

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6471660号  
(P6471660)

(45) 発行日 平成31年2月20日 (2019. 2. 20)

(24) 登録日 平成31年2月1日 (2019. 2. 1)

(51) Int. Cl.	F I
<b>G06F 3/12 (2006.01)</b>	G06F 3/12 3 2 4
<b>B41J 5/30 (2006.01)</b>	G06F 3/12 3 2 0
	G06F 3/12 3 4 3
	G06F 3/12 3 1 1
	G06F 3/12 3 4 7
	請求項の数 5 (全 17 頁) 最終頁に続く

(21) 出願番号 特願2015-182847 (P2015-182847)  
 (22) 出願日 平成27年9月16日 (2015. 9. 16)  
 (65) 公開番号 特開2017-58926 (P2017-58926A)  
 (43) 公開日 平成29年3月23日 (2017. 3. 23)  
 審査請求日 平成30年7月17日 (2018. 7. 17)

(73) 特許権者 000005496  
 富士ゼロックス株式会社  
 東京都港区赤坂九丁目7番3号  
 (74) 代理人 110000752  
 特許業務法人朝日特許事務所  
 (72) 発明者 原 裕樹  
 神奈川県横浜市西区みなとみらい六丁目1  
 番 富士ゼロックス株式会社内

審査官 豊田 真弓

(56) 参考文献 特開2015-001866 (JP, A)  
 )  
 特開平08-337014 (JP, A)

最終頁に続く

(54) 【発明の名称】 情報処理装置及びプログラム

(57) 【特許請求の範囲】

【請求項1】

複数のページの画像を表す印刷データを描画データに変換する際に前記複数のページにわたって共通に利用される共通描画資源を記憶する第1の記憶部から、前記共通描画資源を分割して分割描画資源として順次取得し、第2の記憶部に記憶する描画資源取得部と、

前記第2の記憶部に記憶された分割描画資源を利用して、画像形成装置において取り扱い可能な形式の描画データを生成する描画データ生成部と、

前記描画資源取得部により一の分割描画資源が取得されて前記第2の記憶部に記憶された場合に、当該一の分割描画資源が前記描画データ生成部により利用されて描画データが生成された後であり、かつ、前記描画資源取得部により前記一の分割描画資源の後に取得された他の分割描画資源が前記第2の記憶部に記憶される前に、当該一の分割描画資源の記憶領域の全部又は一部を解放する記憶領域開放部と

を備える情報処理装置。

【請求項2】

前記描画資源取得部は、前記描画データ生成部による前記一の分割描画資源に基づく描画データの生成が完了する前に、前記他の分割描画資源の取得を開始することを特徴とする請求項1に記載の情報処理装置。

【請求項3】

前記共通描画資源は、入れ子状に階層化された複数の部分共通描画資源により構成され

前記描画資源取得部は、前記第1の記憶部から、前記複数の部分共通描画資源の各々を分割して分割描画資源として順次取得し、前記第2の記憶部に記憶し、

前記複数の部分共通描画資源のうち前記描画資源取得部により最初に取得される部分共通描画資源を起算点として $n$ 番目の階層の部分共通描画資源から分割される分割描画資源のサイズは、 $n + 1$ 番目階層の部分共通描画資源から分割される分割描画資源のサイズよりも大きい

ことを特徴とする請求項1又は2に記載の情報処理装置。

【請求項4】

前記複数の部分共通描画資源のうち $n$ 番目の階層の部分共通描画資源から分割される分割描画資源のサイズは、予め定められた値を2の $n$ 乗で除することにより求められる値以下であることを特徴とする請求項3に記載の情報処理装置。

10

【請求項5】

コンピュータを、

複数のページの画像を表す印刷データを描画データに変換する際に前記複数のページにわたって共通に利用される共通描画資源を記憶する第1の記憶部から、前記共通描画資源を分割して分割描画資源として順次取得し、第2の記憶部に記憶する描画資源取得部と、

前記第2の記憶部に記憶された分割描画資源を利用して、画像形成装置において取り扱可能な形式の描画データを生成する描画データ生成部と、

前記描画資源取得部により一の分割描画資源が取得されて前記第2の記憶部に記憶された場合に、当該一の分割描画資源が前記描画データ生成部により利用されて描画データが生成された後であり、かつ、前記描画資源取得部により前記一の分割描画資源の後に取得された他の分割描画資源が前記第2の記憶部に記憶される前に、当該一の分割描画資源の記憶領域の全部又は一部を解放する記憶領域開放部

20

として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報処理装置及びプログラムに関する。

【背景技術】

【0002】

印刷ジョブでページ共通のフォーム又はイメージを使用するいわゆるバリエブル印刷では、印刷データをページ共通の再利用可能な共通部分と可変部分とに分けて解釈し、各々の中間データを出力する。中間データをビットマップ画像データに展開する際には、可変部分から参照される共通部分を収集し結合してゆく。

30

【0003】

バリエブル印刷の際に、必要とされるメモリの容量を低減したり中間データの転送量を低減したりするための技術が従来提案されている。例えば、特許文献1では、フォーム情報をビットマップに展開して圧縮し、キャッシュとして記憶しておき、フォームオーバーレイが指示された時に、その他の画像データと記憶されているフォーム情報とを合成して画像を形成できる画像形成方法が提案されている。特許文献2では、フォームデータをコンピュータ端末装置であらかじめ印刷装置固有の色変換および $n$ 値化まで実施した後に保存し、フォームデータの情報量低減を図るようにした印刷処理方法が提案されている。特許文献3では、印刷作業が行われない時期に印刷装置側にバリエブルデータをストアすることで、印刷時にバリエブルデータの転送量を減らすことができ、バリエブル印刷の高速化を可能にするバリエブル印刷システムが提案されている。

40

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特開平6 - 106810号公報

【特許文献2】特開2003 - 319167号公報

50

【特許文献3】特開2008-33634号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

中間データをビットマップ画像データに展開する際に可変部分から参照される共通部分は、キャッシュ利用されるフォームやイメージ等で構成されるため、可変部分に対してデータサイズが大きくなることがある。

【0006】

本発明は、かかる事情に鑑みてなされたものであり、印刷データを描画データに変換する際に複数のページにわたって共通に利用される共通描画資源を記憶するための記憶領域のサイズを抑制することを目的とする。

10

【課題を解決するための手段】

【0007】

上記の課題を解決するため、本発明の請求項1に係る情報処理装置は、複数のページの画像を表す印刷データを描画データに変換する際に前記複数のページにわたって共通に利用される共通描画資源を記憶する第1の記憶部から、前記共通描画資源を分割して分割描画資源として順次取得し、第2の記憶部に記憶する描画資源取得部と、前記第2の記憶部に記憶された分割描画資源を利用して、画像形成装置において取り扱い可能な形式の描画データを生成する描画データ生成部と、前記描画資源取得部により一の分割描画資源が取得されて前記第2の記憶部に記憶された場合に、当該一の分割描画資源が前記描画データ生成部により利用されて描画データが生成された後であり、かつ、前記描画資源取得部により前記一の分割描画資源の後に取得された他の分割描画資源が前記第2の記憶部に記憶される前に、当該一の分割描画資源の記憶領域の全部又は一部を解放する記憶領域開放部とを備える。

20

【0008】

本発明の請求項2に係る情報処理装置は、請求項1に記載の構成において、前記描画資源取得部は、前記描画データ生成部による前記一の分割描画資源に基づく描画データの生成が完了する前に、前記他の分割描画資源の取得を開始することを特徴とする。

【0009】

本発明の請求項3に係る情報処理装置は、請求項1又は2に記載の構成において、前記共通描画資源は、入れ子状に階層化された複数の部分共通描画資源により構成され、前記描画資源取得部は、前記第1の記憶部から、前記複数の部分共通描画資源の各々を分割して分割描画資源として順次取得し、前記第2の記憶部に記憶し、前記複数の部分共通描画資源のうち前記描画資源取得部により最初に取得される部分共通描画資源を起算点としてn番目の階層の部分共通描画資源から分割される分割描画資源のサイズは、n+1番目階層の部分共通描画資源から分割される分割描画資源のサイズよりも大きいことを特徴とする。

30

【0010】

本発明の請求項4に係る情報処理装置は、請求項3に記載の構成において、前記複数の部分共通描画資源のうちn番目の階層の部分共通描画資源から分割される分割描画資源のサイズは、予め定められた値を2のn乗で除することにより求められる値以下であることを特徴とする。

40

【0011】

本発明の請求項5に係るプログラムは、コンピュータを、複数のページの画像を表す印刷データを描画データに変換する際に前記複数のページにわたって共通に利用される共通描画資源を記憶する第1の記憶部から、前記共通描画資源を分割して分割描画資源として順次取得し、第2の記憶部に記憶する描画資源取得部と、前記第2の記憶部に記憶された分割描画資源を利用して、画像形成装置において取り扱い可能な形式の描画データを生成する描画データ生成部と、前記描画資源取得部により一の分割描画資源が取得されて前記第2の記憶部に記憶された場合に、当該一の分割描画資源が前記描画データ生成部により

50

利用されて描画データが生成された後であり、かつ、前記描画資源取得部により前記一の分割描画資源の後に取得された他の分割描画資源が前記第2の記憶部に記憶される前に、当該一の分割描画資源の記憶領域の全部又は一部を解放する記憶領域開放部として機能させる。

【発明の効果】

【0012】

請求項1又は5に係る発明によれば、印刷データを描画データに変換する際に複数のページにわたって共通に利用される共通描画資源を記憶するための記憶領域のサイズを、共通描画資源全体が取得されてから描画データが生成される場合と比較して、抑制することができる。

10

請求項2に係る発明によれば、一の分割描画資源に基づく描画データの生成が完了してから他の分割描画資源の取得が開始される場合と比較して、共通描画資源に基づく描画データの生成に要する時間が低減される。

請求項3又は4に係る発明によれば、共通描画資源を記憶するための記憶領域の上限が定められる。

【図面の簡単な説明】

【0013】

【図1】印刷システム1の構成の一例を示す図である。

【図2】RIP部41とBEP部51の構成の一例を示す図である。

【図3】ラスタライズ処理の一例を示すフローチャートである。

20

【図4】共通リソース取得処理の一例を示すフローチャートである。

【図5】ページデータを構成する各オブジェクトがラスタライズされる過程の一例を示す図である。

【図6】ネスト構造を有する共通リソースの一例を示す図である。

【図7】ラスタライズ処理の一例を示すフローチャートである。

【図8】共通リソース取得処理の一例を示すフローチャートである。

【発明を実施するための形態】

【0014】

以下、本発明の実施の形態について図面を参照しつつ説明する。

1. 実施形態

30

1-1. 構成

図1は、本発明の一実施形態に係る印刷システム1の構成の一例を示す図である。印刷システム1は、複数のクライアント装置10と、印刷装置20と、印刷画像処理システム100とを備える。印刷画像処理システム100は、中央制御処理装置30、解釈処理装置40及びバックエンド処理装置50を備える。クライアント装置10は、中央制御処理装置30と通信回線60を介して接続され、中央制御処理装置30は、解釈処理装置40と通信回線70を介して接続され、解釈処理装置40は、バックエンド処理装置50と通信回線80を介して接続され、バックエンド処理装置50は、印刷装置20と通信回線90を介して接続される。通信回線60、70、80及び90は、例えば、有線又は無線のLAN(Local Area Network)である。なお、解釈処理装置40、バックエンド処理装置50はソフトウェアによって実現することも可能で、そのような場合の通信回線はいわゆるプロセス間通信になる。

40

【0015】

クライアント装置10は、印刷画像処理システム100に対して印刷データを含む印刷指示を送信するコンピュータ装置である。クライアント装置10から送信される印刷データは、ページ記述言語(Page Description Language、略称PDL)で記述される。

【0016】

印刷装置20は、用紙等の記録媒体に画像を形成する画像形成装置である。印刷装置20は、例えば、連続紙プリンタやカット紙プリンタである。印刷装置20の印刷方式は、例えば、電子写真方式やインクジェット方式である。

50

## 【 0 0 1 7 】

印刷画像処理システム 100 を構成する中央制御処理装置 30、解釈処理装置 40 及びバックエンド処理装置 50 は、各々コンピュータ装置である。各装置は、制御部と、記憶部と、通信部とを備える。制御部は、CPU (Central Processing Unit) と、ROM (Read Only Memory) と、RAM (Random Access Memory) とを備える。CPU は、ROM 又は記憶部から RAM 上にプログラムをロードして実行することにより、後述する各種の機能を実現する。記憶部は、ハードディスクドライブ等の記憶装置である。記憶部は、CPU により実行されるプログラムや各種のデータを記憶する。通信部は、データ通信カード等の通信インタフェースである。通信部は、他の装置の通信を制御する。以下、各装置について個別に説明する。

10

## 【 0 0 1 8 】

中央制御処理装置 30 は、CEP (Central Processor) 部 31 を備える。CEP 部 31 は、中央制御処理装置 30 の制御部により実現される機能である。CEP 部 31 は、クライアント装置 10 から印刷データを含んだ印刷指示を受信すると、その印刷指示に応じた印刷処理を実行するために、当該印刷データを解釈処理装置 40 に対して送信する。当該印刷データは、PDL で記述されたデータである。

## 【 0 0 1 9 】

解釈処理装置 40 は、RIP (Raster Image Processor) 部 41 を備える。RIP 部 41 は、解釈処理装置 40 の制御部により実現される機能である。RIP 部 41 は、CEP 部 31 から受信した印刷データを解釈して、印刷データが表す各ページの画像を表現する中間データを生成する。そして、生成した各ページの中間データをバックエンド処理装置 50 に送信する。解釈処理装置 40 は複数設けられて、並列処理が行われてもよい。

20

## 【 0 0 2 0 】

バックエンド処理装置 50 は、BEP (Back End Processor) 部 51 を備える。BEP 部 51 は、バックエンド処理装置 50 の制御部により実現される機能である。BEP 部 51 は、RIP 部 41 から受信した各ページの中間データを処理して、それら各ページのラスタデータを生成する。BEP 部 51 は、生成したラスタデータを印刷装置 20 に送信して印刷を実行させる。バックエンド処理装置 50 は、本発明に係る「情報処理装置」の一例である。バックエンド処理装置 50 は複数設けられて、並列処理が行われてもよい。

## 【 0 0 2 1 】

図 2 は、RIP 部 41 と BEP 部 51 の構成の一例を示す図である。RIP 部 41 は、印刷データ取得部 411 と、印刷データ解釈部 412 と、スプール 413 と、ページデータ送信部 414 と、共通リソース送信部 415 とを備える。BEP 部 51 は、ページデータ取得部 511 と、共通リソース取得部 512 と、バッファ 513 と、ラスタデータ生成部 514 と、ページメモリ 515 と、記憶領域開放部 516 と、ラスタデータ送信部 517 とを備える。

30

## 【 0 0 2 2 】

BEP 部 51 の印刷データ取得部 411 は、CEP 部 31 から送信される印刷データを取得する。この印刷データは、PDL で記述されている。

## 【 0 0 2 3 】

印刷データ解釈部 412 は、印刷データ取得部 411 により取得された印刷データを解釈して、印刷データが表す各ページについて中間データ (以下、「ページデータ」という。) を生成する。

40

## 【 0 0 2 4 】

印刷データが表す各ページは、1 以上のオブジェクト (言い換えると、画像要素) を含む。オブジェクトには、例えば、文字列、グラフィックス (線画、図形)、イメージ (連続調画像) などの種類がある。ページ中に含まれるオブジェクトの中には、再利用されるもの、すなわち印刷データの全ページの画像の中に複数回現れるものがある。例えば、PostScript (登録商標) の場合では、フォームやイメージなどといったタイプのオブジェクトがその一例である。印刷データ解釈部 412 は、印刷データの解釈中に、そのような再

50

利用されるオブジェクトを検出すると、当該オブジェクトの中間データ（以下、「共通リソース」という。）を生成して、スプール413に保存する。そして、共通リソースの代わりに、当該共通リソースを特定するための特定情報をページデータに記述する。ここで、特定情報とは、例えば、当該共通リソースの格納場所を示すURL（Uniform Resource Locator）である。なお、印刷データ解釈部412は、例えば印刷データ中の描画命令の種類に基づいて、当該描画命令が表すオブジェクトが再利用されるタイプか否かを判定する。

**【0025】**

印刷データ解釈部412の分割部4121は、スプール413に保存される共通リソースのサイズが、予め定められたサイズ（以下、「システム固定値」という。）よりも大きい場合には、システム固定値よりも小さくなるように分割してからスプール413に保存する。ここで、共通リソースが分割されて生成される各データを、以下の説明では、「分割共通リソース」という。ただし、特に、分割されない共通リソースと区別する必要がない場合には、単に「共通リソース」という。印刷データ解釈部412は、分割部4121により共通リソースが分割されてスプール413に保存された場合には、分割されて生成された各分割共通リソースを特定する特定情報をページデータに記述する。なお、上記のシステム固定値は、例えば、256MBである。

10

**【0026】**

印刷データ解釈部412は、印刷データの解釈中に、すでにその中間データがスプール413に保存されている再利用オブジェクトになっていることを検出した場合には、当該オブジェクトの中間データの保存を省略してもよい。その場合、印刷データ解釈部412は、すでにスプール413に保存されている共通リソースを特定する特定情報をページデータに記述する。印刷データ解釈部412は、例えば、印刷データに含まれるオブジェクトのIDを参照することにより、すでにその中間データがスプール413に保存されているか否かを判定する。

20

**【0027】**

ページデータ送信部414は、印刷データ解釈部412により生成されたページデータをBEP部51に対して送信する。

**【0028】**

共通リソース送信部415は、BEP部51からの送信要求を受けて、共通リソースをスプール413から読み出してBEP部51に対して送信する。

30

**【0029】**

次に、BEP部51のページデータ取得部511は、RIP部41から送信されるページデータを取得する。

**【0030】**

共通リソース取得部512は、共通リソースを記憶するRIP部41のスプール413から、共通リソースを分割して分割共通リソースとして順次取得し、バッファ513に記憶する。ここで、スプール413は、本発明に係る「第1の記憶部」の一例であり、バッファ513は、本発明に係る「第2の記憶部」の一例である。共通リソース取得部512は、本発明に係る「描画資源取得部」の一例である。

40

**【0031】**

共通リソース取得部512は、ラスタデータ生成部514からの取得要求を受けて、RIP部41のスプール413から、共通リソースを分割して順次取得して、バッファ513に記憶する。

**【0032】**

共通リソース取得部512は、ラスタデータ生成部514による一の分割共通リソースに基づくラスタデータの生成が完了する前に、他の分割共通リソースの取得を開始する。

**【0033】**

ここで、ラスタデータの生成が完了する前とは、具体的には、ラスタデータがページメモリ515に書き込まれる前である。または、ラスタデータの生成が開始される前である

50

。他の分割共通リソースは、例えば、一の分割共通リソースと同一の共通リソースから分割されて生成された分割共通リソースである。他の分割共通リソースは、必ずしも一の分割共通リソースの直後に共通リソース取得部 5 1 2 により取得されたものに限られない。分割共通リソースの取得を開始するとは、具体的には、共通リソース取得部 5 1 2 が、R I P 部 4 1 の共通リソース送信部 4 1 5 に対して送信要求を送信することをいう。または、共通リソース取得部 5 1 2 が、共通リソース送信部 4 1 5 からの共通リソースの受信を開始することをいう。

**【 0 0 3 4 】**

ラスタデータ生成部 5 1 4 は、ページデータ取得部 5 1 1 により取得されたページデータを解釈して、各ページのラスタデータを生成する（すなわち、ラスタライズする）。具体的には、ページデータを構成する各オブジェクトを解釈して、オブジェクトごとにラスタデータを生成する。その際、ラスタデータ生成部 5 1 4 は、ページデータ内のデータだけで完結するオブジェクトについては、そのままラスタライズを行う。一方、共通リソースを参照するオブジェクトについては、共通リソース取得部 5 1 2 を介して分割取得される共通リソースを利用してラスタライズを行う。ラスタデータ生成部 5 1 4 は、オブジェクトの解釈の際に、共通リソースを特定する特定情報を検出すると、共通リソース取得部 5 1 2 に対して当該共通リソースの取得要求を発行する。ラスタデータ生成部 5 1 4 は、ラスタデータを生成すると、生成したラスタデータをページメモリ 5 1 5 に書き込む。ラスタデータ生成部 5 1 4 は、本発明に係る「描画データ生成部」の一例である。

**【 0 0 3 5 】**

記憶領域開放部 5 1 6 は、バッファ 5 1 3 の記憶領域の全部又は一部を開放する。ここで、記憶領域を開放するとは、当該記憶領域に新たなデータが記憶可能な状態にすることをいう。例えば、当該記憶領域に記憶されるデータを削除することをいう。または、当該記憶領域に記憶されるデータに対して上書きが可能な状態にすることをいう。

**【 0 0 3 6 】**

記憶領域開放部 5 1 6 は、共通リソース取得部 5 1 2 により一の分割共通リソースが取得されてバッファ 5 1 3 に記憶された場合に、当該一の分割共通リソースがラスタデータ生成部 5 1 4 により利用されてラスタデータが生成された後であり、かつ、共通リソース取得部 5 1 2 により当該一の分割共通リソースの後に取得された他の分割共通リソースがバッファ 5 1 3 に記憶される前に、当該一の分割共通リソースの記憶領域の全部又は一部を解放する。

**【 0 0 3 7 】**

ここで、ラスタデータが生成された後とは、具体的には、ラスタデータがページメモリ 5 1 5 に書き込まれた後である。他の分割共通リソースは、例えば、一の分割共通リソースと同じ共通リソースから分割された分割共通リソースである。他の分割共通リソースは、必ずしも一の分割共通リソースが取得された直後に共通リソース取得部 5 1 2 により取得されたものに限られない。

**【 0 0 3 8 】**

ラスタデータ送信部 5 1 7 は、ページメモリ 5 1 5 に書き込まれたラスタデータを印刷装置 2 0 に対して送信する。

**【 0 0 3 9 】****1 - 2 . 動作**

次に、B E P 部 5 1 の動作について説明する。具体的には、B E P 部 5 1 のラスタデータ生成部 5 1 4 により実行されるラスタライズ処理と、共通リソース取得部 5 1 2 により実行される共通リソース取得処理について説明する。図 3 は、ラスタライズ処理の一例を示すフローチャートであり、図 4 は、共通リソース取得処理の一例を示すフローチャートである。共通リソース取得部 5 1 2 とラスタデータ生成部 5 1 4 とは、同一のページデータを参照して各処理を実行する。

**【 0 0 4 0 】**

まず、ラスタデータ生成部 5 1 4 は、共通リソース取得要求を共通リソース取得部 5 1

10

20

30

40

50

2 に対して発行する (ステップ S a 1)。この共通リソース取得要求は、共通リソース取得部 5 1 2 に対する割り込み要求として作用する。

【 0 0 4 1 】

次に、ラスタデータ生成部 5 1 4 は、ページデータを構成するオブジェクトを 1 つ読み込む (ステップ S a 2)。その際、ラスタデータ生成部 5 1 4 は、例えば、ページデータの先頭からオブジェクトの読み込みを開始する。

【 0 0 4 2 】

ラスタデータ生成部 5 1 4 は、読み込んだオブジェクトが、共通リソースを参照するオブジェクトであるか否かを判断する (ステップ S a 3)。この判断の結果、読み込んだオブジェクトが、共通リソースを参照しないオブジェクトである場合には (ステップ S a 3 : N O)、ラスタデータ生成部 5 1 4 は、当該オブジェクトをラスタライズして、生成したラスタデータをページメモリ 5 1 5 に書き込む (ステップ S a 4)。

10

【 0 0 4 3 】

次に、ラスタデータ生成部 5 1 4 は、読み込んだオブジェクトが、ページデータにおいて最後のオブジェクトであるか否かを判断する (ステップ S a 5)。この判断の結果、読み込んだオブジェクトが最後のオブジェクトでない場合には (ステップ S a 5 : N O)、ラスタデータ生成部 5 1 4 はステップ S a 2 に戻り、次のオブジェクトを読み込む。一方、この判断の結果、読み込んだオブジェクトが最後のオブジェクトである場合には (ステップ S a 5 : Y E S)、ラスタデータ生成部 5 1 4 は、終了要求を割り込み要求として共通リソース取得部 5 1 2 に対して発行して (ステップ S a 6)、本処理を終了する。

20

【 0 0 4 4 】

上記のステップ S a 3 の判断の結果、読み込んだオブジェクトが、共通リソースを参照するオブジェクトである場合には (ステップ S a 3 : Y E S)、ラスタデータ生成部 5 1 4 は、共通リソース取得部 5 1 2 から、共通リソース取得通知を取得しているか否かを判断する (ステップ S a 7)。この判断の結果、まだ共通リソース取得通知を取得していない場合には (ステップ S a 7 : N O)、ラスタデータ生成部 5 1 4 は待機する。一方、この判断の結果、共通リソース取得通知を取得している場合には (ステップ S a 7 : Y E S)、ラスタデータ生成部 5 1 4 は、次の共通リソースの取得要求を共通リソース取得部 5 1 2 に対して発行する (ステップ S a 8)。ここで、ラスタデータ生成部 5 1 4 は、すでに取得された共通リソースに基づいてラスタデータを生成する前に、次の共通リソースの取得を共通リソース取得部 5 1 2 に対して要求する。

30

【 0 0 4 5 】

共通リソースの取得要求の発行後、ラスタデータ生成部 5 1 4 は、共通リソース取得部 5 1 2 により取得された共通リソースをバッファ 5 1 3 から取得する。そして、当該共通リソースを解釈してラスタデータを生成し、ページメモリ 5 1 5 に書き込む (ステップ S a 4)。なお、バッファ 5 1 3 における当該共通リソースの記憶領域は、ラスタデータのページメモリ 5 1 5 への書き込み後に、記憶領域開放部 5 1 6 により開放される。ラスタデータを生成すると、ラスタデータ生成部 5 1 4 は、すでに説明したステップ S a 5 を実行する。

以上が、ラスタライズ処理について説明である。

40

【 0 0 4 6 】

次に、共通リソース取得部 5 1 2 の方は、ラスタデータ生成部 5 1 4 から発行される割り込み要求を取得したか否かを判断する (ステップ S b 1)。この判断の結果、割り込み要求を取得していない場合には (ステップ S b 1 : N O)、共通リソース取得部 5 1 2 は待機する。一方、この判断の結果、割り込み要求を取得している場合には (ステップ S b 1 : Y E S)、共通リソース取得部 5 1 2 は、取得した割り込み要求が終了要求であるか否かを判断する (ステップ S b 2)。この判断の結果、取得した割り込み要求が終了要求である場合には (ステップ S b 2 : Y E S)、共通リソース取得部 5 1 2 は本処理を終了する。一方、この判断の結果、取得した割り込み要求が終了要求でない場合には (すなわち、取得した割り込み要求が共通リソース取得要求である場合には) (ステップ S b 2 :

50

NO)、共通リソース取得部512は、ページデータを構成するオブジェクトを1つ読み込む(ステップS b 3)。その際、共通リソース取得部512は、例えば、ページデータの先頭からオブジェクトの読み込みを開始する。

【0047】

共通リソース取得部512は、読み込んだオブジェクトが、共通リソースを参照するオブジェクトであるか否かを判断する(ステップS b 4)。この判断の結果、読み込んだオブジェクトが、共通リソースを参照しないオブジェクトである場合には(ステップS b 4: NO)、共通リソース取得部512は、読み込んだオブジェクトが、ページデータにおいて最後のオブジェクトであるか否かを判断する(ステップS b 5)。この判断の結果、読み込んだオブジェクトが最後のオブジェクトでない場合には(ステップS b 5: NO)、共通リソース取得部512はステップS b 3に戻り、次のオブジェクトを読み込む。一方、この判断の結果、読み込んだオブジェクトが最後のオブジェクトである場合には(ステップS b 5: YES)、ステップS b 1に戻る。

10

【0048】

上記のステップS b 4の判断の結果、読み込んだオブジェクトが、共通リソースを参照するオブジェクトである場合には(ステップS b 4: YES)、共通リソース取得部512は、当該共通リソースを取得する(ステップS b 6)。具体的には、RIP部41に対して共通リソースの送信要求を送信し、RIP部41から共通リソースを取得すると、バッファ513に記憶する。共通リソースを取得すると、共通リソース取得部512は、共通リソース取得通知をラスタデータ生成部514に対して発行する(ステップS b 7)。その後、共通リソース取得部512は、ステップS b 1に戻る。

20

以上が、共通リソース取得処理について説明である。

【0049】

次に、以上説明したラスタライズ処理及び共通リソース取得処理の具体例について説明する。図5は、ページデータを構成する各オブジェクトがラスタライズされる過程の一例を示す図である。同図に示す例では、ページデータを構成するオブジェクトobjのうち、オブジェクトobj4は、共通リソースmng1を参照するオブジェクトであり、オブジェクトobj5は、共通リソースmng2を参照するオブジェクトである。共通リソースmng1及びmng2は、各々分割して共通リソース取得部512により取得される。

【0050】

ラスタデータ生成部514は、先頭のオブジェクトobj1をラスタライズする前に、まず、共通リソース取得部512に対して、共通リソース取得要求を発行する(ステップS a 1)。この取得要求を受けて、共通リソース取得部512は、オブジェクトobj4により参照される分割共通リソースmng1-1をRIP部41から取得する(ステップS b 6)。そして、取得すると、共通リソース取得通知をラスタデータ生成部514に対して発行する(ステップS b 7)。

30

【0051】

ラスタデータ生成部514は、共通リソース取得要求の発行後、共通リソースmngを参照しないオブジェクトobj1~obj3をそれぞれラスタライズする(ステップS a 4)。

40

【0052】

オブジェクトobj1~obj3をラスタライズ後、ラスタデータ生成部514は、オブジェクトobj4については共通リソースmng1を参照するオブジェクトであるから、共通リソースmng1の取得後にラスタライズを行う。具体的には、共通リソース取得部512から、分割共通リソースmng1-1についての取得通知を取得しているか否かを判断し(ステップS a 7)、取得している場合には(ステップS a 7: YES)、次の共通リソースの取得要求を共通リソース取得部512に対して発行する(ステップS a 8)。この取得要求を受けて、共通リソース取得部512は、オブジェクトobj4により参照される分割共通リソースmng1-2をRIP部41から取得する(ステップS b 6)。この取得は、本具体例において、分割共通リソースmng1-1に基づくラスタデー

50

タの生成と並行して実行される。共通リソース取得部512は、分割共通リソースmng1-2を取得すると、共通リソース取得通知をラスタデータ生成部514に対して発行する(ステップSb7)。

**【0053】**

共通リソース取得要求の発行後、ラスタデータ生成部514は、分割共通リソースmng1-1をバッファ513から取得する。そして、分割共通リソースmng1-1を解釈してラスタデータを生成し、ページメモリ515に書き込む(ステップSa4)。生成されたラスタデータがページメモリ515に書き込まれると、記憶領域開放部516は、バッファ513において分割共通リソースmng1-1により占められていた記憶領域を開放する。この記憶領域の開放は、本具体例において、分割共通リソースmng1-3がバッファ513に記憶される前に実行される。

10

**【0054】**

分割共通リソースmng1-1に基づいてラスタデータを生成すると、ラスタデータ生成部514は、次に、分割共通リソースmng1-2についての取得通知を取得しているか否かを判断し(ステップSa7)、取得している場合には(ステップSa7:YES)、次の共通リソースの取得要求を共通リソース取得部512に対して発行する(ステップSa8)。この取得要求を受けて、共通リソース取得部512は、オブジェクトobj4により参照される分割共通リソースmng1-3をRIP部41から取得する(ステップSb6)。この取得は、本具体例において、分割共通リソースmng1-2に基づくラスタデータの生成と並行して実行される。共通リソース取得部512は、分割共通リソースmng1-3を取得すると、共通リソース取得通知をラスタデータ生成部514に対して発行する(ステップSb7)。

20

**【0055】**

共通リソース取得要求の発行後、ラスタデータ生成部514は、分割共通リソースmng1-2をバッファ513から取得する。そして、分割共通リソースmng1-2を解釈してラスタデータを生成し、ページメモリ515に書き込む(ステップSa4)。生成されたラスタデータがページメモリ515に書き込まれると、記憶領域開放部516は、バッファ513において分割共通リソースmng1-2により占められていた記憶領域を開放する。この記憶領域の開放は、本具体例において、分割共通リソースmng2-1がバッファ513に記憶される前に実行される。

30

**【0056】**

分割共通リソースmng1-2に基づいてラスタデータを生成すると、ラスタデータ生成部514は、次に、分割共通リソースmng1-3についての取得通知を取得しているか否かを判断し(ステップSa7)、取得している場合には(ステップSa7:YES)、次の共通リソースの取得要求を共通リソース取得部512に対して発行する(ステップSa8)。この取得要求を受けて、共通リソース取得部512は、オブジェクトobj5により参照される分割共通リソースmng2-1をRIP部41から取得する(ステップSb6)。そして、取得すると、共通リソース取得通知をラスタデータ生成部514に対して発行する(ステップSb7)。

40

**【0057】**

共通リソース取得要求の発行後、ラスタデータ生成部514は、分割共通リソースmng1-3をバッファ513から取得する。そして、分割共通リソースmng1-3を解釈してラスタデータを生成し、ページメモリ515に書き込む(ステップSa4)。この際、オブジェクトobj4により参照される共通リソースmng1はすべて取得済みであるので、ラスタデータ生成部514は当該オブジェクトをラスタライズする。生成されたラスタデータがページメモリ515に書き込まれると、記憶領域開放部516は、バッファ513において分割共通リソースmng1-3により占められていた記憶領域を開放する。

**【0058】**

以上は、オブジェクトobj4をラスタライズするまでの説明であるが、オブジェクト

50

obj 5についてもオブジェクトobj 4と同様にラスタライズ処理が実行される。

【0059】

以上説明したラスタライズ処理及び共通リソース取得処理によれば、RIP部41に記憶される共通リソースは、BEP部51により分割して取得される。そして、解釈されラスタデータが生成された分割共通リソースから順次その記憶領域が開放される。そのため、共通リソース全体が取得されてからその解釈が行われてラスタデータが生成される場合と比較して、共通リソースを記憶するための記憶領域のサイズが小さく済む。

【0060】

また、以上説明したラスタライズ処理及び共通リソース取得処理によれば、BEP部51において、一の分割共通リソースの取得が、他の分割共通リソースのラスタライズと並行して実行される。そのため、一の分割共通リソースのラスタライズが完了してから他の分割共通リソースの取得が開始される場合と比較して、ラスタライズ処理に要する時間が低減される。

【0061】

## 2. 変形例

上記の実施形態は下記のように変形してもよい。また、下記の変形例は他の1以上の変形例と組み合わせてもよい。

【0062】

### 2-1. 変形例1

上記の実施形態において、共通リソースはネスト構造を有してもよい。すなわち、共通リソースは、入れ子状に階層化された複数の共通リソースにより構成されてもよい。図6は、ネスト構造を有する共通リソースの一例を示す図である。

【0063】

同図に示す例において、共通リソースmngはネスト構造を有している。共通リソースmngは、入れ子状に階層化された複数の部分共通リソースmng1、mng2及びmng3により構成されている。部分共通リソースmng1は、ページデータを構成するオブジェクトobj 4により参照され、オブジェクトobj 5～obj 8と、部分共通リソースmng2とにより構成される。部分共通リソースmng2は、オブジェクトobj 8により参照され、オブジェクトobj 9～obj 11と、部分共通リソースmng3とにより構成される。部分共通リソースmng3は、オブジェクトobj 11により参照され、オブジェクトobj 12及びobj 13により構成される。以下の説明では、ページデータにより直接参照される部分共通リソースmng1を第1階層（または、ネスト1）とする。そして、第n階層（または、ネストn）の部分共通リソースmngにより参照される部分共通リソースmngを第n+1階層（または、ネストn+1）とする。

【0064】

RIP部41の印刷データ解釈部412は、図6に示すページデータを生成する場合、参照先の部分共通リソースmngをスプール413に保存するとともに、当該部分共通リソースmngを特定する特定情報を参照元の部分共通リソースmng又はページデータに記述する。具体的には、まず、部分共通リソースmng3をスプール413に保存するとともに、部分共通リソースmng3を特定する特定情報を部分共通リソースmng2に記述する。次に、部分共通リソースmng2をスプール413に保存するとともに、部分共通リソースmng2を特定する特定情報を部分共通リソースmng1に記述する。最後に、部分共通リソースmng1をスプール413に保存するとともに、部分共通リソースmng1を特定する特定情報をページデータに記述する。

【0065】

印刷データ解釈部412の分割部4121は、スプール413に保存される部分共通リソースmngのサイズが、分割サイズよりも大きい場合には、分割サイズよりも小さくなるように分割してからスプール413に保存する。ここで、当該分割サイズは、本変形例では、階層に応じて決定される。第n階層の分割サイズVは下式により求められる。

$$V = X / 2^n$$

10

20

30

40

50

ここで、変数 $X$ は、システム固定値である。例えば、システム固定値を「1GB」とすると、第1階層の分割サイズは「512MB」となり、第2階層の分割サイズは「256MB」となり、第3階層の分割サイズは「128MB」となる。

【0066】

印刷データ解釈部412は、分割部4121により、参照先の部分共通リソース $mng$ が分割されてプール413に保存された場合には、分割されて生成された各分割共通リソースを特定する特定情報を、参照元の部分共通リソース $mng$ 又はページデータに記述する。

【0067】

BE P部51の共通リソース取得部512は、RIP部41のプール413から、部分共通リソース $mng$ の各々を分割して分割共通リソースとして順次取得し、バッファ513に記憶する。部分共通リソース $mng$ のうち共通リソース取得部512により最初に取得される部分共通リソース $mng$ を起算点として $n$ 番目の階層の部分共通リソース $mng$ から分割される分割共通リソースのサイズは、予め定められた値を2の $n$ 乗で除することにより求められる値以下となる。

【0068】

次に、本変形例に係るBE P部51の動作について説明する。図7は、本変形例に係るラスタライズ処理の一例を示すフローチャートである。同図に示すラスタライズ処理は、図3に示すラスタライズ処理と比較して、ステップSc1が追加されている点において相違している。以下ではこの相違点についてのみ説明する。図8は、本変形例に係る共通リソース取得処理の一例を示すフローチャートである。同図に示す共通リソース取得処理は、図4に示す共通リソース取得処理と比較して、ステップSd1が追加されている点において相違している。以下ではこの相違点についてのみ説明する。

【0069】

ラスタデータ生成部514は、共通リソース取得通知を取得すると(ステップSa7: YES)、共通リソース取得部512により取得された共通リソースをバッファ513から取得する。そして、取得した共通リソースについて、図7に示すラスタライズ処理(図7に示すステップSa1~Sa8及びSc1)を実行する。すなわち、ラスタデータ生成部514は、取得した共通リソースを構成する各オブジェクトを読み込んで、共通リソースを参照しないオブジェクトである場合には、そのままラスタライズし、共通リソースを参照するオブジェクトである場合には、共通リソースを取得してからラスタライズする。なお、その際、ラスタデータ生成部514は、取得した共通リソースについて、図7に示すラスタライズ処理(図7に示すステップSa1~Sa8及びSc1)を実行する。このように再帰的にラスタライズ処理が実行されるのは、本変形例に係る共通リソースがネスト構造を有しているからである。

【0070】

共通リソース取得部512は、共通リソースを取得して(ステップSb4: YES)、共通リソース取得通知をラスタデータ生成部514に対して発行すると、(ステップSb7)、取得した共通リソースについて、図8に示す共通リソース取得処理(図8に示すステップSb1~Sb7及びSd1)を実行する。すなわち、共通リソース取得部512は、ラスタデータ生成部514からの取得要求を受けて、取得した共通リソースを構成する各オブジェクトを読み込み、共通リソースを参照するオブジェクトである場合には、その参照される共通リソースを取得する。なお、その際、共通リソース取得部512は、取得した共通リソースについて、図8に示すラスタライズ処理(図8に示すステップSb1~Sb7及びSd1)を実行する。このように再帰的に共通リソース取得処理が実行されるのは、本変形例に係る共通リソースがネスト構造を有しているからである。

【0071】

以上説明した変形例に係るラスタライズ処理及び共通リソース取得処理によれば、ネスト構造の階層が深くなるにつれて、取得される分割共通リソースのサイズが小さくなってゆく。そのため、共通リソースを記憶するための記憶領域の上限を定められる。

10

20

30

40

50

## 【 0 0 7 2 】

## 2 - 2 . 変形例 2

上記の実施形態に係る共通リソースは、本発明に係る「共通描画資源」の一例である。共通描画資源とは、複数のページの画像を表す印刷データを描画データに変換する際に、複数のページにわたって共通に利用される描画資源である。ここで、印刷データは、PDLで記述されたデータである。例えば、PostScript（登録商標）で記述されたデータである。印刷データにより表される各ページは、1以上のオブジェクト（換言すると、画像要素）を含む。オブジェクトには、文字列、グラフィックス（線画、図形等）、イメージ（連続調画像等）などの種類がある。描画データは、画像形成装置において取り扱い可能な形式のデータである。例えば、ラスタ形式のデータである。描画データは、印刷データから中間データを経て生成される。ここで、中間データは、印刷データと描画データの中間の粒度のデータである。中間データでは、PDLで記述されたオブジェクトが単純な形状の微小要素に細分化されて表現される。中間データの形式には、ディスプレイリスト形式が含まれる。

10

## 【 0 0 7 3 】

共通描画資源は、オブジェクトを表すデータである。特に、複数のページ間で再利用されるオブジェクト（すなわち、共通部分）を表すデータである。再利用されるオブジェクトとは、フォームやイメージである。より具体的には、罫線や枠線や企業ロゴ等である。共通描画資源は、オブジェクトの中間データである。

20

## 【 0 0 7 4 】

## 2 - 3 . 変形例 3

上記の実施形態に係る分割共通リソースは、本発明に係る「分割描画資源」の一例である。分割描画資源とは、共通描画資源を構成する一部分である。分割描画資源は、例えば、共通描画資源を分割して生成される。分割描画資源のデータサイズは、予め定められた値以下に設定される。例えば、256MBに設定される。

## 【 0 0 7 5 】

分割描画資源は、可変描画資源により参照される。具体的には、可変描画資源内に記述される特定情報により特定される。特定情報とは、分割描画資源の記憶場所を示す情報であり、例えばURLである。ここで、可変描画資源は、オブジェクトを表すデータである。特に、ページごとに異なるオブジェクト（すなわち、バリエブル部又は可変部分）を表すデータである。可変描画資源は、オブジェクトの中間データである。上記の実施形態に係るページデータは、可変描画資源の一例である。

30

## 【 0 0 7 6 】

## 2 - 4 . 変形例 4

上記の変形例 1 に係る部分共通リソース  $mng$  は、本発明に係る「部分共通描画資源」の一例である。部分共通描画資源とは、入れ子状に階層化された共通描画資源の一部分である。部分共通描画資源は、可変描画資源により参照される。また、部分共通描画資源は、他の部分共通描画資源を参照する。

## 【 0 0 7 7 】

## 2 - 5 . 変形例 5

上記の変形例 1 に係る、部分共通リソース  $mng$  の分割サイズを算出するための計算式は一例にすぎない。第  $n + 1$  階層の分割サイズが第  $n$  階層の分割サイズよりも小さくなる式であれば他の計算式を採用してもよい。

40

## 【 0 0 7 8 】

## 2 - 6 . 変形例 6

上記の実施形態又は変形例において解釈処理装置 40 又はバックエンド処理装置 50 の制御部により実行されるプログラムは、磁気テープ、磁気ディスク、フレキシブルディスク、光ディスク、光磁気ディスク、メモリ等の記憶媒体に記憶された状態で提供されてもよい。また、当該プログラムは、インターネット等の通信回線を介してダウンロードされてもよい。

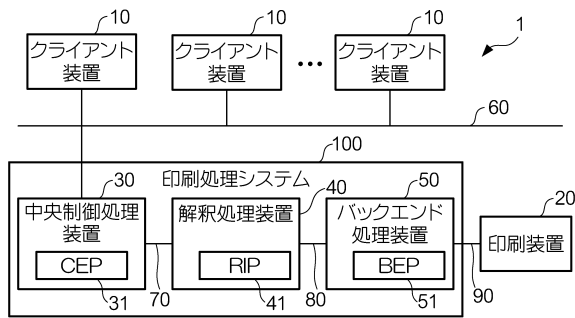
50

【符号の説明】

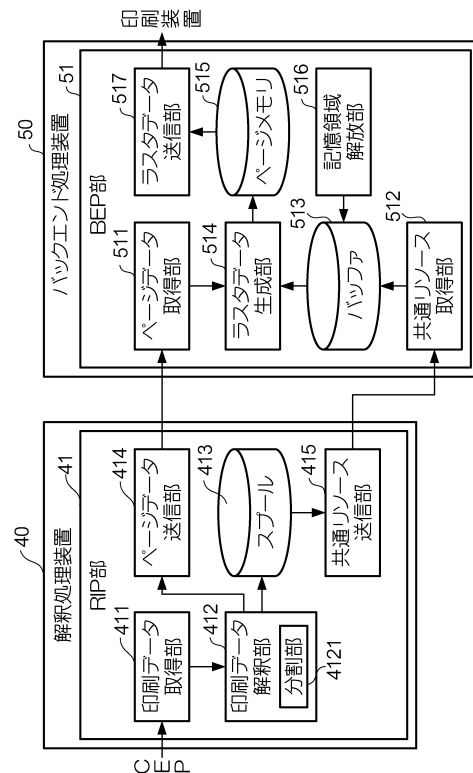
【0079】

1 ...印刷システム、10 ...クライアント装置、20 ...印刷装置、30 ...中央制御処理装置、31 ...CEP部、40 ...解釈処理装置、41 ...RIP部、50 ...バックエンド処理装置、51 ...BEP部、60 ...通信回線、70 ...通信回線、80 ...通信回線、90 ...通信回線、100 ...印刷画像処理システム、411 ...印刷データ取得部、412 ...印刷データ解釈部、413 ...スプール、414 ...ページデータ送信部、415 ...共通リソース送信部、511 ...ページデータ取得部、512 ...共通リソース取得部、513 ...バッファ、514 ...ラスタデータ生成部、515 ...ページメモリ、516 ...記憶領域開放部、517 ...ラスタデータ送信部、4121 ...分割部

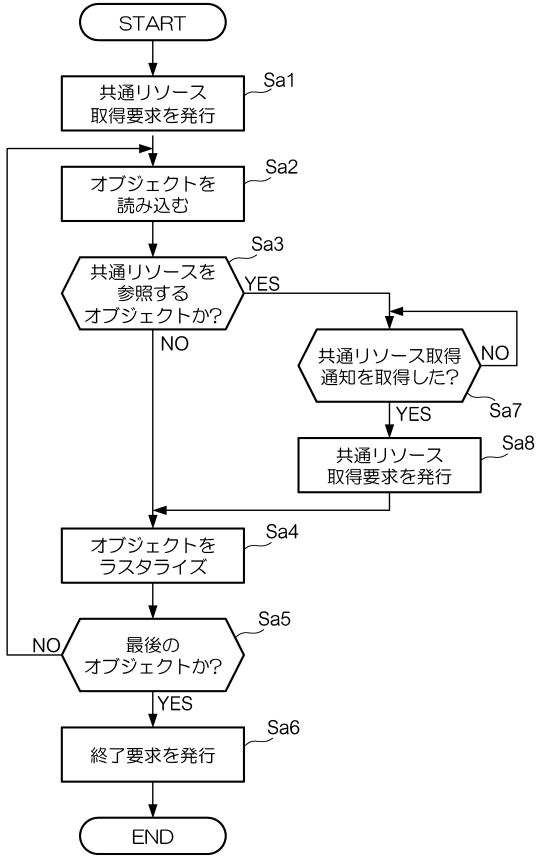
【図1】



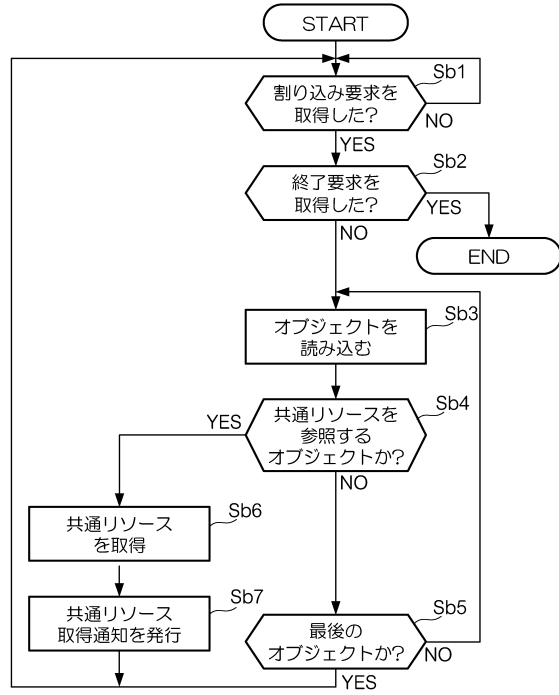
【図2】



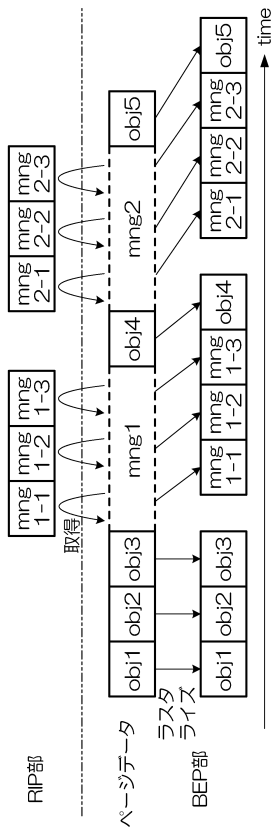
【図3】



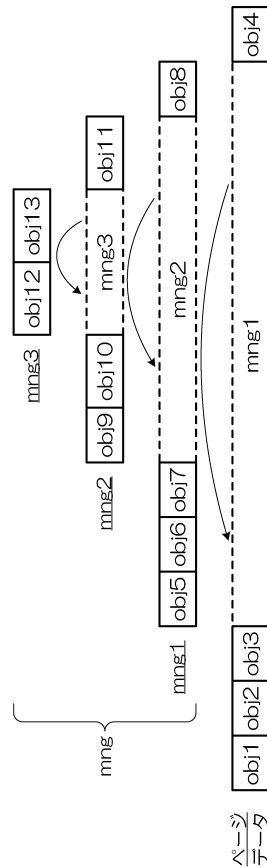
【図4】



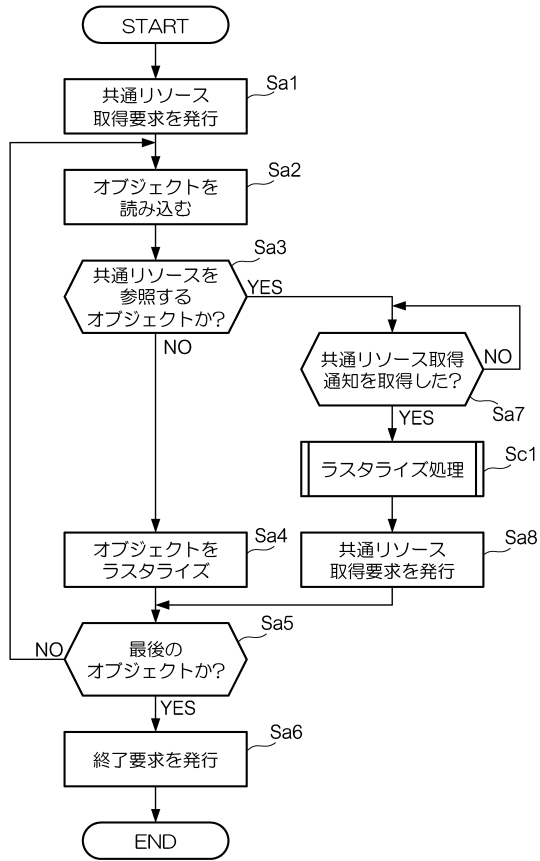
【図5】



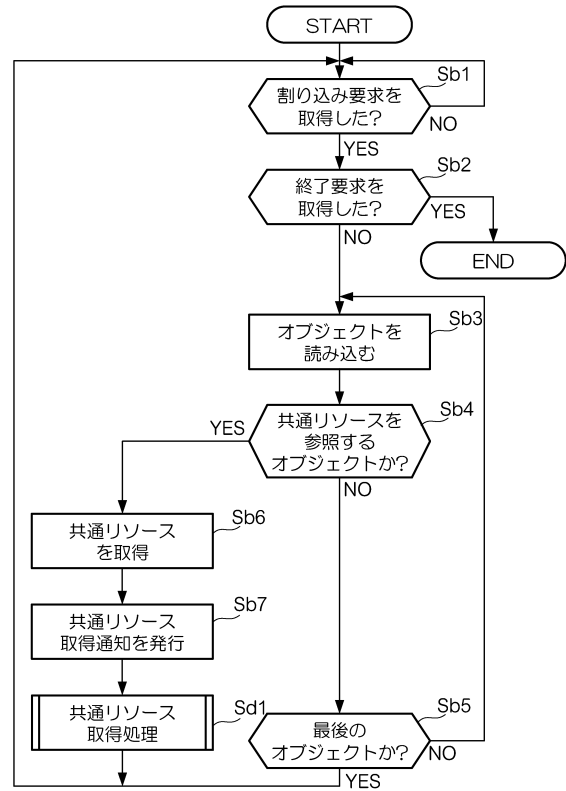
【図6】



【図7】



【図8】



---

フロントページの続き

(51)Int.Cl. F I  
B 4 1 J 5/30 Z

(58)調査した分野(Int.Cl. , D B名)  
G 0 6 F 3 / 1 2  
B 4 1 J 5 / 3 0