

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5451331号
(P5451331)

(45) 発行日 平成26年3月26日(2014.3.26)

(24) 登録日 平成26年1月10日(2014.1.10)

(51) Int.Cl. F1
G06T 1/20 (2006.01) G06T 1/20 A

請求項の数 8 (全 17 頁)

| | | | |
|-----------|-------------------------------|-----------|-------------------|
| (21) 出願番号 | 特願2009-263344 (P2009-263344) | (73) 特許権者 | 000001007 |
| (22) 出願日 | 平成21年11月18日 (2009.11.18) | | キヤノン株式会社 |
| (65) 公開番号 | 特開2011-108047 (P2011-108047A) | | 東京都大田区下丸子3丁目30番2号 |
| (43) 公開日 | 平成23年6月2日 (2011.6.2) | (74) 代理人 | 100076428 |
| 審査請求日 | 平成24年11月13日 (2012.11.13) | | 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】 画像処理装置及びその制御方法、並びにプログラム

(57) 【特許請求の範囲】

【請求項 1】

複数の外部装置と連携して画像データに対する画像処理を実行する画像処理装置であって、

前記画像データに対して実行する連携処理の手順が予め定義された連携処理フローを、該連携処理フローを管理する外部の記憶装置から取得する取得手段と、

取得した前記連携処理フローに含まれる複数の機能について、前記複数の外部装置のそれぞれが実行可能か否かを確認する確認手段と、

前記複数の機能のそれぞれについて、実行することを予め定義された外部装置が実行不可能であると確認されると、実行可能であると確認された他の外部装置が代わりに当該機能を実行するように、前記連携処理フローの定義を変更する変更手段と、

前記変更手段によって変更された連携処理フローに従った連携処理を、該連携処理フローに定義された外部装置に実行させる実行制御手段とを備え、

前記変更手段は、

前記複数の機能のそれぞれについて、前記複数の外部装置の2つ以上で実行可能であると、前記連携処理フローに含まれる前記複数の機能のうち実行可能な機能の数が最も多い外部装置において当該機能を優先的に実行するように、該連携処理フローの定義を変更することを特徴とする画像処理装置。

【請求項 2】

前記確認手段は、

前記複数の外部装置のうち前記連携処理フローに予め定義されている外部装置について、前記複数の機能が実行可能か否かを確認することを特徴とする請求項 1 に記載の画像処理装置。

【請求項 3】

前記変更手段は、

前記複数の機能のそれぞれについて、実行することを予め定義された外部装置だけでなく、他の外部装置も実行不可能であると確認されると、実行不可能な当該機能を前記連携処理フローから削除する削除手段を備えることを特徴とする請求項 1 又は 2 に記載の画像処理装置。

10

【請求項 4】

前記変更手段は、

前記複数の機能のそれぞれについて、実行することを予め定義された外部装置だけでなく、他の外部装置も実行不可能であると確認されると、前記連携処理フローの実行を禁止する禁止手段を備えることを特徴とする請求項 1 又は 2 に記載の画像処理装置。

【請求項 5】

前記実行制御手段は、

前記画像データと前記変更手段によって変更された連携処理フローとを、前記複数の機能のうち最初の実行対象の機能を実行する外部装置に対して送信することにより、該連携処理フローの定義に従った処理を開始させることを特徴とする請求項 1 乃至 4 の何れか 1

20

【請求項 6】

前記連携処理フローは、原稿から画像を読み取る際の読取条件をさらに含み、

前記画像データを生成するために、前記連携処理フローに含まれる前記読取条件に従って、原稿から画像を読み取る読取手段をさらに備えることを特徴とする請求項 1 乃至 5 の何れか 1 項に記載の画像処理装置。

【請求項 7】

複数の外部装置と連携して画像データに対する画像処理を実行する画像処理装置の制御方法であって、

前記画像データに対して実行する連携処理の手順が予め定義された連携処理フローを、該連携処理フローを管理する外部の記憶装置から取得する取得ステップと、

30

取得した前記連携処理フローに含まれる複数の機能について、前記複数の外部装置のそれぞれが実行可能か否かを確認する確認ステップと、

前記複数の機能のそれぞれについて、実行することを予め定義された外部装置が実行不可能であると確認されると、実行可能であると確認された他の外部装置が代わりに当該機能を実行するように、前記連携処理フローの定義を変更する変更ステップと、

前記変更ステップで変更された連携処理フローに従った連携処理を、該連携処理フローに定義された外部装置に実行させる実行制御ステップと

を含み、

前記変更ステップでは、

40

前記複数の機能のそれぞれについて、前記複数の外部装置の 2 つ以上で実行可能であると、前記連携処理フローに含まれる前記複数の機能のうち実行可能な機能の数が最も多い外部装置において当該機能を優先的に実行するように、該連携処理フローの定義を変更することを特徴とする画像処理装置の制御方法。

【請求項 8】

請求項 7 に記載の画像処理装置の制御方法における各ステップをコンピュータに実行させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

50

本発明は、画像処理装置及びその制御方法、並びにプログラムに関するものである。

【背景技術】

【0002】

近年、スキャナ、ファクシミリ装置、複写機又はこれらを複合した複合機、パーソナルコンピュータ（PC）、メールサーバ等の装置をネットワークで接続し、各装置が有する機能を連携して処理を実行する画像処理システムが知られている。画像処理システムでは、一般的に、連携処理の手順を定義した連携処理フロー（ワークフロー）が、業務手順に合わせて予め作成される。ユーザは、作成された連携処理フローを実行することにより、当該システムの機能を効率的に組み合わせることができる。

【0003】

このような画像処理システムにおいて、ネットワーク上の各装置のサービス（機能）が追加又は変更された場合には、予め作成された連携処理フローを変更することが必要となる。例えば、特許文献1では、そのような場合に、ネットワークを介して取得した各装置のサービス情報に基づいて、ユーザに新たなワークフローを容易に作成させることが可能な手法が提案されている。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特開2004-151893号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、上述の従来技術には、以下のような問題がある。例えば、画像処理システムに含まれる各装置の機能が変更された場合に、当該変更内容に合わせて手作業で連携処理フローを変更する必要があるため、ユーザにとっては煩雑な作業が求められてしまう。また、そのような場合に、ユーザが連携処理フローを変更せずに実行すると、処理が途中で停止してしまい、画像処理システムの処理効率を低下させてしまう。

【0006】

本発明は、上述の問題に鑑みてなされたものであり、連携処理フローを実行する装置の機能に変更が生じた場合に、ユーザに煩雑な作業を求めることなく、当該連携処理フローの定義を変更して実行する画像処理装置及びその制御方法、並びにプログラムを提供することを目的としている。

【課題を解決するための手段】

【0007】

本発明は、例えば、画像処理装置として実現できる。画像処理装置は、複数の外部装置と連携して画像データに対する画像処理を実行する画像処理装置であって、画像データに対して実行する連携処理の手順が予め定義された連携処理フローを、連携処理フローを管理する外部の記憶装置から取得する取得手段と、取得した連携処理フローに含まれる複数の機能について、複数の外部装置のそれぞれが実行可能か否かを確認する確認手段と、複数の機能のそれぞれについて、実行することを予め定義された外部装置が実行不可能であると確認されると、実行可能であると確認された他の外部装置が代わりに当該機能を実行するように、連携処理フローの定義を変更する変更手段と、変更された連携処理フローに従った連携処理を、該連携処理フローに定義された外部装置に実行させる実行制御手段とを備え、変更手段は、複数の機能のそれぞれについて、複数の外部装置の2つ以上で実行可能であると、連携処理フローに含まれる複数の機能のうち実行可能な機能の数が最も多い外部装置において当該機能を優先的に実行するように、連携処理フローの定義を変更することを特徴とする。

【発明の効果】

【0008】

本発明によれば、例えば、連携処理フローを実行する装置の機能に変更が生じた場合に

10

20

30

40

50

、ユーザに煩雑な作業を求めることなく、当該連携処理フローの定義を変更して実行する画像処理装置及びその制御方法、並びにプログラムを提供できる。

【図面の簡単な説明】

【0009】

【図1】第1の実施形態に係る画像処理システム1000の構成例を示す図である。

【図2】第1の実施形態に係る画像処理システム1000におけるスキャナ10のハードウェア構成例を示す図である。

【図3】第1の実施形態に係る画像処理システム1000における各サーバ20、40、60、70のハードウェア構成例を示す図である。

【図4】第1の実施形態に係る画像処理システム1000におけるチケット生成クライアント30のハードウェア構成例を示す図である。 10

【図5】第1の実施形態に係るチケットの一例を示す図である。

【図6】第1の実施形態に係るチケットの一例を示す図である。

【図7】第1の実施形態に係るチケットの一例を示す図である。

【図8】第1の実施形態に係るチケットの一例を示す図である。

【図9】第1の実施形態に係るチケット一覧の一例を示す図である。

【図10】第1の実施形態に係るスキャナ10の操作部105に表示される表示画面の一例を示す図である。

【図11】第1の実施形態に係る各処理サーバが有する機能管理テーブルの一例である。

【図12】第1の実施形態に係るスキャナ10におけるチケットの実行処理の手順を示すフローチャートである。 20

【図13】第1の実施形態に係る各処理サーバにおけるチケットの確認処理の手順を示すフローチャートである。

【図14】第1の実施形態に係る各処理サーバにおけるチケットの実行処理の手順を示すフローチャートである。

【図15】第2の実施形態に係るスキャナ10におけるチケットの変更処理の手順を示すフローチャートである。

【図16】第2の実施形態に係る機能状態テーブルとチケットの一例を示す図である。

【発明を実施するための形態】

【0010】

30

以下に本発明の一実施形態を示す。以下で説明される実施形態は、本発明の上位概念、中位概念及び下位概念など種々の概念を理解するために役立つであろう。また、本発明の技術的範囲は、特許請求の範囲によって確定されるのであって、以下の実施形態によって限定されるわけではない。

【0011】

[第1の実施形態]

<画像処理システム1000の構成>

以下では、図1乃至図14を参照して、本発明における第1の実施形態について説明する。まず、図1を参照して、画像処理システム1000の構成例について説明する。画像処理システム1000は、画像処理装置(スキャナ)10、チケット管理サーバ20、チケット生成クライアント30、文書管理サーバ40、及び処理サーバ60、70を備え、これらはネットワーク(LAN)50を介して接続されている。スキャナ10は、原稿から画像を読み取るスキャナ機能を有する装置であればよく、例えば、MFP(Multiple Function Peripheral)でもよい。チケット管理サーバ20は、データを記憶する機能を有する装置であればよく、記憶装置の他にも、例えば、アクティブ・ディレクトリのようなディレクトリ・サービスを提供可能な装置でもよい。また、処理サーバの数は、図1に示すように2台(処理サーバ60、70)である必要はなく、LAN50上に何台存在してもよい。 40

【0012】

画像処理システム1000において、スキャナ10は、LAN50を介して接続された 50

複数の外部装置（処理サーバ60、70）と連携して画像データに対する画像処理を実行する。具体的には、複数の外部装置の有する機能を連携して実行する連携処理の手順が予め定義された連携処理フロー（以下では、「チケット」と称する。）に従って、当該画像処理が実行される。このチケットは、チケット生成クライアント30で生成され、チケット管理サーバ20に記憶されている。

【0013】

後述するように、チケットは、スキャナ10と連携する複数の外部装置が有する複数の機能及びその処理内容に関する定義を含む。チケットは、例えば、スキャナ10によって生成された画像データに対するファイル形式の変換（例えば、PDF形式への変換）、E-mail送信、ファイル送信、OCR処理等の処理に関する設定を含む。また、チケットは、スキャナ10において原稿から画像を読み取る際の、解像度・サイズ等の読取条件に関する設定を含んでいてもよい。

10

【0014】

（スキャナ10の構成）

次に、図2を参照して、スキャナ10のハードウェア構成例について説明する。図2では、特に、コントローラユニット118の構成例について説明する。コントローラユニット118において、システムバス106上には以下のデバイスが配置される。CPU101は、スキャナ10全体を制御するプロセッサである。NVMEM100は、不揮発性のメモリであり、設定情報等を記憶する。例えば、NVMEM100は、CPU101が動作するためのシステムワークメモリ、プログラムを記録するためのプログラムメモリ、及び画像データを一時記録するための画像メモリとして機能する。FlashROM102は、書き換え可能な不揮発性メモリであり、システムを制御するための各種制御プログラムを記憶する。USBポート110は、USBデバイス111との接続を可能にする。

20

【0015】

操作部I/F104は、操作部105とのインタフェース部である。操作部I/F104は、操作部105に表示する操作画面等の画像データを、操作部105に対して出力し、また、操作部105から入力された情報を、CPU101に伝達する。操作部105は、システムの操作画面を表示するLCD表示部、及び操作キーを有し、LCD表示部上にはタッチパネルシートが貼られている。操作部105は、ユーザによって操作キーが押下された場合の入力情報、及びタッチパネルシートが押下された場合の位置情報を、操作部I/F104を介してCPU101に伝達する。

30

【0016】

LAN I/F107は、LANに接続するためのインタフェースである。スキャナ10は、LAN I/F107を介して、LAN50上の他の装置（PCや各サーバ）に対してデータを送信し、又は他の装置からデータを取得する。また、画像バス（IMAGE BUS）I/F108は、システムバス106と画像バス112とを接続し、データ構造を変換するバスブリッジである。画像バス112は、PCIバスやIEEE1394で構成され、画像データを高速で転送可能である。画像バス112上には以下のデバイスが配置される。

40

【0017】

ラストイメージプロセッサRIP113は、PDLコードのようなベクトルデータをビットマップイメージに展開する。スキャナI/F115は、画像読取部として機能するスキャナ部117とコントローラユニット118とを接続するインタフェース部であり、画像データの変換を行う。スキャナ部117は、原稿から画像を読み取ることにより、画像データを生成する。当該読取方式は、例えば、レーザを用いた方式である。コントローラユニット118は、スキャナI/F115を介して、スキャナ部117との間で画像データやデバイス情報等の入出力を行う。

【0018】

画像処理部116は、入力された画像データに対して、補正、加工、又は編集といった画像処理を実行する。例えば、スキャナ部117において生成された画像データに対して

50

、補正、解像度変換等を行う。さらに、画像処理部 116 は、入力された画像データに対して回転処理、及び、多値画像データに対しては J P E G、2 値画像データに対しては J B I G、M M R、M H 等の圧縮・伸張処理を施すことが可能である。

【 0 0 1 9 】

(サーバ 20、40、60、70 の構成)

次に、図 3 を参照して、チケット管理サーバ 20 のハードウェア構成例について説明する。なお、文書管理サーバ 40 及び処理サーバ 60、70 に関しても同様である。チケット管理サーバ 20 において、C P U 2 1 は、メインバス 25 を介して、R A M 2 2、記憶装置 23、及びネットワーク装置 24 を制御する。R A M 2 2 は、C P U 2 1 に対してワークエリアを提供する。記憶装置 23 は、各種のプログラムや設定を記憶する。ネットワーク装置 24 は、L A N 5 0 を介して接続された他の装置との間で通信を行う。

10

【 0 0 2 0 】

(チケット生成クライアント 30 の構成)

次に、図 4 を参照して、チケット生成クライアント 30 のハードウェア構成例について説明する。チケット生成クライアント 30 において、C P U 3 1 は、メインバス 25 を介して、C P U 3 1、R A M 3 2、記憶装置 33、ユーザ入力装置 34、U I 表示装置 35、及びネットワーク装置 36 を制御する。R A M 3 2 は、C P U 3 1 に対してワークエリアを提供する。記憶装置 33 は、例えば、ハードディスクや N V R A M であり、各種のプログラムや設定を記憶する。ユーザ入力装置 34 は、ユーザがコマンドの入力を行うための入力装置である。U I 表示装置 35 は、画面表示を行う装置である。ネットワーク装置 36 は、L A N 5 0 を介して接続された他の装置との間で通信を行う。

20

【 0 0 2 1 】

ここで、図 5 を参照して、チケット生成クライアント 30 において生成されるチケットについて説明する。本実施形態では、チケットとして X M L 形式のファイルを一例として採り上げるが、構造化データを表現可能なファイル形式であれば何れの形式でもよい。チケット 501 は、チケット生成クライアント 30 において生成され、チケット管理サーバ 20 において保存及び管理される。チケット管理サーバ 20 は、図 5 に示すチケットと同種のチケットを多数管理しており、要求に応じてスキャナ 10 に対して何れかのチケットを提供する。

【 0 0 2 2 】

スキャン設定 502 (S c a n C o n f i g u r a t i o n 要素) には、スキャナ 10 において原稿から画像を読み取る際の読取条件を示す設定情報が定義される。当該設定情報には、例えば、カラーモード (C o l o r M o d e)、解像度 (R e s o l u t i o n)、両面設定 (D u p l e x) 等の読み取りに必要な設定が含まれるが、他の設定が含まれていてもよい。

30

【 0 0 2 3 】

フロー設定 503 (I m a g i n g P r o c e s s 要素) には、スキャナ 10 で生成された画像データに対して複数の外部装置 (処理サーバ 60、70) において実行する画像処理の内容及びその実行順序が定義される。具体的には、フロー設定 503 には、当該画像処理において使用する機能に関する設定 (プロセス設定) が定義される。ここで、プロセス設定は複数定義されてもよく、各プロセス設定には、実行対象となる機能及び処理内容、当該機能を実行する順番、並びに当該機能を実行する外部装置に関する設定等が含まれる。

40

【 0 0 2 4 】

図 5 では、当該設定として、プロセス設定 504、505、506 (P r o c e s s 要素) が定義されている。各プロセス設定 504、505、506 には、実行される順番 (i d)、処理名 (N a m e)、処理サーバの I D (S e r v e r I D) 及び U R L (S e r v e r U R L)、並びに処理に必要な設定情報 (P a r a m 要素) が定義されている。

【 0 0 2 5 】

保存先 507 (D e s t i n a t i o n 要素) には、フロー設定 503 に従って処理さ

50

れた画像データの保存先のサーバのアドレス、フォルダパス、及び通信プロトコルが記載される。また、文書属性508(Document Property要素)には、保存される画像データに関する属性情報が定義され、例えば、文書名、更新日時、作成者に関する情報が含まれる。

【0026】

<スキャナ10におけるチケットの実行処理>

次に、図12を参照して、本実施形態に係るスキャナ10におけるチケットの実行処理の手順について説明する。まず、S101で、スキャナ10のCPU101は、外部のチケット管理サーバ20との通信により、チケットの一覧を要求し、当該一覧を取得する。その後、S102へ移行する。ここで、CPU101は、図9に示すようなチケット一覧901をチケット管理サーバ20から取得し、NVMEM100に格納する。なお、CPU101は、例えば、スキャナ10を使用するユーザのログイン認証結果に基づいて、当該ユーザに関連付けられたチケットの一覧を要求する。その場合、CPU101は、チケット管理サーバ20に対してユーザの識別情報又は認証情報を送信する。

10

【0027】

S102及びS103で、CPU101は、各チケットについて、定義に含まれる複数の機能(プロセス)が、各処理サーバにおいて実行可能か否かを確認する処理を実行する。まず、S102で、CPU101は、チケット一覧901から、確認処理を行っていないチケットを選択し、確認処理を実行する。具体的には、CPU101は、選択したチケットを、当該チケットに予め定義されている処理サーバに対して送信する。これにより、送信先の処理サーバに、確認対象のチケットに定義された各プロセスを実行可能か否かを確認させる。チケットを受信した処理サーバは、当該チケットに定義された全てのプロセスについて確認処理を実行するとともに、その結果を当該チケットに付加する。

20

【0028】

ここで、S102における処理は、スキャナ10が、確認対象のチケットに定義された何れかの処理サーバに対して当該チケットを送信し、定義されたサーバ間で当該チケットを転送することにより実現され得る。例えば、図5のチケット501を確認対象とすると、スキャナ10は、当該チケットにおいて最初に定義されているプロセス設定504を実行する処理サーバ60に対して、当該チケットを送信する。処理サーバ60は、確認処理の実行後、チケット501に定義された他の処理サーバ70に対して、当該チケットを転送する。チケット501に定義された全ての処理サーバにおける確認処理の実行後、当該チケットはスキャナ10へ返送される。その後、S103へ移行する。なお、S102における処理は、スキャナ10が各処理サーバに対して個別にチケットを送受信することによっても実現され得る。

30

【0029】

ここで、図7に示すチケット701は、チケット501を確認対象として、各処理サーバにおける確認処理の後に、最終的にスキャナ10へ返送されたチケットの一例である。チケット501には、チケットに含まれるプロセスを実行する処理サーバとして、処理サーバ60(ServerID="AAA")、処理サーバ70(ServerID="BBB")が定義されている。これに対して、返送されたチケット701には、各プロセス設定702~703に、当該機能を処理サーバ60、70においてそれぞれ実行可能か否かを示す情報712~713が付加されている。各情報712~713において、"not install"は、サーバIDに指定された処理サーバにおいて、当該機能を実行不可能であることを示す。なお、各処理サーバにおけるチケットの確認処理に関しては図13を用いて後述する。

40

【0030】

次に、S103で、CPU101は、チケット一覧901に含まれる全てのチケットの確認処理が終了したか否かを判定する。ここで、未だ確認処理が終了していないチケットが存在すると判定した場合には、S102へ戻って確認処理を継続する。一方で、全てのチケットの確認処理が終了したと判定した場合には、S104へ移行する。

50

【0031】

S104で、CPU101は、チケット一覧901に基づいて、操作部105のLCD表示部にチケットの一覧表示を行う。操作部105には、例えば、図10に示す表示画面1001が表示される。表示画面1001には、チケットの一覧表示1002、表示切替ボタン1003、及び実行指示ボタン1004がソフトキーとして表示されている。その後、S105へ移行する。

【0032】

S105で、CPU101は、ユーザによってチケットが選択され、チケットの実行指示が与えられたか否かを判定する。当該判定処理は、例えば、表示画面1001の実行指示ボタン1004が押下されたことを検知したか否かによって実現される。ここで、CPU101は、当該実行指示が与えられていないと判定した場合には、S105の処理を繰り返す。一方で、当該実行指示が与えられたと判定した場合には、S106へ移行し、チケットの実行処理を開始する。

10

【0033】

S106で、CPU101は、実行対象のチケットに定義された複数のプロセスのうち、実行することを予め定義された処理サーバが実行不可能であると確認されているプロセスについて、可能な場合には当該プロセスを実行する処理サーバを変更する。より具体的には、CPU101は、実行不可能と確認されたプロセスが、他の処理サーバにおいて実行可能と確認されている場合には、当該プロセスを当該他の処理サーバで実行するように、チケットの定義を変更する。その後、S107へ移行する。

20

【0034】

S107で、CPU101は、実行対象のチケットに、設定された処理サーバだけでなく、他の何れの処理サーバにおいても実行不可能と確認されたプロセス（実行不可プロセス）が存在するか否かを判定する。ここで、実行不可プロセスは存在しないと判定した場合には、S108へ移行する。一方で、実行不可プロセスが存在すると判定した場合には、S110へ移行する。S110で、CPU101は、実行対象のチケットにおいて、実行不可プロセスに関する設定を削除して、S108へ移行する。例えば、チケット701の場合、プロセス設定703における情報713が削除対象となる。

【0035】

S108で、CPU101は、上述の処理が実行されたチケットに含まれる読取条件（例えば、図7のスキャン設定705）に従って、原稿から画像を読み取ることにより、画像データを生成する。その際、外部の処理サーバにおいて展開可能なファイル形式（例えば、JPEG、TIFF、PDF等）で画像データを生成する。その後、S109で、CPU101は、チケットに含まれる複数のプロセスのうち、最初の実行対象のプロセスを実行する処理サーバに対して、当該チケット及び画像データを送信する。ここで、S109の処理は、実行制御手段による処理に相当する。チケット701の場合、CPU101は、プロセス設定702を実行する処理サーバ60に対して、それらのデータを送信する。これにより、CPU101は、最初の処理サーバ60にプロセス設定702の処理を開始させるとともに、当該チケットの定義に従った連携処理を各処理サーバに実行させる。

30

【0036】

なお、CPU101は、図10において、表示切替ボタン1003のチェックが外された場合には、実行不可プロセスを含むチケットを表示画面1001に表示しないように制御してもよい。これにより、実行不可プロセスを含むチケットが表示画面1001において選択不可能となるため、当該プロセスの実行を禁止することができる。

40

【0037】

<各処理サーバにおけるチケットの確認処理>

次に、図13を参照して、本実施形態に係る各処理サーバにおける確認処理（図12におけるS102の処理）の実行手順について、処理サーバ60における処理を一例として説明する。まず、S201で、処理サーバのCPU21は、スキャナ10から送信、又は他の処理サーバから転送された確認対象のチケットを受信する。ここでは、処理サーバ6

50

0 がスキャナ 10 からチケット 501 を受信した場合を想定する。その後、S202 へ移行する。

【0038】

S202 で、CPU21 は、受信したチケットから、未確認のプロセス設定を抽出する。さらに、S203 で、CPU21 は、当該プロセス設定に定義された機能（プロセス）を実行可能か否か判定する。当該判定処理は、例えば、図11 に示すように、処理サーバが記憶装置33 やRAM32 等に記憶している機能管理テーブル1101、1102 を参照することにより、実行される。確認対象の機能が機能管理テーブル1101、1102 に存在する場合、CPU21 は、当該機能が実行可能であると判定する。ここで、当該機能が実行可能であると判定した場合、S204 で、CPU21 は、確認対象のプロセス設定に対して、実行可能であることを示す情報を付加する。一方で、当該機能が実行不可能であると判定した場合、S205 で、CPU21 は、当該プロセス設定に対して、実行不可能であることを示す情報を付加する。S204 及びS205 の何れかの処理の実行後、S206 へ移行する。

10

【0039】

S206 で、CPU21 は、確認対象のチケットにおいて、未だ確認処理を行っていないプロセス設定が存在するか否かを判定する。ここで、未確認のプロセス設定が存在すると判定した場合には、S202 へ戻り、確認処理を継続する。一方で、未確認のプロセス設定が存在しないと判定した場合には、S207 へ移行する。

【0040】

S207 で、CPU21 は、確認対象のチケットに定義されている処理サーバのうち、未だ上述の確認処理を実行していない処理サーバが存在するか否かを判定する。当該判定処理は、例えば、チケットに予め定義された処理サーバのURL やサーバID等を抽出するとともに、各プロセス設定に既に付加された情報に含まれるURL やサーバIDとの比較によって実現される。ここで、確認処理を実行していない処理サーバが存在すると判定した場合には、S208 へ移行する。一方で、確認処理を実行していない処理サーバが存在しないと判定した場合には、S209 へ移行する。

20

【0041】

ここで、図6 に示すチケット601 は、確認対象のチケット501 に対して、処理サーバ60 においてS202 ~ S206 の処理の実行後のチケットの一例である。チケット601 には、予め定義されたプロセス設定602、603、604 に対して、サーバ60 (ServerID = "AAA") における各機能の実行の可否を示す情報612、613、614 がそれぞれ付加されている。なお、チケット601 には、定義された各機能を実行する処理サーバとして、サーバ60 の他にサーバ70 (ServerID = "BBB") も予め定義されている。

30

【0042】

S208 で、CPU21 は、チケットに定義されている処理サーバのうち、確認処理を実行していない処理サーバの何れかに対して、チケットを転送する。例えば、チケット601 の場合、CPU21 は、確認処理を実行していない処理サーバ70 (ServerID = "BBB") に対して転送する。また、S209 で、CPU21 は、確認処理の実行が完了したチケットを、スキャナ10 へ返送する。あるいは、送信元の処理サーバへ返送することを繰り返すことにより、チケットに定義された各処理サーバを中継してスキャナ10 へ返送してもよい。S208 及びS209 の何れかの処理の後、処理が終了する。

40

【0043】

< 各処理サーバにおけるチケットの実行処理 >

図12 のS109 においてスキャナ10 からチケット及び画像データが送信されると、当該チケットの定義に従って、複数の処理サーバにおいて当該画像データに対する画像処理が実行される。以下では、図14 を参照して、本実施形態に係る各処理サーバにおけるチケットの実行処理の手順について説明する。

【0044】

50

まず、S301で、処理サーバのCPU21は、スキャナ10から送信、又は他の処理サーバから転送される、実行対象のチケット及び当該チケットに従った画像処理の対象となる画像データを受信する。これらのデータを受信すると、S302で、CPU21は、受信したチケットのプロセス設定を予め定義された実行順序に従って確認し、未実行のプロセス設定を抽出する。その後、S303へ移行する。

【0045】

S303で、CPU21は、抽出したプロセス設定が、実行対象のプロセス設定が否かを判定する。当該判定は、抽出したプロセス設定に含まれるサーバIDやURLに基づいて実行される。ここで、当該プロセス設定が実行対象ではないと判定した場合には、S308へ移行する。S308で、CPU21は、当該プロセス設定の機能を実行することを
10

【0046】

S304で、CPU21は、抽出したプロセス設定に従って、定義された機能を実行することにより、画像データに対して画像処理を行う。さらに、S305で、CPU21は、チケットにおける抽出したプロセス設定の部分に対して、当該プロセス設定が実行済であることを示す情報を付加する。本実施形態では、図8に示すチケット801において、設定802(Result要素)がこれに相当する。当該処理の後、S306へ移行する
20

【0047】

S306で、CPU21は、チケットに未実行のプロセスが存在するか否かを判定する。ここで、未実行のプロセスが存在すると判定した場合には、S302へ戻り、上述の処理を繰り返す。一方で、未実行のプロセスが存在しないと判定した場合には、チケットに定義された保存先のサーバへチケット及び画像を転送し、処理を終了する。

【0048】

以上説明したように、第1の実施形態に係る画像処理装置は、複数の外部装置との連携処理によって画像データに対する画像処理を実行するために、当該連携処理の手順が予め定義された連携処理フローを外部の記憶装置から取得する。また、取得した連携処理フローに従った処理の実行開始前に、当該フローに含まれる複数の機能を、当該フローに予め
30

定義されている複数の外部装置のそれぞれが実行可能か否かを確認する。さらに、実行することを予め定義された外部装置が実行不可能であると確認された機能について、実行可能であると確認された他の外部装置が変わりに当該機能を実行するように、当該フローの定義を変更した後に、当該フローに従った処理の実行を開始する。これにより、連携処理フローを実行する外部装置の機能に変更が生じた場合に、ユーザに煩雑な作業を求めることなく、当該フローの定義を変更して実行することができる。また、連携処理フローに従った処理がエラーにより中止されることを抑制することができる。その結果、画像処理装置は、複数の外部装置とともに、連携処理フローに従った処理の効率の低下を防止できる
40

【0049】

なお、本実施形態に係る画像処理装置は、連携処理フローに含まれる各機能について、実行することを予め定義された外部装置のみならず、他の何れの外部装置においても実行不可能であると確認された場合には、当該機能を当該フローから削除してもよい。その場合、画像処理装置は、削除による変更後の当該フローの定義に従った処理の実行を開始する。あるいは、当該機能を削除するのではなく、当該フローによる処理の実行を禁止してもよい。これにより、連携処理フローを実行する外部装置の機能の変更起因して、何れの外部装置においても実行不可能な機能が定義に含まれた場合にも、当該フローに従った処理の途中で、エラーで処理が中止されることを抑制し、処理効率の低下を防止できる。

【0050】

[第2の実施形態]

10

20

30

40

50

次に、図15及び図16を参照して、本発明における第2の実施形態について説明する。第1の実施形態では、連携処理フローによる処理の実行開始前に、当該フローの定義に含まれる各機能が実行可能か否か予め確認するとともに、当該確認結果に基づいて当該フローの定義を変更する。その場合に、連携処理フローに含まれる各機能を定義された順に各外部装置が実行する際に、当該変更結果に依存して、外部装置間におけるチケット及び画像データの転送が何度も繰り返される可能性がある。そこで、本実施形態では、画像処理装置において連携処理フローを変更する際に、当該フローに含まれる複数の機能のうち実行可能な機能の数が最も多い外部装置が、優先的に各機能を実行するように、当該フローを変更することを特徴とする。

【0051】

本実施形態に係るスキャナ10におけるチケットの実行処理の手順は、第1の実施形態にの図12と基本的に同様であるので、異なる部分についてのみ説明する。なお、本実施形態において、各処理サーバにおけるチケットの確認処理及び実行処理の手順は第1の実施形態の図13及び図14と同様であるため、その説明を省略する。

【0052】

図12のS106において、CPU101は以下の処理を実行する。S106で、CPU101は、第1の実施形態と同様の処理に加えて、実行することを予め定義された処理サーバが実行不可能であると確認されたプロセスについて、当該プロセスが他の複数の処理サーバで実行可能である場合に、チケットに含まれるプロセスを最も多く実行可能な処理サーバが優先的に各プロセスを実行するように、当該フローを変更する。なお、S106で、CPU101は、実行することを予め定義された処理サーバが、定義どおりにプロセスを実行可能である場合においても、優先される処理サーバが当該プロセスを実行可能であるときは、上記と同様に当該フローを変更するとよい。以下では、S106の処理について、図15を参照して、より詳細に説明する。

【0053】

S501で、CPU101は、処理対象のチケットに含まれる全てのプロセス設定を抽出する。次に、S502で、CPU101は、抽出した情報に基づいて、図16に一例として示すような機能状態テーブル1601を作成する。ここで、機能状態テーブル1601は、処理対象のチケット1611に基づいて作成されたテーブルである。CPU101は、チケットに定義された処理サーバごとに、各プロセスの実行可否を確認することにより、機能状態テーブル1601を作成する。ここで、チケット1611には、処理サーバ60を示すServerID="AAA"と処理サーバ70を示すServerID="BBB"が含まれている。このため、CPU101は、プロセス設定1612、1613、1614ごとに、処理サーバ60、70における実行の可否を確認する。例えば、プロセス設定1612の場合、各サーバに対応する状態(STATUS要素)がともに"available"であるので、機能状態テーブル1601には何れも実行可能であることを示す情報が設定される。

【0054】

その後、S503で、CPU101は、作成した機能状態テーブル1601において、各処理サーバが実行可能なプロセスの数(処理スコア)を計算する。さらに、計算した処理スコアに基づいて、S504で、CPU101は、最も処理スコアの高い処理サーバを、最優先の処理サーバと決定する。その後、S505へ移行する。機能状態テーブル1601においては、処理サーバ60、70の処理スコアはそれぞれ1及び3と計算される。従って、処理サーバ70が最優先の処理サーバと決定される。

【0055】

S505で、CPU101は、プロセス設定に含まれるプロセスID(id)の順に、チケット1611から未選択のプロセス設定を選択する。さらに、S506で、CPU101は、当該プロセスが最優先の処理サーバで実行可能か否かを判定する。ここで、実行可能であると判定した場合には、S507へ移行し、当該プロセスを最優先の処理サーバで実行するように、プロセス設定を変更する。その後、S508へ移行する。一方で、S

10

20

30

40

50

506で、CPU101は、実行不可能であると判定した場合には、プロセス設定を変更せずに、S508へ移行する。

【0056】

例えば、チケット1611のプロセス設定1612では、最優先の処理サーバ70において実行可能であるため、CPU101は、当該プロセスを処理サーバ70が実行するように、処理サーバ70のID及びURL(1622)を、プロセスを実行する処理サーバのID及びURL(1621)に設定する。

【0057】

S508で、CPU101は、チケット1611に未選択のプロセスが存在するか否かを判定する。ここで、未選択のプロセスが存在すると判定した場合には、S505へ戻り、変更処理を継続する。一方で、未選択のプロセスが存在しないと判定した場合には、変更処理を終了する。

10

【0058】

以上説明したように、第2の実施形態に係る画像処理装置は、連携処理フローによる処理の実行開始前に、当該フローの定義に含まれる各外部装置における、各機能の実行可否の確認結果に基づいて、連携処理フローを変更する。その際に、当該フローに含まれる複数の機能のうち実行可能な機能の数が最も多い外部装置が、優先的に各機能を実行するように、当該フローを変更する。これにより、本実施形態に係る画像処理装置は、連携処理フローに含まれる各機能を実行する処理サーバの数を可能な限り減少させることが可能になる。その結果、第1の実施形態における効果に加えて、連携処理フローによる処理の実行中に、処理サーバ間で転送されるチケット及び画像データの転送回数を減少させることができ、ネットワーク上で送受信されるトラフィック量の低減効果を得ることができる。

20

【0059】

なお、本実施形態において、図15の処理は、スキャナ10で実行される場合について説明したが、当該処理は何れかの処理サーバで実行されてもよい。即ち、図13のS209で、各処理サーバから送信元の装置へチケットを順に中継することによって、チケットをスキャナ10へ返送する場合には、最後にスキャナ10へ中継する処理サーバにおいて、当該処理を実行することが可能である。その場合にも、上記と同等の効果を得ることができる。

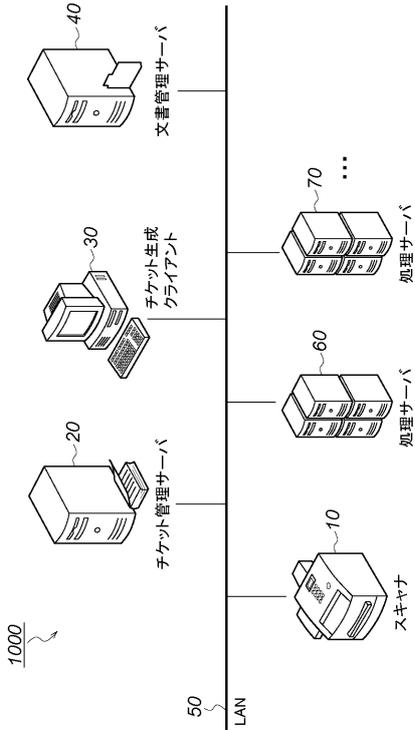
【0060】

[その他の実施形態]

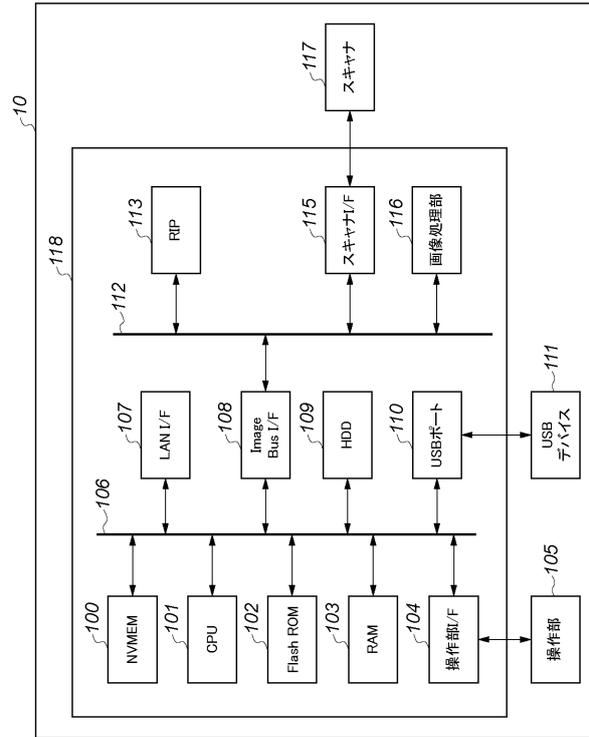
また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア(プログラム)を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ(またはCPUやMPU等)がプログラムを読み出して実行する処理である。

30

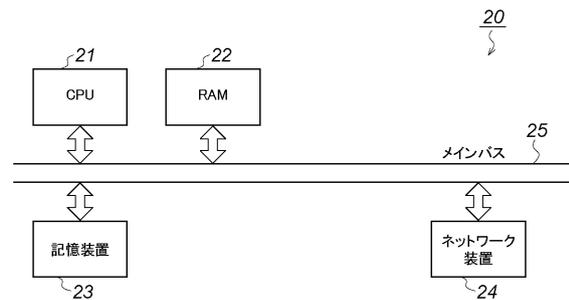
【図1】



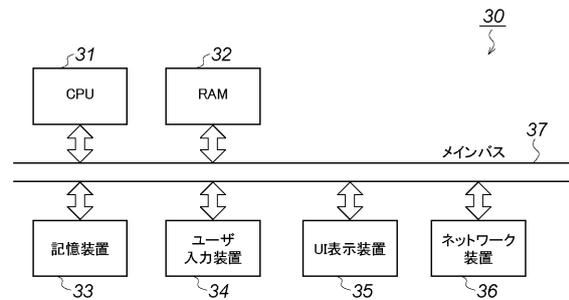
【図2】



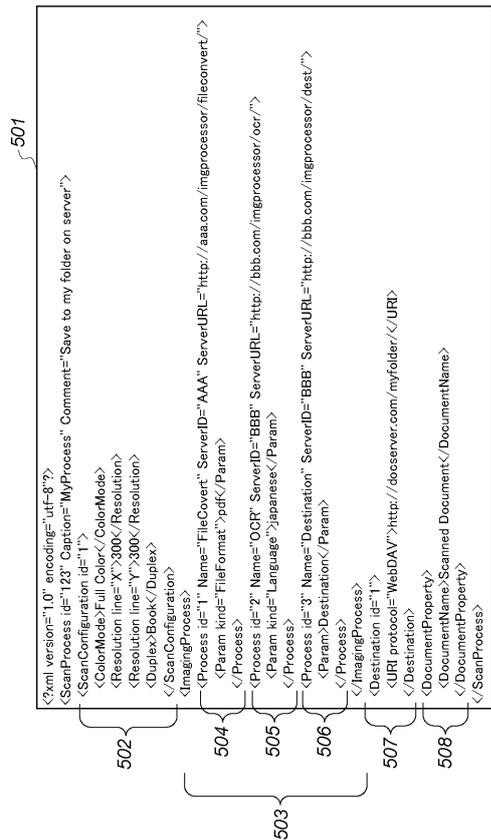
【図3】



【図4】



【図5】



【 図 6 】

```

<?xml version="1.0" encoding="utf-8"?>
<ScanProcess id="123" Caption="MyProcess" Comment="Save to my folder on server">
  <ColorMode>Full Color</ColorMode>
  <Resolution line="X">300</Resolution>
  <Resolution line="Y">300</Resolution>
  <Duplex>Book</Duplex>
  <ImagingProcess>
    <Process id="1" Name="FileCovert" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">
      <Param kind="FileFormat">pdf</Param>
      <Status id="1" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">available</Status>
    </Process>
    <Process id="2" Name="OCR" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">
      <Param kind="Language">japanese</Param>
      <Status id="2" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">not install</Status>
    </Process>
    <Process id="3" Name="Destination" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">
      <Param kind="Destination">Param>
      <Status id="3" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">not install</Status>
    </Process>
  </ImagingProcess>
  <Destination id="1">
    <URI protocol="WebDAV">http://docserver.com/myfolder</URI>
  </Destination>
  <DocumentName>Scanned Document</DocumentName>
  </DocumentName>
</ScanProcess>

```

【 図 7 】

```

<?xml version="1.0" encoding="utf-8"?>
<ScanProcess id="123" Caption="MyProcess" Comment="Save to my folder on server">
  <ColorMode>Full Color</ColorMode>
  <Resolution line="X">300</Resolution>
  <Resolution line="Y">300</Resolution>
  <Duplex>Book</Duplex>
  <ImagingProcess>
    <Process id="1" Name="FileCovert" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">
      <Param kind="FileFormat">pdf</Param>
      <Status id="1" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">available</Status>
    </Process>
    <Process id="2" Name="OCR" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">
      <Param kind="Language">japanese</Param>
      <Status id="2" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">not install</Status>
    </Process>
    <Process id="3" Name="Destination" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">
      <Param kind="Destination">Param>
      <Status id="3" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">not install</Status>
    </Process>
  </ImagingProcess>
  <Destination id="1">
    <URI protocol="WebDAV">http://docserver.com/myfolder</URI>
  </Destination>
  <DocumentName>Scanned Document</DocumentName>
  </DocumentName>
</ScanProcess>

```

【 図 8 】

```

<?xml version="1.0" encoding="utf-8"?>
<ScanProcess id="123" Caption="MyProcess" Comment="Save to my folder on server">
  <ColorMode>Full Color</ColorMode>
  <Resolution line="X">300</Resolution>
  <Resolution line="Y">300</Resolution>
  <Duplex>Book</Duplex>
  <ImagingProcess>
    <Process id="1" Name="FileCovert" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">
      <Param kind="FileFormat">pdf</Param>
      <Status id="1" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">available</Status>
    </Process>
    <Process id="2" Name="OCR" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">
      <Param kind="Language">japanese</Param>
      <Status id="2" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">not install</Status>
    </Process>
    <Process id="3" Name="Destination" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">
      <Param kind="Destination">Param>
      <Status id="3" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">available</Status>
    </Process>
  </ImagingProcess>
  <Destination id="1">
    <URI protocol="WebDAV">http://docserver.com/myfolder</URI>
  </Destination>
  <DocumentName>Scanned Document</DocumentName>
  </DocumentName>
</ScanProcess>

```

【 図 9 】

901

| | |
|--------|-------------------|
| チケット一覧 | |
| 1 | My Process |
| 2 | Send to my friend |
| 3 | Send to my PC |

1003 実行不可プロセスを含むチケットを表示する

1004 スキャン実行

【 図 10 】

1001

チケット表示画面

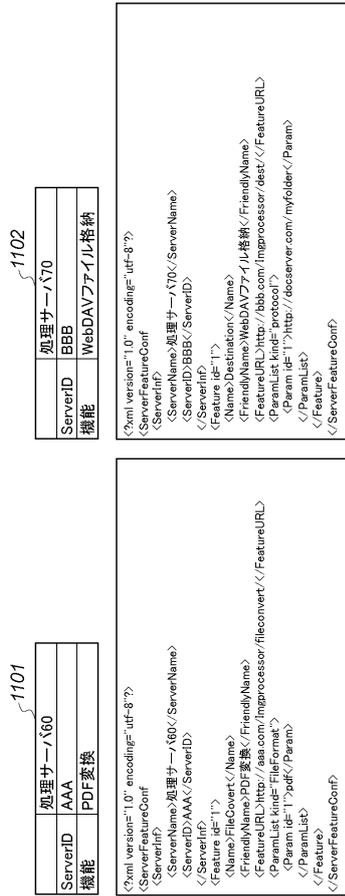
My Process

Send to my friend

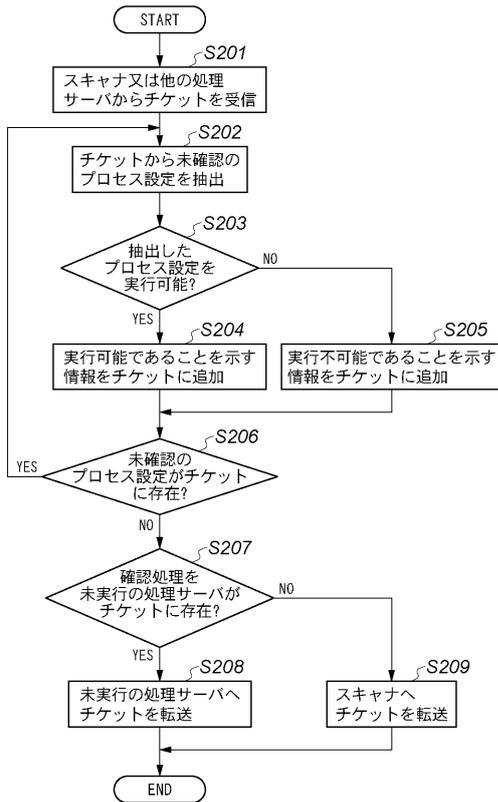
Send to my PC

1002

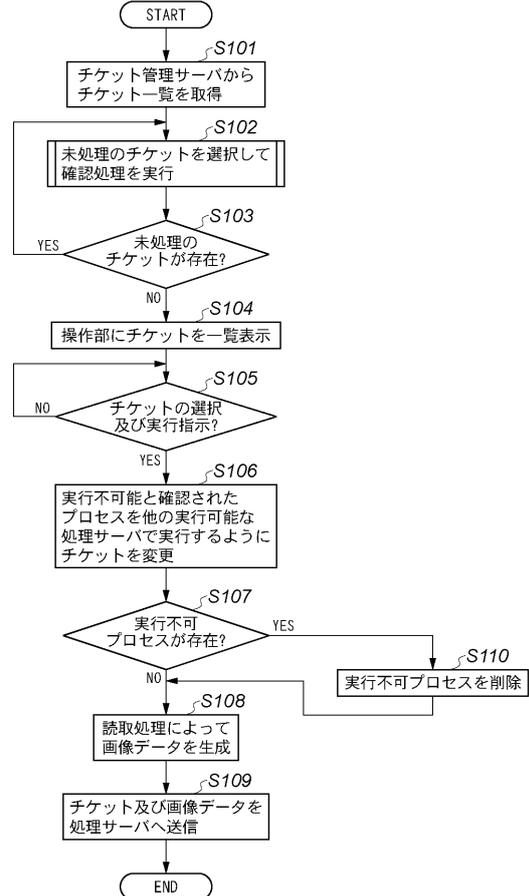
【図 1 1】



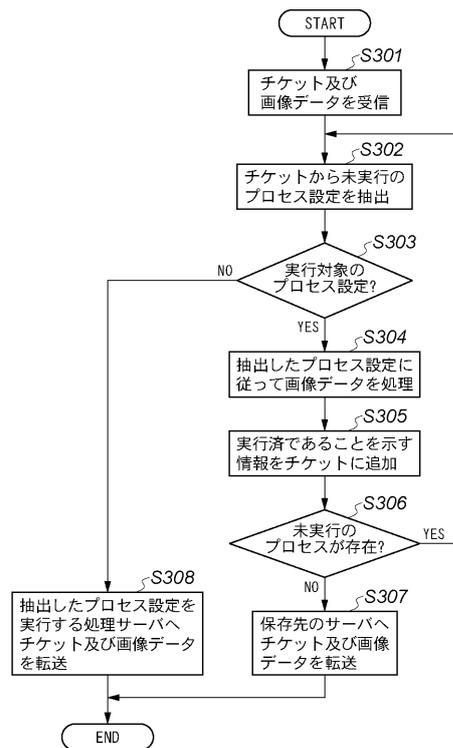
【図 1 3】



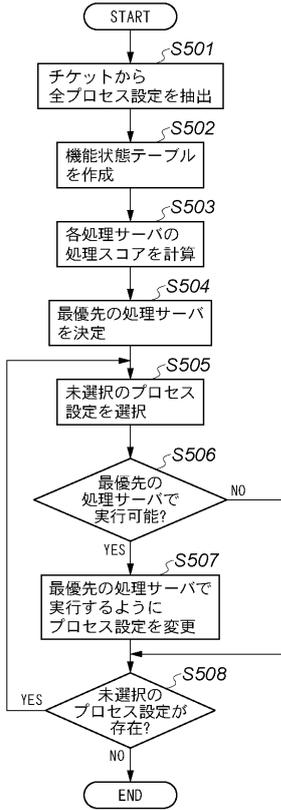
【図 1 2】



【図 1 4】



【図15】



【図16】

| | | | |
|------|---------------------|--------------|--------------|
| 1612 | FileConvert(pdf) | AAA(処理サーバ60) | BBB(処理サーバ70) |
| 1613 | OCR | 実行不可 | 実行可能 |
| 1614 | Destination(WebDAV) | 実行不可 | 実行可能 |

| | | |
|-------|---|---|
| 処理スコア | 1 | 3 |
|-------|---|---|

```

    <?xml version="1.0" encoding="utf-8"?>
    <ScanProcess id="123" Caption="MyProcess" Comment="Save to my folder on server">
    <ScanConfiguration id="1">
    <ColorMode>Full Color</ColorMode>
    <Resolution line="X">300</Resolution>
    <Resolution line="Y">300</Resolution>
    <Duplex>Book</Duplex>
    </ScanConfiguration>
    <ImagingProcess>
    <Process id="1" Name="FileCovert" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">
    <Param kind="FileFormat">pdf</Param>
    <Statuses>
    <Status id="1" ServerID="AAA" ServerURL="http://aaa.com/imgprocessor/fileconvert/">available</Status>
    <Status id="2" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/fileconvert/">available</Status>
    </Statuses>
    </Process>
    <Process id="2" Name="OCR" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">
    <Param kind="Language">japanese</Param>
    <Statuses>
    <Status id="1" ServerID="AAA">not install</Status>
    <Status id="2" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/ocr/">available</Status>
    </Statuses>
    </Process>
    <Process id="3" Name="Destination" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">
    <Param>Destination</Param>
    <Statuses>
    <Status id="1" ServerID="AAA">not install</Status>
    <Status id="2" ServerID="BBB" ServerURL="http://bbb.com/imgprocessor/dest/">available</Status>
    </Statuses>
    </Process>
    </ImagingProcess>
    <Destination id="1">
    <URI protocol="WebDAV">http://docserver.com/myfolder/</URI>
    <DocumentProperty>
    <DocumentName>Scanned Document</DocumentName>
    </DocumentProperty>
    </Destination>
    </ScanProcess>
  
```

フロントページの続き

(72)発明者 中脇 純
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 板垣 有紀

(56)参考文献 特開2005-275476(JP,A)
特開平10-147043(JP,A)
特開2005-266930(JP,A)
特開2008-282336(JP,A)

(58)調査した分野(Int.Cl., DB名)
G06T 1/20
G06F 9/50
H04N 1/00