

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7589613号
(P7589613)

(45)発行日 令和6年11月26日(2024.11.26)

(24)登録日 令和6年11月18日(2024.11.18)

(51)国際特許分類

G 0 6 F	3/12 (2006.01)	F I	G 0 6 F	3/12	3 4 7
H 0 4 N	1/00 (2006.01)		G 0 6 F	3/12	3 1 4
			G 0 6 F	3/12	3 6 0
			G 0 6 F	3/12	3 4 0
			G 0 6 F	3/12	3 7 3

請求項の数 8 (全29頁) 最終頁に続く

(21)出願番号 特願2021-49168(P2021-49168)
 (22)出願日 令和3年3月23日(2021.3.23)
 (65)公開番号 特開2022-147777(P2022-147777)
 A)
 (43)公開日 令和4年10月6日(2022.10.6)
 審査請求日 令和6年1月19日(2024.1.19)

(73)特許権者 000006747
 株式会社リコー
 東京都大田区中馬込1丁目3番6号
 (74)代理人 100089118
 弁理士 酒井 宏明
 本田 正
 東京都大田区中馬込1丁目3番6号 株
 式会社リコー内
 審査官 佐賀野 秀一

最終頁に続く

(54)【発明の名称】 データ変換処理システム、データ変換処理方法及びデータ変換処理プログラム

(57)【特許請求の範囲】

【請求項1】

異なるデータ形式へのデータ変換処理を指定するキューが蓄積されるキューバ装置と、

前記キューにより指定されるデータ変換処理を行う変換部と、

オペレーションシステム上のユーザ空間であるコンテナに設けられた変換部数調整モジュールと、を備え、

前記変換部数調整モジュールは、前記キューバ装置に閾値以上の数のキューが蓄積された場合に、前記変換部と同じデータ変換処理を行う少なくとも一つの仮想変換部を前記コンテナに形成して、前記閾値以上の数となったキューに対応するデータ変換処理を行い、設定ファイルに設定されている起動条件に基づいて、前記仮想変換部を形成し、
前記設定ファイルには、前記キューの過去の処理実績に基づく起動条件が設定され、
前記起動条件は、前記キューの過去の処理実績に応じて逐次修正されること
を特徴とするデータ変換処理システム。

【請求項2】

前記変換部数調整モジュールは、前記設定ファイルに設定されている起動条件に基づいて、定期的に前記仮想変換部を形成して、前記キューバ装置に蓄積されているキューに対応するデータ変換処理を行うこと

を特徴とする請求項1に記載のデータ変換処理システム。

【請求項3】

前記設定ファイルには、前記起動条件として少なくとも起動日時が設定されており、前記変換部数調整モジュールは、前記設定ファイルに設定されている起動日時に前記仮想変換部を形成して、前記キューサーバ装置に蓄積されているキューに対応するデータ変換処理を行うこと

を特徴とする請求項1に記載のデータ変換処理システム。

【請求項4】

前記変換部数調整モジュールは、前記キューの過去の処理実績に基づいて検出した、キューの処理が停滞する時間帯を、前記起動日時として前記設定ファイルに設定することで、キューの処理が停滞する時間帯に前記仮想変換部を形成して、前記キューサーバ装置に蓄積されているキューに対応するデータ変換処理を行うこと

を特徴とする請求項3に記載のデータ変換処理システム。

【請求項5】

異なるデータ形式へのデータ変換処理を指定するキューが蓄積されるキューサーバ装置と、前記キューにより指定されるデータ変換処理を行う変換部と、
オペレーションシステム上のユーザ空間であるコンテナに設けられた変換部数調整モジュールと、を備え、

前記変換部数調整モジュールは、前記キューサーバ装置に閾値以上の数のキューが蓄積された場合に、前記変換部と同じデータ変換処理を行う少なくとも一つの仮想変換部を前記コンテナに形成して、前記閾値以上の数となったキューに対応するデータ変換処理を行い、前記変換部によるデータ変換処理が所定時間以上停滞しているキューが、前記キューサーバ装置に蓄積されている場合に、前記仮想変換部を形成して、所定時間以上停滞しているキューに対応するデータ変換処理を行うこと

を特徴とするデータ変換処理システム。

【請求項6】

異なるデータ形式へのデータ変換処理を指定するキューが蓄積されるキューサーバ装置と、前記キューにより指定されるデータ変換処理を行う変換部と、
オペレーションシステム上のユーザ空間であるコンテナに設けられた変換部数調整モジュールと、を備え、

前記変換部数調整モジュールは、前記キューサーバ装置に閾値以上の数のキューが蓄積された場合に、前記変換部と同じデータ変換処理を行う少なくとも一つの仮想変換部を前記コンテナに形成して、前記閾値以上の数となったキューに対応するデータ変換処理を行い、前記変換部によるデータ変換処理がエラーのステータスとなっているキューが、前記キューサーバ装置に蓄積されている場合に、前記仮想変換部を形成して、エラーのステータスとなっているキューに対応するデータ変換処理を行うこと

を特徴とするデータ変換処理システム。

【請求項7】

異なるデータ形式へのデータ変換処理を指定するキューが蓄積されるキューサーバ装置と、前記キューにより指定されるデータ変換処理を行う変換部とを備えたデータ変換処理システムのデータ変換処理方法であって、

オペレーションシステム上のユーザ空間であるコンテナに、変換部数調整モジュールを設け、

前記キューサーバ装置に閾値以上の数のキューが蓄積された場合に、前記変換部数調整モジュールが、前記変換部と同じデータ変換処理を行う少なくとも一つの仮想変換部を前記コンテナに形成して、前記閾値以上の数となったキューに対応するデータ変換処理を行い、

前記変換部数調整モジュールは、設定ファイルに設定されている起動条件に基づいて、前記仮想変換部を形成し、

前記設定ファイルには、前記キューの過去の処理実績に基づく起動条件が設定され、前記起動条件は、前記キューの過去の処理実績に応じて逐次修正されること

を特徴とするデータ変換処理方法。

10

20

30

40

50

【請求項 8】

異なるデータ形式へのデータ変換処理を指定するキューが蓄積されるキューサーバ装置と、前記キューにより指定されるデータ変換処理を行う変換部とを備えたデータ変換処理システムのデータ変換処理プログラムであって、

コンピュータに、

オペレーションシステム上のユーザ空間であるコンテナに、変換部数調整モジュールを形成するステップと、

前記キューサーバ装置に閾値以上の数のキューが蓄積された場合に、前記変換部数調整モジュールが、前記変換部と同じデータ変換処理を行う少なくとも一つの仮想変換部を前記コンテナに形成して、前記閾値以上の数となったキューに対応するデータ変換処理を行うステップとを実行させ、

10

前記変換部数調整モジュールは、設定ファイルに設定されている起動条件に基づいて、前記仮想変換部を形成し、

前記設定ファイルには、前記キューの過去の処理実績に基づく起動条件が設定され、

前記起動条件は、前記キューの過去の処理実績に応じて逐次修正されること

を特徴とするデータ変換処理プログラム。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、データ変換処理システム、データ変換処理方法及びデータ変換処理プログラムに関する。

20

関する。

【背景技術】**【0002】**

従来、ワードプロセッサアプリケーション等で作成した電子文書ファイルを、プリンタで解釈可能なデータ形式の電子データ（ページ記述言語（PDL）形式のプリントデータ）へプリンタドライバで変換し、プリンタへ送信して出力することが行われている。

【0003】

また、例えばオフィス文書をプリントデータへと変換するプリンタドライバのように、あるデータ形式の電子データを他のデータ形式の電子データに変換することで、出力時の用途に適した形式の電子データへと変換するデータ変換処理は存在する。また、単一のコンピュータにおけるデータ変換処理に限らず、複数台のコンピュータを用いたコンピュータシステムにてデータ変換処理を行うことも従来知られている（例えば、特許文献1又は特許文献2）。

30

【0004】

あるデータ形式から他のデータ形式に変換する際、プログラムの独立性やメンテナンス性を高めるために、データ形式を変換する変換部（データ変換プログラム又はデータ変換ライブラリ）が用いられる。

【発明の概要】**【発明が解決しようとする課題】****【0005】**

しかし、従来は、予め設けられている変換部で処理可能なジョブ数以上の数のジョブの処理が依頼されると、必然的に処理が停滞する問題があった。

40

【0006】

本発明は、上述の課題に鑑みてなされたものであり、変換部に対するジョブ数の増加に柔軟に対応可能なデータ変換処理システム、データ変換処理方法及びデータ変換処理プログラムの提供を目的とする。

【課題を解決するための手段】**【0007】**

上述した課題を解決し、目的を達成するために、本発明は、異なるデータ形式へのデータ

50

タ変換処理を指定するキューが蓄積されるキューサーバ装置と、キューにより指定されるデータ変換処理を行う変換部と、オペレーションシステム上のユーザ空間であるコンテナに設けられた変換部数調整モジュールと、を備え、変換部数調整モジュールは、キューサーバ装置に閾値以上の数のキューが蓄積された場合に、変換部と同じデータ変換処理を行う少なくとも一つの仮想変換部を前記コンテナ内に形成して、閾値以上の数となったキューに対応するデータ変換処理を行い、設定ファイルに設定されている起動条件に基づいて、仮想変換部を形成し、設定ファイルには、キューの過去の処理実績に基づく起動条件が設定され、起動条件は、キューの過去の処理実績に応じて逐次修正される。

【発明の効果】

【0008】

本発明によれば、変換部に対するジョブ数の増加に柔軟に対応できるという効果を奏する。

【図面の簡単な説明】

【0009】

【図1】図1は、クラウドプリントを説明するための図である。

【図2】図2は、クラウドプリントを実現するコンピュータシステムの構成例を示すプロック図である。

【図3】図3は、図2の利用手順を例示するフローチャートである。

【図4】図4は、実施形態の機能構成を示すブロック図である。

【図5】図5は、図4の変換部の詳細な構成を示すブロック図である。

【図6】図6は、図4の非同期変換サーバの動作を示すタイミングチャートである。

【図7】図7は、図6におけるデータの遷移を例示する概念図である。

【図8】図8は、実施形態のデータ変換システムのサーバ構成を説明するための図である。

【図9】図9は、Linux（登録商標）系の変換部を示す図である。

【図10】図10は、Windows（登録商標）系の変換部を示す図である。

【図11】図11は、メッセージキューのOCR処理のキュー、及び、OCR処理を行う変換部を示す図である。

【図12】図12は、実施形態のデータ変換システムのコンテナの構成を示す図である。

【図13】図13は、コンテナに設けられている変換部数調整モジュールによるCaas変換部の起動動作の流れを示すフローチャートである。

【図14】図14は、実施の形態のデータ変換システムに設けられている変換部数調整モジュールにおける設定ファイルの書き換え処理を説明するためのフローチャートである。

【図15】図15は、実施の形態のデータ変換システムに設けられている変換部数調整モジュールにおける設定ファイルの書き換え処理の他の例を説明するためのフローチャートである。

【図16】図16は、過去の実績調査の動作の流れを示すフローチャートである。

【図17】図17は、変換部数調整モジュールにおける、既存の変換部の状態を判断する処理を説明するためのフローチャートである。

【図18】図18は、変換部数調整モジュールにおける、既存の変換部の状態を判断する処理を詳細に説明するためのフローチャートである。

【図19】図19は、変換部数調整モジュールにおける、ステータスがエラーとなったジョブの再実行動作を説明するためのフローチャートである。

【図20】図20は、変換部数調整モジュールによるCaas変換部のスケジュール起動動作を説明するためのフローチャートである。

【図21】図21は、変換部数調整モジュールによるスケジュール起動処理の詳細な動作の流れを示すフローチャートである。

【図22】図22は、Caas変換部を自動起動するスケジュールを設定するために行う、過去の実績の自動調査を説明するためのフローチャートである。

【発明を実施するための形態】

【0010】

10

20

30

40

50

以下、本発明の実施形態について図面を参照しながら説明する。まず、本実施形態の利用形態を述べ、次に、本実施形態の機能的構成及び動作について説明する。

【0011】

(クラウドプリント)

本明細書で「クラウドプリント」と呼んでいるものは、クラウドコンピューティング型のプリントサービスである。概念について図1を参照して説明する。図1において、インプット側機器（例えばパーソナルコンピュータ（以下「PC」）12）で、ワードプロセッサアプリケーションで作成した電子文書ファイルなどを、RPCS（株式会社リコーが開発したページ記述言語）等のプリントデータに変換し（A1）、USBやLAN経由でアウトプット側機器（例えば複合機（以下「MFP」）13）に送信し（A2）、出力する。

10

【0012】

クラウドプリントでは、A1の手順をクラウド10上で実行する（A4）。インプット側機器もアウトプット側機器も、クラウド10側から見るとクライアントとなり、これらクライアントがクラウド10上にある仮想的なサーバ11と通信を行い、所与の目的を達成する。

【0013】

A1のデータ変換処理を行う必要がないので、ローカルクライアントにデータ変換エンジンを持つ必要がない。ところが、クラウド10側で用意するデータ変換エンジンは、多種多様な電子文書ファイルに対応する必要があり、その種類は今後も増えていく可能性がある。出力に関しては、アウトプット側機器がMFPのみならずプロジェクタや、或いは全く未知の機器への出力に対応する可能性がある。別のクラウドサービスに接続できるデータ形式への変換もあり得る。データ変換処理の種類の増加に柔軟に対応可能とすることが好ましい。

20

【0014】

図1を用いて説明したようなクラウドプリントを、図2に示すようなクラウドコンピューティング型のコンピュータシステム（データ変換システム）で構成する。図2において、ローカル側では汎用的なPC12やMFP13が存在し、LANのゲートウェイ等に設置されているファイヤーウォール103を介して、インターネットに接続する。PC12及びMFP13が同一のネットワーク上にあってもなくても、クラウドプリントは有効である。以下では、同一のネットワーク上にない場合と仮定して説明する。ローカル側の機器同士でデータの受け渡しが簡単にできない場合の方がクラウドプリントの利便性がよく理解されるためである。

30

【0015】

図2において、クラウド側は、データ変換システム100、印刷ジョブ管理サーバ102、リバースプロキシ104、認証サーバ105、認証用データベース106を備える。データ変換システム100は、非同期変換サーバ101と共有ファイルストレージ107とジョブ情報管理データベース108を有する。共有ファイルストレージ107をデータ変換システム100の外側に置く設計をしてもよい。ジョブ情報管理データベース108も、図4に示すように、非同期変換サーバ101を構成する一部となるように設計してもよい。

40

【0016】

ローカル側機器は、通信がリバースプロキシ104によって仲介されることで、クラウド側のサーバ群やストレージの内部構造を意識した処理や通信をする必要がない。認証サーバ105と認証用データベース106は、ユーザ及び/又はローカル側機器の認証を行う。印刷ジョブ管理サーバ102とデータ変換システム100は、ローカル側機器がアップロードした印刷ジョブを管理/登録する。ローカル側機器から印刷要求があった場合にはプリントデータを送信する。

【0017】

印刷ジョブには、ワードプロセッサアプリケーションなどで作成した電子文書ファイルが含まれる。データ変換システム100は、印刷ジョブに含まれる電子文書ファイルを、

50

例えばMFP13で出力可能なデータ形式（ページ記述言語（PDL））の電子データであるプリントデータに変換する。

【0018】

図2に示したコンピュータシステムの利用例について図3を参照して説明する。図3は、図2の利用手順を例示するフローチャートである。ユーザはまず、PC12から本クラウドプリントサービスにログインする（S101）。認証処理は認証サーバ105が実行する。次に、PC12から印刷したい電子文書ファイルを本クラウドプリントサービスにアップロードする（S102）。これが印刷ジョブの登録である。印刷ジョブの登録処理は、印刷ジョブ管理サーバ102が実行する。ここまでがPC12で行われる手順である。

【0019】

印刷ジョブが登録されると、本クラウドプリントサービスではバックグラウンドでデータ変換が行われる。印刷ジョブ管理サーバ102がデータ変換システム100に登録された電子文書ファイルのデータ変換を依頼する（S103）。ここで行われる処理の詳細については後述する。

【0020】

ユーザ印刷出力する場合、MFP13にて以下の一連の手順を行う。まず、MFP13が本クラウドプリントサービスに対応の機器であるかを確かめる。機器としての認証（S104）である。次に、ユーザは、MFP13を操作してMFP13から本クラウドプリントサービスにログインする。ユーザとしての認証（S105）である。

【0021】

認証後、ユーザは、ジョブ一覧を本クラウドプリントサービスから取得し、MFP13の図示しない表示部に表示する（S106）。次に、表示されているジョブ一覧の中から印刷出力を所望する印刷ジョブを選択する（S107）。そうすると、MFP13が本クラウドプリントサービスから印刷ジョブを取得し、出力する（S108）。

【0022】

上述したように、PC12とMFP13が物理的に離れていてもよく、PC12でMFP13のプリンタドライバ（つまりデータ変換エンジン）を用意する必要がないので、例えば、会議資料を作成／アップロードしておき、海外出張時に現地で印刷出力するような用途に有效である。

【0023】

（データ変換システム）

本実施形態のデータ変換システム100は、多種多様なデータ変換要求に対応し、また、データ変換要求の種類及び数の増加に柔軟に対応するため、以下に説明する構成を備える。図4に、データ変換システム100の機能構成を示す。

【0024】

図4に示すように、データ変換システム100は、印刷ジョブ管理サーバ102の一機能であるフロントエンドアプリケーション102aと、共有ファイルストレージ107と、非同期変換サービスを提供する非同期変換サーバ101とを含む。非同期変換サーバ101は、データ変換要求管理部111と、メッセージキューリースト112と、ジョブ情報管理データベース108と、複数の変換部120を含む構成である。

【0025】

印刷ジョブ管理サーバ102は、図3を参照して説明したようなクラウドプリントのソリューションにおいて、PC12などインプット側機器から送信された印刷ジョブを受信して、印刷ジョブに含まれる電子文書ファイル等を共有ファイルストレージ107に登録する機能を備える。また、MFP13などアウトプット側機器から印刷ジョブの出力指示を受信して、出力指示の実行に必要なファイル変換を判断し、データ変換システム100に変換を実行させる制御を行う。変換後のファイルが共有ファイルストレージ107に登録されるので、印刷ジョブ管理サーバ102は、登録された変換後のファイルを、出力指示を送信したMFP13に送信する。

【0026】

10

20

30

40

50

フロントエンドアプリケーション 102a は、印刷ジョブ管理サーバ 102 の機能のうち、出力指示の実行に必要なファイル変換を判断する機能を備える。例えば、登録されている電子ファイルがマイクロソフト社のワード（登録商標）ドキュメントファイルであり、出力指示を送信した MFP 13 が RPCS（株式会社リコーが開発したページ記述言語）対応の機種である場合、ワードドキュメントファイルのデータ形式のデータから RPCS のデータ形式のデータへの変換を判断する。

【0027】

フロントエンドアプリケーション 102a は、判断したデータ変換の種類を、例えば、「type = “doc2rpcs”」のようなパラメータにして、非同期変換サーバ 101 に送信するリクエストに埋め込む。同様に、変換元のデータの URI (Uniform Resource Indicator) もパラメータとしてリクエストに埋め込む。ここで、URI は、共有ファイルストレージ 107 におけるデータの存在する場所の情報である。

【0028】

データ変換の種類とデータの URI を少なくとも含むリクエストを受け取った非同期変換サーバ 101 では、まず、データ変換要求管理部 111 がリクエストを受け取る。データ変換要求管理部 111 は、データ変換に係るリクエストを解析して、データ変換の種類に合致する変換部 120 があるか否かを判断する。ある場合には、その変換部 120 に対応するメッセージキュー 112 に、キューを入れる。ない場合には、複数の別種の変換部 120 を連携させて、所望のデータ変換の種類ができる組み合わせを判断し、最初の変換部 120 に対応するメッセージキュー 112 に、キューを入れる。

【0029】

データ変換要求管理部 111 は、メッセージキュー 112 へキューを入れることと同時に並行して、ジョブ情報管理データベース 108 に、当該リクエストの情報を登録又は更新する。ジョブ情報管理データベース 108 に登録される情報とは、ファイルの場所（共有ファイルストレージ 107 上の URI）、リクエストに含まれるデータ変換の種類、当該リクエストの状態である。リクエストの状態としては、例えば「データ変換実行中」、「データ変換中継中」、「データ変換完了」などがあり、データ変換の処理状況を示すステータスを表す。

【0030】

メッセージキュー 112 は、複数の変換部 120 の各々に対応するキューを有する。各変換部 120 は、その変換部 120 に対応するキューを監視しており、キューにメッセージが入り次第、そのメッセージに対応するジョブ情報管理データベース 108 に登録されたリクエストの情報を取りに行き、取得したリクエストの情報に含まれる URI に基づいて、共有ファイルストレージ 107 上のデータを取得する。そして、取得したデータの変換を行う。

【0031】

各変換部 120 は、変換後のデータを共有ファイルストレージ 107 上に登録し、ジョブ情報管理データベース 108 上に登録されたリクエストの情報を書き換えて更新する。また、連携してデータ変換する場合に、その変換部でデータ変換した後、新たなリクエストを次にデータ変換を行う変換部のキューに入れる。

【0032】

複数の変換部 120 は、各々異なる重複しない変換部（第 1 の変換部 121、第 2 の変換部 122、・・・、第 n の変換部 12n）である。各変換部 120 は、図 5 に示すように、変換処理ライブラリ 120a と変換エンジン 120b を備える構成である。

【0033】

変換処理ライブラリ 120a は、その変換部 120 に対応するキューを監視する処理や、変換後のデータを共有ファイルストレージ 107 上に登録する処理や、ジョブ情報管理データベース 108 上に登録されたリクエストの情報を書き換えて更新する処理や、連携してデータ変換する場合にデータ変換後に新たなリクエストを次にデータ変換を行う変換

10

20

30

40

50

部のキューに入れる処理などを実行する。

【0034】

変換エンジン120bとしては、例えば、以下のようなものがある。

・`o f f i c e 2 r p c s`：マイクロソフト（登録商標）社の提供するオフィスソフトで作成した電子文書ファイルをRPCS（株式会社リコーが開発したページ記述言語）に変換するエンジン

・`p d f 2 r p c s`：PDF（Portable Document Format）をRPCSに変換するエンジン

・`i m g 2 p d f`：画像ファイルをPDFに変換するエンジン

・`h t m l 2 p d f`：HTML（Hyper Text Markup Language）で記述されたページファイルをPDFに変換するエンジン

【0035】

（非同期変換サービス）

図4に示す非同期変換サーバ101による非同期変換サービスの流れについて、図6のタイミングチャートを参照して説明する。

【0036】

非同期変換サービスは、フロントエンドアプリケーション102aが共有ファイルストレージ107に登録しておいた（S201）データに対するデータ変換の要求（リクエスト）を、データ変換システム100を構成するデータ変換要求管理部111に送信する（S202）と始まる。

【0037】

S202で送信されるリクエストは、フロントエンドアプリケーション102aが判断したデータ変換の種類が、例えば、「`t y p e = " d o c 2 r p c s "`」のようなパラメータにて埋め込まれている。また、リクエストには、変換しようとしているデータのURI（Uniform Resource Indicator）もパラメータで埋め込まれている。

【0038】

リクエストを受け取ったデータ変換要求管理部111は、データ変換に係るリクエストを解析して、データ変換の種類に合致する変換部120があるか否かを判断する（S203）。ある場合には、その変換部120に対応するメッセージキュー112に、キューを入れる。ない場合には、複数の別種の変換部120を連携させて、所望のデータ変換の種類ができる組み合わせを判断し、最初の変換部120に対応するメッセージキュー112に、キューを入れる。

【0039】

図6に依拠する本例では、データ変換の種類に合致する変換部120がない場合であって、2種の変換部120を連携させれば、所望のデータ変換が可能なケースを考える。例えば、図7に示すように、「`o f f i c e 2 r p c s`（マイクロソフト社の提供するオフィスソフトで作成した電子文書ファイルをRPCS（株式会社リコーが開発したページ記述言語）に変換するエンジン）」がない場合であって、「`o f f i c e 2 p d f`」と「`p d f 2 r p c s`」を連携させれば、オフィス文書からプリント記述言語へのデータ変換が可能な場合などである。

【0040】

データ変換要求管理部111は、メッセージキュー112へキューを入れること（S205）と同時に並行して、ジョブ情報管理データベース108に、当該リクエストの情報を登録する（S204）。

【0041】

キューを監視している各変換部120は、キューにメッセージが入ったことを検知すると（S206）、そのメッセージに対応するジョブ情報管理データベース108に登録されたリクエストの情報を取りに行き（S207、S208）、取得したリクエストの情報に含まれるURIに基づいて、共有ファイルストレージ107上のデータを取得する（S209、S210）。そして、取得したデータの変換を行う（S211）。

10

20

30

40

50

【0042】

各変換部120は、変換後のデータを共有ファイルストレージ107上に登録又は更新する(S212)と同時に、変換したことを表すために、ジョブ情報管理データベース108に登録されている当該リクエストの情報を更新する(S213)。このとき、リクエストの状態が、“データ変換中継中”に更新される。

【0043】

データ変換要求管理部111は、ジョブ情報管理データベース108の更新を監視している。S203で2段階に分けたデータ変換のプロセスの1段階目が終了したことを検知すると(S214)、データ変換要求管理部111は、2段階目のデータ変換のメッセージを生成して、メッセージキュー112へキューを入れる(S215)。

10

【0044】

なお、各変換部120がメッセージキュー112へ直接キューを入れる構成としてもよい。この場合、変換部120(ないし変換処理ライブラリ120a)がその変換部におけるデータ変換後に、新たなリクエストを次にデータ変換を行う変換部のキューに入れる処理を実行する。

【0045】

キューを監視している各変換部120は、キューにメッセージが入ったことを検知すると(S216)、そのメッセージに対応するジョブ情報管理データベース108に登録されたリクエストの情報を取りに行き(S217、S218)、取得したリクエストの情報に含まれるURIに基づいて、共有ファイルストレージ107上のデータを取得する(S219、S220)。そして、取得したデータの変換を行う(S221)。

20

【0046】

各変換部120は、変換後のデータを共有ファイルストレージ107上に登録又は更新する(S222)と同時に、変換したことを表すために、ジョブ情報管理データベース108に登録されている当該リクエストの情報を更新する(S223)。このとき、リクエストの状態が、“データ変換完了”に更新される。

【0047】

S215～S223の処理は、S205～S213の処理と同様のものである。ただし、異なる変換部120(例えば、第1の変換部121と第2の変換部122)が行う(同じデータ変換を2重に行う場合を除く)。

30

【0048】

上述した動作を行うことによって、図7に示したような直接変換する変換エンジンが存在しない変換要求のリクエストに対して、そのようなリクエストを満足する変換エンジン及び変換リクエストを用意するのではなく、複数の既存の変換部を組み合わせ連携することによって、変換要求を処理することができるようになる。

【0049】

データ変換のリクエストの種類は、上述したように、データ変換システム100を立ち上げたあとに増加する可能性がある。これに対して、本実施形態によれば柔軟に対応することができるようになる。

【0050】

なお、変換部の種類としては、上記実施形態のものに限定されない。他に例えば、バーコード画像のデータを、文字列のデータに変換する変換部や、画像ファイルのデータからOCR(文字読み取り)して、文字列のデータに変換する変換部など、種々のものが考えられる。

40

【0051】

(ジョブ数の増加に対する対応)

次に、本実施形態のデータ変換システム100は、このようにデータ変換処理の種類の増加に対応可能であると共に、データ変換処理のジョブ数の増加にも対応可能となっている。すなわち、本実施形態のデータ変換システム100は、キューに溜まったジョブが所定の閾値以上となった際に、必要数の変換部120を起動させることで、ジョブ数の増加

50

にも対応するようになっている。

【0052】

(サーバ構成)

まず、本実施形態のデータ変換システム100は、図8に示すサーバ構成を有している。すなわち、本実施形態のデータ変換システム100は、図8に示すように、アップサーバ装置30、フロントサーバ装置31、キューサーバ装置32、第1の印刷ジョブ管理サーバ装置33、第2の印刷ジョブ管理サーバ装置34、データベースサーバ装置(DBサーバ装置)35及びファイルサーバ装置36を有している。

【0053】

アップサーバ装置30は、上述のフロントエンドアプリ102aを備えている。フロントサーバ装置31は、上述のデータ変換要求管理部111を備えている。キューサーバ装置32には、メッセージキュー112が蓄積される。第1の印刷ジョブ管理サーバ装置33及び第2の印刷ジョブ管理サーバ装置34は、予め定められた数の変換部120-1~120-n(nは、2以上の自然数)が設けられている。DBサーバ装置35には、上述のジョブ管理DB108が設けられている。ファイルサーバ装置36には、上述の共有ファイルストレージ107が設けられている。

10

【0054】

なお、この例では、2台の印刷ジョブ管理サーバ装置33、34を設けているが(冗長構成)、1台又は3台以上の印刷ジョブ管理サーバ装置を設けてもよい。印刷ジョブ管理サーバ装置を複数台設けることで、一つのサーバ装置が異常になんしても、他のサーバ装置の変換部120でドキュメント処理等を継続することができる。また、アップサーバ装置30、フロントサーバ装置31、キューサーバ装置32、DBサーバ装置35及びファイルサーバ装置36も複数台設ける構成としてもよい。

20

【0055】

(ジョブ情報テーブルの構成)

DBサーバ装置35のジョブ情報管理DB108には、ジョブ情報テーブル37及びタスクステータステーブル38が設けられている。ジョブ情報テーブル37は、以下の表1に示す構成となっている。

【0056】

30

40

50

【表1】

(表1) ジョブ情報テーブル37						
ジョブID	ユーザID	ステータス	投入日時	完了日時	ジョブタイプ	パラメータ
1	AAA	done	2018/11/17 17:59	2018/11/17 18:03	office2pdf	param1
2	AAA	error	2018/11/17 18:02	2018/11/17 18:05	office2pdf	param2
3	BBB	doing	2018/11/18 13:55		image2pdf	param3
4	BBB	doing	2018/11/18 13:57		image2pdf	param4
6	BBB	accepted	2018/11/18 13:58		image2pdf	param5
7	CCC	accepted	2018/11/18 14:01		send_email	param6
8	DDD	accepted	2018/11/18 14:02		ocr	param7
9	CCC	accepted	2018/11/18 14:05		send_email	param8

【0057】

すなわち、この表1に示すように、ジョブ情報テーブル37には、どのジョブかを識別するためのジョブID、どのユーザのジョブかを識別するユーザID、いつジョブがリクエストされたかを示す投入日時が記録される。また、ジョブ情報テーブル37には、いつ処理が終了したかを示す完了日時、何のジョブかを示すジョブタイプ、処理を実行する際のパラメータ、そして処理を実行した結果が記録される。

【0058】

(変換部の構成)

変換部120には、一例として、図9に示すLinux(登録商標)系の変換部120

と、図10に示すWindows(登録商標)系の変換部120が設けられている。Linux(登録商標)系及びWindows(登録商標)系は、冗長構成で例えば第1のインスタンス及び第2のインスタンス等のように、2台ずつサーバインスタンスが設けられている。また、各インスタンスが担当する処理は、図9及び図10に示すように分かれている。

【0059】

また、キューにも「Nomal(処理時間が短いキュー)」と「Heavy(処理時間が長くなるキュー)」という2種類のキューがある。処理に時間を要する機能は、Heavyキューに溜めることで、短時間で処理ができるキューをスムーズに処理できるようにしている。

10

【0060】

サーバインスタンス内の変換部120は、全ての変換部120が全てのキューを監視しているわけではない。例えば、図9に示すように、Linux(登録商標)系の変換部120では、第1のインスタンスの第1の変換処理プロセス及び第2の変換処理プロセスでOCR(Optical Character Recognition)及びメール処理を担当する。また、第2のインスタンスの第7の変換処理プロセス及び第8の変換処理プロセスでOCR(Optical Character Recognition)及びメール処理を担当する。

【0061】

また、図9に示すLinux(登録商標)系の変換部120では、第1のインスタンスの第3の変換処理プロセス及び第4変換処理プロセスでバーコード、PDF及びメール処理を担当する。また、第2のインスタンスの第9の変換処理プロセス及び第10の変換処理プロセスでバーコード、PDF及びメール処理を担当する。

20

【0062】

また、図9に示すLinux(登録商標)系の変換部120では、第1のインスタンスの第5の変換処理プロセス及び第6変換処理プロセスで画像変換、テキスト変換及びメール処理を担当する。また、第2のインスタンスの第11の変換処理プロセス及び第12の変換処理プロセスで画像変換、テキスト変換及びメール処理を担当する。

【0063】

同様に、図10に示すように、Windows(登録商標)系の変換部120では、第1のインスタンスの第1の変換処理プロセス及び第2の変換処理プロセスでofffce変換処理及びPDL(Page Description Language)変換処理を担当する。また、第2のインスタンスの第7の変換処理プロセス及び第8の変換処理プロセスでofffce変換処理及びPDL(Page Description Language)変換処理を担当する。

30

【0064】

また、図10に示すWindows(登録商標)系の変換部120では、第1のインスタンスの第3の変換処理プロセス及び第4変換処理プロセスで、日本語OCR処理(Nomal及びHeavy)を担当する。また、第2のインスタンスの第9の変換処理プロセス及び第10の変換処理プロセスで、日本語OCR処理(Nomal及びHeavy)を担当する。

40

【0065】

また、図10に示すWindows(登録商標)系の変換部120では、第1のインスタンスの第5の変換処理プロセス及び第6変換処理プロセスで、バーコード処理(Nomal及びHeavy)及び帳票処理を担当する。また、第2のインスタンスの第11の変換処理プロセス及び第12の変換処理プロセスで、バーコード処理(Nomal及びHeavy)及び帳票処理を担当する。

【0066】

このように処理を担当するキューの種類と、キューの処理を担当する変換部120が、予め設定されている。このような処理の分担設定は、データ変換システム100のシステム稼働中に変わることは無い。

【0067】

50

(キュー数の増加による不都合)

ここで、一例として、図 11 に、メッセージキュー 112 の OCR のキュー、及び、OCR 处理を行う変換部 (OCR 変換部 120 - 1 ~ 120 - n) を示す。上述のように、OCR 处理は、Linux (登録商標) 系の第 1、第 2、第 7 及び第 8 の、計 4 つの変換処理プロセスで行うように予め設定されている。このため、メッセージキュー 112 の 4 つの OCR キュー (OCR - 1 ~ OCR - 4) は、すぐに処理を開始できる。しかし、OCR - 5 の OCR キューは、4 つの変換処理プロセスが全て処理を実行中であるため、処理を待たされることとなる。4 つの変換処理プロセスでの処理が長引くことで、所定時間が経過すると、OCR - 5 の OCR キューは処理されることなく、タイムアウトエラーとして返却される不都合を生ずる。

10

【 0068 】

(変換部の自動増加構成)

本実施形態のデータ変換システム 100 は、今後、ユーザ数が増えることで、処理が待機状態となるキュー及びタイムアウトとなるキューの発生を防止するために、自動的に変換部 120 の数を増やす構成を備えている。

【 0069 】

すなわち、本実施形態のデータ変換システム 100 は、図 12 に示すように、いわゆる CaaS (Containers as a Service) のコンテナ 40 に変換部数調整モジュール 41 を備えている。また、コンテナ 40 には、ドッカー (Docker (登録商標)) イメージの CaaS 変換部 42 が設けられている。変換部数調整モジュール 41 は、必要となった際にタスクとしてドッカーイメージを起動して一つ又は複数の CaaS 変換部 42 (仮想変換部の一例) を形成する (ドッカーによるコンテナ型仮想化)。

20

【 0070 】

なお、コンテナ 40 は、オペレーションシステム (OS) 上に、仮想的に提供されるユーザ空間である。また、ユーザ空間とは、ユーザがアプリケーションを実行するための、ひと揃いのリソースが提供される空間である。

【 0071 】

例えば、4 つ等の既存の OCR 変換部 120 - 1 ~ 120 - n で 4 つの OCR キュー (OCR - 1 ~ OCR - 4) の処理を開始したが、キュー サーバ装置 32 に 11 個の OCR キューが蓄積されたとする。変換部数調整モジュール 41 は、キュー サーバ装置 32 に蓄積される OCR キューの数を定期的に認識し、OCR キューが例えば 5 個等の所定の閾値以上となったか否かを判別する。そして、OCR キューが所定の閾値以上となった際に、変換部数調整モジュール 41 は、既存の OCR 変換部 120 - 1 ~ 120 - n とは別に、OCR キューの処理を行う変換部である CaaS 変換部 42 を起動する。そして、この CaaS 変換部 42 に OCR キューの処理を実行させる。

30

【 0072 】

CaaS 変換部 42 は、例えば AWS (登録商標) (Amazon Web Service) で提供されるドッカーコンテナという技術を使って構成され、以下の特徴を有している。

【 0073 】

1. 既存の変換部 120 が常に存在し、キュー サーバ装置 32 を所定時間毎にポーリングし、キュー サーバ装置 32 にキューが存在していた場合に、そのキューを処理する。これに対し、CaaS 変換部 42 は、起動されると、例えば 1 度だけキュー サーバ装置 32 を見に行き、キュー サーバ装置 32 に処理するキューが存在する場合に、このキューのジョブを処理する。そして、CaaS 変換部 42 は、ジョブの処理が完了するとプロセスを停止して消える。

40

【 0074 】

2. 既存の変換部 120 のサーバ装置を形成する場合、オペレーションシステムの他、様々なモジュール及びソフトウェアをインストールしデプロイをする必要がある。換言すると、変換部 120 が存在しないサーバ装置に対して、変換部 120 を設ける作業が必要となる。このため、既存の変換部 120 は、増設に例えば 15 分 ~ 30 分の時間を要する

50

。これに対し、CaaS変換部42の場合、CaaS変換部42も含めてイメージ化されており、かつ、短時間で立ち上げ可能なコンテナで実現できるため、例えば2~3分で増設することができる。従って、既存の変換部120に追加して、変換部120のサーバ装置を増設するよりも、効率よく変換部120を増設することができる。

【0075】

3. 既存の変換部120で処理困難な場合のみ、CaaS変換部42を動的に増やして処理困難なキューを処理することで、効率よくキューの処理を行うことができ、キューの処理の待ち時間が長時間となり、又は、タイムアウトエラーとなる不都合を低減できる。

【0076】

(CaaS変換部の起動数)

10

変換部数調整モジュール41は、上述の閾値を超えたキューの数に応じた所定数のCaaS変換部42を起動する。キュー閾値と起動するCaaS変換部42の数は、以下の表2に示す設定ファイルに起動条件の一つとして記憶されている。

【0077】

【表2】

(表2)

機能(キュー)	キュー閾値	起動Worker数
OCR	5	5
バーコード	11	3
...		
PDF	15	1

20

【0078】

一例ではあるが、この表2に示すように、例えばOCR処理のように、処理に時間を要し、また、リソースを多く使う処理は、キュー閾値が小さく、また、起動するCaaS変換部42の数が多く設定されている。反対に、PDF変換処理のように短時間で処理が完了し、リソースを使わない処理に対しては、キュー閾値が大きく、また、起動するCaaS変換部42の数も少なく設定されている。

30

【0079】

変換部数調整モジュール41は、最初にこの設定ファイルの読み込みを行い、設定ファイルに記載されているキュー閾値に対応する数のCaaS変換部42を起動する。例えば、OCRキューの場合、キュー閾値が「5個」で、起動するCaaS変換部42の数が「5個」となっている。このため、変換部数調整モジュール41は、キューサーバ装置32に5個以上のOCRキューが蓄積された際に、5個のCaaS変換部42を起動する。

30

【0080】

(CaaS変換部の起動動作)

40

図13は、変換部数調整モジュール41によるCaaS変換部42の起動動作の流れを示すフローチャートである。この図13に示すように、変換部数調整モジュール41は、まず、表2に示した設定ファイルに記載されているOCR処理に対するキュー閾値及び起動するCaaS変換部42の数を認識する(ステップS1)。

【0081】

次に、変換部数調整モジュール41は、キューサーバ装置32に対して、設定ファイルに記載されている閾値以上の数のOCRキューが蓄積されているか否かを判別する(ステップS2)。

【0082】

キューサーバ装置32に対して、設定ファイルに記載されている閾値以上の数のOCRキューが蓄積されている場合(ステップS2: Yes)、変換部数調整モジュール41は

50

、設定ファイルに記載されている数の C a a S 変換部 4 2 を起動する（ステップ S 3）。表 2 の例の場合は、5 つの C a a S 変換部 4 2 が起動される。これにより、起動された C a a S 変換部 4 2 により、処理が停滯していた O C R キューの処理が行われる。

【 0 0 8 3 】

これに対して、設定ファイルに記載されている閾値以上の数の O C R キューが、キュー サーバ装置 3 2 に蓄積されていない場合（ステップ S 2 : N o ）、変換部数調整モジュール 4 1 は、表 2 に示した設定ファイルに記載されているバーコード処理に対するキュー閾値及び起動する C a a S 変換部 4 2 の数を認識する（ステップ S 4 ）。

【 0 0 8 4 】

次に、変換部数調整モジュール 4 1 は、キュー サーバ装置 3 2 に対して、設定ファイルに記載されている閾値以上の数のバーコードキューが蓄積されているか否かを判別する（ステップ S 5 ）。

【 0 0 8 5 】

キュー サーバ装置 3 2 に対して、設定ファイルに記載されている閾値以上の数のバーコードキューが蓄積されている場合（ステップ S 5 : Y e s ）、変換部数調整モジュール 4 1 は、設定ファイルに記載されている数の C a a S 変換部 4 2 を起動する（ステップ S 6 ）。表 2 の例の場合は、3 つの C a a S 変換部 4 2 が起動される。これにより、起動された C a a S 変換部 4 2 により、処理が停滯していたバーコードキューの処理が行われる。

【 0 0 8 6 】

これに対して、設定ファイルに記載されている閾値以上の数のバーコードキューが、キュー サーバ装置 3 2 に蓄積されていない場合（ステップ S 5 : N o ）、変換部数調整モジュール 4 1 は、表 2 に示した設定ファイルに記載されている P D F 変換処理に対するキュー閾値及び起動する C a a S 変換部 4 2 の数を認識する（ステップ S 7 ）。

【 0 0 8 7 】

次に、変換部数調整モジュール 4 1 は、キュー サーバ装置 3 2 に対して、設定ファイルに記載されている閾値以上の数の P D F キューが蓄積されているか否かを判別する（ステップ S 8 ）。

【 0 0 8 8 】

キュー サーバ装置 3 2 に対して、設定ファイルに記載されている閾値以上の数の P D F キューが蓄積されている場合（ステップ S 8 : Y e s ）、変換部数調整モジュール 4 1 は、設定ファイルに記載されている数の C a a S 変換部 4 2 を起動する（ステップ S 9 ）。表 2 の例の場合は、1 つの C a a S 変換部 4 2 が起動される。これにより、起動された C a a S 変換部 4 2 により、処理が停滯していた P D F キューの処理が行われる。

【 0 0 8 9 】

これに対して、設定ファイルに記載されている閾値以上の数のバーコードキューが、キュー サーバ装置 3 2 に蓄積されていない場合（ステップ S 8 : N o ）、変換部数調整モジュール 4 1 は、処理をステップ S 1 に戻し、上述の各処理を繰り返し行う。

【 0 0 9 0 】

このように変換部数調整モジュール 4 1 は、O C R 処理 ハーコード処理 . . . P D F 変換処理の処理毎に、キュー サーバ装置 3 2 に蓄積されているキューの数を順次検出し、閾値以上の数のキューが蓄積されていた場合に、予め設定されている数の C a a S 変換部 4 2 を起動する。これにより、停滯していたキューの処理を順次行うことができ、処理の停滯を解消することができる。

【 0 0 9 1 】

（ 設定ファイルの書き換え処理 ）

次に、表 2 に示したキュー閾値、及び、起動する C a a S 変換部 4 2 の数の設定の仕方を説明する。

【 0 0 9 2 】

過去の実績をベースに、処理の重さ・リソースの使われ具合を調べ、キュー閾値と起動する C a a S 変換部 4 2 の数等を起動条件として予め設定ファイルに記載しておく。そし

10

20

30

40

50

て、変換部数調整モジュール41が、この設定ファイルで示される数のCaaS変換部42を起動してもよい。

【0093】

ここで、例えば急にOCRキューが100件投入されたとする。しかし、起動するCaaS変換部42の数は、設定ファイルに記載されている起動するCaaS変換部の数に制限される。このため、100件のOCRキューを処理可能とするだけの数のCaaS変換部42を起動することが困難となり、キューの処理待ちの時間が長時間となり、また、タイムアウトエラーが発生する。

【0094】

このため、本実施形態のデータ変換システム100は、自動的に起動条件の一つであるキュー閾値と起動するCaaS変換部42の数を変更して、投入又は蓄積されたキューの数に対応する数のCaaS変換部42を起動可能としている（設定ファイルの書き換え処理）。

10

【0095】

図14は、設定ファイルの書き換え処理を含めたかたちの、CaaS変換部42の起動動作の流れを示すフローチャートである。この図14のフローチャートにおけるステップS1～ステップS9の各処理は、図13のフローチャートのステップS1～ステップS9の各処理と同じ処理である。図14のフローチャートにおいて、ステップS3で、閾値以上の数のOCRキューがキューサーバ装置32に蓄積されていることで、CaaS変換部42を起動させると、処理がステップS11に進む。

20

【0096】

ステップS11では、変換部数調整モジュール41が、OCR処理の過去の実績調査を行う。この実績調査の詳細な動作は、図16のフローチャートを用いて後述する。次に、変換部数調整モジュール41は、OCR処理の過去の実績調査結果に基づいて、表2に示した設定ファイルのOCR処理に対するキュー閾値、及び、起動するCaaS変換部42の数を書き換え処理する（ステップS12：起動条件の書き換え処理）。

30

【0097】

この後、変換部数調整モジュール41は、バーコード処理の過去の実績調査結果に基づいて（ステップS13）、表2に示した設定ファイルのバーコード処理に対するキュー閾値、及び、起動するCaaS変換部42の数を書き換え処理する（ステップS14）。さらに、変換部数調整モジュール41は、PDF変換処理の過去の実績調査結果に基づいて（ステップS15）、表2に示した設定ファイルのPDF変換処理に対するキュー閾値、及び、起動するCaaS変換部42の数を書き換え処理する（ステップS16）。

【0098】

このように、変換部数調整モジュール41は、過去の実績調査を処理毎に順次行い、実績調査結果に対応する値に、設定ファイルのキュー閾値、及び、起動するCaaS変換部42の数を書き換え処理する。これにより、投入又は蓄積されたキューの数に対応する数のCaaS変換部42を、動的に起動可能とすることができる、キューの処理待ちの時間が長時間となる不都合、及び、タイムアウトエラーが発生する不都合を防止できる。

【0099】

40

（設定ファイルの書き換え処理の他の例）

このような設定ファイルの書き換え処理は、図14のフローチャートを用いて説明したように、OCR又はバーコード等の処理毎に順次行ってもよいが、図15のフローチャートに示すように、一括して行ってもよい。この場合、図15のフローチャートに示すように、変換部数調整モジュール41は、ステップS1～ステップS9の処理が終了すると、ステップS21に進める。ステップS21では、変換部数調整モジュール41が、OCR、バーコード等の各処理に対する過去の実績調査を、一括して行う。そして、変換部数調整モジュール41は、各処理の実績調査結果に基づいて、各処理のキュー閾値、及び、起動するCaaS変換部42の数を書き換え処理する（ステップS22）。

【0100】

50

これにより、上述と同様に、投入又は蓄積されたキューの数に対応する数の C a a S 変換部 4 2 を、動的に起動可能とすることができ、キューの処理待ちの時間が長時間となる不都合、及び、タイムアウトエラーが発生する不都合を防止できる。

【0101】

(過去の実績調査)

次に、図 14 のフローチャートのステップ S 1 1、ステップ S 1 3、ステップ S 1 5、及び、図 15 のフローチャートのステップ S 2 1 の処理となる過去の実績調査の説明をする。図 16 は、過去の実績調査の動作の流れを示すフローチャートである。

【0102】

まず、ステップ S 3 1 では、変換部数調整モジュール 4 1 が、O C R、バーコード等の各処理のジョブ情報を収集する。具体的には、図 8 に示したジョブ情報管理 D B 1 0 8 内のジョブ情報テーブル 3 7 には、表 1 に示したように、ジョブタイプ、投入日時、完了日時、「結果」等の各種の情報が記憶されている。変換部数調整モジュール 4 1 は、このジョブ情報テーブル 3 7 から、例えば O C R 処理に関する全ての情報を取得する。なお、投入日時で指定することで、過去 1 ヶ月等の期間を指定して情報を取得してもよい。

10

【0103】

次に、ジョブ情報テーブル 3 7 から取得した「結果」の中には、以下の表 3 に示すよう に、総ファイルサイズ、ファイル数、及び、総ページ数の各情報が記憶されている。

【0104】

【表 3】

20

(表3)

総ファイルサイズ	ファイル数	総ページ数
10,245,327	3	37

【0105】

変換部数調整モジュール 4 1 は、この表 3 に示した総ファイルサイズ、ファイル数、及び、総ページ数の各情報に基づいて、O C R 処理等の処理毎に、処理能力を示す平均秒速を算出する。つまり、変換部数調整モジュール 4 1 は、以下の演算を行い、全ジョブで平均化する。

30

【0106】

ファイルサイズ秒速 = 総ファイルサイズ ÷ (完了日時 - 投入日時) (秒)

ファイル数秒速 = ファイル数 ÷ (完了日時 - 投入日時) (秒)

ページ数秒速 = 総ページ数 ÷ (完了日時 - 投入日時) (秒)

【0107】

変換部数調整モジュール 4 1 は、このように算出した各秒速が一定値を超えた場合に、キュー閾値を下げ、起動する C a a S 変換部 4 2 の数を増加させるように、設定ファイルの書き換えを行う。

40

【0108】

一例をあげて具体的に説明すると、変換部数調整モジュール 4 1 は、例えばファイルサイズ秒速の場合、以下のように算出する。

【0109】

キュー閾値 = (ファイルサイズ秒数 ÷ 1 M B)

起動する C a a S 変換部 4 2 の数 = 5 - (ファイルサイズ秒数 ÷ 1 M B)

【0110】

例えば、ファイルサイズ秒速が 5 M B / 秒の場合、処理能力が高いため、変換部数調整モジュール 4 1 は、キュー閾値を「5」、起動する C a a S 変換部 4 2 の数を「1」とする。また、ファイルサイズ秒速が 0.1 M B / 秒の場合、処理能力が低いため、変換部数調整モジュール 4 1 は、キュー閾値を「1」、起動する C a a S 変換部 4 2 の数を「10」

50

」とする。

【0111】

なお、変換部数調整モジュール41は、小数点以下は切り上げて計算し、「1」より小さくなった場合は「1」とする。また、上述の「1MB」及び起動するCaaS変換部42の数の初期値である「5」の数値は、あくまでも一例であり、上述の過去の実績等に基づいて任意の数値を設定すればよい。

【0112】

また、変換部数調整モジュール41は、ファイル数秒速については、下記の数式に基づいて算出する。

【0113】

$$\text{キュー閾値} = (\text{ファイル数秒数} \div 1)$$

$$\text{起動するCaaS変換部42の数} = 5 - (\text{ファイル数秒数} \div 1)$$

【0114】

また、変換部数調整モジュール41は、ページ数秒速については、下記の数式に基づいて算出する。

【0115】

$$\text{キュー閾値} = (\text{ページ数秒数} \div 100)$$

$$\text{起動するCaaS変換部42の数} = 5 - (\text{ページ数秒数} \div 100)$$

【0116】

変換部数調整モジュール41は、このように算出したキュー閾値及び起動するCaaS変換部42の数を、設定ファイルのOCR等の対応する処理の項目に書き込む。これにより、次回からは新しい設定値に基づいて変換部数調整モジュール41を動作させることができる。

【0117】

なお、以上の説明は、図14を用いて説明したようにOCR、バーコード等の処理毎にファイルサイズ秒速及びファイル数秒速等を調査（算出）して、処理毎に設定ファイルの設定値を更新する例である。これに対して、図15に示した例の場合、変換部数調整モジュール41は、ステップS21で、OCR、バーコード等の全処理のファイルサイズ秒速及びファイル数秒速等を一括して調査（算出）し、設定ファイルの全処理の設定値を一括して更新処理する。

【0118】

（既存の変換部の状態判断に基づくCaaS変換部の起動処理）

また、変換部数調整モジュール41は、図17のステップS38に示すように既存の変換部120の状態を判断し、この状態の判断結果に基づいて、CaaS変換部42を起動してもよい。なお、既存の変換部120の状態の判断処理は、上述の実績調査と組み合わせて行ってもよい。

【0119】

すなわち、図18は、ステップS38の詳細な動作の流れを示すフローチャートである。この図18のフローチャートに示すように、変換部数調整モジュール41は、まず、図8に示したジョブ情報テーブル37を参照し、ステータスが「データ変換実行中（doing）」になっているジョブ情報を検出する（ステップS41）。また、変換部数調整モジュール41は、検出した「データ変換実行中（doing）」のジョブの投入日時が、現在日時から所定時間以上経過しているか否かを判別する（ステップS42）。「データ変換実行中（doing）」のジョブの投入日時が、現在日時から所定時間以上経過していた場合（ステップS42：Yes）、そのジョブに対応する処理のCaaS変換部42を起動し、停滞しているジョブを再実行させる（ステップS43）。

【0120】

これにより、既存の変換部120で処理することができず、投入から一定時間が経過しているジョブを、CaaS変換部42で処理することができる。

【0121】

10

20

30

40

50

(ステータスがエラーのジョブの再実行動作)

また、変換部数調整モジュール41は、既存の変換部120で処理がエラーとなったジョブを、CaaS変換部42で再実行させて処理を完了させる。この場合、変換部数調整モジュール41は、図19のフローチャートに示すように、図8に示したジョブ情報テーブル37を参照し、ステータスが「エラー(error)」になっているジョブ情報を検出する(ステップS45)。また、変換部数調整モジュール41は、検出した「エラー(error)」のジョブの「結果」をジョブ情報テーブル37から参照する。この「結果」には、例えば下記の表4に示すようにジョブが「エラー(error)」となった理由が記録されている。

【0122】

【表4】

(表4)

総ファイルサイズ	ファイル数	総ページ数	エラー時の理由
10,245,327	3	37	Memory_Empty

【0123】

この理由が、例えば「メモリの残量不足」、「メモリリーク」、「HDDの残量不足」等の所定の理由の場合(ステップS46: Yes)、そのジョブに対応する処理のCaaS変換部42を起動し、エラーとなったジョブを再実行させる(ステップS47)。

【0124】

これにより、既存の変換部120で処理がエラーとなったジョブを、CaaS変換部42で処理することができる。

【0125】

(CaaS変換部のスケジュール起動動作)

次に、既存の変換部120での処理が停滞することが、過去の実績から分かっている場合、変換部数調整モジュール41は、この日時に合わせてCaaS変換部42を定期的に起動してもよい。このような、言わばスケジュールに沿ったCaaS変換部42の定期的な起動処理は、図14又は図15を用いて説明した設定ファイルの書き換え処理等と組み合わせて実行してもよい。

【0126】

すなわち、例えば毎週月曜日の9:00~10:00の間は、ジョブが大量に投入されることが過去の実績から分かっていたとする。この場合、その時間帯に予めCaaS変換部42を必要数起動しておけば、よりスムーズに大量のジョブを処理できる。このため、図20のフローチャートのステップS50に示すように、CaaS変換部42のスケジュール起動処理を行うステップを設ける。そして、スケジュールで設定された日時になった際に、変換部数調整モジュール41が、スケジュールで設定されている数のCaaS変換部42を起動する。

【0127】

具体的には、下記の表5に示すように起動日時、起動する処理、及び、起動するCaaS変換部42の数を予め設定ファイルに設定する。これは、ジョブが大量に投入される日時又はジョブが停滞する日時等を過去の実績に基づいて調べ、設定ファイルに設定する。

【0128】

10

20

30

40

50

【表5】

(表5)

起動日時	起動する処理	起動Worker数
平日 09:00~10:00	OCR	5
毎週金曜日 17:00~18:00	image2pdf	3

【0129】

この表5の例は、平日の9:00~10:00の間にOCR処理のCaas変換部42を5つ起動する設定を示している。また、表5の例は、毎週金曜日の17:00~18:00の間に画像情報をPDFデータに変換処理するCaas変換部42を、3つ起動する設定を示している。

10

【0130】

図21は、ステップS50のスケジュール起動処理の詳細な動作の流れを示すフローチャートである。この図21において、変換部数調整モジュール41は、まず、ステップS51で表5に例示したような設定ファイルを読み込む。そして、変換部数調整モジュール41は、現在の日時が、設定ファイルに設定されているCaas変換部42の起動日時から例えば10分以内であるか否かを判別する(ステップS52)。

20

【0131】

変換部数調整モジュール41は、現在の日時が、設定ファイルに設定されているCaas変換部42の起動日時の例えば10分前以内であると判別した際に(ステップS52: Yes)、ステップS53に処理を進める。そして、変換部数調整モジュール41は、設定ファイルに設定されている処理に対応するCaas変換部42を、設定ファイルに設定されている数分、起動する(ステップS53)。

30

【0132】

(1ジョブモード)

ここで、変換部数調整モジュール41は、起動したCaas変換部42に処理するキューが存在しない場合、そのCaas変換部42は、すぐ終了させる。しかし、スケジュール起動処理の場合は、過去の実績から設定された数のCaas変換部42を起動している。そして、現在、処理するキューが存在しない場合でも、設定された時間内にキューが投入される可能性が高い。このため、変換部数調整モジュール41は、スケジュール起動処理で起動したCaas変換部42は、1つのキューを処理し終わるまではCaas変換部42を終了させない(1ジョブモード)。

30

【0133】

この1ジョブモードは、例えば平日の9:00~10:00の間等のように、設定ファイルで設定された時間帯で実行されるモードである。変換部数調整モジュール41は、1ジョブモードで動作している間、キューサーバ装置32に蓄積されるキューを所定時間毎にポーリングし、投入されたキューの処理をCaas変換部42(又は既存の変換部120)で処理する。そして、変換部数調整モジュール41は、10:00を過ぎた際に、1ジョブモードを終了する。1ジョブモードを終了する際には、変換部数調整モジュール41は、キューを処理していないCaas変換部42も終了させる。

40

【0134】

これにより、ジョブが大量に投入される日時又はジョブの処理が停滞する日時にCaas変換部42を必要数分起動して、ジョブの処理をスムーズに完了させることができる。

【0135】

(過去の実績の自動調査)

次に、Caas変換部42を起動させるスケジュールを設定するための実績調査であるが、これは、図22のフローチャートのステップS54~ステップS56に示すように、変換部数調整モジュール41が自動で行ってよい。

50

【0136】

この場合、図22のフローチャートに示すように、上述のステップS52で、現在の日時が、設定ファイルに設定されているCaaS変換部42の起動日時から例えば10分以内ではないと判別された際に(ステップS52:No)、処理をステップS54に進める。ステップS54では、変換部数調整モジュール41が、ジョブ情報テーブル37を参照することで、OCR処理、バーコード処理等の処理毎に、例えば1時間単位又は10分単位等の所定時間単位で何件のジョブが処理されたかを、投入日時とジョブ数から集計する(ステップS54)。

【0137】

次に、変換部数調整モジュール41は、表2に例示した設定ファイルを参照する。そして、変換部数調整モジュール41は、例えばOCR処理であれば5ジョブ以上、バーコード処理であれば11ジョブ以上又はPDF変換処理であれば15ジョブ以上等のように、キュー閾値以上のキューが処理されている時間帯を検出する(ステップS55)。

10

【0138】

次に、変換部数調整モジュール41は、設定ファイルに対して、検出された全ての時間帯、CaaS変換部42をスケジュール起動させる日時、OCR処理等の処理の種類、及び、起動するCaaS変換部42の数を設定する(ステップS56)。

【0139】

これにより、次回のスケジュール起動の際には、最新の起動日時/処理の種類/起動するCaaS変換部42の数が反映されて、スケジュールに従って、CaaS変換部42を起動させることができる。すなわち、過去の処理実績に応じて、CaaS変換部42を起動させるスケジュールを自動的に更新して、キューの処理を行うことができる。

20

【0140】

(実施の形態の効果)

以上の説明から明らかなように、本実施形態のデータ変換システム100は、CaaSのコンテナ40に、ドッカーアイメージとしてCaaS変換部42を用意しておく。そして、必要となった際に、タスクとしてドッカーアイメージを起動することで、既存の変換部120と同じ処理を行うCaaS変換部42を増設して、ジョブの処理を行う。これにより、キューサーバ装置32に蓄積されるキューの数に応じて、CaaS変換部42を増設し、蓄積されたキューを処理できる。このため、キューの処理に長時間を要し、又は、タイムアウトエラーとなる不都合を防止できる。従って、ジョブが停滞して顧客に迷惑を及ぼす不都合を防止できる。

30

【0141】

最後に、上述の実施の形態は、一例として提示したものであり、本発明の範囲を限定することは意図していない。この新規な実施の形態は、その他の様々な形態で実施されることが可能であり、発明の要旨を逸脱しない範囲で、種々の省略、置き換え、変更を行うことも可能である。また、実施の形態及び各実施の形態の変形は、発明の範囲や要旨に含まれると共に、特許請求の範囲に記載された発明とその均等の範囲に含まれるものである。

【符号の説明】

【0142】

40

10 クラウド

11 サーバ

12 PC(パーソナルコンピュータ)

13 MFP(複合機)

30 アップサーバ装置

31 フロントサーバ装置

32 キューサーバ装置

33 第1の印刷ジョブ管理サーバ装置

34 第2の印刷ジョブ管理サーバ装置

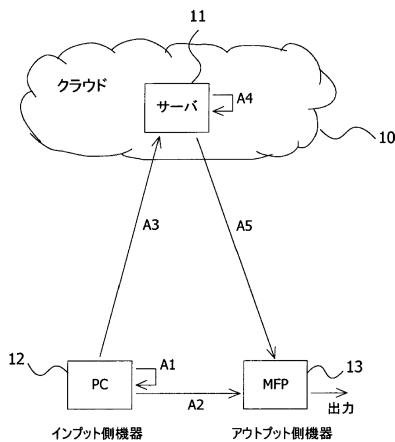
35 データベースサーバ装置(DBサーバ装置)

50

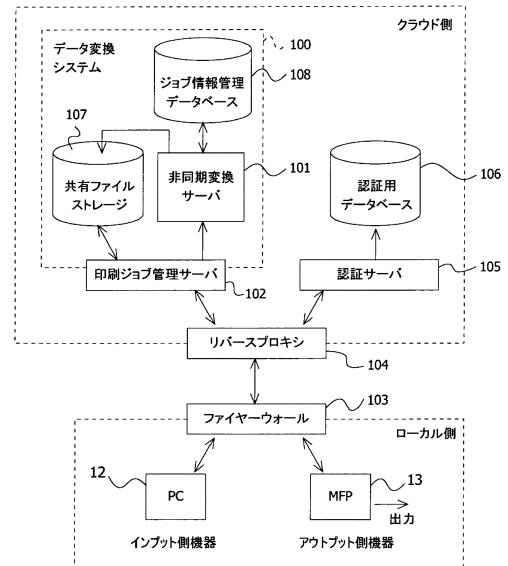
3 6	ファイルサーバ装置	
3 7	ジョブ情報テーブル	
4 0	コンテナ	
4 1	変換部数調整モジュール	
4 2	C a a S 変換部	
1 0 0	データ変換システム	
1 0 1	非同期変換サーバ	
1 0 2	印刷ジョブ管理サーバ	
1 0 3	ファイヤーウォール	
1 0 4	リバースプロキシ	10
1 0 5	認証サーバ	
1 0 6	認証用データベース	
1 0 7	共有ファイルストレージ	
1 0 8	ジョブ情報管理データベース	
1 1 1	データ変換要求管理部	
1 1 2	メッセージキュー	
1 2 0	変換部	
【先行技術文献】		
【特許文献】		
【0 1 4 3】		
【文献】特開 2 0 0 6 - 0 8 1 0 8 1 号公報		
【文献】特開 2 0 0 9 - 0 7 5 6 8 9 号公報		

【図面】

【図 1】



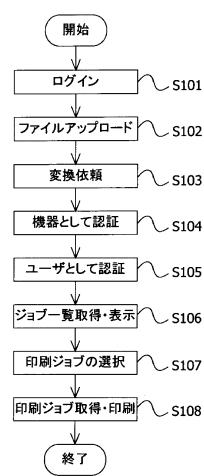
【図 2】



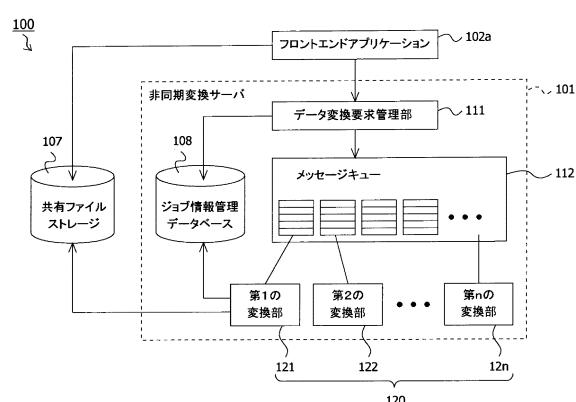
10

20

【図 3】



【図 4】

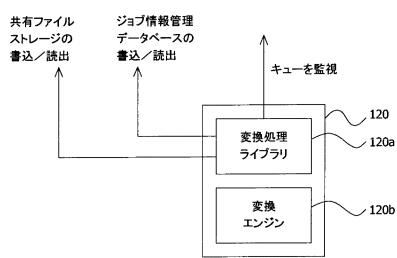


30

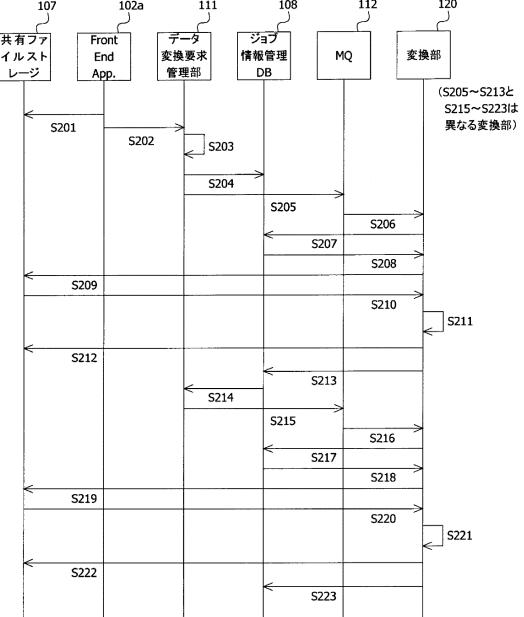
40

50

【図5】



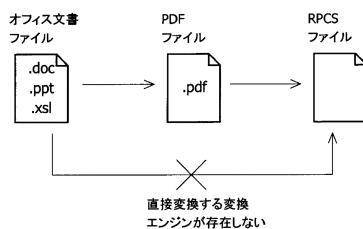
【図6】



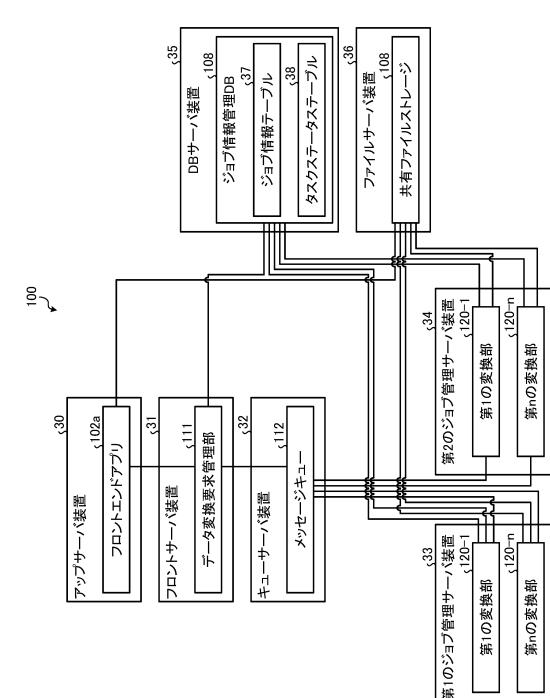
10

20

【図7】



【図8】

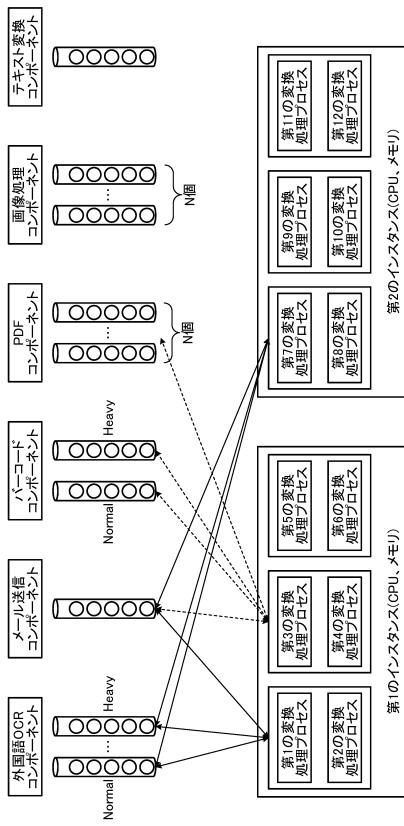


30

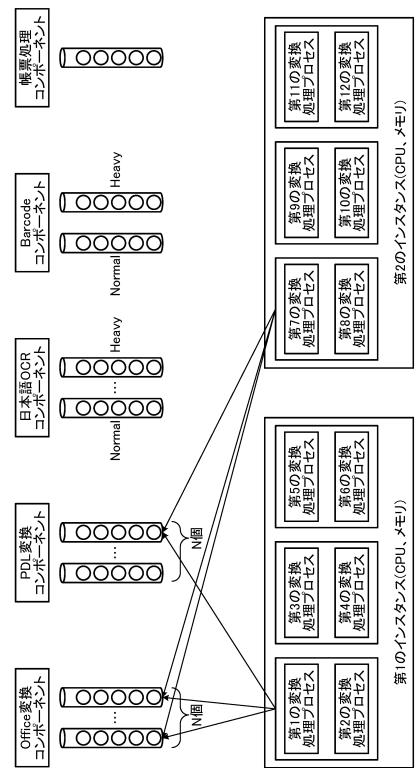
40

50

【図 9】



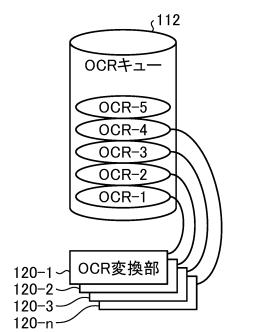
【図 10】



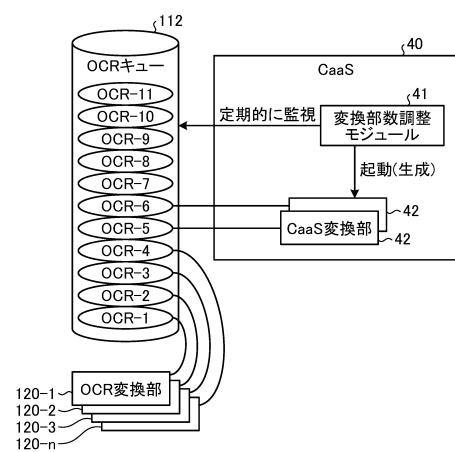
10

20

【図 11】



【図 12】

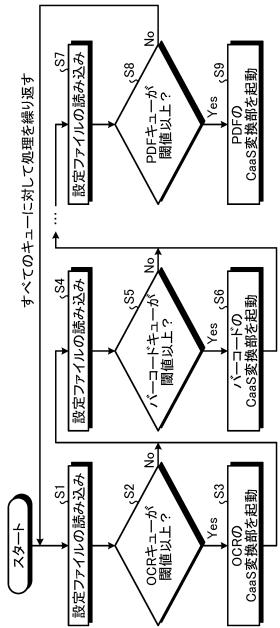


30

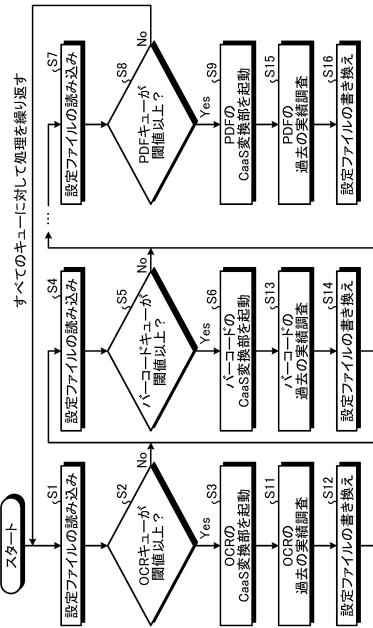
40

50

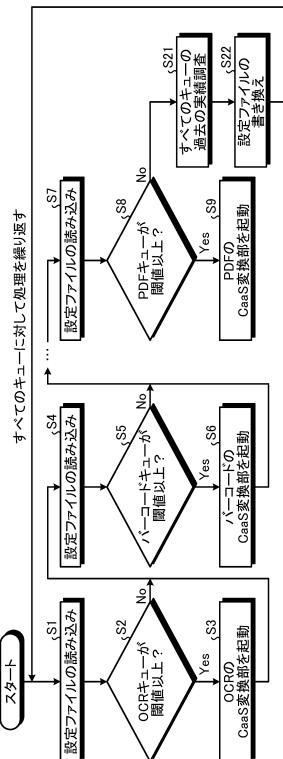
【図13】



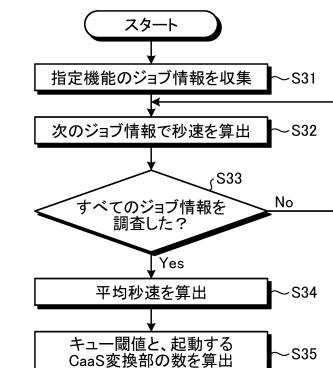
【図14】



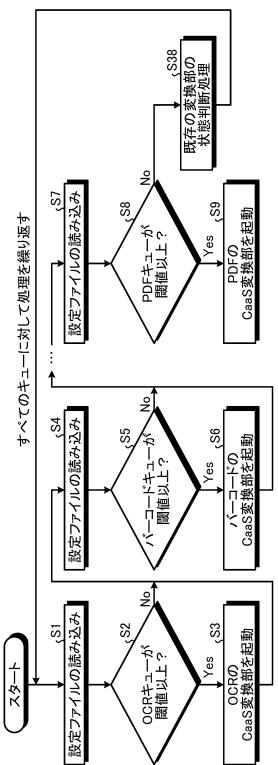
【図15】



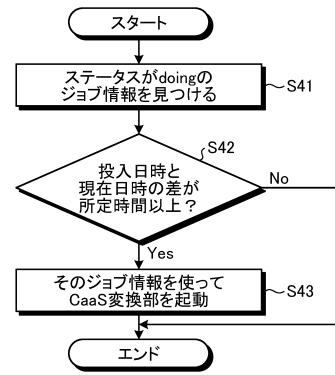
【図16】



【図17】



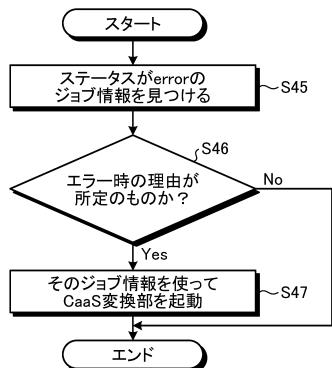
【図18】



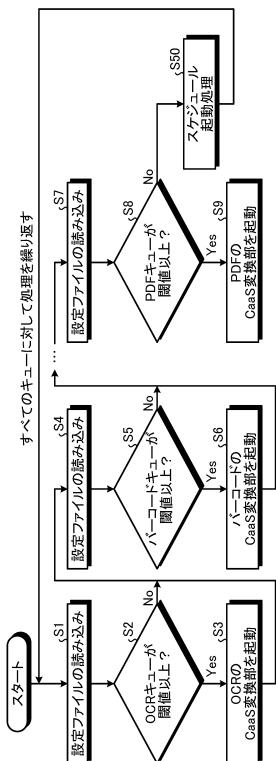
10

20

【図19】



【図20】

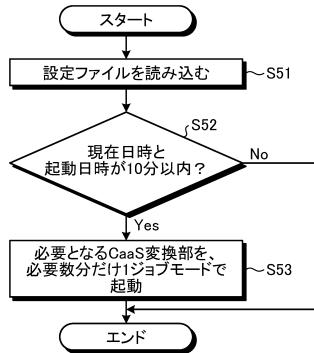


30

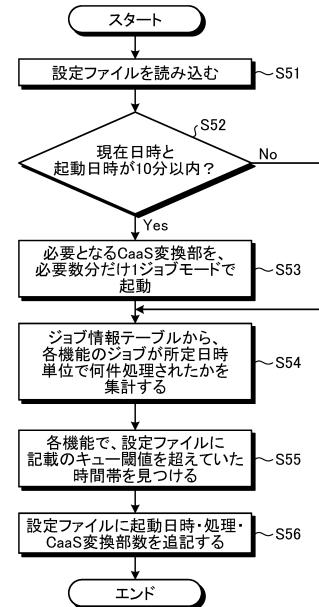
40

50

【図 2 1】



【図 2 2】



10

20

30

40

50

フロントページの続き**(51)国際特許分類**

F I
G 0 6 F 3/12 3 8 7
H 0 4 N 1/00 1 2 7 A

(56)参考文献

特開2013-178748 (JP, A)
特開2017-146791 (JP, A)
特開2000-341504 (JP, A)
特開2015-125740 (JP, A)
特開2015-049550 (JP, A)

(58)調査した分野 (Int.Cl., DB名)

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