、"split" が指定された場合に実行されるものである。

まず、ステップ S 6 では、ページ領域の横方向に対して画像が収まるか否かを調べる。この結果、ページ領域に収容可能ならばステップ S 8 に進む。また、ステップ S 6 の判定の結果、画像をページ領域内に収容できないと判定した場合にはステップ S 7 に進む。

【 0 0 6 9 】

ステップ S 7 では、横方向が収容可能になる割合で画像を縮小する処理を実行する。

次に、ステップ S 8 では、画像がページ領域の縦方向に対して収容可能な大きさであるか否かを調べる。この結果、収容可能ならそのまま配置する。また、収容できない場合にはステップ S 9 に進み、画像を複数に分割し、さらに改ページをしながら配置する。

【 0 0 7 0 】

本実施の形態では、前述した実施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、インターネットなどのネットワークを介して画像形成装置に供給し、画像形成装置のコントローラ 3 1 (または CPU や MPU) が記憶媒体に格納されたプログラムコードを読み出し実行することによって達成できる。

【 0 0 7 1 】

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コントローラ 3 1 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。

10

20

30

40

50

## 【 0 0 7 2 】

プログラムコードを供給するための記憶媒体としては、例えば、ハードディスク、ROMなどを用いることができる。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、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼動しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。

## 【 0 0 7 3 】

さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムコードの指示にもとづき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される。

10

## 【 0 0 7 4 】

本実施の形態を前記記憶媒体に適用する場合、その記憶媒体には、先に説明したフローチャートに対応するプログラムコードを格納することになるが、簡単に説明すると、本実施の形態の画像形成装置に不可欠なモジュールを、記憶媒体に格納することになる。

## 【 0 0 7 5 】

したがって、本発明の機能処理をコンピュータで実現するために、前記コンピュータにインストールされるプログラムコード自体も本発明を実現するものである。つまり、本発明は、本発明の機能処理を実現するためのコンピュータプログラム自体も含まれる。

20

## 【 0 0 7 6 】

その場合、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、OSに供給するスクリプトデータ等の形態であってもよい。

## 【 0 0 7 7 】

プログラムを供給するための記録媒体としては、例えば、フロッピー（登録商標）ディスク、ハードディスク、光ディスク、光磁気ディスク、MO、CD-ROM、CD-R、CD-RW、磁気テープ、不揮発性のメモリカード、ROM、DVD（DVD-ROM、DVD-R）などがある。

## 【 0 0 7 8 】

その他、プログラムの供給方法としては、クライアントコンピュータのブラウザを用いてインターネットのホームページに接続し、前記ホームページから本発明のコンピュータプログラムそのもの、もしくは圧縮され自動インストール機能を含むファイルをハードディスク等の記録媒体にダウンロードすることによっても供給できる。

30

## 【 0 0 7 9 】

また、本発明のプログラムを構成するプログラムコードを複数のファイルに分割し、それぞれのファイルを異なるホームページからダウンロードすることによっても実現可能である。つまり、本発明の機能処理をコンピュータで実現するためのプログラムファイルを複数のユーザに対してダウンロードさせるWWWサーバも、本発明に含まれるものである。

40

## 【 0 0 8 0 】

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

## 【 0 0 8 1 】

また、コンピュータが、読み出したプログラムを実行することによって、前述した実施の形態の機能が実現される他、そのプログラムの指示に基づき、コンピュータ上で稼動しているOSなどが、実際の処理の一部または全部を行い、その処理によっても前述した実施の形態の機能が実現され得る。

50

## 【 0 0 8 2 】

さらに、記録媒体から読み出されたプログラムが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によっても前述した実施の形態の機能が実現される。

## 【図面の簡単な説明】

## 【 0 0 8 3 】

【図1】本実施の形態の印刷システム全体の概要構成を示す図である。

【図2】要素ツリーと構造体を示す図である。

10

【図3】ページ情報リストの構築を説明する図である。

【図4】ページ内レイアウトにおけるページ構造体の操作を説明する図であり、1頁目を説明する図である。

【図5】ページ内レイアウトにおけるページ構造体の操作を説明する図であり、2頁目を説明する図である。

【図6】page-breakの一般形を説明する図である。

【図7 - a】page-break-before: always;の例を示す図である。

【図7 - b】page-break-before: left、right;の例を示す図である。

【図7 - c】page-break-before: avoid;の例を示す図である。

【図7 - d】page-break-inside: avoid;の例を示す図である。

20

【図7 - e】page-break-after: always、left、right;の例を示す図である。

【図7 - f】page-break-after: avoid;の例を示す図である。

【図8】改ページ処理における原点位置について説明する図である。

【図9】全要素ツリーを説明する図である。

【図10】ページ内要素ツリーを説明する図である。

【図11】テキストデータの要素ブロックの改ページ処理を説明する図である。

【図12】従来の画像データの要素ブロックの改ページ処理を説明する図である。

【図13】実施の形態を示し、画像データの要素ブロックの改ページ処理を説明する図である。

【図14】実施の形態を示し、分割による画像データの要素ブロックの改ページ処理を説明する図である。

30

【図15】実施の形態を示し、縮小による画像データの要素ブロックの改ページ処理を説明する図である。

【図16】画像ブロックの改ページ処理における配置手順を説明するフローチャートである。

【図17】画像ブロックの改ページ処理(avoid)における配置手順を説明するフローチャートである。

【図18】画像ブロックの改ページ処理(clip)における配置手順を説明するフローチャートである。

【図19】画像ブロックの改ページ処理(split)における配置手順を説明するフローチャートである。

40

【図20】テキストデータの要素ブロックの改ページ処理における配置を説明するための図である。

【図21】画像ブロックのクリップによる改ページ処理における配置手順を説明するための図である。

【図22】要素ブロックの改ページ処理における配置手順を説明するフローチャートである。

【図23】テキストデータの要素ブロックの改ページ処理における配置手順を説明するフローチャートである。

## 【符号の説明】

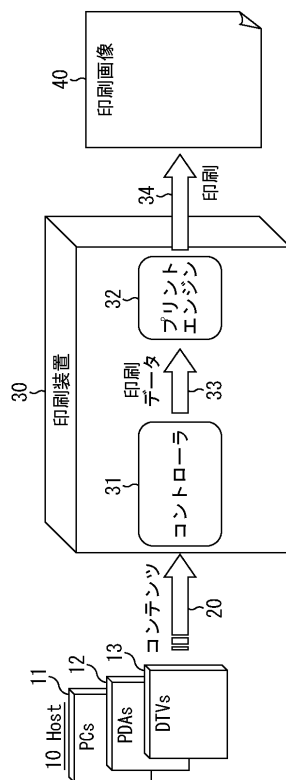
50

## 【 0 0 8 4 】

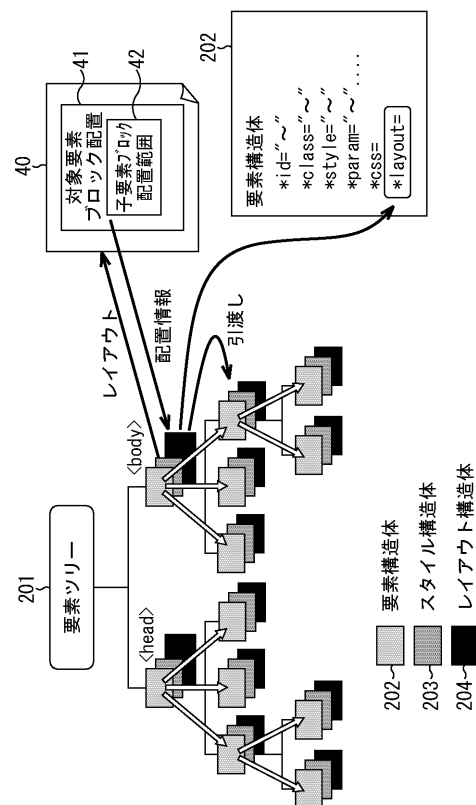
- 1 0    ホスト装置
- 1 1    P C ( パーソナルコンピュータ )
- 1 2    P D A ( personal digital assistant : 個人向け携帯情報機器の総称 )
- 1 3    デジタル放送機器
- 2 0    コンテンツ
- 3 0    印刷装置
- 3 1    コントローラ
- 3 2    プリントエンジン
- 3 3    印刷データ
- 3 4    印刷処理の矢印
- 4 0    メディア
- 4 1    レイアウト ( 自要素配置 )
- 4 2    子要素配置可能範囲
- 2 0 1    要素ツリー
- 2 0 2    要素構造体
- 2 0 3    スタイル構造体
- 2 0 4    レイアウト構造体

10

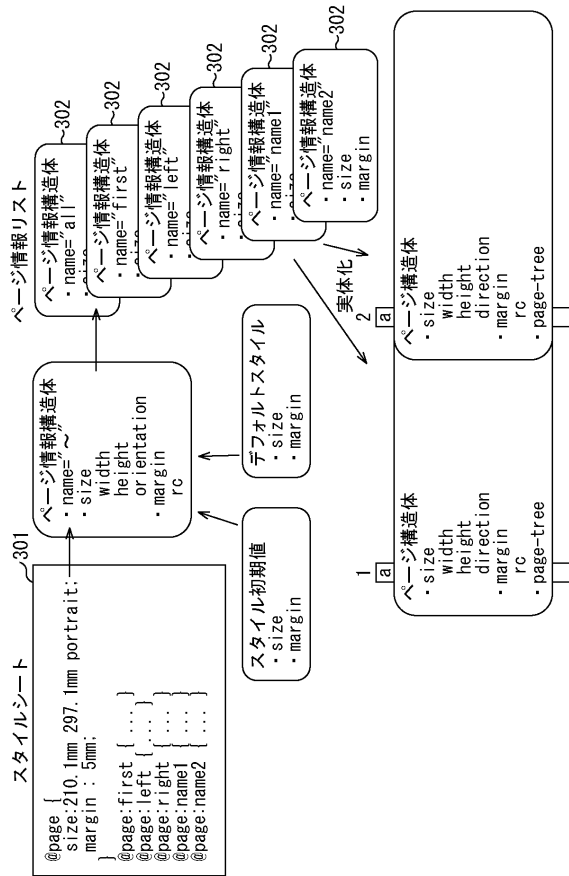
【 図 1 】



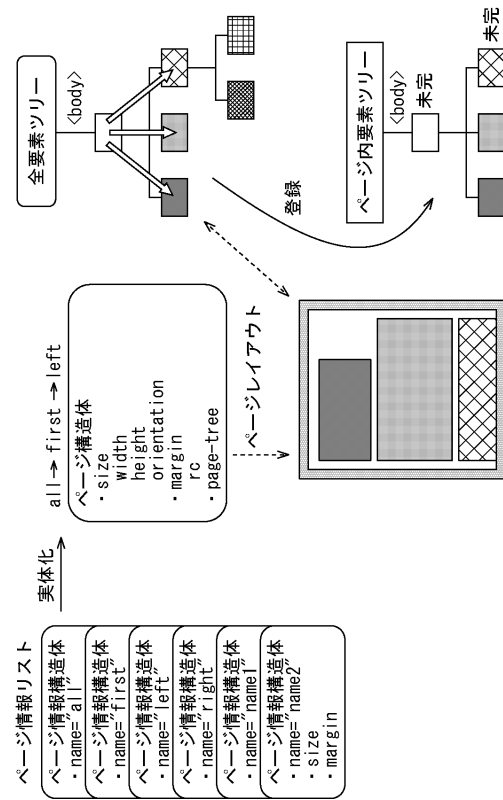
【 図 2 】



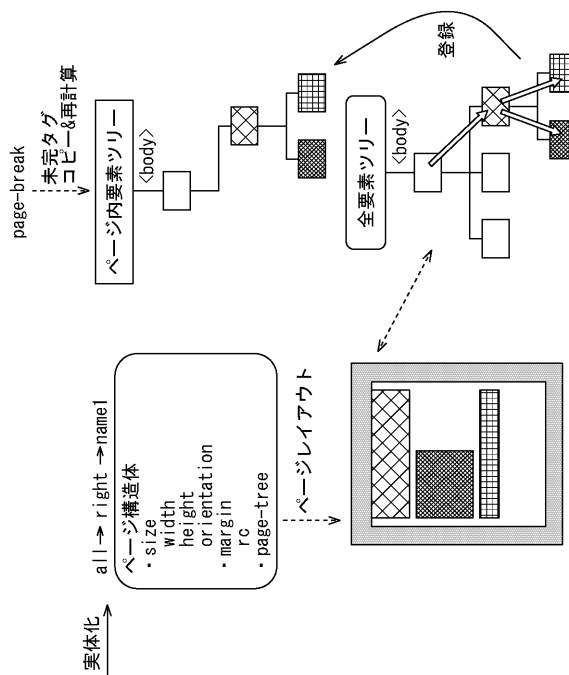
【図 3】



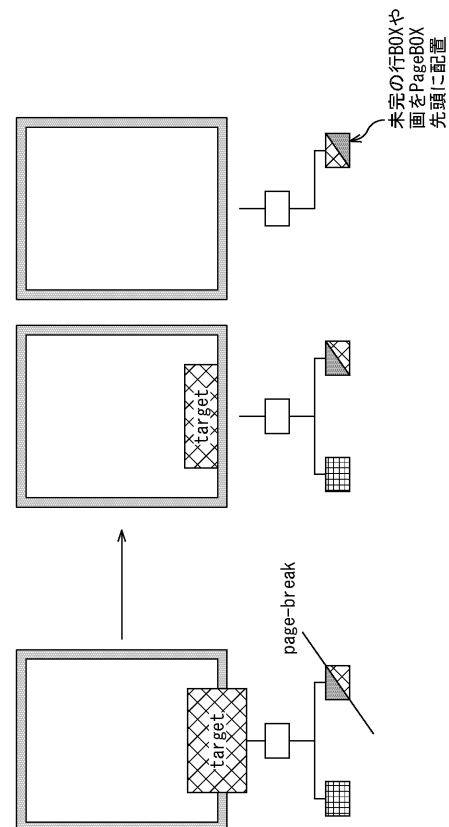
【図 4】



【図 5】

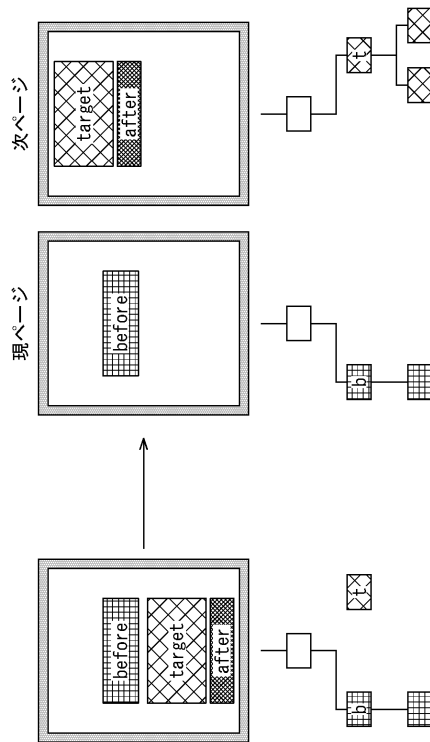


【図 6】

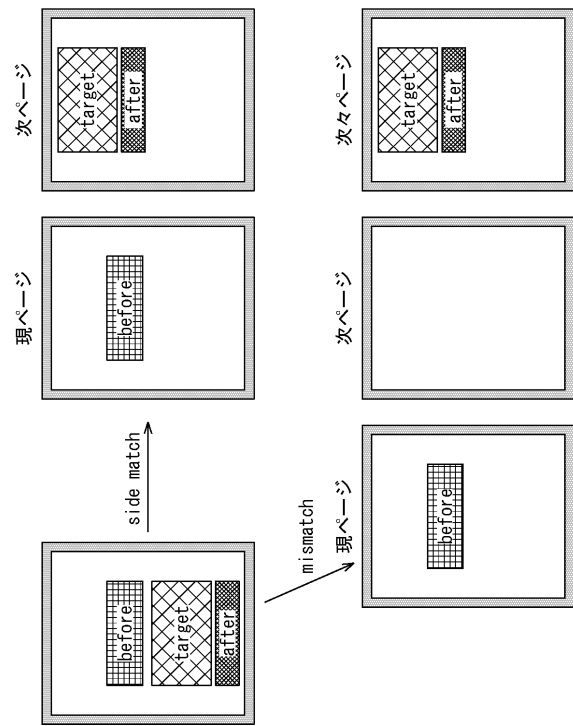




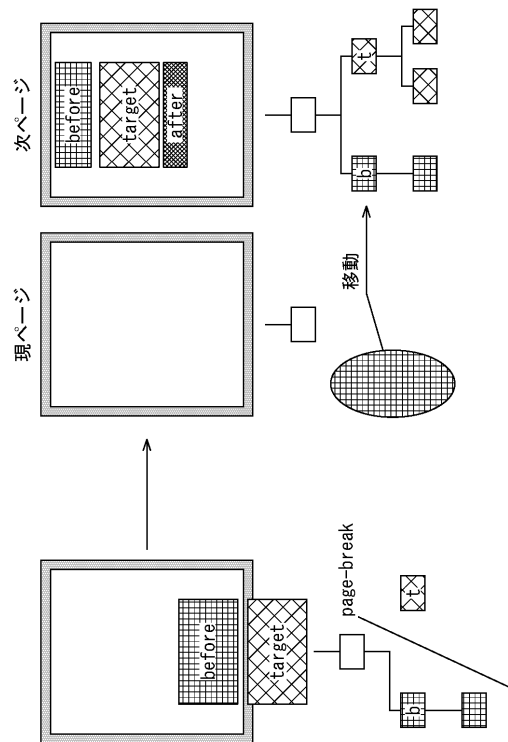
【図 7 - a】



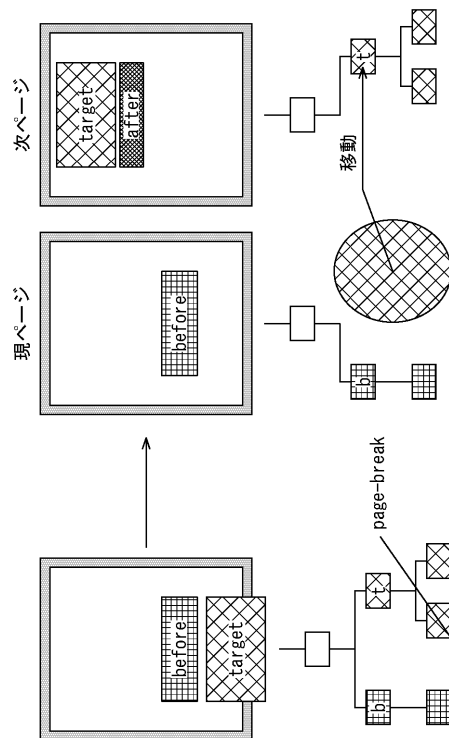
【図 7 - b】



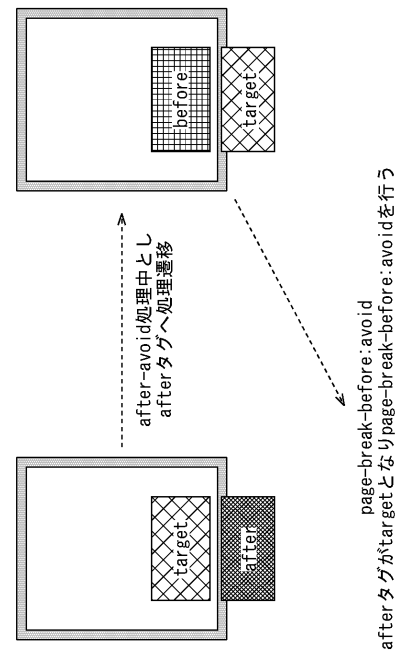
【図 7 - c】



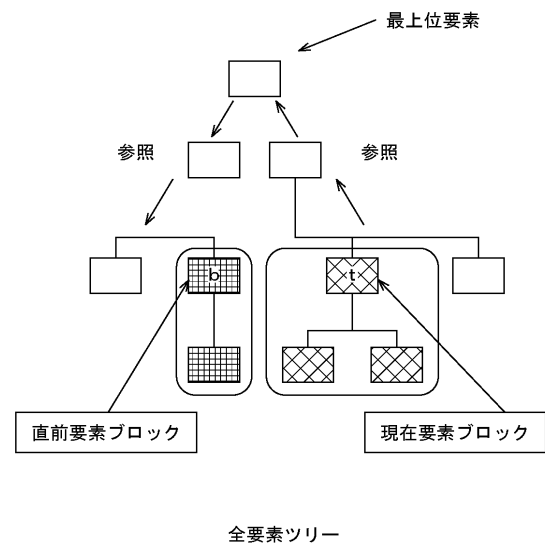
【図 7 - d】



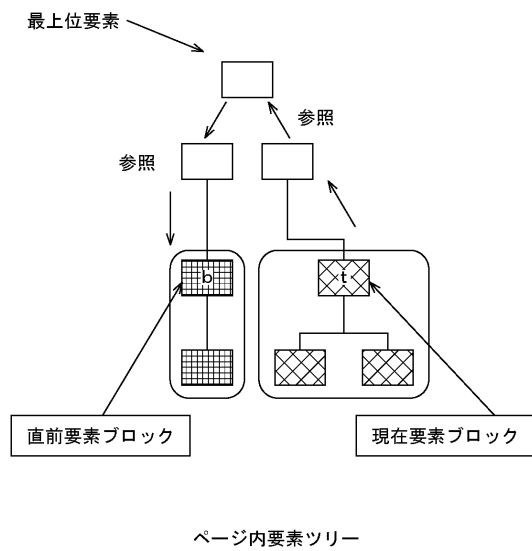
【 図 7 - f 】



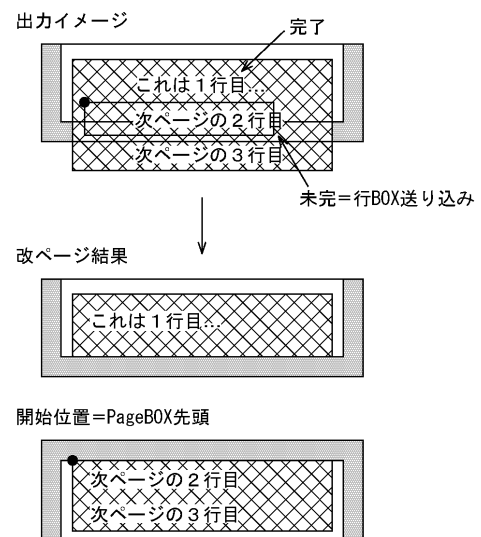
【 図 9 】



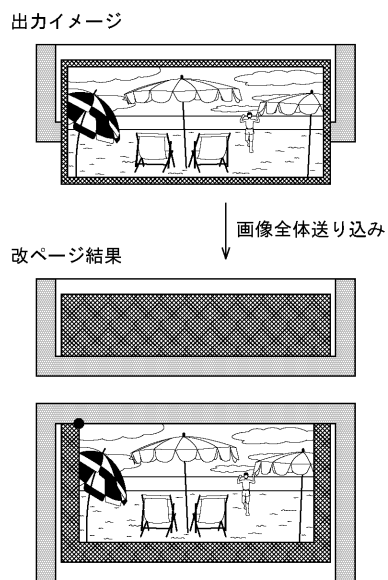
【図 10】



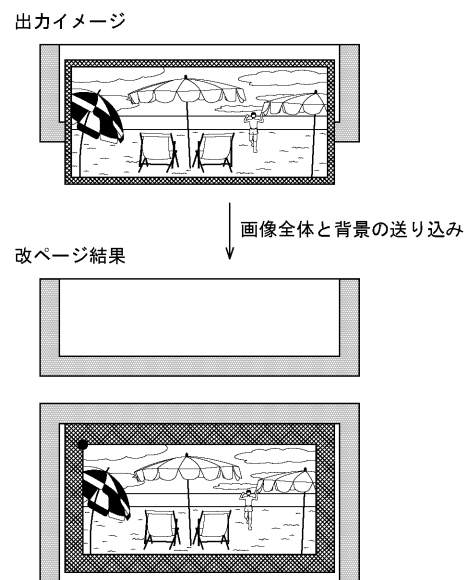
【図 11】



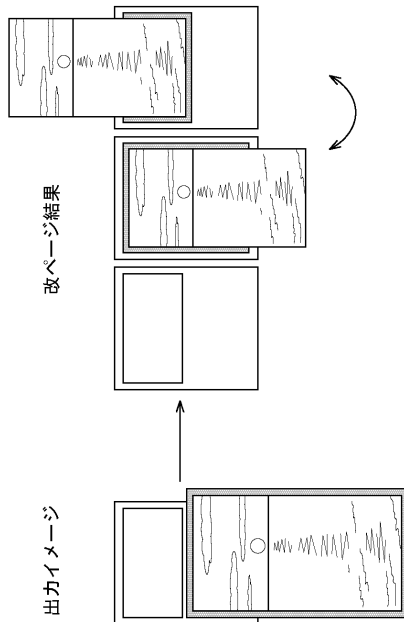
【図 12】



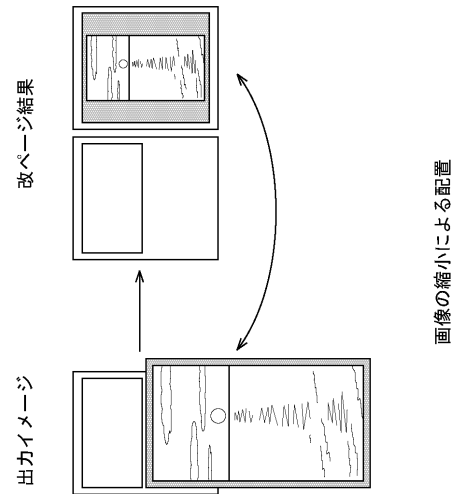
【図 13】



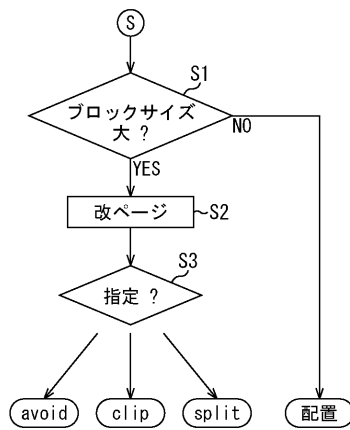
【図 14】



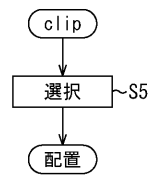
【図 15】



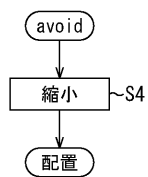
【図 16】



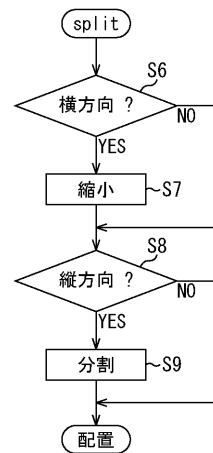
【図 18】



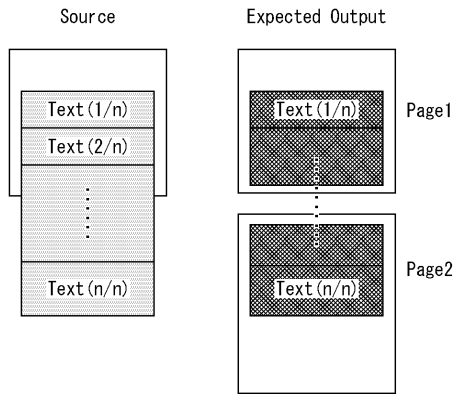
【図 17】



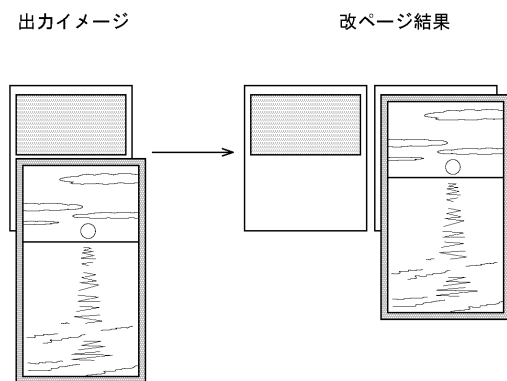
【図 19】



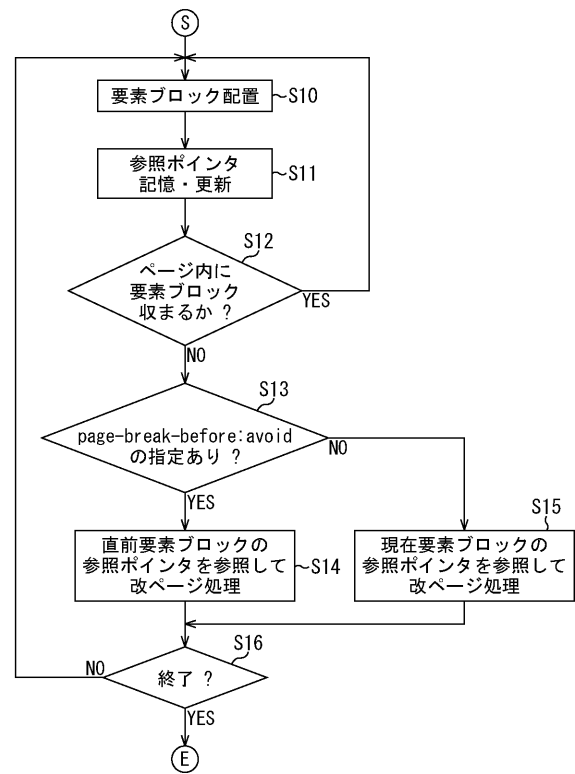
【図 20】



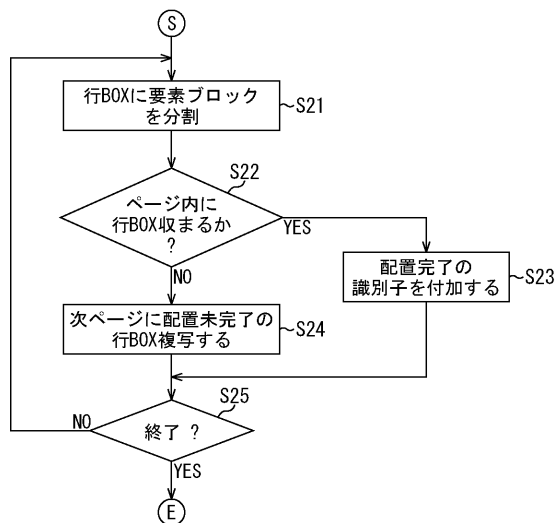
【図 21】



【図 22】



【図 23】



---

フロントページの続き

(56)参考文献 特開平 1 1 - 0 1 5 7 4 9 ( J P , A )  
特開昭 6 3 - 0 3 7 4 6 0 ( J P , A )  
特開 2 0 0 4 - 0 2 1 3 8 3 ( J P , A )  
特開 2 0 0 4 - 1 3 9 4 6 6 ( J P , A )  
特開平 0 5 - 0 3 5 7 3 6 ( J P , A )  
特開平 0 1 - 3 1 8 1 4 5 ( J P , A )

(58)調査した分野(Int.Cl. , D B 名)

B 4 1 J      2 1 / 0 0  
G 0 6 F      3 / 1 2  
H 0 4 N      1 / 3 8 - 1 / 3 9 3  
G 0 6 F      1 7 / 2 1