

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5319237号
(P5319237)

(45) 発行日 平成25年10月16日 (2013.10.16)

(24) 登録日 平成25年7月19日 (2013.7.19)

(51) Int. Cl.	F I
G 0 6 F 3/12 (2006.01)	G O 6 F 3/12 D
H 0 4 N 1/00 (2006.01)	H O 4 N 1/00 1 O 7 Z
B 4 1 J 29/38 (2006.01)	B 4 1 J 29/38 Z

請求項の数 10 (全 25 頁)

(21) 出願番号	特願2008-277377 (P2008-277377)	(73) 特許権者	000001007
(22) 出願日	平成20年10月28日 (2008.10.28)		キヤノン株式会社
(65) 公開番号	特開2010-108104 (P2010-108104A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成22年5月13日 (2010.5.13)	(74) 代理人	100076428
審査請求日	平成23年10月18日 (2011.10.18)		弁理士 大塚 康德
		(74) 代理人	100112508
			弁理士 高柳 司郎
		(74) 代理人	100115071
			弁理士 大塚 康弘
		(74) 代理人	100116894
			弁理士 木村 秀二
		(74) 代理人	100130409
			弁理士 下山 治
		(74) 代理人	100134175
			弁理士 永川 行光

最終頁に続く

(54) 【発明の名称】 印刷システム及びその制御方法

(57) 【特許請求の範囲】

【請求項 1】

複数のプリントサーバと、少なくとも1つのプリントデバイスとを含む印刷システムであって、

プリントサーバは、

ドキュメントデータを識別するドキュメント識別子を含む印刷ジョブの印刷リクエストを受け付ける受付手段と、

当該印刷ジョブの登録をプリントデバイスに要求する手段と、

前記印刷ジョブの登録を要求した後、当該印刷ジョブを破棄する手段と、

前記プリントデバイスからの指示に応答して、前記指示された印刷ジョブを取得し、当該印刷ジョブに基づき印刷データを生成して前記プリントデバイスに転送する手段とを有し、

前記プリントデバイスは、

前記印刷ジョブの登録の要求を受け付けたことに応じて、前記ドキュメント識別子を含む前記印刷ジョブをスケジュールリストに登録する登録手段と、

前記スケジュールリストに登録された登録内容に基づき、前記複数のプリントサーバのうちのいずれかのプリントサーバに対して、印刷データの生成を指示する指示手段とを有することを特徴とする印刷システム。

【請求項 2】

前記指示手段は、前記プリントデバイスで連続して印刷ジョブが処理されるように予め

10

20

決められたプリントサーバに前記印刷データの生成を指示することを特徴とする請求項 1 に記載の印刷システム。

【請求項 3】

前記プリントデバイスは、前記プリントサーバの負荷状況に基づいて、負荷のかかっていない他のプリントサーバへ前記印刷データの生成の指示を分散させる手段を更に有することを特徴とする請求項 1 に記載の印刷システム。

【請求項 4】

前記指示手段は、指示する印刷ジョブの数が前記プリントデバイスに設定されている連続印刷制限数よりも少ない場合、前記プリントサーバに前記印刷データの生成を指示することを特徴とする請求項 1 に記載の印刷システム。

10

【請求項 5】

前記プリントデバイスは、

前記プリントデバイスと前記プリントサーバとが接続されている状態か否かを判定する手段と、

前記プリントデバイスと前記プリントサーバとが接続されていない状態と判定した場合、前記スケジュール登録の際に通知されるアドレスに印刷結果を通知する手段と、

を更に有することを特徴とする請求項 1 に記載の印刷システム。

【請求項 6】

前記プリントサーバは、印刷結果通知で通知される情報に基づいて印刷結果を登録する手段を更に有することを特徴とする請求項 5 に記載の印刷システム。

20

【請求項 7】

前記指示手段は、前記プリントサーバとの接続が切断された場合、前記プリントデバイスが前記スケジュールリストを参照して指示がなされたか否かを判定し、既に指示がなされている場合に、再度、指示を行うことを特徴とする請求項 1 に記載の印刷システム。

【請求項 8】

複数のプリントサーバと、少なくとも 1 つのプリントデバイスとを含む印刷システムの制御方法であって、

プリントサーバにおいて、

ドキュメントデータを識別するドキュメント識別子を含む印刷ジョブの印刷リクエストを受け付ける受付工程と、

30

当該印刷ジョブの登録をプリントデバイスに要求する工程と、

前記印刷ジョブの登録を要求した後、当該印刷ジョブを破棄する工程と、

前記プリントデバイスからの指示に応答して、前記指示された印刷ジョブを取得し、当該印刷ジョブに基づき印刷データを生成して前記プリントデバイスに転送する工程とを有し、

前記プリントデバイスにおいて、

前記印刷ジョブの登録の要求を受け付けたことに応じて、前記ドキュメント識別子を含む前記印刷ジョブをスケジュールリストに登録する登録工程と、

前記スケジュールリストに登録された登録内容に基づき、前記複数のプリントサーバのうちのいずれかのプリントサーバに対して、印刷データの生成を指示する指示工程とを有することを特徴とする印刷システムの制御方法。

40

【請求項 9】

請求項 8 に記載の印刷システムの制御方法の各工程をコンピュータに実行させるためのプログラム。

【請求項 10】

請求項 9 に記載のプログラムを記録した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、印刷システム及びその制御方法に関するものである。

50

【背景技術】

【0002】

従来、複数台のプリントサーバで印刷処理を分散する技術が提案されている（例えば、特許文献1参照）。

【0003】

この特許文献1によれば、プリントサーバへの印刷処理の振り分けを行う管理サーバが論理デバイスを占有し、印刷リクエストをプリントサーバに割り振る。そして、プリントサーバが割り振られた順番に印刷データを取得し、プリントデバイスに対して配信印刷を行うことが開示されている。

【特許文献1】特開2006-313442号公報

10

【発明の開示】

【発明が解決しようとする課題】

【0004】

全国のおフィスに位置する何千台ものプリントデバイスに対して配信印刷を行う際に、大量の印刷ジョブの制御を1台のプリントサーバで行うと、プリントサーバが高負荷となり、各ジョブの排紙までに多大な時間がかかってしまう。これを回避するためには、複数のプリントサーバでクラスタ化し、負荷分散（ロードバランシング）して稼働させる必要がある。印刷システムがクラスタ構成により分散されるため、プリントサーバ間で共通のジョブ情報やプリンタ情報などの印刷情報を共通ディスクで集中管理を行っている。

【0005】

20

しかし、大規模プリントサーバシステムにおいて、印刷処理が集中してしまう。これを解決するプリントサーバシステムの必須要件として印刷順序制御がある。プリントサーバからプリントデバイスへの印刷実行時に、プリントデバイスとプリントサーバが一对一の関係となるため、1台のプリントデバイスへの印刷順序を守るためには、1台のプリントサーバが連続して印刷する必要がある。

【0006】

そのため、1台のプリントデバイスへ印刷リクエストが集中した場合、それを処理するプリントサーバは1台に限定され、印刷処理がプリントサーバ1台に集中してしまう。

【0007】

負荷が集中するとクラスタ化したメリットが損なわれ、負荷集中したプリントサーバへ印刷リクエストされたジョブはなかなか印刷されないことになりプリントサーバシステムの印刷パフォーマンスが劣化する恐れがある。

30

【0008】

上述した特許文献1の方法によって印刷順序を制御した場合、特定のプリントデバイスへの印刷処理を行うプリントサーバが固定となる。そのため、1台のプリントデバイスへの印刷リクエストが集中した場合には、1台のプリントサーバの負荷が増大してしまい、印刷処理集中の課題を解決することができない。

【0009】

本発明は、印刷処理が1台のプリントサーバに集中することなく、指示通りの印刷順序で印刷することを目的とする。

40

【課題を解決するための手段】

【0010】

本発明は、複数のプリントサーバと、少なくとも1つのプリントデバイスとを含む印刷システムであって、

プリントサーバは、

ドキュメントデータを識別するドキュメント識別子を含む印刷ジョブの印刷リクエストを受け付ける受付手段と、

当該印刷ジョブの登録をプリントデバイスに要求する手段と、

前記印刷ジョブの登録を要求した後、当該印刷ジョブを破棄する手段と、

前記プリントデバイスからの指示に応答して、前記指示された印刷ジョブを取得し、当

50

該印刷ジョブに基づき印刷データを生成して前記プリントデバイスに転送する手段とを有し、

前記プリントデバイスは、

前記印刷ジョブの登録の要求を受け付けたことに応じて、前記ドキュメント識別子を含む前記印刷ジョブをスケジュールリストに登録する登録手段と、

前記スケジュールリストに登録された登録内容に基づき、前記複数のプリントサーバのうちのいずれかのプリントサーバに対して、印刷データの生成を指示する指示手段とを有することを特徴とする。

【発明の効果】

【0012】

本発明によれば、印刷処理が1台のプリントサーバに集中することなく指示通りの印刷順序で印刷することができる。

【発明を実施するための最良の形態】

【0013】

以下、図面を参照しながら発明を実施するための最良の形態について詳細に説明する。

【0014】

<システム構成>

図1は、本実施形態における印刷システムの構成の一例を示すブロック図である。図1において、複数の印刷システムを制御するためのサーバコンピュータシステムとプリントデバイス及びクライアントコンピュータがWAN(Wide Area Network)110を介して広域に接続されている。

【0015】

DBMS(DataBase Management System)104は、ストレージ105に保存されているドキュメントデータ、印刷ジョブやプリンタ情報といったデータを管理するサーバコンピュータシステムである。他のサーバから各種データの更新、取得、追加、削除といったオペレーションを受け付け実行する。また、大規模な印刷システムの場合は負荷分散するためのクラスタ構成や障害発生に備えて冗長性を持たせたシステム構成を採用することもある。

【0016】

ストレージ105は、DBMS104とDAS(Direct Attached Storage)として直接接続する形態をとっても良いし、SAN(Storage Area Network)としてネットワークを介して接続する形態をとっても良い。

【0017】

プリントサーバシステム101は、印刷システムにおけるプリントデバイスの監視や管理、印刷ジョブの制御や監視、プリントデバイスへの印刷ジョブの転送を司るサーバコンピュータシステムである。このプリントサーバシステム101も大規模な印刷システムの場合は負荷分散するためのクラスタ構成もしくは冗長性を持たせたクラスタ構成となっている。また、LAN(Local Area Network)109Aを介してDBMS104や後述するWebサーバシステム102と接続し、印刷ジョブデータや印刷制御指示、管理している印刷ジョブの情報などを交換する。

【0018】

Webサーバシステム102は、WAN110を介して接続する複数のフロントエンドのクライアント108へDBMS104で管理しているジョブ情報やドキュメント情報を送信する。Webサーバシステム102はサーバコンピュータシステムであり、大規模な印刷システムの場合は負荷分散のためのクラスタ構成や冗長構成となることが通例である。また、クライアント108とからの印刷指示、印刷ジョブの操作を受け付け、プリントサーバシステム101へ指示を転送する。

【0019】

また、Webサーバシステム102は、後述するディレクトリサーバ103と連携してクライアント108及びプリントデバイス107から行われるログインの認証管理を行う

10

20

30

40

50

。Webサーバシステム102は、予めシステム設計者かもしくはドキュメント生成者が設定したユーザの権限を保持している。権限情報としては、例えばドキュメントの参照、印刷、削除権限、あるプリンタへの印刷権限、印刷ジョブの削除権限などが挙げられる。また、このユーザの権限情報はWebサーバシステム102で保存、管理しても良いし、上述したDBMS104で保存、管理しても良い。

【0020】

ディレクトリサーバ103は、ユーザ名及びパスワードの組み合わせによってユーザの認証管理を行うサーバである。また、各々のユーザはドメイングループに参加するような構成も可能であり、例えばドメイングループ内のユーザ同士での権限をユーザ単体の権限とは違う範囲で設定することもできる。また、各々のユーザは複数のドメイングループに参加するように設計することも可能である。

10

【0021】

システム管理サーバ106は、プリントサーバシステム101、Webサーバシステム102、ディレクトリサーバ103などのシステムを管理するサーバである。

【0022】

プリントデバイス107(107A、107B、107C、107D)は印刷デバイスである。プリントデバイス107は不図示のネットワークインタフェースを介してLAN109BもしくはLAN109Cに接続されている。また、LAN109B、109CはWAN110と接続され、プリントデバイス107はLAN及びWANを介して上述したプリントサーバシステム101やWebサーバシステム102と通信が可能に構成されている。

20

【0023】

このプリントデバイス107としては、電子写真方式を採用したレーザビームプリンタやインクジェット方式を採用したインクジェットプリンタを適宜適用することができる。また、プリントデバイス107は後述する印刷スケジュールを管理するスケジュールアプリケーションを備える。

【0024】

クライアント108(108A、108B、108C、108D)は、情報処理を行うためのクライアントコンピュータである。クライアント108は不図示のネットワークインタフェースを介してLAN109BもしくはLAN109Cに接続され、WAN110を介して各種サーバシステムと通信が可能に構成されている。尚、クライアント108は、それぞれ永続記憶装置を接続したコンピュータであっても良いし、一時記憶装置のみによって構成されたThin Client構成をとっても良い。

30

【0025】

<サーバコンピュータ及びクライアントコンピュータのハードウェア構成>

図2は、サーバコンピュータ又はクライアントコンピュータのハードウェア構成の一例を示す図である。CPU203は、HDD206に記憶されたプログラムをRAM205に読み込み、そのプログラムに従って処理を実行し、本装置(サーバコンピュータ、又はクライアントコンピュータ)を制御する。ROM204は、本装置の各処理に関わるプログラムや制御データを記憶する書き換え不可能なメモリである。RAM205は、本装置の各処理に関わる一時的なデータを記憶する書き換え可能なメモリである。

40

【0026】

尚、本装置の各処理に関わるプログラムや制御データ及び一時的なデータなどはHDD206にも記憶される。

【0027】

入力装置207は、本装置への指示入力を受け付けるキーボードやポインティングデバイスである。表示装置208は、本装置の動作状況や、本装置上で動作する各プログラムが出力する情報を表示する。Network I/F202は、ネットワーク経由でLAN及びインターネットに接続し、外部と情報交換を行うためのインタフェースである。外部機器I/F201は外部記憶機器などを接続するためのインタフェースである。

50

【 0 0 2 8 】

< W e bサーバシステム >

W e bサーバシステム 1 0 2 は上述の通り、負荷分散するためのクラスタ構成が通例であり、複数のW e bサーバから構成される。W e bサーバは、W W W (World Wide Web) システムにおいて情報通信を行うサーバであり、ソフトウェアである。W e bサーバシステム 1 0 2 は以下の機能を備える。

【 0 0 2 9 】

フロントエンドのクライアント 1 0 8 からのログインによりユーザアカウント情報及びパスワードを得てディレクトリサーバ 1 0 3 に認証を依頼する。ログイン認証に成功すると、ユーザのドキュメントの参照権限を承認する。承認された参照権限に従ってD B M S 1 0 4 からドキュメントの情報を取得し、ログインしているクライアント 1 0 8 へ情報を送信する。また、クライアント 1 0 8 からドキュメントの印刷を指示されると、印刷指示をプリントサーバシステム 1 0 1 へ送信する。

10

【 0 0 3 0 】

また、上述した参照権限の承認ポリシーとして、例えば所属するドメイン全てのユーザのドキュメントが参照可能であるポリシーや、ユーザ本人のドキュメントのみ参照可能といったポリシーが考えられる。また、それらは印刷システムを導入するユーザのポリシーによって柔軟に設計可能である。また、この承認については、ドキュメントの参照のみでなく、ドキュメントの印刷権限、印刷ジョブの削除権限、プロモート権限といったような権限設定が可能である。

20

【 0 0 3 1 】

< D B M S >

D B M S 1 0 4 はカード型、リレーショナル型、オブジェクト型といった様々なデータ管理方法を選択することができるが、本実施形態においては最も広く普及しているリレーショナル型として説明する。D B M S 1 0 4 はジョブ情報 3 0 0 (図 3)、デバイス情報 4 0 0 (図 4)、ドキュメント情報 5 0 0 (図 5)などをテーブル情報としてストレージ 1 0 5 に保持している。

【 0 0 3 2 】

図 3 は、本実施形態におけるD B M Sで管理されるジョブ情報の一例を示す図である。図 3 に示すように、ジョブ情報 3 0 0 には、ジョブID 3 0 1、ドキュメント名 3 0 2、ユーザ名 3 0 3、受付日時 3 0 4、ドキュメントID 3 0 5、ステータス 3 0 6、デバイス名 3 0 7、デバイス内ジョブID 3 0 8が含まれる。ここで、ユーザ名 3 0 3 は印刷を指示したユーザのアカウント名である。ドキュメントID 3 0 5 はD B M S 1 0 4 で管理されているドキュメントの識別子である。ステータス 3 0 6 は印刷ジョブのステータスである。

30

【 0 0 3 3 】

ステータスとしては、印刷データをプリントデバイス 1 0 7 へ転送している「転送中」、プリントデバイス 1 0 7 で印刷が行われている「印刷中」、印刷順序待ちである「スケジュール待機」といった状態がある。

【 0 0 3 4 】

また、デバイス名 3 0 7 は印刷を行うプリントデバイス 1 0 7 の名称である。デバイス内ジョブID 3 0 8 はプリントデバイス 1 0 7 内のスケジュールアプリケーション(詳細は図 8 を用いて後述する)により発行されるジョブの識別子である。

40

【 0 0 3 5 】

図 4 は、本実施形態におけるD B M Sで管理されるデバイス情報の一例を示す図である。デバイス情報 4 0 0 として、印刷対象となるプリントデバイス 1 0 7 の各種情報が保持されている。図 4 に示すように、デバイス情報 4 0 0 には、デバイス名 4 0 1、デバイスの通信先のアドレス 4 0 2、通信に用いられるポート番号 4 0 3、印刷データの転送方法 4 0 4、デバイスのステータス 4 0 5、プリントサーバアドレス 4 0 6 が含まれる。

【 0 0 3 6 】

50

ここで、印刷データの転送方法 4 0 4 には、データをストリームとして T C P / I P により連続送信する R A W、L P R (Line P Rinter deamon protocol)、或いは H T T P により送信する方法などがある。

【 0 0 3 7 】

また、デバイス情報 4 0 0 は、プリントサーバシステム 1 0 1 のデバイスマネージャが管理しても良く、データを保持する場所は問わない。

【 0 0 3 8 】

また、プリントサーバアドレス 4 0 6 には、プリントサーバシステム 1 0 1 内の複数のプリントサーバの 1 つのアドレスが登録される。ここで、プリントサーバアドレス 4 0 6 にアドレスが登録されている場合には、そのプリントサーバが印刷リクエストの処理中であることを示す。プリントデバイス 1 0 7 での印刷順序を守るため、スケジュール登録はプリントサーバアドレス 4 0 6 に登録されているプリントサーバで占有される。

【 0 0 3 9 】

図 5 は、本実施形態における D B M S で管理されるドキュメント情報の一例を示す図である。図 5 に示すように、ドキュメント情報 5 0 0 には、ドキュメントの識別子であるドキュメント I D 5 0 1、ドキュメント名 5 0 2、ドキュメントの登録者のユーザ名 5 0 3、実際にドキュメントが保存されている場所を示すデータパス 5 0 4 が含まれる。また、本実施形態では、ドキュメント情報 5 0 0 としてリファレンスが保持され、データは別の箇所で保存しているが、直接保持しても良い。

【 0 0 4 0 】

D B M S 1 0 4 は、例えば以下のような処理に利用される。

(1) W e b サーバシステム 1 0 2 から指定したユーザ名と一致する全てのドキュメント情報やジョブ情報の取得。

(2) W e b サーバシステム 1 0 2 からデバイス名の指定によって一致するデバイス情報の取得。

(3) プリントサーバ 6 0 1 からドキュメント I D の指定によって一致するドキュメントの情報の取得。

(4) プリントサーバ 6 0 1 からの印刷実行に必要なデバイス情報の取得。

(5) プリントサーバ 6 0 1 からの新規ドキュメントの登録、ジョブの登録。

(6) プリントサーバ 6 0 1 からジョブやデバイスの状態が変化した際のジョブステータスやデバイスステータスの更新。

【 0 0 4 1 】

< プリントサーバシステム >

図 6 は、プリントサーバシステム 1 0 1 の構成の一例を示すブロック図である。図 6 において、プリントサーバシステム 1 0 1 は大規模な印刷システムに対応するために、負荷分散構成が採用されている。そして、複数台のプリントサーバ 6 0 1 が L A N 1 0 9 A や W A N 1 1 0 と負荷分散を行うためのスイッチ機構であるロードバランサ 6 0 2 を介して接続されている。ロードバランサ 6 0 2 は、複数台のプリントサーバ 6 0 1 の負荷状況を把握し、負荷のかかっていないサーバに対してリクエストを優先的に発行するような分散を行う。

【 0 0 4 2 】

負荷状況の把握方法としては、単純にリクエストを順次各サーバに回していくラウンドロビンと呼ばれる方法や、規定のリクエストを定期的に複数台のプリントサーバ 6 0 1 に送出し、返信までかかった時間を負荷として判断する方法などが考えられる。

【 0 0 4 3 】

ここで、プリントサーバ 6 0 1 の概略構成を説明する。プリントサーバ 6 0 1 は A P I 6 0 3、D B ドライバ 6 0 4、ジョブマネージャ 6 0 5、デバイスマネージャ 6 0 6 から構成されている。A P I は Application Program Interface の略である。A P I 6 0 3 では、W e b サーバシステム 1 0 2 や他の印刷可能なドキュメントを生成するサーバからのドキュメント登録、ドキュメントの印刷リクエストや印刷ジョブの制御リクエストを受け

付ける。また、この他の印刷可能なドキュメントを生成するサーバとしては、例えば定型帳票を作成するような帳票サーバや、ユーザのドキュメントを一括管理するドキュメント管理サーバなどがある。

【 0 0 4 4 】

A P I 6 0 3 が印刷リクエストを受け付けると、リクエスト情報より D B M S 1 0 4 で保持するデバイス情報 4 0 0 を特定し、このプリントサーバのアドレスがプリントサーバアドレス 4 0 6 に登録される。また、印刷リクエスト C B (Call Back) する際に、プリントサーバアドレス 4 0 6 をクリアする。ここで、プリントサーバアドレス 4 0 6 を登録する際に、既に他のプリントサーバアドレスが登録されている場合には印刷リクエストをそのプリントサーバアドレスへ転送する。

10

【 0 0 4 5 】

また、D B ドライバ 6 0 4 はドキュメント登録リクエストを受けて、D B M S 1 0 4 へドキュメントを登録する場合、印刷リクエストを受けて印刷データを D B M S 1 0 4 から取得する場合など、D B M S 1 0 4 との通信を行うモジュールである。

【 0 0 4 6 】

次に、ジョブマネージャ 6 0 5 は次のような機能を有する。A P I 6 0 3 で受け付けたドキュメントを、D B ドライバ 6 0 4 を介して D B M S 1 0 4 へ登録する。A P I 6 0 3 が印刷リクエストを受け付けると、D B ドライバ 6 0 4 を介して D B M S 1 0 4 の印刷するドキュメント情報を参照し印刷可否を判断する。印刷可能な場合にはジョブマネージャ 6 0 5 がスケジュールアプリケーションにスケジュール登録するようデバイスマネージャ 6 0 6 へ指示する。

20

【 0 0 4 7 】

デバイスマネージャ 6 0 6 でスケジュール登録が行われると、ジョブマネージャ 6 0 5 がデバイスマネージャ 6 0 6 から通知を受ける。その後、印刷リクエスト情報とスケジュール登録 C B で受け取る情報をジョブ情報 3 0 0 として D B M S 1 0 4 に登録する。D B M S 1 0 4 に登録された後、ジョブ情報はプリントサーバ 6 0 1 内で保持する必要はなく破棄される。

【 0 0 4 8 】

次に、デバイスマネージャ 6 0 6 の処理について説明する。デバイスマネージャ 6 0 6 は D B M S 1 0 4 が保持しているデバイス情報 4 0 0 を取得し、一時保存 (キャッシュ) する。デバイスマネージャ 6 0 6 でキャッシュされるデバイス情報 4 0 0 は印刷実行時に使用され、印刷開始時に毎回取得しても良いし、定期的に取得しても良く、デバイス情報 4 0 0 のキャッシュを更新するタイミングは問わない。

30

【 0 0 4 9 】

また、デバイスマネージャ 6 0 6 はジョブマネージャ 6 0 5 からジョブのスケジュール登録リクエストを受け付け、プリントデバイス 1 0 7 内のスケジュールアプリケーションとの通信を開始してスケジュール登録を行う。スケジュール登録が完了し、スケジュールアプリケーションからスケジュール登録 C B を受け取ると、ジョブマネージャ 6 0 5 にスケジュール登録 C B 情報を通知する。尚、スケジュール登録シーケンスの詳細については後述する。

40

【 0 0 5 0 】

デバイスマネージャ 6 0 6 がジョブマネージャ 6 0 5 からの印刷指示を受け付けると、印刷指示で受け取るスケジュール情報に基づいてドキュメントデータを取得する。次に、デバイスマネージャ 6 0 6 は保持しているデバイス情報 4 0 0 のキャッシュを用いドキュメントデータをプリントデバイス 1 0 7 で印刷可能な形式に変換し、プリントデバイス 1 0 7 へ印刷データ転送を開始する。プリントデバイス 1 0 7 が印刷可能な形式には P D L (Page Discription Language) や P D F (Portable Document Format) などがある。

【 0 0 5 1 】

ここでプリントデバイス 1 0 7 が P D F を直接受け付けて印刷できる P D F ダイレクトプリント機能を有する場合など形式変換が必要ない場合もある。尚、デバイスマネージャ

50

606が印刷指示を受け付けてから印刷データをプリントデバイス107へ転送し、印刷が完了するまでの印刷シーケンスの詳細については後述する。

【0052】

また、デバイスマネージャ606はスケジュールアプリケーションから印刷結果を受け付ける機能を有する。そして、スケジュールアプリケーションからの印刷結果を受け付け、DBドライバ604を介してDBMS104に印刷結果の登録を行う。

【0053】

<プリントデバイスのハードウェア構成>

図7は、プリントデバイス107のハードウェア構成の一例を示す図である。イメージリーダ702は原稿給送部701で原稿を読み込む。画像形成部703は読み込んだ原稿やネットワーク経由で受信したデータを印刷画像に変換し印刷出力する。排紙部704は印刷出力した紙を排出し、ソートやステイブルなどの処理を施す。Network I/F 705はネットワーク経由でLAN及びインターネットに接続し、外部と情報交換を行うためのインタフェースである。

【0054】

CPU706はROM707又はHDD709に記憶されたプログラムをRAM708に読み込み、そのプログラムに従って処理を実行し、本装置（プリントデバイス）を制御する。不揮発性のROM707は本装置の各処理に関わるプログラムやデータを記憶するメモリである。書き換え可能なRAM708は本装置の各処理に関わる一時的なデータを記憶するメモリである。HDD709は本装置の各処理に関わるプログラムやデータ及び一時的なデータ等を記憶する。操作部710は後述する画面を表示したり、画面を介したユーザの操作指示を受け付けたりする。

【0055】

<プリントデバイス>

図8は、プリントデバイス107の構成を階層的に示す図である。図8において、画像形成部813は、紙のハンドリングや画像転写・定着等の一連の画像形成プロセスを実行して、記録紙等の記録媒体上に画像を形成する。この画像形成部813は、例えばインクジェット方式や電子写真方式の画像形成ユニットを備えている。

【0056】

画像読み取り部817は、スキャナ等を備え、原稿画像を光学的に読み取ってデジタル画像情報に変換する。また、画像読み取り部817は、そのデジタル画像情報を画像形成部813に出力して画像を形成したり、ファックス部812やネットワークインタフェース部814等に渡して回線を介して伝送したりする。

【0057】

デバイスコントローラ810は、画像形成部813と、画像読み取り部817とのそれぞれの動作を制御し、例えば画像読み取り部817で読み取った原稿情報を画像形成部813で複写するよう制御する。また、デバイスコントローラ810は、ネットワークインタフェース部814、プリント処理部815、ファックス部812、操作部制御部811を有し、これら各部の間での情報のやり取りも制御する。

【0058】

ファックス部812は、ファクシミリ画像の送受信、即ち画像読み取り部817で読み取ったデジタル画像情報を送信したり、また逆に、受信したファクシミリ信号を復号し、画像形成部813で記録したりする等の処理を実行する。

【0059】

操作部制御部811は、不図示の操作部の操作パネルを使用したユーザの操作に応じた信号を発生したり、また操作部（又は表示部）等に各種データやメッセージ等を表示するように制御する。プリント処理部815は、例えばネットワークインタフェース部814を介して入力された印刷データを処理して画像形成部813に出力し、印刷する等の制御を行う。ネットワークインタフェース部814は、通信回線を介して他の通信端末との間のデータの送受信を制御する。

10

20

30

40

50

【 0 0 6 0 】

仮想マシン（Virtual Machine）8 0 5 は、デバイスコントローラ 8 1 0 の上位に位置し、この仮想マシン 8 0 5 からデバイスコントローラ 8 1 0 を制御できるように構成されている。

【 0 0 6 1 】

また、ネットワークインタフェース部 8 1 4 は、デバイスコントローラ 8 1 0 と、仮想マシン 8 0 5 との双方から直接利用でき、それぞれ独立して外部ネットワークにアクセスすることができるよう構成されている。

【 0 0 6 2 】

更に、仮想マシン 8 0 5 の上位には、仮想マシン 8 0 5 が提供する A P I に対応したプログラミング言語で記述されたアプリケーション 8 0 1 ~ 8 0 3 が存在している。これらのアプリケーション 8 0 1 ~ 8 0 3 は、仮想マシン 8 0 5 を介して間接的にデバイスコントローラ 8 1 0 に働きかけることができ、また画像形成部 8 1 3 や画像読み取り部 8 1 7 を動作させることができる。

10

【 0 0 6 3 】

本実施形態では、アプリケーション 8 0 1 として、特にスケジュールアプリケーションを備える。また、アプリケーション 8 0 1 ~ 8 0 3 は、仮想マシン 8 0 5 上からアンインストールしたり、更に新たにアプリケーション 8 0 4 をインストールしたりするといったことが可能なように構成されている。

【 0 0 6 4 】

20

また、これらのアプリケーション 8 0 1 ~ 8 0 3 は、本実施形態ではプリントデバイス 1 0 7 にインストールされるものとして説明したが、ハードウェアとして備えられていても良い。更に、プリントデバイス 1 0 7 と通信可能に接続された外部装置（コンピュータ）上のアプリケーションとして備えられていても良い。

【 0 0 6 5 】

外部記憶装置制御部 8 1 6 は、画像読み取り部 8 1 7 で読み取られた画像を画像形成部 8 1 3 で外部記憶装置に保存可能なデータフォーマットに変換し、外部記憶装置に保存する。また、外部記憶装置制御部 8 1 6 は、外部記憶装置に保存されたデータを読み出し、画像形成部 8 1 3 を介して印刷処理を行ったり、ネットワークインタフェース部 8 1 4 を介して外部にネットワーク送信したりする。

30

【 0 0 6 6 】

また、本実施形態では、プリントデバイス 1 0 7 へのログインの方法として I C カードリーダー 8 1 8 が通信可能に接続されている。ログインの方法としては、他の方法でも勿論問題なく、例えば操作部制御部 8 1 1 が管理するデバイスパネルからの入力といった方法でも良い。

【 0 0 6 7 】

< スケジュールアプリケーション >

図 9 は、スケジュールアプリケーションの構成を示すブロック図である。スケジュールアプリケーション 8 0 1 は、通信マネージャ 9 0 1、ジョブマネージャ 9 0 2、デバイスマネージャ 9 0 3 から構成される。

40

【 0 0 6 8 】

ここで、通信マネージャ 9 0 1 はプリントサーバ 6 0 1 からの接続やリクエストを受け付け、デバイスコントローラ 8 1 0 から通知される印刷ジョブのステータス変化の通知を行う。ジョブマネージャ 9 0 2 はプリントサーバ 6 0 1 から受け付けたスケジュール情報をスケジュールリストとして保持し、印刷ジョブを管理するモジュールで、スケジュールリストに登録された順序通りに印刷が行われるように制御する。また、プリントサーバ 6 0 1 からの印刷ジョブの削除やプロモート（優先度変更）といった操作を受け付け、印刷ジョブをコントロールする。尚、印刷ジョブの削除やプロモートの詳細な説明は本発明と関係しないため省略する。

【 0 0 6 9 】

50

デバイスマネージャ 903 はデバイスコントローラ 810 との通信を行うモジュールで、印刷ジョブの削除を行ったり、印刷ジョブの状態の変化をキャッチしたりする。

【0070】

図 10 は、ジョブマネージャ 902 が管理する印刷ジョブのスケジュール情報を示す図である。スケジュール情報 1000 には、ドキュメント名 1001、ユーザ名 1002、受付日時 1003、ドキュメント ID 1004、ステータス 1005、デバイス名 1006、デバイス内ジョブ ID 1007、リクエスト先アドレス 1008 が含まれる。

【0071】

スケジュール情報 1000 は、基本的には上述した DBMS 104 で管理されるジョブ情報 300 と共通する情報であり、詳細な説明は省略する。但し、ステータス 1005 にプリントサーバ 601 への印刷指示が行われた状態を示す「リクエスト済」が増える。

10

【0072】

図 11 は、デバイスマネージャ 903 が管理する連続印刷制限数に関する情報を示す図である。連続印刷制限情報 1100 には、連続印刷制限数 1101 が含まれる。ここで、スケジュールアプリケーション 801 からプリントサーバ 601 へ印刷指示が複数ジョブまとめて行われ、プリントサーバ 601 は複数の印刷データを同一セッションで連続して転送する。連続印刷制限数 1101 はその最大ジョブ数である。

【0073】

つまり、連続印刷制限数 1101 は、プリントサーバ 601 が一度に印刷処理する最大ジョブ数となるため、連続印刷制限数 1101 でプリントサーバ 601 毎の負荷分散状況を調整することができる。このように、プリントサーバシステム 101 内で負荷分散されるプリントサーバ 601 の台数や性能によって連続印刷制限数 1101 を調整することにより印刷処理のパフォーマンスの最適化を行うことができる。

20

【0074】

ここで、上述した構成を備える本実施形態における印刷システムの印刷フローについて詳細に説明する。

【0075】

< プリントサーバ 601 の印刷リクエスト受付フロー >

図 12 は、プリントサーバ 601 が印刷リクエストを受け付けた際の処理を示すフローチャートである。まず、プリントサーバ 601 における API 603 が印刷リクエストを受け付ける (F1201)。この印刷リクエストによりドキュメント ID 及び印刷対象のプリンタ名称もしくはプリンタのアドレスが通知される。次に、ジョブマネージャ 605 が通知されたドキュメント ID に基づき DB ドライバ 604 を介して DBMS 104 からドキュメント情報 500 を取得し、ジョブ情報を生成する (F1202)。ここで生成されるジョブ情報は DBMS 104 が保持するジョブ情報 300 と同じ情報であるが、スケジュールアプリケーション 801 から発行されるデバイス内ジョブ ID 308 はまだ確定していない。

30

【0076】

次に、上述の F1202 で取得したドキュメント情報よりドキュメントデータの存在を確認し、印刷可能か否かを判定する (F1203)。ここで、ドキュメントデータが存在しない場合には、印刷不可能と判定し、この印刷リクエストに対する印刷リクエスト C B としてエラーを返す (F1208)。

40

【0077】

一方、ドキュメントデータが存在する場合には、印刷可能と判定し、そのドキュメントデータの印刷を行うプリントデバイス 107 を特定し、スケジュールアプリケーション 801 に対してスケジュール要求を行う (F1204)。このスケジュール要求で通知するスケジュール情報リストは F1202 で生成したジョブ情報とリクエスト先アドレスから生成される。

【0078】

本実施形態では、リクエスト先アドレスはロードバランサ 602 のアドレスを指定する

50

。また、リクエスト先アドレスの指定はロードバランサ 6 0 2 のアドレスに固定し、負荷分散をロードバランサ 6 0 2 に任せる。ここで、リクエスト先アドレスに複数のプリントサーバ 6 0 1 のアドレスを振り分けて登録し、ロードバランサ 6 0 2 の機能を使用せずに負荷分散を行うこともできる。

【 0 0 7 9 】

上述した F 1 2 0 4 のスケジュール要求により、スケジュールアプリケーション 8 0 1 がスケジュール登録の処理を行い、スケジュール登録 C B が返される。このスケジュール登録 C B には、スケジュールアプリケーション 8 0 1 で発行されたデバイス内ジョブ I D が含まれる。

【 0 0 8 0 】

次に、スケジュール登録 C B により受け取ったデバイス内ジョブ I D とジョブ情報とを D B M S 1 0 4 のジョブ情報 3 0 0 に保存する (F 1 2 0 5)。ここで、D B M S 1 0 4 に保存されたジョブ情報 3 0 0 のステータス 3 0 6 はプリントサーバ 6 0 1 により随時更新される。そして、W e b サーバシステム 1 0 2 は D B M S 1 0 4 が保持しているジョブ情報 3 0 0 を参照してユーザにジョブ実行状況を伝えることができる。

【 0 0 8 1 】

ジョブ情報 3 0 0 の保存が完了すると、プリントサーバ 6 0 1 はスケジュール登録に使用するために、F 1 2 0 2 で生成したジョブ情報を破棄する (F 1 2 0 6)。印刷実行に必要なジョブ情報は、スケジュールアプリケーション 8 0 1 からの印刷指示を受け付けた際の情報として受け取ることができるため、プリントサーバ 6 0 1 はジョブ情報を保持し続ける必要がない。

【 0 0 8 2 】

そして、F 1 2 0 7 において、プリントサーバ 2 0 1 は印刷リクエスト受付完了として印刷リクエスト C B を返す。以上をもってプリントサーバ 2 0 1 の印刷リクエスト受付が完了する。

【 0 0 8 3 】

< スケジュールアプリケーション 8 0 1 のスケジュール登録動作フロー >

図 1 3 は、スケジュールアプリケーション 8 0 1 がプリントサーバ 6 0 1 からスケジュール要求を受け付けた際の処理を示すフローチャートである。スケジュールアプリケーション 8 0 1 の通信マネージャ 9 0 1 がスケジュール要求を受け付け、ジョブマネージャ 9 0 2 にスケジュール要求を受け付けた際に渡されるスケジュール情報リストを通知する (F 1 3 0 1)。そして、スケジュール情報リストを受け取ったジョブマネージャ 9 0 2 がスケジュール情報リストに登録されている全てのジョブに対してデバイス内ジョブ I D を発行する (F 1 3 0 2)。

【 0 0 8 4 】

次に、ジョブマネージャ 9 0 2 が管理するスケジュールリストを確認し (F 1 3 0 3)、ステータス 1 0 0 5 がスケジュール待機となっているジョブがあるか否かを確認する (F 1 3 0 4)。ここで、既にスケジュールリストにスケジュール待機のジョブがある場合は、スケジュールリストの最後にスケジュール要求で受け取ったスケジュール情報リストを追加し (F 1 3 0 7)、この処理を終了する。

【 0 0 8 5 】

一方、スケジュールリストにスケジュール待機のジョブがない場合は、スケジュール要求で受け取ったスケジュール情報リストをスケジュールリストに登録し、ステータス 1 0 0 5 をスケジュール待機とする (F 1 3 0 5)。そして、スケジュール情報リストの先頭のスケジュール情報でジョブ実行を開始する (F 1 3 0 6)。ジョブ実行開始についての詳細は後述する。

【 0 0 8 6 】

このように、スケジュールアプリケーション 8 0 1 がプリントサーバ 6 0 1 からのスケジュール要求を受け付け、スケジュール情報リストの登録を行う。

【 0 0 8 7 】

< スケジュールアプリケーション 801 のジョブ実行開始フロー >

図 14 は、スケジュールアプリケーション 801 がプリントサーバ 601 に対して印刷指示を行う際の処理を示すフローチャートである。まず、ジョブマネージャ 902 で管理されるスケジュールリストに登録済みであるスケジュール情報 1000 でジョブ実行開始が通知される (F1401)。次に、通知されたスケジュール情報のステータス 1005 をリクエスト済に更新する (F1402)。そして、プリントサーバ 601 への印刷指示の際に通知するジョブ実行リストをヌル (NULL) に初期化し (F1403)、ジョブ実行開始されたスケジュール情報をジョブ実行リストに追加する (F1404)。

【0088】

次に、ジョブマネージャ 902 で管理されるスケジュールリストにおいて、ジョブ実行開始されたスケジュール情報の次のスケジュール情報についてスケジュールリストの最後まで処理を繰り返す (F1405 ~ F1411)。

【0089】

まず、リクエスト先アドレス 1008 がジョブ実行開始されたスケジュール情報リストのリクエスト先アドレスと同じか否かを判定する (F1406)。ここで、リクエスト先アドレスが異なる場合には、連続印刷することができないため、F1405 ~ F1411 の処理を終了する。

【0090】

一方、リクエスト先アドレスが同じ場合には、プリントサーバ 601 からの連続印刷の対象となり、プリントサーバ 601 の連続印刷数を制限する。即ち、ジョブ実行リストの登録数がスケジュールアプリケーション 801 で保持する連続印刷制限数 1101 以上か否かを判定する。判定の結果、ジョブ実行リストの登録数が連続印刷制限数 1101 以上の場合には、これ以上ジョブ実行リストに登録させず、F1405 ~ F1411 の処理を終了する。

【0091】

一方、連続印刷制限数 1101 以上でない場合には、連続印刷可能としてジョブ実行リストにスケジュール情報を追加する (F1409)。そして、追加したスケジュール情報のステータス 1005 をリクエスト済に更新する (F1410)。

【0092】

その後、スケジュールリストに登録されている全てのスケジュール情報リストの処理が終了すると (F1411)、ジョブ実行リストでリクエスト先アドレス 1008 に対して印刷指示を行う (F1412)。尚、印刷指示を行う際のスケジュールアプリケーション 801 とプリントサーバ 601 と間のシーケンスについては後述する。

【0093】

このように、スケジュールアプリケーション 801 はプリントサーバ 601 に対して印刷指示を行う。

【0094】

< プリントサーバ 601 の印刷フロー >

図 15 は、スケジュールアプリケーション 801 からの印刷指示を受け付けたプリントサーバ 601 の処理を示すフローチャートである。スケジュールアプリケーション 801 からの印刷指示にはジョブ実行リストが含まれており、複数のスケジュール情報 1000 が登録されている。

【0095】

プリントサーバ 601 が印刷指示を受け付けると、ジョブ実行リストに登録されている全てのスケジュール情報リストを実行する (F1502 ~ F1505)。

【0096】

まず、スケジュール情報リストに基づきドキュメントデータを取得する (F1503)。次に、取得したドキュメントデータを印刷実行するプリントデバイス 107 で印刷可能な印刷データを生成する (F1504)。この生成処理では、プリンタドライバを用いて PDL を生成しても良いし、PDF を変換せずに、そのまま使用しても良い。

10

20

30

40

50

【 0 0 9 7 】

そして、ジョブ実行リストに登録されているスケジュール情報 1 0 0 0 についての印刷データの生成処理が完了すると (F 1 5 0 2 ~ F 1 5 0 5)、プリントデバイス 1 0 7 に印刷データの転送を開始する (F 1 5 0 6)。

【 0 0 9 8 】

プリントサーバ 6 0 1 が印刷データをプリントデバイス 1 0 7 に転送し、印刷が行われて印刷結果が通知されるまでの、プリントサーバ 6 0 1、プリントデバイス 1 0 7、スケジュールアプリケーション 8 0 1 の間の詳細なシーケンスは後述する。

【 0 0 9 9 】

このように、プリントサーバ 6 0 1 はスケジュールアプリケーション 8 0 1 からの印刷指示を受け付け、プリントデバイス 1 0 7 へ印刷データ転送を開始し印刷を行う。

10

【 0 1 0 0 】

< スケジュール登録シーケンス >

図 1 6 は、プリントサーバ 6 0 1 とスケジュールアプリケーション 8 0 1 との間のスケジュール登録時のシーケンスを示す図である。プリントサーバ 6 0 1 は印刷リクエストを受け付けた際に、このシーケンスでスケジュールアプリケーション 8 0 1 にスケジュール登録を行う。

【 0 1 0 1 】

まず、S 1 6 0 1 で、プリントサーバ 6 0 1 及びスケジュールアプリケーション 8 0 1 間で通信セッションを開始する。通信セッションは T C P / I P や H T T P といった通信プロトコルによって確立される。次に、S 1 6 0 2 で、プリントサーバ 6 0 1 がスケジュールアプリケーション 8 0 1 に対して接続要求を発行する。そして、スケジュールアプリケーション 8 0 1 における通信マネージャ 9 0 1 がこの接続要求を受け付ける。

20

【 0 1 0 2 】

次に、S 1 6 0 3 で、スケジュールアプリケーション 8 0 1 のジョブマネージャ 9 0 2 で管理されているスケジュール情報 1 0 0 0 のリスト取得を行う。取得したスケジュール情報 8 0 0 のリストは、例えばプリントサーバ 6 0 1 に接続する他のアプリケーションサーバへ通知するか、プリントサーバ 6 0 1 を管理するためのコンソールへ出力するといった利用方法がある。

【 0 1 0 3 】

また、W e b サーバシステム 1 0 2 を介して印刷ジョブの情報をクライアント 1 0 8 へ提供する機能を実現することも可能である。これら機能は、本発明とは関係しないため、詳細な説明は省略する。

30

【 0 1 0 4 】

次に、S 1 6 0 4 で、プリントサーバ 6 0 1 がスケジュールアプリケーション 8 0 1 に対して印刷ジョブのスケジュール要求を発行する。このスケジュール要求にはジョブ情報 3 0 0 が含まれる。スケジュール要求を受けたスケジュールアプリケーション 8 0 1 は、S 1 6 0 5 でスケジュール登録を行う。

【 0 1 0 5 】

スケジュールアプリケーション 8 0 1 がスケジュール登録を完了すると、S 1 6 0 6 でプリントサーバ 6 0 1 にスケジュール要求 C B を返す。このスケジュール要求 C B では、スケジュール登録時に発行されるデバイス内ジョブ I D を通知する。

40

【 0 1 0 6 】

次に、S 1 6 0 7 で、プリントサーバ 6 0 1 はスケジュールアプリケーション 8 0 1 に切断要求を発行し、S 1 6 0 8 で、S 1 6 0 1 で確立させたセッションを終了する。

【 0 1 0 7 】

このように、プリントサーバ 2 0 1 はスケジュールアプリケーション 8 0 1 にスケジュール登録を行うことができる。

【 0 1 0 8 】

< 印刷シーケンス >

50

図17は、プリントサーバ601、ロードバランサ602、スケジュールアプリケーション801、デバイスコントローラ810間の印刷シーケンスを示す図である。この印刷シーケンスは、スケジュールアプリケーション801がプリントサーバ601に印刷指示を行い、印刷指示を受け付けたプリントサーバ601が印刷処理を実行し、終了するまでのシーケンスである。

【0109】

ここでは、リクエスト先アドレス1008をロードバランサ602に固定しているため、スケジュールアプリケーション801からの印刷指示がロードバランサ602によって負荷分散される。また、不図示であるが、スケジュールアプリケーション801が起動時に、デバイスコントローラ810に対してイベント登録を行っている。従って、デバイス

10

【0110】

まず、S1701で、ロードバランサ602及びスケジュールアプリケーション801間で通信のセッションを開始する。通信セッションはTCP/IPやHTTPといった通信プロトコルによって確立される。次に、S1702で、ロードバランサ602がプリントサーバ601とセッションを開始する。ここで、セッションを開始するプリントサーバ601はロードバランサ602によって負荷分散される。負荷分散方法としては、単純にリクエストを順次各プリントサーバに回していくラウンドロビンや、規定のリクエストを定期的に各プリントサーバ601に送出し、返信までかかった時間を負荷として判断する

20

【0111】

次に、ロードバランサ602を介して開始されたスケジュールアプリケーション801とプリントサーバ601のセッションは、S1718、S1719で、セッションが終了するまで維持される。ここで、S1701、S1702からS1718、S1719までのスケジュールアプリケーション801とプリントサーバ601の通信はロードバランサ602を介して行われる。

【0112】

上述のセッションが開始されると、S1703でスケジュールアプリケーション801はプリントサーバ601に対して接続要求を行う。プリントサーバ601はデバイスマネージャ606にて接続要求を受け付ける。

30

【0113】

次に、S1704で、プリントサーバ601は印刷ジョブ及びプリントデバイスのステータスが変化した場合に通知を受けるべくスケジュールアプリケーション801に対してイベント登録を行う。そして、S1705で、現在スケジュールアプリケーション801のジョブマネージャ902にて管理されているスケジュール情報1000のリスト取得を行う。

【0114】

ここで取得したスケジュール情報1000のリストは、例えばプリントサーバ601に接続する他のアプリケーションサーバへ通知する他、プリントサーバ601を管理するためのコンソールへ出力するといった利用方法が考えられる。また、Webサーバシステム102を介して印刷ジョブの情報をクライアント108へ提供する機能を実現することも可能である。これら機能は、本発明とは関係しないため、詳細は省略する。

40

【0115】

次に、S1706で、スケジュールアプリケーション801はプリントサーバ601に対して印刷指示をする。印刷指示を受けたプリントサーバ601は、S1707で、スケジュールアプリケーション801に印刷データの転送開始を通知し、S1708で、デバイスコントローラ810に対して印刷ジョブの転送を開始する。この印刷ジョブの転送はデバイス情報400に設定された転送方法404にて行われる。また、転送開始を受け付けたスケジュールアプリケーション801はそのスケジュール情報1000のステータス

50

８０６を転送中に変更する。

【０１１６】

一方、Ｓ１７０９で、デバイスコントローラ８１０が受信した印刷データを順次解析し、印刷を開始する。そして、Ｓ１７１０で、イベント登録を行っているスケジュールアプリケーション８０１に対して印刷開始にステータスが変更されたことを通知する。また、印刷開始通知を受けたスケジュールアプリケーション８０１は自身で管理しているスケジュール情報１０００のステータス１００５を印刷中に変更する。

【０１１７】

更に、Ｓ１７１１で、スケジュールアプリケーション８０１は、イベント登録しているプリントサーバ６０１に対して印刷開始通知を行う。こうして、プリントサーバ６０１は自身で印刷した印刷ジョブの状態を把握することが可能であり、ＤＢＭＳ１０４で保持するジョブ情報３００のステータス３０６を更新する。

10

【０１１８】

この印刷データの転送が完了すると、Ｓ１７１２で、プリントサーバ６０１は転送終了をスケジュールアプリケーション８０１に通知する。通知を受けたスケジュールアプリケーション８０１は転送開始時と同様に当該印刷ジョブのスケジュール情報１０００のステータス１００５を転送済みに変更する。

【０１１９】

一方、プリントデバイス１０７において、Ｓ１７１３で印刷が終了すると、Ｓ１７１４で、デバイスコントローラ８１０が印刷開始時と同様に、スケジュールアプリケーション８０１に対して印刷結果を通知する。そして、印刷結果の通知を受けたスケジュールアプリケーション８０１は、Ｓ１７１５でプリントサーバ６０１に印刷結果を通知する。

20

【０１２０】

ここで、印刷結果を受けたプリントサーバ６０１は、予定していた全ての印刷ジョブの印刷が完了した場合は、Ｓ１７１６でスケジュールアプリケーション８０１に対してイベントの解除を行い、続いてＳ１７１７で切断要求を行う。

【０１２１】

このイベントの解除及び切断要求を受けたスケジュールアプリケーション８０１はプリントサーバ６０１との通信の終了処理を行う。最後に、Ｓ１７１８、Ｓ１７１９で、Ｓ１７０１、Ｓ１７０２で確立させたセッションを終了する。

30

【０１２２】

このように、スケジュールアプリケーション８０１はロードバランサ６０２を介してプリントサーバ２０１へ印刷指示を行い、プリントサーバ２０１はスケジュールアプリケーション８０１を介してプリントデバイス１０７で印刷を行うことが可能となる。

【０１２３】

<ジョブ転送開始通知時のフロー>

図１８は、プリントサーバ６０１より印刷データの転送開始通知がなされた場合のスケジュールアプリケーション８０１の処理を示すフローチャートである。尚、スケジュールアプリケーション８０１は印刷データの転送開始が通知されたタイミングで転送開始されたジョブの次ジョブの印刷指示を行う。

40

【０１２４】

まず、図１７に示すＳ１７０７で、プリントサーバ６０１から転送開始通知を受け付ける（Ｆ１８０１）。この転送開始通知はＳ１７０３で接続要求により確立した接続で通知されるため、スケジュールアプリケーション８０１は転送開始された印刷データのデバイス内ジョブＩＤを特定できる。このデバイス内ジョブＩＤよりジョブマネージャ９０２で管理されるスケジュールリストのスケジュール情報１０００を特定し、ステータス１００５を印刷中に更新する（Ｆ１８０２）。

【０１２５】

次に、スケジュールリストにおいて転送開始通知されたスケジュール情報の次のスケジュール情報を参照する（Ｆ１８０３）。そして、次のスケジュール情報のステータス１０

50

05 がリクエスト済であるか否かを判定し (F 1804)、リクエスト済でなかった場合にジョブ実行開始し (F 1805)、リクエスト済だった場合には処理を終了する。尚、ジョブ実行開始 (F 1805) の詳細は上述したジョブ実行開始処理である。

【0126】

<印刷結果通知フロー>

図19は、スケジュールアプリケーション801がデバイスコントローラ810から印刷結果を受け取った場合の処理を示すフローチャートである。まず、スケジュールアプリケーション801にデバイスコントローラ810より印刷結果通知がなされた時に、印刷を開始したプリントサーバ601とのセッションが維持されているか確認する (F 1901)。ここで、印刷開始したプリントサーバ601とのセッションが維持されている場合は、接続しているプリントサーバ601に印刷結果通知を行う (F 1904)。

10

【0127】

一方、印刷開始したプリントサーバ601とのセッションが切れてしまっている場合は、リクエスト先アドレス1008との接続を確立し印刷結果を通知する (F 1903)。その後、上述の印刷結果の通知が終了すると、スケジュールリストよりスケジュール情報を削除する (F 1905)。

【0128】

本実施形態では、リクエスト先アドレス1008にロードバランサ602のアドレスを指定しているため、印刷結果通知がロードバランサ602により負荷分散され、プリントサーバ601に通知される。

20

【0129】

図20は、プリントサーバ601がスケジュールアプリケーション801から印刷結果通知を受け付けた場合の処理を示すフローチャートである。まず、プリントサーバ601がスケジュールアプリケーション801から印刷結果通知を受け付ける (F 2001)。そして、通知されるスケジュール情報に基づいて、DBMS104で保持するジョブ情報300を特定し、印刷結果を登録する (F 2002)。

【0130】

印刷結果が通知されるプリントサーバ601はロードバランサ602によって決まる。しかし、DBMS104にクラスタ構成のプリントサーバ601で共通情報としてジョブ情報300を保持しているため、印刷結果を登録するプリントサーバ601は固定されない。

30

【0131】

印刷開始したプリントサーバ601とスケジュールアプリケーション801の間の接続が切れた場合であっても、他のプリントサーバ601が印刷結果の登録を行うため、接続の回復を待つ必要がない。また、プリントサーバ601が停止した場合であってもロードバランサ602により稼働しているプリントサーバ601に印刷結果が通知されるため、停止したプリントサーバ601の復帰を待つ必要もなく、印刷結果をすぐにユーザに通知することができる。

【0132】

<プリントサーバとのセッション切断時フロー>

40

図21は、転送終了通知がなされる前にプリントサーバ601とスケジュールアプリケーション801との間のセッション切断を検知した場合のスケジュールアプリケーション801の処理を示すフローチャートである。

【0133】

スケジュールアプリケーション801がプリントサーバ601との間のセッション切断を検知し (F 2101)、セッションを切断したプリントサーバ601が印刷を開始する予定だったジョブの再印刷指示を行う。

【0134】

次に、印刷指示時に必要なジョブ実行リストをヌル (NULL) に初期化する (F 2102)。そして、スケジュールアプリケーション801で管理するスケジュールリストに

50

登録されている全てのスケジュール情報 1 0 0 0 を処理する (F 2 1 0 3 ~ F 2 1 0 6)
。

【 0 1 3 5 】

まず、スケジュールリストに登録されているスケジュール情報 1 0 0 0 のうち、ステータス 1 0 0 5 がリクエスト済となっているものをジョブ実行リストに追加する (F 2 1 0 4、F 2 1 0 5)。そして、全てのスケジュール情報 1 0 0 0 の処理を終了すると (F 2 1 0 6)、スケジュール情報 1 0 0 0 に登録されているリクエスト先アドレス 1 0 0 8 にジョブ実行リストをもって印刷指示を行う (F 2 1 0 7)。

【 0 1 3 6 】

本実施形態では、リクエスト先アドレス 1 0 0 8 にロードバランサ 6 0 2 のアドレスを指定しているため、印刷指示は負荷分散される。ここで、スケジュールアプリケーション 8 0 1 からの印刷指示の詳細は上述した通りである。

10

【 0 1 3 7 】

印刷開始前にプリントサーバ 6 0 1 とのセッションが切断してしまい、印刷を開始できない状況になったとしても、再度、印刷指示を行うことにより印刷が実行される。また、プリントサーバ 6 0 1 とスケジュールアプリケーション 8 0 1 の間の接続が切れた場合も、他のプリントサーバ 6 0 1 が印刷実行を行うため、セッション切断したプリントサーバ 6 0 1 との接続の回復を待つ必要がない。

【 0 1 3 8 】

更に、プリントサーバ 6 0 1 が停止している場合であっても、ロードバランサ 6 0 2 により稼働しているプリントサーバ 6 0 1 に印刷指示されるため、停止したプリントサーバ 6 0 1 の復帰を待つ必要もなく、他のプリントサーバ 6 0 1 で印刷実行される。

20

【 0 1 3 9 】

尚、本発明は複数の機器 (例えば、ホストコンピュータ、インターフェース機器、リーダー、プリンタなど) から構成されるシステムに適用しても、1つの機器からなる装置 (例えば、複写機、ファクシミリ装置など) に適用しても良い。

【 0 1 4 0 】

また、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記録した記録媒体を、システム或いは装置に供給し、そのシステム或いは装置のコンピュータ (C P U 若しくは M P U) が記録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。これによっても、本発明の目的が達成されることは言うまでもない。

30

【 0 1 4 1 】

この場合、コンピュータ読み取り可能な記録媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記録媒体は本発明を構成することになる。

【 0 1 4 2 】

このプログラムコードを供給するための記録媒体として、例えばフレキシブルディスク、ハードディスク、光ディスク、光磁気ディスク、C D - R O M、C D - R、磁気テープ、不揮発性のメモリカード、R O Mなどを用いることができる。

【 0 1 4 3 】

40

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、次の場合も含まれることは言うまでもない。即ち、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 O S (オペレーティングシステム) などが実際の処理の一部又は全部を行い、その処理により前述した実施形態の機能が実現される場合である。

【 0 1 4 4 】

更に、記録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込む。その後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わる C P U などが実際の処理の一部又は全部を行い、その処理により前述した実施形態の機能

50

が実現される場合も含まれることは言うまでもない。

【図面の簡単な説明】

【 0 1 4 5 】

【図 1】本実施形態における印刷システムの構成の一例を示すブロック図である。

【図 2】サーバコンピュータ又はクライアントコンピュータのハードウェア構成の一例を示す図である。

【図 3】本実施形態における D B M S で管理されるジョブ情報の一例を示す図である。

【図 4】本実施形態における D B M S で管理されるデバイス情報の一例を示す図である。

【図 5】本実施形態における D B M S で管理されるドキュメント情報の一例を示す図である。

10

【図 6】プリントサーバシステム 1 0 1 の構成の一例を示すブロック図である。

【図 7】プリントデバイス 1 0 7 のハードウェア構成の一例を示す図である。

【図 8】プリントデバイス 1 0 7 の構成を階層的に示す図である。

【図 9】スケジュールアプリケーションの構成を示すブロック図である。

【図 1 0】ジョブマネージャ 9 0 2 が管理する印刷ジョブのスケジュール情報を示す図である。

【図 1 1】デバイスマネージャ 9 0 3 が管理する連続印刷制限数に関する情報を示す図である。

【図 1 2】プリントサーバ 6 0 1 が印刷リクエストを受け付けた際の処理を示すフローチャートである。

20

【図 1 3】スケジュールアプリケーション 8 0 1 がプリントサーバ 6 0 1 からスケジュール要求を受け付けた際の処理を示すフローチャートである。

【図 1 4】スケジュールアプリケーション 8 0 1 がプリントサーバ 6 0 1 に対して印刷指示を行う際の処理を示すフローチャートである。

【図 1 5】スケジュールアプリケーション 8 0 1 からの印刷指示を受け付けたプリントサーバ 6 0 1 の処理を示すフローチャートである。

【図 1 6】プリントサーバ 6 0 1 とスケジュールアプリケーション 8 0 1 との間のスケジュール登録時のシーケンスを示す図である。

【図 1 7】プリントサーバ 6 0 1、ロードバランサ 6 0 2、スケジュールアプリケーション 8 0 1、デバイスコントローラ 8 1 0 間の印刷シーケンスを示す図である。

30

【図 1 8】プリントサーバ 6 0 1 より印刷データの転送開始通知がなされた場合のスケジュールアプリケーション 8 0 1 の処理を示すフローチャートである。

【図 1 9】スケジュールアプリケーション 8 0 1 がデバイスコントローラ 8 1 0 から印刷結果を受け取った場合の処理を示すフローチャートである。

【図 2 0】プリントサーバ 6 0 1 がスケジュールアプリケーション 8 0 1 から印刷結果通知を受け付けた場合の処理を示すフローチャートである。

【図 2 1】転送終了の通知がなされる前にプリントサーバ 6 0 1 とスケジュールアプリケーション 8 0 1 との間のセッション切断を検知した場合のスケジュールアプリケーション 8 0 1 の処理を示すフローチャートである。

【符号の説明】

40

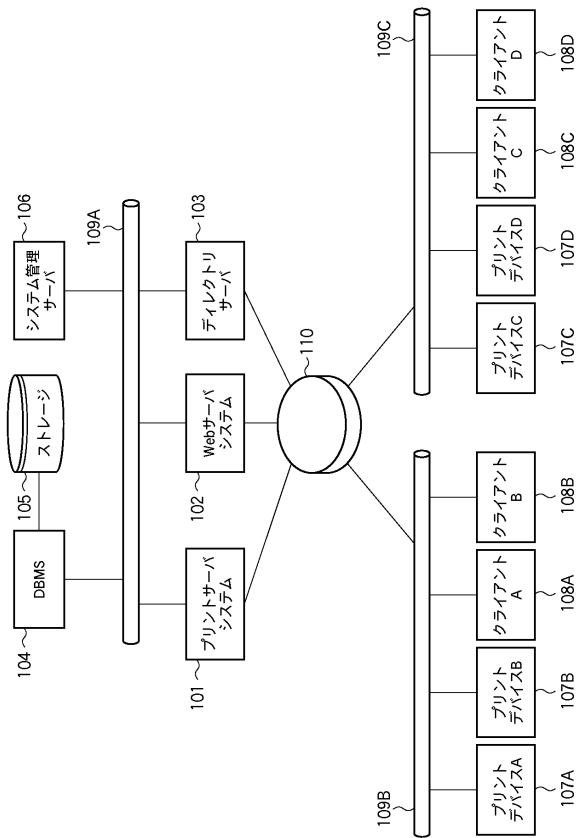
【 0 1 4 6 】

- 1 0 1 プリントサーバシステム
- 1 0 2 W e bサーバシステム
- 1 0 3 ディレクトリサーバ
- 1 0 4 D B M S (DataBase Management System)
- 1 0 5 ストレージ
- 1 0 6 システム管理サーバ
- 1 0 7 プリントデバイス
- 1 0 8 クライアント
- 1 0 9 L A N (Local Area Network)

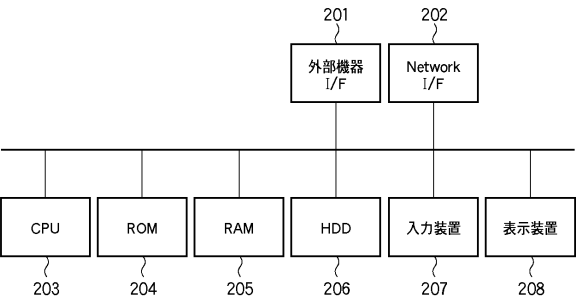
50

1 1 0 W A N (Wide Area Network)

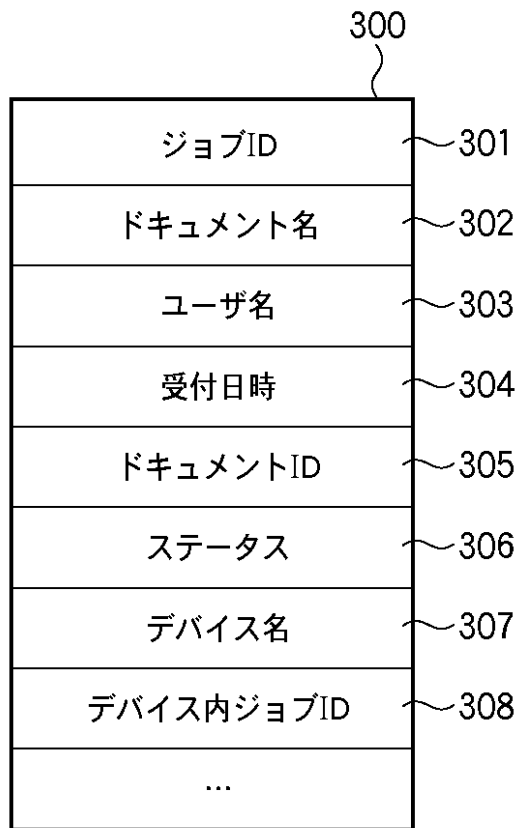
【 図 1 】



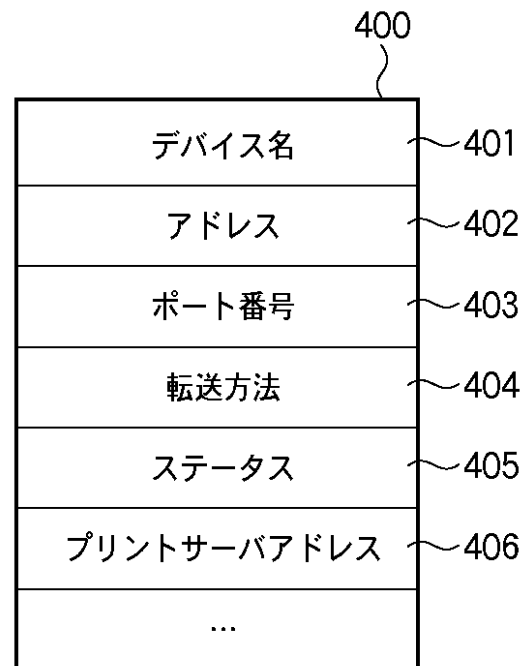
【 図 2 】



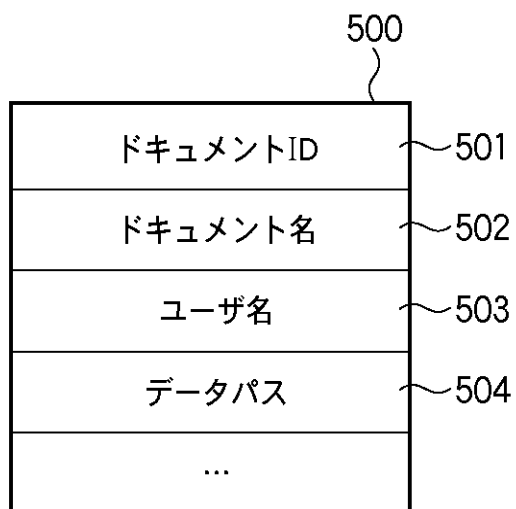
【図 3】



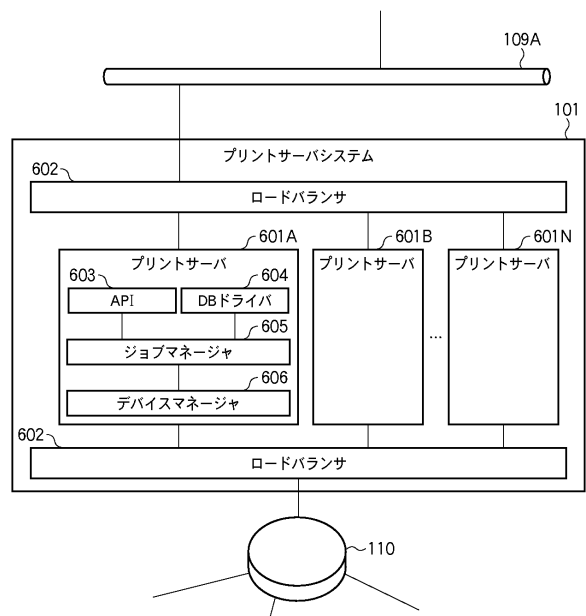
【図 4】



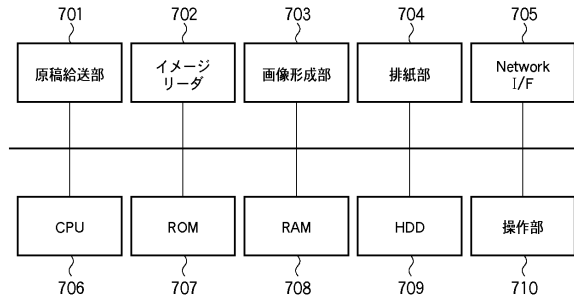
【図 5】



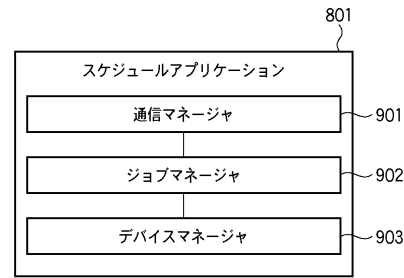
【図 6】



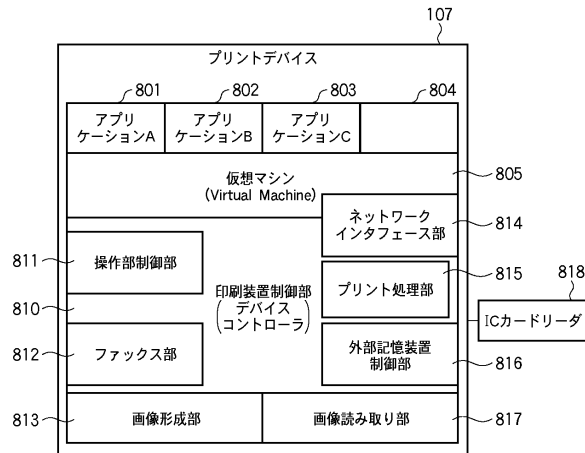
【図 7】



【図 9】



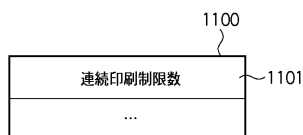
【図 8】



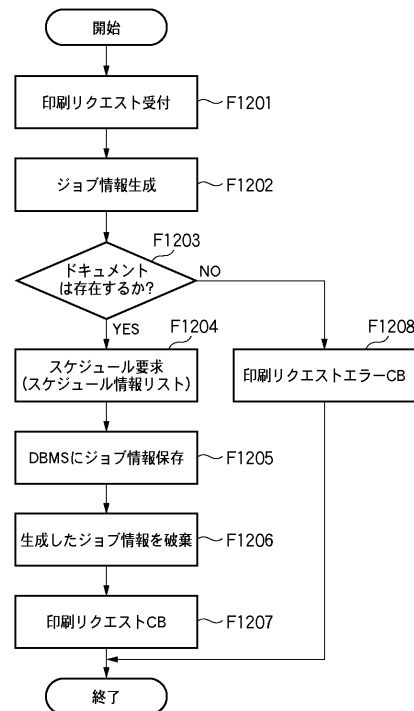
【図 10】



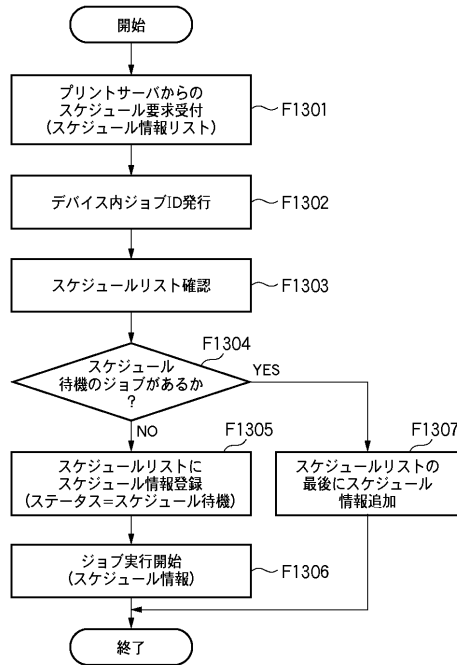
【図 11】



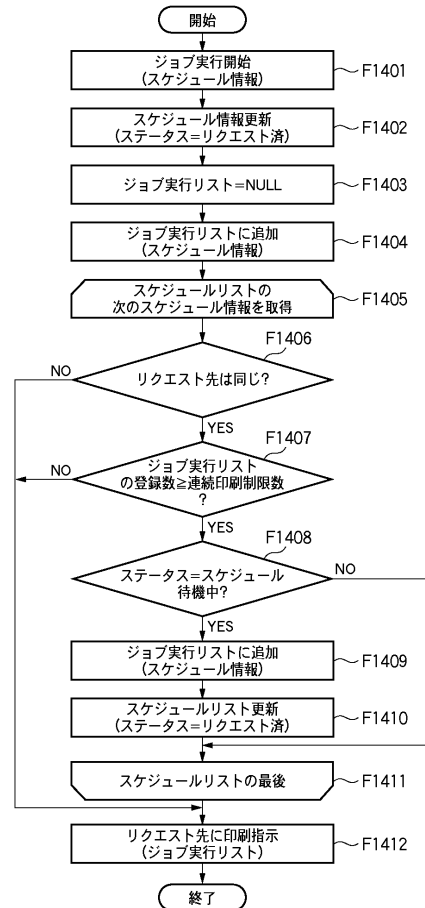
【図 12】



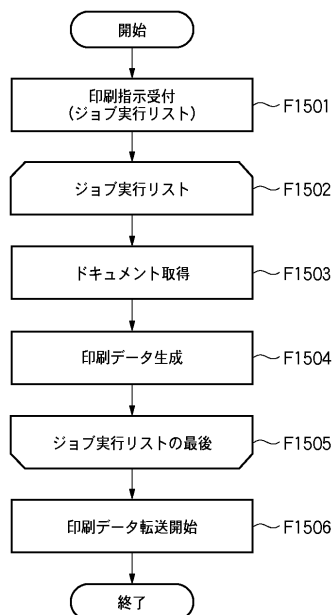
【図 13】



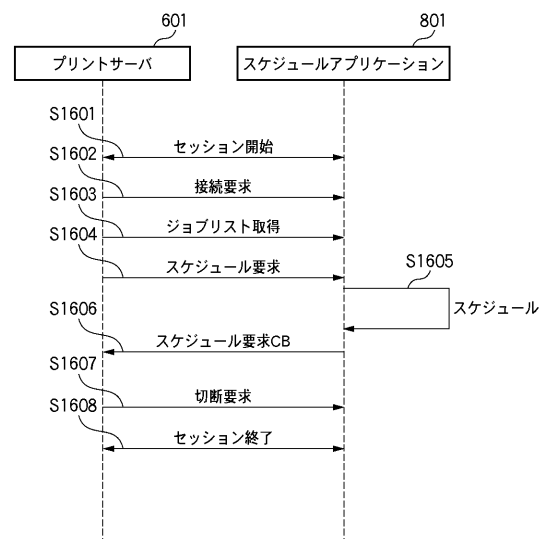
【図 14】



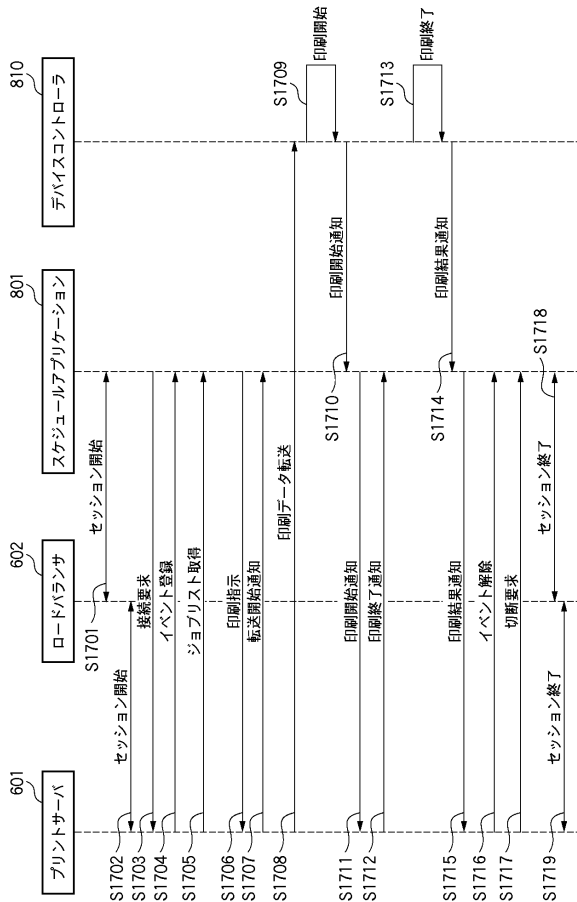
【図 15】



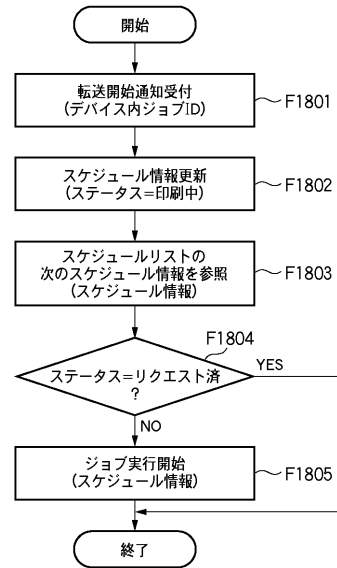
【図 16】



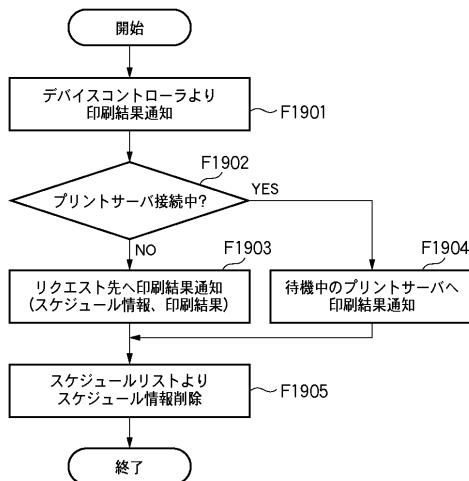
【図 17】



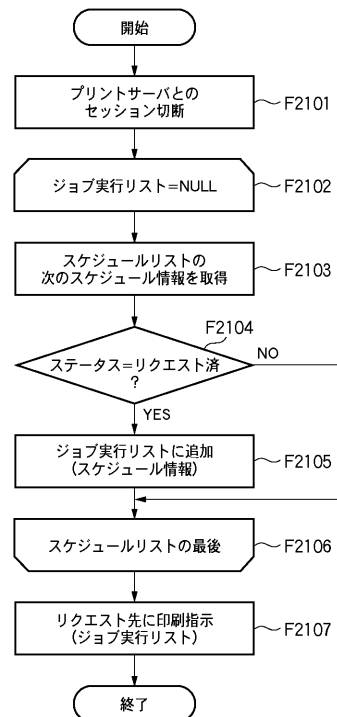
【図 18】



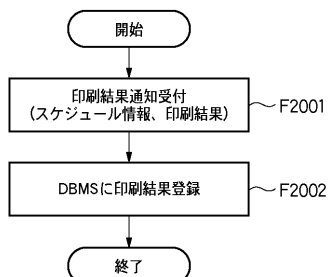
【図 19】



【図 21】



【図 20】



フロントページの続き

(72)発明者 堤 健介
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 池田 聡史

(56)参考文献 特開2008-107921(JP,A)
特開平07-219907(JP,A)
特開2008-158767(JP,A)

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