

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4297811号  
(P4297811)

(45) 発行日 平成21年7月15日(2009.7.15)

(24) 登録日 平成21年4月24日(2009.4.24)

(51) Int.Cl.

F I

<b>G06F 13/00</b>	<b>(2006.01)</b>	G06F 13/00	550A
<b>G06F 3/048</b>	<b>(2006.01)</b>	G06F 3/00	655B
<b>G06F 3/14</b>	<b>(2006.01)</b>	G06F 3/14	350B
<b>G06F 12/00</b>	<b>(2006.01)</b>	G06F 12/00	515B
<b>G06F 17/30</b>	<b>(2006.01)</b>	G06F 12/00	546A

請求項の数 15 (全 29 頁) 最終頁に続く

(21) 出願番号 特願2004-65119 (P2004-65119)  
 (22) 出願日 平成16年3月9日(2004.3.9)  
 (65) 公開番号 特開2004-303218 (P2004-303218A)  
 (43) 公開日 平成16年10月28日(2004.10.28)  
 審査請求日 平成18年6月27日(2006.6.27)  
 (31) 優先権主張番号 特願2003-70908 (P2003-70908)  
 (32) 優先日 平成15年3月14日(2003.3.14)  
 (33) 優先権主張国 日本国(JP)

(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 100070150  
 弁理士 伊東 忠彦  
 (72) 発明者 中村 学  
 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 西出 隆二

最終頁に続く

(54) 【発明の名称】 情報提供装置、情報提供方法、及びプログラム

(57) 【特許請求の範囲】

【請求項1】

ネットワークを介して接続している端末からの要求に基づいて、前記端末にWebページを送信する情報提供装置であって、

前記端末の要求に基づいて目次情報を表示するメニューフレームと、前記目次情報の選択に対応するWebページを表示するワークフレームとを含むフレームページを提供するフレームページ提供手段を有し、

前記フレームページ提供手段は、

前記端末より送信される第一のURLに前記ワークフレームに表示させる情報を特定するページ指定情報が引数情報として含まれている場合、前記ワークフレームに対するURLを前記ページ指定情報によって特定されるWebページのURLとし、かつ、前記メニューフレームに対するURLに前記ページ指定情報を引数情報として含む前記フレームページの定義データを生成し、該定義データを前記端末に送信する定義データ生成手段と、

前記定義データに基づいて前記端末より送信される前記メニューフレームに対するURLを受信し、前記メニューフレームに対するURLに含まれている引数情報に従って前記ワークフレームに表示させる情報に応じた目次情報を表示させるWebページを生成し、該Webページを前記端末に送信するメニューページ提供手段とを有する情報提供装置。

【請求項2】

前記目次情報は、ツリー形式によって表示されることを特徴とする請求項1記載の情報提供装置。

10

20

**【請求項 3】**

前記メニューページ提供手段は、前記メニューフレームに対するURLに含まれる言語識別情報に応じた言語で前記目次情報を表示させるWebページを生成する請求項1又は2記載の情報提供装置。

**【請求項 4】**

前記フレームページ提供手段は、

前記フレームページの定義データに基づいて前記端末より送信される前記ワークフレームに対するURLを受信し、前記ワークフレームに対するURLによって特定されるWebページを生成し、該Webページを前記端末に送信するワークページ提供手段を有する請求項1乃至3いずれか一項記載の情報提供装置。

10

**【請求項 5】**

前記ワークページ提供手段は、前記ワークフレームに対するURLに含まれる言語識別情報に応じた言語で前記ワークフレームに対するURLによって特定されるWebページを生成する請求項4記載の情報提供装置。

**【請求項 6】**

前記第一のURLに引数情報として含まれる前記ページ指定情報に基づいて前記メニューフレームに表示させる情報が決定されるように、前記フレームページの書式情報が定義された書式情報ファイルを更に有し、

前記定義データ生成手段は、前記第一のURLに含まれる引数情報を前記書式情報ファイルに適用することによって、前記定義データを生成することを特徴とする請求項1乃至5いずれか一項記載の情報提供装置。

20

**【請求項 7】**

前記書式情報ファイルには、前記第一のURLに引数情報として含まれる前記ページ指定情報に基づいて前記ワークフレームに表示させる情報が決定されるように、前記フレームページの書式情報が定義されている請求項6記載の情報提供装置。

**【請求項 8】**

ネットワークを介して接続している端末からの要求に基づいて、前記端末にWebページを送信する情報提供装置が実行する情報提供方法であって、

前記端末の要求に基づいて目次情報を表示するメニューフレームと、前記目次情報の選択に対応するWebページを表示するワークフレームとを含むフレームページを提供するフレームページ提供手順を有し、

30

前記フレームページ提供手順は、

前記端末より送信される第一のURLに前記ワークフレームに表示させる情報を特定するページ指定情報が引数情報として含まれている場合、前記ワークフレームに対するURLを前記ページ指定情報によって特定されるWebページのURLとし、かつ、前記メニューフレームに対するURLに前記ページ指定情報を引数情報として含む前記フレームページの定義データを生成し、該定義データを前記端末に送信する定義データ生成手順と、

前記定義データに基づいて前記端末より送信される前記メニューフレームに対するURLを受信し、前記メニューフレームに対するURLに含まれている引数情報に従って前記ワークフレームに表示させる情報に応じた目次情報を表示させるWebページを生成し、該Webページを前記端末に送信するメニューページ提供手順とを有する情報提供方法。

40

**【請求項 9】**

前記目次情報は、ツリー形式によって表示されることを特徴とする請求項8記載の情報提供方法。

**【請求項 10】**

前記メニューページ提供手順は、前記メニューフレームに対するURLに含まれる言語識別情報に応じた言語で前記目次情報を表示させるWebページを生成する請求項8又は9記載の情報提供方法。

**【請求項 11】**

前記フレームページ提供手順は、

50

前記フレームページの定義データに基づいて前記端末より送信される前記ワークフレームに対するURLを受信し、前記ワークフレームに対するURLによって特定されるWebページを生成し、該Webページを前記端末に送信するワークページ提供手順を有する請求項8乃至10いずれか一項記載の情報提供方法。

**【請求項12】**

前記ワークページ提供手順は、前記ワークフレームに対するURLに含まれる言語識別情報に応じた言語で前記ワークフレームに対するURLによって特定されるWebページを生成する請求項11記載の情報提供方法。

**【請求項13】**

前記第一のURLに引数情報として含まれる前記ページ指定情報に基づいて前記メニューフレームに表示させる情報が決定されるように、前記フレームページの書式情報が定義された書式情報ファイルを更に有し、

前記定義データ生成手順は、前記第一のURLに含まれる引数情報を前記書式情報ファイルに適用することによって、前記定義データを生成することを特徴とする請求項8乃至12いずれか一項記載の情報提供方法。

**【請求項14】**

前記書式情報ファイルには、前記第一のURLに引数情報として含まれる前記ページ指定情報に基づいて前記ワークフレームに表示させる情報が決定されるように、前記フレームページの書式情報が定義されている請求項13記載の情報提供方法。

**【請求項15】**

コンピュータに、

ネットワークを介して接続している端末の要求に基づいて目次情報を表示するメニューフレームと、前記目次情報の選択に対応するWebページを表示するワークフレームとを含むフレームページを提供するフレームページ提供手順を実行させ、

前記フレームページ提供手順は、

前記端末より送信される第一のURLに前記ワークフレームに表示させる情報を特定するページ指定情報が引数情報として含まれている場合、前記ワークフレームに対するURLを前記ページ指定情報によって特定されるWebページのURLとし、かつ、前記メニューフレームに対するURLに前記ページ指定情報を引数情報として含む前記フレームページの定義データを生成し、該定義データを前記端末に送信する定義データ生成手順と、

前記定義データに基づいて前記端末より送信される前記メニューフレームに対するURLを受信し、前記メニューフレームに対するURLに含まれている引数情報に従って前記ワークフレームに表示させる情報に応じた目次情報を表示させるWebページを生成し、該Webページを前記端末に送信するメニューページ提供手順とを有するプログラム。

**【発明の詳細な説明】**

**【技術分野】**

**【0001】**

本発明は、情報提供装置に関し、特にネットワークを介して接続している端末からの要求に基づいて、前記端末にWebページを送信する情報提供装置に関する。

**【0002】**

また、本発明は、前記情報提供装置から送信されたWebページを表示する情報表示装置に関する。

**【背景技術】**

**【0003】**

近年のWeb関連技術の発展により、Webページの操作性も向上している。かかるWebページの操作性を向上させるための機能として、今や一般的に利用されているものにフレームがある(例えば、特許文献1)。

**【0004】**

図1は、フレームに分割されたWebページの表示例を示す図である。図1において、

10

20

30

40

50

WebブラウザにはWebページ500が表示されている。また、Webページ500は3つのフレーム(501、502、503)に分割されている。

【0005】

フレームを利用することの利点の一つとして、ドキュメントのナビゲーション機能を提供できることが挙げられる。例えば、図1において、フレーム501には検索機能、フレーム502には目次機能を持たせ、フレーム503には、フレーム501で検索指示をした結果、又はフレーム502の目次において選択したドキュメントを表示させるという構成をとることができる。この場合、フレーム501に入力した値や、フレーム502における目次の選択状態はそのままにして、フレーム503の内容のみが更新される。従って、ユーザは、現在参照している情報の位置付けを容易に確認することができ、くもの巣状に張られた情報のリンクの中で迷子になることはない。

10

【特許文献1】特開2001-34527号公報

【特許文献2】特開平11-296463

【発明の開示】

【発明が解決しようとする課題】

【0006】

しかしながら、フレームに分割されたWebページにおいては各フレームに最初に表示させるドキュメントは静的に決められているため、ユーザが、目的とするドキュメントを表示させるためには、更に数回の操作を必要とするという問題がある。

【0007】

20

例えば、図1の例において、ユーザがWebブラウザにWebページ500のURLを入力すると、WebブラウザにWebページ500が表示される。この状態でフレーム503には、Webページ500を参照する際には最初に必ず表示される、いわゆるトップページ、又はホームページと呼ばれるドキュメントが表示される。従って、ユーザは、フレーム502の目次機能を利用して、目的とするドキュメントを表示させるための操作(マウスクリック等)を一回以上行う必要がある。

【0008】

特に、目次機能においては、情報の階層構造を視覚的に表現するため、ツリー構造をもったインタフェースが提供されている場合がある。その場合には、階層を一つ一つたどっていく必要があるため、更に操作が煩雑となる。

30

【0009】

なお、何度も同じドキュメントを参照する必要があるのであれば、ブックマークの利用も考えられる。しかし、フレームを利用したWebページにおいて、ブックマークに記録されるのは、フレームのレイアウトを規定したWebページのURLであって、各フレーム内に表示されたドキュメントのURLではない。図1の例では、ブックマークにはWebページ500のURLが記録され、フレーム503に表示されたドキュメントのURLは記録されない。従って、フレーム503に、目的とするドキュメントを表示させた状態でブックマークに記録しても、次回そのブックマークを利用した際にフレーム503に表示されるのは、やはりトップページである。

【0010】

40

ところで、近年においては汎用的なコンピュータのみならず、特定の用途に特化した機器についてもWebサーバとしての機能を有し、図1のWebページ500のようなホームページ(以下、Webページ500として説明する。)を提供することにより、機器に対する各種の遠隔操作を可能としているものもある。ユーザは、Webブラウザに表示されたWebページ500のフレーム502における目次機能によって、機器の異常の詳細内容を確認するためのページ(以下、「異常確認ページ」という。)や、機器に対してパラメータを設定するためのページ等をフレーム503に表示させることができる。このような機器の中には、異常の発生等を電子メールによって管理者等に自動的に通知するものがある。かかる電子メールには、異常確認ページに対するURLが記載されており、当該URLにアクセスして、異常の詳細内容を確認するよう促しているものもある。

50

## 【0011】

しかしながら、異常確認ページに対するURLをクリックしてWebブラウザを起動しても、Webブラウザには異常確認ページのみが全面に表示されてしまう。すなわち、Webページ500によって提供される操作環境（検索環境や目次機能等）は得ることはできず、当該操作環境を得るにはWebページ500に対するURLを入力しなければならない。

## 【0012】

なお、特許文献2には、フレームによって分割されたWebページをブックマークへ登録する際に、各フレームに表示されているページのURL等を記録しておくことにより、次回当該ブックマークが選択された場合に目的とするページをフレーム内にダイレクトに表示させることができる技術が開示されている。しかしながら、当該技術は、クライアント側において各フレームに対するURL等を記録しておくといった点において、サーバ側において機能を実現する本願発明とその構成を大きく異にする。

10

## 【0013】

本発明は、上記の点に鑑みてなされたものであって、Webページのフレーム内に、目的とするWebページを簡便に表示させることができる情報提供装置、情報提供方法、及び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】

## 【0014】

そこで上記課題を解決するため、本発明は、請求項1に記載されるように、ネットワークを介して接続している端末からの要求に基づいて、前記端末にWebページを送信する情報提供装置であって、前記端末からの要求に対して、複数のフレームに分割され、前記複数のフレームのうちの所定のフレームに前記端末からの要求において指定されたURLに含まれる識別情報に基づいて特定される情報を表示させるフレームページを生成するフレームページ生成手段を有することを特徴とする。

20

## 【0015】

また、上記課題を解決するため、本発明は、請求項17に記載されるように、ネットワークを介して接続している情報提供装置にWebページの送信を要求するWebページ要求手段と、前記Webページ要求手段が送信した要求に対して前記情報提供装置から送信された、複数のフレームに分割され、前記複数のフレームのうちの所定のフレームに前記要求において指定したURLに含まれる識別情報に基づいて特定される情報が表示されるフレームページを表示させる情報表示手段とを有することを特徴とする。

30

## 【0016】

このような情報提供装置又は情報表示装置では、URLに所定の識別情報を指定することで、Webページのフレーム内に目的とするWebページを簡便に表示させることができる。

## 【発明の効果】

## 【0017】

本発明によれば、Webページのフレーム内に、目的とするWebページを簡便に表示させることができる情報提供装置、情報提供方法、及びプログラムを提供することができる。

40

## 【発明を実施するための最良の形態】

## 【0018】

以下、図面に基づいて本発明の実施の形態を説明する。本実施の形態においては、画像処理装置を情報提供装置の一例として説明する。図2は、本発明の実施の形態における画像処理装置の機能構成例を示す図である。図2において、画像処理装置10は、プロッタ

50

エンジン 1025 及びスキャナエンジン 1026 等の個別の画像処理を行うためのハードウェア資源と、エンジン制御ボード 1024 と、OS 1023 と、システム管理サービス 1016 と、ネットワークサービス 1017 と、メモリ管理サービス 1018 と、オペパネ管理サービス 1019 と、エンジン管理サービス 1020 と、配信サービス 1021 と、FAX 管理サービス 1022 と、コピーアプリ 1011、ファックスアプリ 1012、プリンタアプリ 1013、Web アプリ 1014 及び機器情報レポートアプリ 1015 等の各種アプリケーションとから構成される。

【0019】

エンジン制御ボード 1024 は、プロッタエンジン 1025 及びスキャナエンジン 1026 等の画像処理装置 10 の個別機能を制御し、エンジン I/F (インタフェース) を通して、エンジン制御ボード 1024 の機能を OS 1023 等に提供する。

10

【0020】

OS 1023 は、UNIX (登録商標) 等のオペレーティング・システムであり、コピーアプリ 1011、ファックスアプリ 1012 等、アプリケーションとして実装されている各種ソフトウェアをプロセスとして並列実行する。

【0021】

システム管理サービス 1016、ネットワークサービス 1017、メモリ管理サービス 1018、オペパネ管理サービス 1019、エンジン管理サービス 1020、配信サービス 1021、及び FAX 管理サービス 1022 は、API (Application Programming Interface) を通して、コピーアプリ 1011 等の上位アプリケーションにそれぞれの機能を提供するものである。即ち、システム管理サービス 1016 は、システム管理に関する機能を提供し、ネットワークサービス 1017 は、ネットワーク通信に関する機能を提供する。また、メモリ管理サービス 1018 は、メモリ管理に関する機能を提供し、オペパネ管理サービス 1019 は、画像処理装置 10 に設置された操作パネル (オペレーションパネル。以下、「オペパネ」という。) に関する機能を提供する。また、エンジン管理サービス 1020 は、エンジン制御ボード 1024 を介してプロッタエンジン 1025 やスキャナエンジン 1026 を制御するための機能を提供する。更に、配信サービス 1021 は、画像処理装置 10 において発生したイベント (異常の発生等) を通知するための電子メール (以下、「イベント通知メール」という。) の送信処理を実行するための機能を提供し、FAX 管理サービス 1022 は、FAX の送受信を制御するための機能を提供する。

20

30

【0022】

コピーアプリ 1011 は、コピー用アプリケーションであり、ファックスアプリ 1012 は、ファクシミリ用アプリケーションである。プリンタアプリ 1013 は、ページ記述言語 (PDL)、PCL 及びポストスクリプト (PS) を有するプリンタ用アプリケーションである。Web アプリ 1014 は、本発明の特徴が実装されているアプリケーションであり、画像処理装置 10 にネットワークを介して接続している端末に対して、端末のユーザに適した言語で各種の情報を提供するためのアプリケーションである。機器情報レポートアプリ 1015 は、イベント通知メールの生成に必要な情報の収集等、イベント通知メールの送信処理全体を制御するアプリケーションである。

40

【0023】

次に、上述した Web アプリ 1014 及びネットワークサービス 1017 等により実現される画像処理装置 10 の Web サーバとしての機能について説明する。図 3 は、Web サーバとしての画像処理装置の機能構成例を示す図である。

【0024】

図 3 において、画像処理装置 10 は、Web サーバプログラム 15 と、Web アプリ 11、12、及び 13 等と、XSLT プロセッサ 14 とから構成される。Web サーバプログラム 15 は、一般に HTTPd (HTTP daemon) と呼ばれるプログラムに相当し、画像処理装置 10 を Web サーバとして機能させるためのものである。Web サーバプログラム 15 は、インターネット、LAN (Local Area Network) 等のネットワークを介して接

50

続されている P C ( Personal Computer ) 等の端末 2 0 の W e b ブラウザ 2 1 から送信される H T T P リクエストを受信し、 H T T P リクエストにおいて指定された U R L に対応した W e b アプリ 1 1 等を起動する。また W e b サーバプログラム 1 5 は、 W e b アプリ 1 1 等が生成した H T M L ( HyperText Markup Language ) データ ( W e b ページ ) を W e b ブラウザ 2 1 に送信する。なお、 W e b サーバプログラム 1 5 は、図 2 において、ネットワークサービス 1 0 1 7 の機能の一部として実装されている。

**【 0 0 2 5 】**

W e b アプリ 1 1 は、画像処理装置 1 0 に関する各種情報を W e b ページによって端末 2 0 に提供するにあたり、入り口となる W e b ページを提供するいわゆる W e b アプリケーションである。 W e b アプリ 1 1 は、ページ呼び分け手段 1 1 1、ロードページモジュール 1 1 2、フレームページモジュール 1 1 3、及びライブラリ 1 1 4 等から構成される。

10

**【 0 0 2 6 】**

ページ呼び分け手段 1 1 1 は、 W e b ブラウザ 2 1 から W e b アプリ 1 1 宛ての H T T P リクエストがあった場合に W e b サーバプログラム 1 5 から呼び出され、要求された U R L に対応するページモジュールの呼び分け等の処理を行う関数群である。ここでページモジュールとは、 W e b ページの種類ごとに定義された関数であり、システムアプリ 1 1 においては、ロードページモジュール 1 1 2 とフレームページモジュール 1 1 3 とが該当する。

**【 0 0 2 7 】**

ロードページモジュール 1 1 2 やフレームページモジュール 1 1 3 は、必要に応じてライブラリ 1 1 4 における関数を呼び出し、それぞれが対応する W e b ページを生成する。ライブラリ 1 1 4 は、ページモジュールに対して種々の機能を提供するための関数群である。

20

**【 0 0 2 8 】**

なお、 W e b アプリ 1 2 及び W e b アプリ 1 3 は、画像処理装置 1 0 には、 W e b アプリ 1 1 以外にも、様々な機能を提供するための W e b アプリ ( W e b ページ生成手段 ) が存在していることを示すために例示したものである。例えば、 F A X 機能、文書管理機能、又は印刷機能に関する情報を提供するもの等が存在する。

**【 0 0 2 9 】**

本実施の形態においては、 W e b アプリ 1 1 を中心として説明するが、他の W e b アプリ ( W e b アプリ 1 2 及び 1 3 ) と明確に区別するために、以下 W e b アプリ 1 1 を「システムアプリ 1 1」と呼ぶこととする。

30

**【 0 0 3 0 】**

上述した各種 W e b アプリは、 X S L T プロセッサ 1 4 を用いて W e b ページを生成する。 X S L T プロセッサ 1 4 ( データ変換手段 ) は、所定の書式情報 ( X S L データ等 ) に従って所定の情報 ( X M L データ等 ) を W e b ページ ( H T M L データ等 ) に変換するためのプログラムである。 X S L T プロセッサ 1 4 は、複数の W e b アプリから共通に利用される。

**【 0 0 3 1 】**

図 4 は、ロードページモジュールを説明するための図である。ロードページモジュール 1 1 2 は、ロードページを生成するためのページモジュールであり、 W e b ブラウザ 2 1 から画像処理装置 1 0 にアクセスがあった場合に、ページ呼び分け手段 1 1 1 から呼び出される。ここで、ロードページとは、 H T M L 形式のいわゆる W e b ページであるが、 W e b ブラウザに情報を表示することを目的としたものではなく、他の W e b ページ、本実施の形態においては後述するフレームページを W e b ブラウザ 2 1 にロードさせるための W e b ページである。

40

**【 0 0 3 2 】**

即ち、ロードページモジュール 1 1 2 によって生成されたロードページには、 W e b ブラウザ 2 1 がロードページを読み込んだ際に、フレームページを自動的にロードさせるた

50

めの命令が J a v a (登録商標) S c r i p t 等により定義されている。

【 0 0 3 3 】

ロードページモジュール 1 1 2 は、ページ呼び分け手段 1 1 1 から呼び出されると、表示言語の選択を行う。上述したように、ロードページは表示を目的とするものではなく、フレームページのロードを目的としたものであるため、ここで選択するのは、フレームページの表示に用いる言語(日本語、英語等)である。

【 0 0 3 4 】

表示言語を選択すると、ロードページモジュール 1 1 2 は、ロードページ生成関数 1 1 5 を利用してロードページを生成する。ロードページ生成関数 1 1 5 は、図 3 におけるライブラリ 1 1 4 に相当する関数であり、ロードページの生成機能を提供するものである。ロードページモジュール 1 1 2 は、更に、ロードページ生成関数 1 1 5 が生成したロードページを HTML データとして完成させるため、ロードページ生成関数 1 1 5 から出力されたロードページに対して HTML ヘッダを付加して、Web サーバプログラム 1 5 に出力する。その後、ロードページは Web サーバプログラム 1 5 によって Web ブラウザ 2 1 に送信される。

【 0 0 3 5 】

図 5 は、フレームページモジュールを説明するための図である。フレームページモジュール 1 1 3 は、フレームページを生成するためのページモジュールであり、Web ブラウザ 2 1 から画像処理装置 1 0 にフレームページの送信要求があった場合に、ページ呼び分け手段 1 1 1 から呼び出される。ここで、フレームページとは、画像処理装置 1 0 が種々の情報を提供するにあたり基盤となる、複数のフレームに分割された Web ページであり、フレームのレイアウトが定義されたものである。即ち、画像処理装置 1 0 の提供する情報は、フレームページによって定義された複数のフレームの中の一に表示される。

【 0 0 3 6 】

フレームページモジュール 1 1 3 は、ページ呼び分け手段 1 1 1 から呼び出されると、Web ブラウザ 2 1 から文字列として渡ってきたデータを、それぞれのデータの意味に従って、フレームページモジュール 1 1 3 や後述するフレームページ処理関数 1 1 6 等の実装に利用されているプログラム言語のデータ型に変換する。例えば、フレームページモジュール 1 1 3 等が C 言語によって実装されている場合、渡って来た文字列が数値として処理されるべきものであれば、フレームページモジュール 1 1 3 は、当該文字列を i n t 型や l o n g 型に変換する。なお、以下においては、フレームページモジュール 1 1 3、フレームページ処理関数 1 1 6 は C 言語で実装されていることとして説明する。

【 0 0 3 7 】

フレームページモジュール 1 1 3 は、C 言語のデータ型に変換されたデータを引数に指定してフレームページ処理関数 1 1 6 を呼び出す。フレームページ処理関数 1 1 6 は、図 3 におけるライブラリ 1 1 4 に相当する関数である。フレームページ処理関数 1 1 5 は、所定の処理を実行し、その処理結果としてフレームページの生成に必要なパラメータをフレームページモジュール 1 1 3 に出力する。

【 0 0 3 8 】

フレームページモジュール 1 1 3 は、フレームページ処理関数 1 1 6 から出力された C 言語のデータ型によるパラメータを D O M (Document Object Model) 形式に変換し、ページ呼び分け手段 1 1 1 に出力する。ページ呼び分け手段 1 1 1 は、D O M 形式のパラメータを X M L 形式に変換し、X M L 形式となったパラメータを入力情報として指定して X S L T プロセッサ 1 4 を呼び出す。

【 0 0 3 9 】

X S L T プロセッサ 1 4 は、ページ呼び分け手段 1 1 1 から渡された X M L 形式のパラメータを図示しない X S L データに適用して HTML 形式のフレームページを生成する。

【 0 0 4 0 】

X S L T プロセッサ 1 4 によって生成されたフレームページは、フレームページ呼び分け手段 1 1 1 によって HTML ヘッダが付加され、Web サーバプログラム 1 5 に出力さ

10

20

30

40

50

れる。その後、フレームページは、Webサーバプログラム15によってWebブラウザ21に送信される。

【0041】

ここで、XSLTプロセッサ14について詳しく説明する。図6は、XSLTプロセッサを説明するための図である。図6において、XSLTプロセッサ14は、入力モジュール141と出力モジュール142とから構成される。

【0042】

入力モジュール141は、XMLデータ145とXSLデータ146とを読み込んで、XMLデータ145及びXSLデータ146の内容をDOM形式に変換する。なお、本実施の形態においては、XMLデータ145は、ページ呼び分け手段111より渡されたものが、XSLデータ146は、フレームページの書式が予め定義されたものが該当する。

【0043】

出力モジュール142は、入力モジュール141によってDOM形式に変換されたデータを入力情報として、XSLT変換を実行し、HTMLデータ147を生成する。なお、HTMLデータ147は、本実施の形態においてはフレームページが該当する。

【0044】

次に、画像処理装置10が提供するWebページについて説明する。図7は、トップページを表示したフレームページの表示例を示す図である。即ち、ユーザがWebブラウザ21のURL入力領域21に画像処理装置10のIPアドレス若しくはドメイン名(以下、両者を総称して「ホスト名」という。)、又はフレームページのURLを入力すると、Webブラウザ21には、図7に示されるフレームページ200が表示される。

【0045】

図7においてフレームページ200は、ヘッダ(header)フレーム201、メニュー(menu)フレーム202、及びワーク(work)フレーム203等の3つのフレームに分割されている。

【0046】

ヘッダフレーム201には、コンボボックス2011とボタン2012とが配置されている。ユーザは、コンボボックス2011を操作することにより、各フレームに表示される情報の表示言語を変更することができる。また、ユーザが、ボタン2012をクリックすると、ユーザIDとパスワードの入力を求められ、認証にパスすると、当該ユーザは、管理者モードで画像処理装置10のサービスを利用することが可能となる。ここで、管理者モードとは、特別な権限を有するユーザ(管理者)のみが利用可能なサービスを利用することができるモードである。管理者モードの他には、一般ユーザモードがある。一般ユーザモードは、利用できるサービスが制限されているユーザ(一般ユーザ)に対応したモードである。なお、初期状態(Webブラウザ21に最初にフレームページを表示させた状態)においては、一般ユーザモードとなっている。

【0047】

メニューフレーム202には、画像処理装置10が提供する情報の目次情報(メニュー)がツリー形式で表示される。従って、初期状態のメニューフレーム202には、情報の種別に対応したルートノードとして状態ノード2021及びドキュメントボックスノード2022が表示されている。

【0048】

ワークフレーム203には、メニューフレーム202で選択されたメニュー(ノード)に対応する情報が表示される。即ち、メニューフレーム202とワークフレーム203とは相互に関連を有している。但し、初期状態においては、ワークフレーム203には所定のトップページが表示される。

【0049】

ここで、ユーザが、画像処理装置10の給紙トレイの情報を英語でかつ管理者モードで参照したい場合に、一般的に以下の操作が必要となる。なお、メニューフレーム202における給紙トレイ情報に対応するノードは、状態ノード2021の直下にあるとする。

10

20

30

40

50

## 【 0 0 5 0 】

( 1 ) Webブラウザ 2 1 に画像処理装置 1 0 のホスト名又はフレームページ 2 0 0 の URL を入力し、Webブラウザ 2 1 にフレームページ 2 0 0 を表示させる。

## 【 0 0 5 1 】

( 2 ) コンボボックス 2 0 1 1 を操作して、表示言語を英語 ( E n g l i s h ) にする。

## 【 0 0 5 2 】

( 3 ) ボタン 2 0 1 2 をクリックして、認証を受ける ( 管理者モードにする ) 。

## 【 0 0 5 3 】

( 4 ) 状態ノード 2 0 2 1 をクリックして状態ノード 2 0 2 1 を開く。

10

## 【 0 0 5 4 】

( 5 ) 状態ノード 2 0 2 1 を開くことによって表示された給紙トレイ情報に対応するノードをクリックする。

## 【 0 0 5 5 】

図 8 は、給紙トレイ情報を英語でかつ管理者モードで表示したフレームページの表示例を示す図である。図 8 において、フレームページ 2 0 0 の各フレーム内の表示は英語になっている。また、メニューフレーム 2 0 2 においては、状態 ( S t a t u s ) ノード 2 0 2 1 が展開されており、給紙トレイ情報に対応するノード 2 1 2 1 が表示されている。更に、ワークフレーム 2 0 3 には、ノード 2 1 2 1 が選択されたことにより給紙トレイ情報を表示するための Web ページ ( 以下、「給紙トレイページ」という。 ) が表示されている。

20

## 【 0 0 5 6 】

上述したように、給紙トレイページを表示させるためにユーザは 5 つのステップを踏む必要がある。かかる操作は、当初より給紙トレイページを参照するつもりであるユーザにとって非常に煩雑なものである。

## 【 0 0 5 7 】

また、給紙トレイページを頻繁に参照するのであれば、ブックマークの利用も考えられる。しかし、URL 入力領域 2 1 1 に表示されているのは給紙トレイページの URL ではなく、フレームページ 2 0 0 の URL であることから明らかなように、図 8 の状態でブックマークに登録をしても、ブックマークへの登録の対象となるのは、フレームページ 2 0 0 に対する URL である。従って、図 8 の状態で登録したブックマークを利用した際に表示されるのは、図 7 の初期状態におけるフレームページ 2 0 0 である。

30

## 【 0 0 5 8 】

更に、給紙トレイページの URL を知ることができて、給紙トレイページの URL を直接 URL 入力領域 2 1 1 に入力した場合には、給紙トレイページは、フレームページ 2 0 0 のワークフレーム 2 0 3 に表示されるのではなく、Webブラウザ 2 1 の全面に表示されてしまう。従って、フレームページ 2 0 0 において作業を継続した場合には、結局のところ図 7 の初期状態のフレームページ 2 0 0 を表示させる必要がある。

## 【 0 0 5 9 】

そこで、本実施の形態においては、ユーザが、画像処理装置 1 0 より用紙切れを通知するイベント通知メールを受信した場合に、給紙トレイページを一回の操作でワークフレーム 2 0 3 に表示させることができるようにした例について説明する。

40

## 【 0 0 6 0 】

まず、画像処理装置 1 0 よりイベント通知メールが送信される段階について説明する。図 9 は、画像処理装置によるイベント通知メールの送信処理を説明するためのブロック図である。図 9 に示される画像処理装置 1 0 においては、図 2 及び図 3 より、システム管理サービス 1 0 1 6、ネットワークサービス 1 0 1 7、Web サーバプログラム 1 5、配信サービス 1 0 2 1、機器情報レポートアプリ 1 0 1 5、Web アプリ 1 1 ( システムアプリ 1 1 )、及び X S L T プロセッサ 1 4 が抜き出されている。更に、図 9 には、N V R A M ( Non Volatile RAM ) 1 7 及びイベント・URL 対応テーブル 1 8 が表示されている。

50

NVRAM 17は、画像処理装置10に関する各種パラメータ（IPアドレス、マシン名、表示言語の判別情報等）や、イベント通知メールの送信先となるメールアドレス等が記録されているメモリである。イベント・URL対応テーブル18は、画像処理装置10において発生する各種のイベントごとに、当該イベントの内容を通知するための情報（Webページ）に対するURLが登録されているテーブルである。

【0061】

図10は、画像処理装置によるイベント通知メールの送信処理を説明するためのフローチャートである。以下、図10の流れに沿って、図9等を参照しつつ説明する。なお、図10におけるステップ番号と図9におけるステップ番号とは対応づけられている。

【0062】

例えば、画像処理装置10の給紙トレイ内の用紙が切れると、その旨をシステム管理サービス1016が検出する（S101）。ステップS101に続いてステップS102に進み、システム管理サービス1016は、機器情報レポートアプリ1015に対して用紙切れが発生した旨を通知する。ステップS102に続いてステップS103に進み、機器情報レポートアプリ1015は、システム管理サービス1016より通知されたイベント（用紙切れ）の内容を表示するWebページ（以下、「対応ページ」という。）に対するURLをイベント・URL対応テーブル18より取得する。

【0063】

図11は、イベント・URL対応テーブルの構成例を示す図である。図11に示されるように、イベント・URL対応テーブル18には、イベントごとに対応ページに対するURLが登録されている。イベント・URL対応テーブル18より、例えば、用紙切れに対するURLは、「websys/status/inTray.cgi」であることが特定される。

【0064】

ステップS103に続いてステップS104に進み、機器情報レポートアプリ1015は、イベント通知メールの送信先とするメールアドレス（送信先アドレス）をNVRAM 17より取得する。ステップS104に続いてステップS105に進み、機器情報レポートアプリ1015は、対応ページのURLを引数として、システムアプリ11に対してイベント通知メールの本文（以下、単に「メール本文」という。）の生成を要求する。S105に続いてステップS106に進み、システムアプリ11は、メール本文の表示言語の判別情報、画像処理装置10のIPアドレス及びマシン名等NVRAM 17より取得する。

【0065】

ステップS106に続いてステップS107に進み、システムアプリ11は、XSLTプロセッサ14を用いて、対応URLとステップS106において取得した情報とを入力情報（以下、対応URLも含めて単に「入力情報」という。）としてメール本文をテキスト形式で生成する。詳しくは、システムアプリ11は、入力情報を各要素の値として含むXMLデータを生成し、当該XMLデータを、予めメール本文の書式が定義されているXSLデータに適用して、XSLTプロセッサ14にXSLT変換を実行させることによりメール本文を生成する。メール本文は、機器情報レポートアプリ1015に対して出力される。

【0066】

ステップS107に続いてステップS108に進み、機器情報レポートアプリ1015は、メール本文を送信先アドレスに対して送信するよう配信サービス1021に対して要求する。ステップS108に続いてステップS109に進み、配信サービス1021は、ネットワークサービス1017を介してイベント通知メールを送信先アドレスに対して送信する。その後、イベント通知メールは、メールサーバ30を介して端末20へ送信される。

【0067】

図12は、用紙切れを通知するイベント通知メールの例を示す図である。図12に示されるように、イベント通知メール500には、用紙を補給する必要がある旨や問い合わせ

10

20

30

40

50

先等が記載されている。また、詳細情報を参照するためのURL 501が、以下の形式で示されている。

【0068】

<ホスト名>/?open=websys/status/inTray.cgi&profile=admin&lang=en・・・(a)

一般にURLは以下の要素から構成されており、本実施の形態においても、各要素を<内の名称によって呼ぶこととする。

【0069】

<ホスト名>/<パス名>?<引数情報>

従って、URL(a)は、ホスト名と引数情報とから構成されパス名は含まれていないと表現することができる。ここで、対応ページのURL(websys/status/inTray.cgi)が、パス名としてではなく引数情報として含まれていることに注意を要する。すなわち、本実施の形態においては、ユーザの便宜を考慮して対応ページをフレームページ200の1のフレーム内に表示させたいところ、対応ページのURLがパス名として直接指定されていると、当該URLがクリックされてWebブラウザが起動した場合に、対応ページがWebブラウザの全面に表示されてしまう。そこで、本実施の形態においては、システムアプリ11が、XSLTプロセッサ14を用いてメール本文を生成する際に、対応ページのURLが引数情報に含まれるようにURL 501を生成しているのである。

【0070】

次に、イベント通知メール500を受信したユーザが、URL 501にアクセスしようとした場合、すなわち、詳細情報を確認しようとした場合の処理について説明する。図13及び図14は、画像処理装置による詳細情報の提供処理を説明するためのシーケンス図である。

【0071】

ステップS11において、ユーザがURL 501(URL(a))をクリックすると、Webブラウザ21は、画像処理装置10に対してURL(a)に対する情報を要求するHTTPリクエストを送信する。ステップS11に続いてステップS12に進み、Webサーバプログラム15は、URL(a)を所定のものに置換する。

【0072】

図15は、WebサーバプログラムによるURLの置換処理を説明するためのフローチャートである。まず、Webブラウザ21から送信されたURL(a)をロードページに対するURLに置換する(S12a)。ここで、ロードページに対するURLは、以下のものであるとする。

【0073】

web/user/zz/websys/load.cgi

更に、URL(a)に引数情報(?以下の記述)が付加されているか判断し(S12b)、付加されている場合は、新たなURLの最後に引数情報の記述を追加する(S12c)。従って、新たに置換されたURLは以下の通りとなる。

【0074】

web/user/zz/websys/load.cgi?open=websys/status/inTray.cgi&profile=admin&lang=en・・・(b)

即ち、ステップS12の処理においてWebブラウザ21からパス名を含まないURLが指定された場合に、Webサーバプログラム15は、ロードページに対する要求であるとみなしているわけである。これによって、ユーザは、ロードページのURLを入力せずとも、画像処理装置10のホスト名を入力するだけでロードページに対する要求を行うことができる。

【0075】

図13に戻り、ステップS12に続いてステップS13に進み、Webサーバプログラム15は、上記URL(b)のパス名における「websys」の記載に基づいて、当該URLはシステムアプリ11に対する要求であると解釈し、システムアプリ11のページ呼び分

10

20

30

40

50

け手段 1 1 1 を呼び出す。即ち、「websys」の記載は、システムアプリ 1 1 に対する識別子であり、Web アプリ 1 2 又は 1 3 等の他の Web アプリに対する URL については、「websys」が記載された箇所に対する記載は、それぞれの Web アプリに対する識別子（例えば「webfax」等）となる。

【 0 0 7 6 】

ステップ S 1 3 に続いてステップ S 1 4 に進み、呼び出されたページ呼び分け手段 1 1 1 は、URL ( b ) のパス名における「load.cgi」の記載に基づいて、当該 URL はロードページモジュール 1 1 2 に対する要求であると解釈し、ロードページモジュール 1 1 2 を呼び出す。

【 0 0 7 7 】

ステップ S 1 4 に続いてステップ S 1 5 に進み、ロードページモジュール 1 1 2 は、情報の表示に利用する言語（表示言語）の選択を行う。

【 0 0 7 8 】

図 1 6 は、ロードページモジュールによる表示言語の選択処理を説明するための処理である。まず、URL ( b ) を参照して表示言語についての指定があるか否かを判断する ( S 1 5 a )。表示言語についての指定とは、URL ( b ) における引数情報において、「lang=en」の記載が該当する。

【 0 0 7 9 】

表示言語が指定されている場合は、指定されている値を取り出し ( S 1 5 b )、表示言語が指定されていない場合は、Web ブラウザ 2 1 からの HTTP リクエストのヘッダに含まれる Accept-Language の値を取り出す ( S 1 5 c )。なお、URL ( b ) には、「lang=en」の指定があるため、英語を意味する「en」の値が取り出される。

【 0 0 8 0 】

更に、ステップ S 1 5 b 又はステップ S 1 5 c で取り出された値に対応する言語を画像処理装置 1 0 がサポートしているか否かを言語情報テーブルを参照して判断する ( S 1 5 d )。

【 0 0 8 1 】

図 1 7 は、言語情報管理テーブルの構成例を示す図である。図 1 7 の言語情報管理テーブル 1 6 は、「サポート言語」と「オペパネ言語」とをデータ項目として有しており、サポート言語ごとに一つのレコードが登録されるテーブルである。「サポート言語」には、画像処理装置 1 0 がサポートしている言語の識別情報が登録されている。「オペパネ言語」は、当該サポート言語がオペパネ言語であるか否かを示す項目である。即ち、当該サポート言語がオペパネ言語である場合には、「1」が登録される。

【 0 0 8 2 】

ここで、オペパネ言語とは、画像処理装置 1 0 の操作パネル（オペパネ）において、現在表示対象となっている言語である。また、操作パネルとは、画像処理装置 1 0 に設置され、画像処理装置 1 0 のユーザに対して画像処理装置 1 0 を操作するための操作情報を表示するためのパネルである。即ち、画像処理装置 1 0 は、予めインストールされている複数の言語の中から、オペパネ言語をユーザの任意によって選択することが可能である。従って、例えば、ユーザがオペパネ言語を日本語に設定した場合、画像処理装置 1 0 は、操作パネルに日本語で操作情報を表示させる。

【 0 0 8 3 】

図 1 7 より、画像処理装置 1 0 は「ja」と「en」で識別される二つの言語（日本語と英語）をサポートしており、オペパネ言語には、日本語が選択されていることが分かる。

【 0 0 8 4 】

ステップ S 1 5 d において、取得した言語がサポートされていると判断した場合は、取得した言語を表示言語とし ( S 1 5 e )、取得した言語はサポートされていないと判断した場合は、オペパネ言語を表示言語とする ( S 1 5 f )。本実施の形態においては、「en」（英語）はサポートされているため、ステップ S 1 5 e においては「en」（英語）が表示言語として選択される。

10

20

30

40

50

## 【 0 0 8 5 】

ここで、オペパネ言語を表示言語とするのは、以下の理由による。近年では、職場に外国人がいるのはあたりまえのようになっているが、例えば、画像処理装置 1 0 がそのような複数の人種によってメンバが構成されているオフィスのフロアに置かれている場合を想定する。この場合、画像処理装置 1 0 の情報を確認したい人は、そのフロアにいる人達にはほぼ限定されるといえる。即ち、実際に操作パネルによって画像処理装置 1 0 を操作する人達である。また、同一のフロア等、ある程度限定された空間の中で勤務している人であれば、普段は共通の言語によってコミュニケーションをとっている可能性がある。従って、画像処理装置 1 0 の操作パネルに表示される言語（オペパネ言語）は、フロア内の人が共通に理解できる言語で設定される可能性がある。よって、オペパネ言語を表示言語とすることで、ユーザが要求した言語と異なる言語であっても、ユーザが理解できる可能性の高い言語で情報を提供することができるといえる。

10

## 【 0 0 8 6 】

図 1 3 に戻り、ステップ S 1 5 に続いてステップ S 1 6 に進み、ロードページモジュール 1 1 2 は、ロードページを生成する。

## 【 0 0 8 7 】

図 1 8 は、ロードページモジュールによるロードページの生成処理を説明するためのフローチャートである。まず、ロードページのテンプレートをバッファ（プログラム内において確保された記憶領域）に読み込む（S 1 6 a）。

20

## 【 0 0 8 8 】

図 1 9 は、ロードページのテンプレートの定義例を示す図である。図 1 9 のテンプレート 1 5 0 において、記述 1 5 1 は、Web ブラウザ 2 1 がロードページを読みこんだ際に所定の関数（jumpToMainFrame()）を呼び出す旨の定義である。また、記述 1 5 2 は、Web ブラウザ 2 1 がロードページを読み込んだ際に実行する所定の関数（jumpToMainFrame()）の定義である。この関数の定義における記述 1 5 3 は、URL 1 5 4 で示される Web ページを呼び出す旨の定義である。従って、テンプレート 1 5 0 には、ロードページを読み込んだ際に、URL 1 5 4 で示される Web ページを呼び出す旨の定義がされていることになる。

## 【 0 0 8 9 】

ここで、URL 1 5 4 は、実行時に処理される実際の値ではなく、雛形（テンプレート）となっている。即ち、URL 1 5 4 には、「\_\_PROFILE\_\_」、「\_\_LANG\_CODE\_\_」、「\_\_JUMP\_TARGET\_\_」という文字列が含まれているが、これらは実際の値を示しているのではなく、引数情報によって置き換えられるマクロである。

30

## 【 0 0 9 0 】

なお、テンプレート 1 5 0 は、ファイルとして保存しておいてもよいし、ロードページモジュール 1 1 2 のプログラムにおいてハードコーディングしておいてもよい。

## 【 0 0 9 1 】

図 1 8 に戻り、ロードページモジュール 1 1 2 は、URL ( b ) にユーザモードの指定があるか否かを判断する（S 1 6 b）。URL ( b ) においては、「profile=admin」の記載がユーザモードの指定に該当する。ユーザモードの指定がある場合は、指定された値に対応するユーザモードとし（S 1 6 c）、ユーザモードの指定が無い場合は、一般ユーザモード（user）とする（S 1 6 d）。URL ( b ) には、「profile=admin」と指定されているため、「admin」（管理者モード）がユーザモードとなる。

40

## 【 0 0 9 2 】

続いて、決定したユーザモード、及び表示言語をそれぞれテンプレート 1 5 0 の URL 1 5 4 における「\_\_PROFILE\_\_」、「\_\_LANG\_CODE\_\_」の部分に当てはめる（S 1 6 e、S 1 6 f）。更に、URL ( b ) にページ指定があるか否かを確認する（S 1 6 g）。ここで、ページ指定とは、ユーザが目的としているページであり、URL ( b ) において、「open=websys/status/inTray.cgi」の記載が該当する。ページ

50

指定がある場合は、指定されたページ (websys/status/inTray.cgi) をワークフレーム 203 における表示対象とし (S 16 h)、ページ指定がない場合は、トップページをワークフレーム 203 における表示対象とする (S 16 i)。そして、テンプレート 150 の URL 154 における「\_\_JUMP\_\_TARGET\_\_」に URL (b) における引数情報 (「open=websys/status/inTray.cgi&profile=admin&lang=en」をそのまま当てはめ (S 16 j)、ロードページを完成させる。

#### 【0093】

図 20 は、完成したロードページの定義例を示す図である。図 20 のロードページ 181 における URL 1812 は、テンプレート 150 の URL 154 における「\_\_PROFILE\_\_」、「\_\_LANG\_\_CODE\_\_」、「\_\_JUMP\_\_TARGET\_\_」が、それぞれ「admin」、「en」、「open= websys/status/inTray.cgi&profile=admin&lang=en」に置き換えられ、

「/web/admin/en/websys/mainFame.cgi?open= websys/status/inTray.cgi&profile=admin&lang=en」

といった値になっている。これによってロードページ 181 には「/web/admin/en/websys/mainFame.cgi」のパスで特定されるフレームページ 200 を「open= websys/status/inTray.cgi」を引数情報として呼び出すという定義がされたことになる。なお、URL 154 において、フレームページ 200 のパスは「/web/admin/en/websys/」となっているが、このパスによって、フレームページの表示言語は英語であり、かつ管理者モードによることが示されている。即ち、「/web/admin/en/websys/」のうちの「admin」によって、ユーザモードは管理者モードであることが特定され、「en」によって表示言語が英語であることが特定される。

#### 【0094】

図 13 に戻り、ステップ S 16 に続いてステップ S 17 に進み、ロードページモジュール 112 は生成したロードページ 181 を Web サーバプログラム 15 に出力する。ステップ S 17 に続いてステップ S 18 に進み、Web サーバプログラム 15 は、ロードページ 181 を Web ブラウザ 21 に送信する。

#### 【0095】

ステップ S 18 に続いて図 14 のステップ S 19 に進み、Web ブラウザ 21 は、ロードページ 181 の定義に従って、フレームページの送信を要求する HTTP リクエストを画像処理装置 10 に送信する。なお、HTTP リクエストによって要求される URL は、ロードページ 181 の URL 1812 に示されている値に基づき、

「<ホスト名>/web/admin/en/websys/mainFrame.cgi?open=websys/status/inTray.cgi&profile=admin&lang=en」・・・(c)

である。なお、<ホスト名> が先頭に付加されているのは、ステップ S 11 において、ユーザによって画像処理装置 10 のホスト名が Web ブラウザ 21 の URL 入力領域 211 に入力されているため、ロードページ 181 における「/web/admin/en/websys/mainFame.cgi」のパス名は、<ホスト名> からの相対パスとして Web ブラウザ 21 に解釈されるからである。

#### 【0096】

ステップ S 19 に続いてステップ S 20 に進み、Web サーバプログラム 15 は、上記 URL (c) の「websys」の記載に基づいて、当該 URL はシステムアプリ 11 に対する要求であると解釈し、システムアプリ 11 のページ呼び分け手段 111 を呼び出す。ステップ S 20 に続いてステップ S 21 に進み、呼び出されたページ呼び分け手段 111 は、URL (c) の「mainFrame.cgi」の記載に基づいて、当該 URL はフレームページモジュール 113 に対する要求であると解釈し、フレームページモジュール 113 を呼び出す。

#### 【0097】

ステップ S 21 に続いてステップ S 22 に進み、フレームページモジュール 113 は、URL (c) における引数情報 (「open=websys/status/inTray.cgi&profile=admin&lang

=en」)をDOM形式に変換し、その結果をページ呼び分け手段111に出力する(S23)。

【0098】

ステップS23に続いてステップS24に進み、ページ呼び分け手段111は、DOM形式にされた引数情報を更にXML形式に変換する。

【0099】

図21は、XML形式に変換された引数情報の例を示す図である。図21において、記述161、記述162、記述163はそれぞれ、URL(c)の引数情報における「lang=en」、「profile=admin」、「open=websys/status/inTray.cgi」に基づいて出力されたものである。なお、以下において、XML形式にされた引数情報を「引数情報XML」という。

10

【0100】

ステップS24に続いてステップS25に進み、ページ呼び分け手段111は、XSLTプロセッサ14に対し、引数情報XMLを入力情報として引渡し、HTML形式のフレームページ200の生成を要求する。

【0101】

ステップS25に続いてステップS26に進み、XSLTプロセッサ14は、引数情報XMLをフレームページ200の書式情報が定義されたXSLデータ(以下、「フレームページXSL」という。)に適用してXSLT変換を実行し、フレームページ200を生成する。

20

【0102】

図22は、XSLTプロセッサによるフレームページの生成処理を説明するためのフローチャートである。まず、引数情報XMLとフレームページXSLとを読み込む(S26a)。

【0103】

図23は、フレームページXSLの定義例を示す図である。図23において、記述171は、フレームページ200のヘッダフレーム201に対する定義である。また、記述172、記述173は、それぞれメニューフレーム202、ワークフレーム203に対する定義である。

30

【0104】

メニューフレーム202に対する定義(記述172)において、記述1721には、引数情報XMLにopen要素(<open>タグで囲まれた要素。図21では、記述163が該当。)が含まれている場合は、メニューフレーム202には、当該open要素の値(websys/status/inTray.cgi)を引数情報としてmenu.cgiを呼び出すようにフレームページ200を生成する旨の定義がされている。また、記述1722には、引数情報XMLにopen要素が無い場合は、メニューフレーム202には、引数情報の指定はせずにmenu.cgiを呼び出すようにフレームページ200を生成する旨の定義がされている。

【0105】

一方、ワークフレーム203に対する定義(記述173)において、記述1731には、引数情報XMLにopen要素がある場合は、ワークフレーム203には、当該open要素の値によって特定されるWebページをワークフレーム203にロードするようにフレームページ200を生成する旨の定義がされている。また、記述1732には、引数情報XMLにopen要素が無い場合は、ワークフレーム203には、topPage.cgi(トップページ)をロードするようにフレームページ200を生成する旨の定義がされている。

40

【0106】

このように、フレームページXSLには、引数情報に基づいて各フレームに表示させる情報が決定されるように、フレームページ200の書式情報が定義されている。

【0107】

従って、ステップS26b以降の処理は、図23におけるフレームページXSLの定義

50

によって規定される。即ち、引数情報XMLにページ指定があるか(open要素があるか)を判断し、ページ指定がある場合は、ワークフレーム203に対するURL(ロードするWebページのURL)をopen要素の値によって特定されるWebページに対するものとし(S26c)、更に、メニューフレーム202に対するURLには、引数情報としてopen要素の値を付加する(S26e)。一方、引数情報XMLにページ指定が無い場合(open要素が無い場合)は、ワークフレームに呼び出すURLはトップページ(topPage.cgi)とする。その他、フレームページXMLに定義されている処理を実行し、フレームページ200を出力する(S26f)。

#### 【0108】

図24は、生成されたフレームページの定義例を示す図である。図24において、記述251は、ヘッダフレーム201に対する定義であり、header.cgiを呼び出す旨が定義されている。また、記述252は、メニューフレーム202に対する定義であり、「open=websys/status/inTray.cgi」という引数情報を付加して、menu.cgiを呼び出す旨が定義されている。更に記述253は、ワークフレーム183に対する定義であり、inTray.cgi(給紙トレイページ)を呼び出す旨が定義されている。

10

#### 【0109】

図14に戻り、ステップS26に続いてステップS27に進み、XSLTプロセッサ14は、生成したフレームページ200をページ呼び分け手段111に出力する。フレームページ200は、Webサーバプログラム15を経てWebブラウザ21に送信される(S28、S29)。

20

#### 【0110】

以降、Webブラウザ21によってフレームページ200の定義が解釈され、図24の記述251、記述252、及び記述253の定義に従って、ヘッダフレーム201、メニューフレーム202、及びワークフレーム203の表示対象となるWebページがロードされる。

#### 【0111】

ここで、menu.cgiとinTray.cgiのパスは、フレームページからの相対パスとしてWebブラウザ21に解釈される。即ち、上述したように、フレームページ(mainframe.cgi)のURLは、「<ホスト名>/web/admin/en/websys/mainFrame.cgi」であるため、Webブラウザ21は、「<ホスト名>/web/admin/en/websys/menu.cgi」によって特定されるmenu.cgiを呼び出し、また、「<ホスト名>/web/admin/en/websys/inTray.cgi」によって特定されるinTray.cgiを呼び出す。従って、それぞれのパスによって、管理者モードであることと表示言語は英語であることが特定されて、menu.cgiとinTray.cgiが呼び出されることになる。

30

#### 【0112】

それぞれのフレームにそれぞれのWebページが読み込まれると、フレームページ200は、図25に示されるように表示される。

#### 【0113】

図25は、引数情報を指定することによってフレームページに給紙トレイ情報を英語でかつ管理者モードで表示させた例を示す図である。図25におけるフレームページ200の表示は、図8におけるそれと全く同じである。即ち、ワークフレーム203には目的とする給紙トレイページが英語で表示されており、ユーザモードも管理者モードとなっている。

40

#### 【0114】

更に、ワークフレーム203と関連を有するメニューフレーム202についても、ワークフレーム203に表示された情報と整合がとられた態様で表示される。即ち、ワークフレーム203に表示された給紙トレイ情報の位置づけが分かるように、状態(Status)ノード2121が開いた状態が表示されている。状態(Status)ノード2121が展開した態様で表示されるのは、メニューフレーム202において、図24の記述252に従って、引数情報「open=websys/status/inTray.cgi」を付加してmenu.cgiを呼び

50

出しているからである。即ち、呼び出されたmenu.cgiは、「open=websys/status/inTray.cgi」という引数情報によって、ワークフレーム203に表示されているのは、inTray.cgiであることを認識することができ、それに合わせて、状態(STATUS)ノード2121を展開させた態様で、メニューページを提供しているからである。

【0115】

以降、ユーザは、フレームページ200上において、メニューフレーム202に表示された目次情報等を操作することにより、画像処理装置10が提供する種々のサービスを利用することができる。なお、この状態でブックマークしたとすると、ブックマークに登録されるURLは、URL(c)「<ホスト名>/web/admin/en/websys/mainFrame.cgi?open=websys/status/inTray.cgi&profile=admin&lang=en」である。従って、かかるブックマークを次回利用した場合は、図14のステップS19以降の処理が実行され、図25の状態でフレームページ200を表示させることができる。

10

【0116】

上述したように、本実施の形態における画像処理装置10によれば、URLに含まれる引数情報に対応した態様で表示されるようにフレームページを生成し、Webブラウザ21に提供するため、ユーザは、URLの引数情報に目的とするWebページ(給紙トレイページ)のパス名、表示言語、及びユーザモード等を引数情報として指定するのみで、デフォルトで提供される一般ユーザモードでの日本語によるトップページを経ることなく、いきなり目的とするWebページを、引数情報で指定した値に対応した態様で所定のフレーム内に表示させることができる。

20

【0117】

また、本実施の形態においては、サーバ側である画像処理装置10によって目的とする機能が実現されるため、クライアント側であるWebブラウザの種別を問わない。したがって、エンドユーザに所定のペングより提供されているWebブラウザの利用を強制する必要もない。

【0118】

また、特許文献2のように、各フレームに対するURLを保存しておく方式では、少なくとも一度はユーザが手動で目的とする情報を表示させる必要があるが、本実施の形態による方式では、URLに含まれている情報(引数情報)に基づいて、所定のフレームに表示させる情報を判断するため、ユーザは、当該情報を一度も参照したことがなくても、引数情報を適切に指定することで当該情報をフレームページ200上にいきなり表示させることができる。また、各フレームに対するURL等の複雑な情報によって構成されるデータではなく、URLさえ入手できれば目的とする情報を直接表示するフレームページ200を得ることができる。したがって、本実施の形態においても説明したように、画像処理装置10からのイベント通知メール内に当該URLを含めることによって、フレームページ200上において詳細情報を確認させるといった運用も可能となる。

30

【0119】

なお、本実施の形態においては、目的とする情報を表示させるフレームが所定のフレーム(ワークフレーム203)に固定されている例について説明したが、表示対象とするフレームを指定可能なようにしてもよい。例えば、URLの引数情報に「frame1=aaacgi frame2=bbb.cgi frame3=ccc.cgi・・・」と指定された場合は、frame1、frame2、frame3の識別子によって特定されるそれぞれのフレームに、aaa.cgi、bbb.cgi、ccc.cgiによって特定される情報(Webページ)を表示させるようにしてもよい。または、「open=aaacgi fopen=bbb.cgi open=ccc.cgi・・・」と指定された場合は、予めフレーム間に規定されている順序関係と、引数情報において指定された順番と対応させて各フレームに表示させる情報を決定するようにしてもよい。かかる機能は、フレームページXSL(図23)の定義を修正することによって容易に実現することができる。

40

【0120】

また、本実施の形態においては、画像処理装置10からのイベント通知メールのトリガーとして用紙切れが発生した場合を例として説明したが、イベント通知メールのトリガー

50

は、トナー切れやFAXの受信等様々なものが考えられる。例えば、FAXを受信した際にイベント通知メール(FAX受信通知メール)を送信する場合について説明する。

【0121】

図26は、画像処理装置によるFAX受信通知メールの送信処理を説明するためのブロック図である。図26中、図9と同一部分には同一符号を付し、その説明は省略する。図26に示されるように、FAXの受信は、FAX管理サービス1022によって検出される(S101)。FAX管理サービス1022からの通知に基づいて、機器情報レポートアプリ1015は、対応URLをイベント・URL対応テーブル18より取得すると共に(S103)、FAX受信の通知先として登録されているメールアドレスの一覧(FAX通知先一覧)をNVRAM17より取得する(S104)。

10

【0122】

図27は、FAX通知先一覧の例を示す図である。図27に示されるように、FAX通知先一覧171には、FAX受信の通知先となるユーザごとにメールアドレスが示されている。

【0123】

機器情報レポートアプリ1015が、メール本文の生成をFAXWebアプリ12に対して要求すると(S105)、以降はシステムアプリ11が行っていた処理をFAXWebアプリ12が実行することを除き、図9及び図10と同様の処理によってメール本文が生成され(S106、S107)、FAX受信通知メールが送信される(S108、S109)。なお、FAXWebアプリ12は、例えば、図3においてWebアプリ12として表示されているものであり、FAX機能に対するWebページを提供するためのWebアプリケーションである。

20

【0124】

図28は、FAX受信通知メールの例を示す図である。図28に示されるように、FAX受信通知メール510には、FAXを受信した旨等が記載されている。また詳細情報を参照するためのURL511が以下の形式で示されている。

【0125】

<ホスト名>/?open=webfax/list.cgi&profile=user&lang=ja

したがって、ユーザがURL511をクリックすると、図13及び図14において説明した処理と同様の処理によって、FAX受信に関する詳細内容を表示するWebページ(webfax/list.cgi)がワークフレーム203に表示されたフレームページ200が、Webブラウザ21に表示される。

30

【0126】

また、実際の運用において、かかるURL(引数情報に所定のフレームに表示させる情報についての指定がされているURL)のWebブラウザ21に対する入力方法は、イベント通知メールに記載されているURLをクリックすることだけには、限られない。例えば、フレームページ200によって提供される機能と密接に関連した専用アプリケーションのユーザインタフェース(メニューやボタン等)に、かかるURLを引数としてWebブラウザを起動するような命令を組み込んでおいてもよい。この場合、ユーザは、当該専用アプリケーションの「給紙トレイ情報」というメニューを選択すると、給紙トレイ情報がワークフレーム203に表示されたフレームページ200を表示させることができる。

40

【0127】

また、ユーザが、確認したい情報に対するURLを予め知っているのであれば、そのURLを引数情報に指定したURLを、ユーザに直接入力させるようにしてもよい。なお、イベント通知メールに記載されたURL、専用アプリケーションのユーザインタフェースに組み込まれたURL、又はユーザによって直接入力されたURLのいずれに基づいてフレームページ200を表示させた場合も、フレームページ200が表示された状態をブックマークに登録しておけば、次回以降は当該ブックマークを選択することによって同じ状態を再現させることができる。

【0128】

50

また、本発明が適用可能な機器は、画像処理装置だけに限られない。一般的なコンピュータ等の情報処理装置はもちろん、通信機能を有していれば、本実施の形態における画像処理装置と同様に記憶容量の制約が厳しいデジタルカメラや携帯電話、いわゆるデジタル家電等にも本発明を適用することができる。

【0129】

以上、本発明の好ましい実施例について詳述したが、本発明は係る特定の実施形態に限定されるものではなく、特許請求の範囲に記載された本発明の要旨の範囲内において、種々の変形・変更が可能である。

【図面の簡単な説明】

【0130】

【図1】フレームに分割されたWebページの表示例を示す図である。

【図2】本発明の実施の形態における画像処理装置の機能構成例を示す図である。

【図3】Webサーバとしての画像処理装置の機能構成例を示す図である。

【図4】ロードページモジュールを説明するための図である。

【図5】フレームページモジュールを説明するための図である。

【図6】XSLTプロセッサを説明するための図である。

【図7】トップページを表示したフレームページの表示例を示す図である。

【図8】給紙トレイ情報を英語でかつ管理者モードで表示したフレームページの表示例を示す図である。

【図9】画像処理装置によるイベント通知メールの送信処理を説明するためのブロック図である。

【図10】画像処理装置によるイベント通知メールの送信処理を説明するためのフローチャートである。

【図11】イベント・URL対応テーブルの構成例を示す図である。

【図12】用紙切れを通知するイベント通知メールの例を示す図である。

【図13】画像処理装置による詳細情報の提供処理を説明するためのシーケンス図である。

【図14】画像処理装置による詳細情報の提供処理を説明するためのシーケンス図である。

【図15】WebサーバプログラムによるURLの置換処理を説明するためのフローチャートである。

【図16】ロードページモジュールによる表示言語の選択処理を説明するための処理である。

【図17】言語情報管理テーブルの構成例を示す図である。

【図18】ロードページモジュールによるロードページの生成処理を説明するためのフローチャートである。

【図19】ロードページのテンプレートの定義例を示す図である。

【図20】完成したロードページの定義例を示す図である。

【図21】XML形式に変換された引数情報の例を示す図である。

【図22】XSLTプロセッサによるフレームページの生成処理を説明するためのフローチャートである。

【図23】フレームページXSLの定義例を示す図である。

【図24】生成されたフレームページの定義例を示す図である。

【図25】引数情報を指定することによってフレームページに給紙トレイ情報を英語でかつ管理者モードで表示させた例を示す図である。

【図26】画像処理装置によるFAX受信通知メールの送信処理を説明するためのブロック図である。

【図27】FAX通知先一覧の例を示す図である。

【図28】FAX受信通知メールの例を示す図である。

【符号の説明】

10

20

30

40

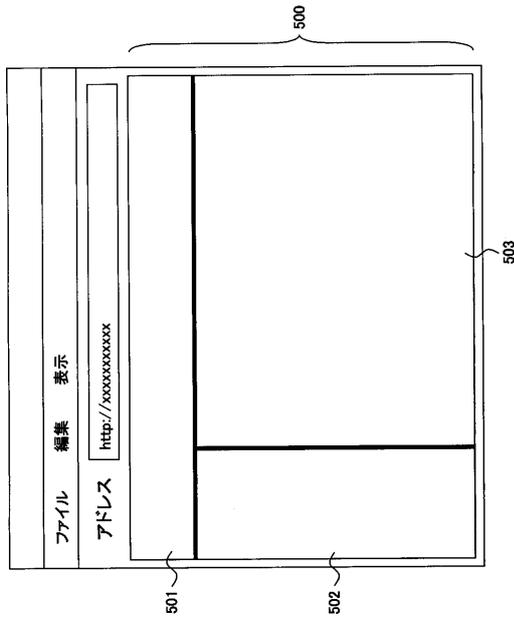
50

## 【 0 1 3 1 】

1 0	画像処理装置	
1 1	システムアプリ	
1 2、1 3	Webアプリ	
1 4	X S L Tプロセッサ	
1 5	Webサーバプログラム	
1 6	言語情報管理テーブル	
1 7	N V R A M	
1 8	イベント・U R L対応テーブル	
2 0	端末	10
2 1	Webブラウザ	
1 1 1	ページ呼び分け手段	
1 1 2	ロードページモジュール	
1 1 3	フレームページモジュール	
1 1 4	ライブラリ	
1 4 1	入力モジュール	
1 4 2	出力モジュール	
1 4 5	X M Lデータ	
1 4 6	X S Lデータ	
1 4 7	H T M Lデータ	20
1 0 1 1	コピーアプリ	
1 0 1 2	ファックスアプリ	
1 0 1 3	プリンタアプリ	
1 0 1 4	情報提供アプリ	
1 0 1 5	機器情報レポートアプリ	
1 0 1 6	システム管理サービス	
1 0 1 7	ネットワークサービス	
1 0 1 8	メモリ管理サービス	
1 0 1 9	オペパネ管理サービス	
1 0 2 0	エンジン管理サービス	30
1 0 2 1	配信サービス	
1 0 2 2	F A X管理サービス	
1 0 2 3	O S	
1 0 2 4	エンジン制御ボード	
1 0 2 5	プロッタエンジン	
1 0 2 6	スキャナエンジン	

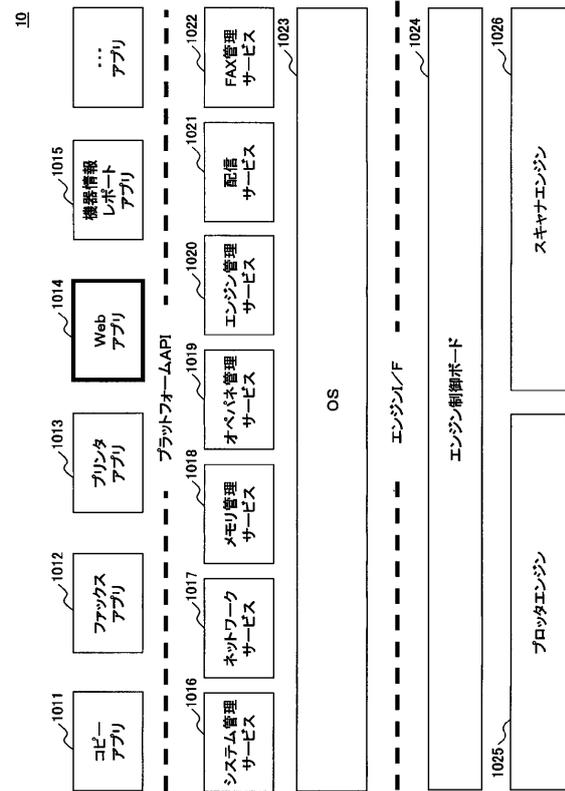
【図1】

フレームに分割されたWebページの表示例を示す図



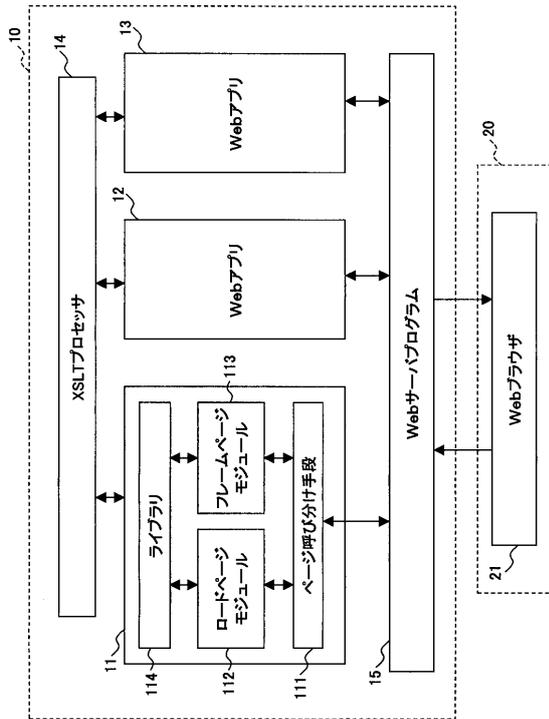
【図2】

本発明の実施の形態における画像処理装置の機能構成例を示す図



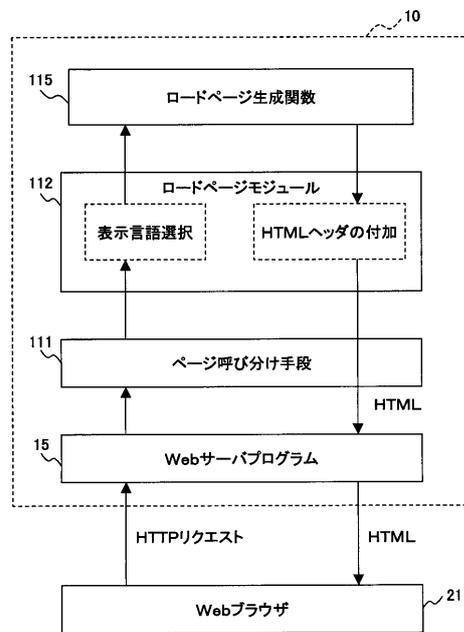
【図3】

Webサーバとしての画像処理装置の機能構成例を示す図



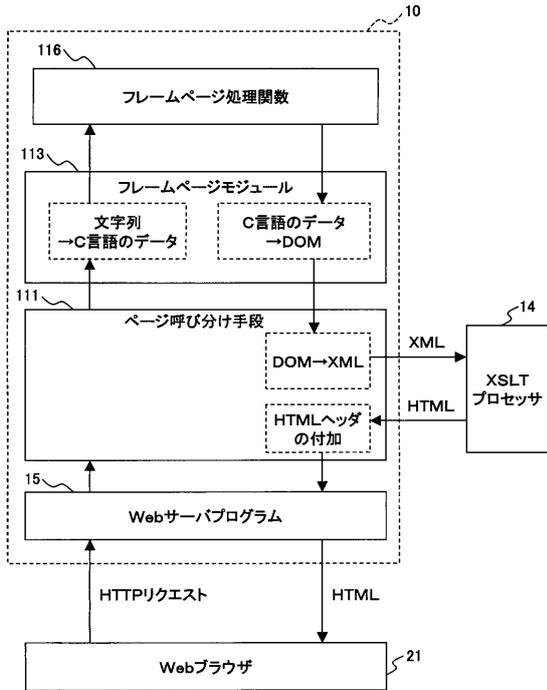
【図4】

ロードページモジュールを説明するための図



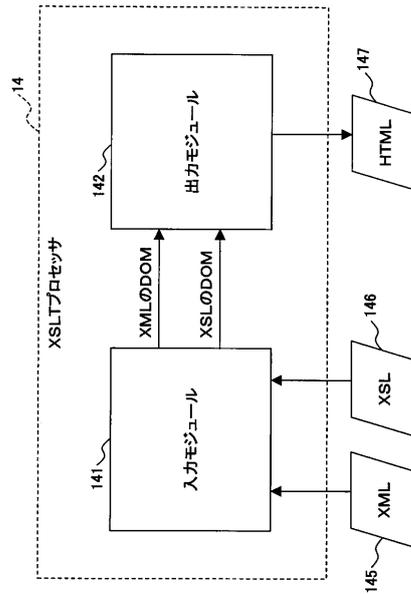
【図5】

フレームページモジュールを説明するための図



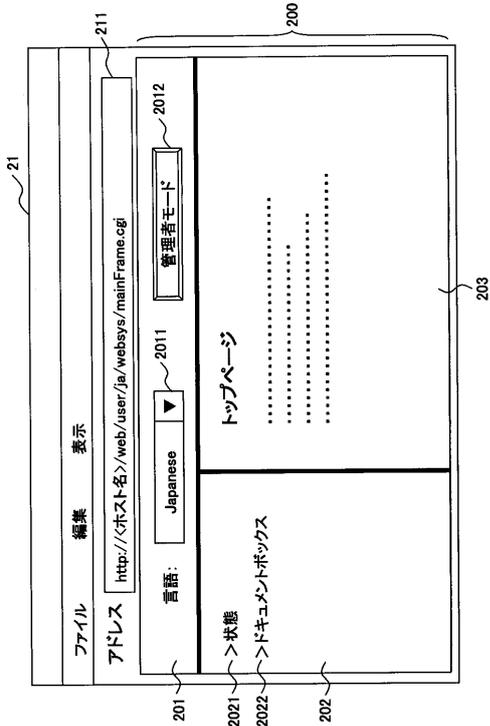
【図6】

XSLTプロセッサを説明するための図



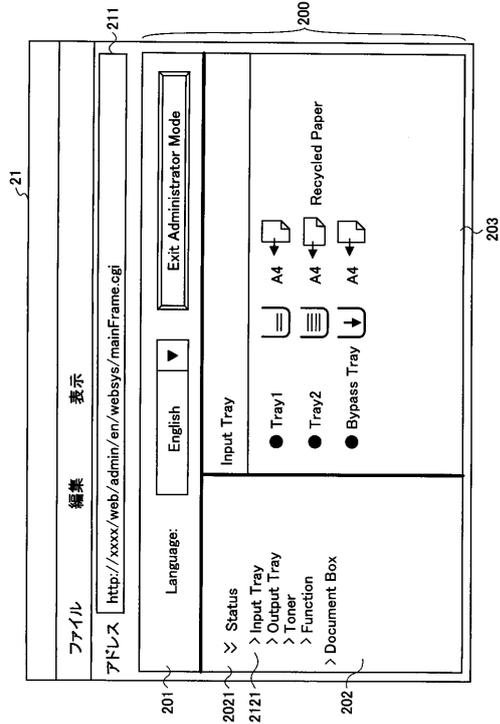
【図7】

トップページを表示したフレームページの表示例を示す図



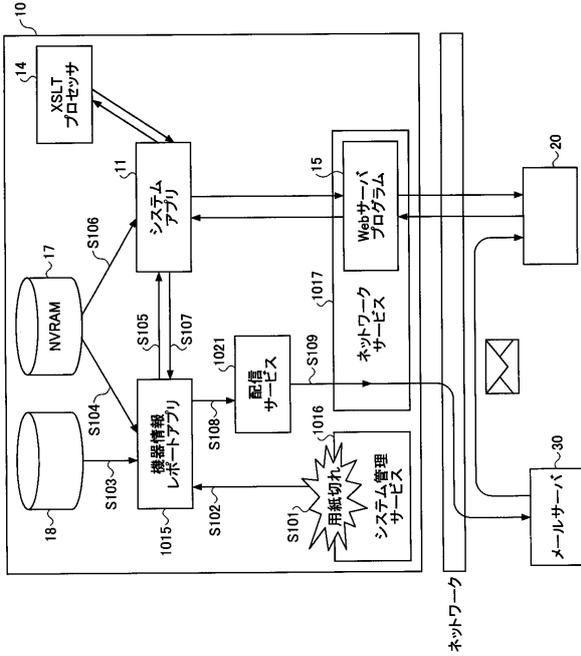
【図8】

給紙トレイ情報を英語でかつ管理者モードで表示したフレームページの表示例を示す図



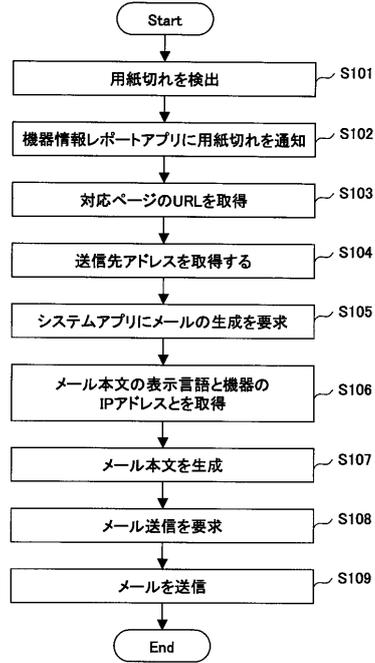
【図9】

画像処理装置によるイベント通知メールの送信処理を説明するためのブロック図



【図10】

画像処理装置によるイベント通知メールの送信処理を説明するためのフローチャート



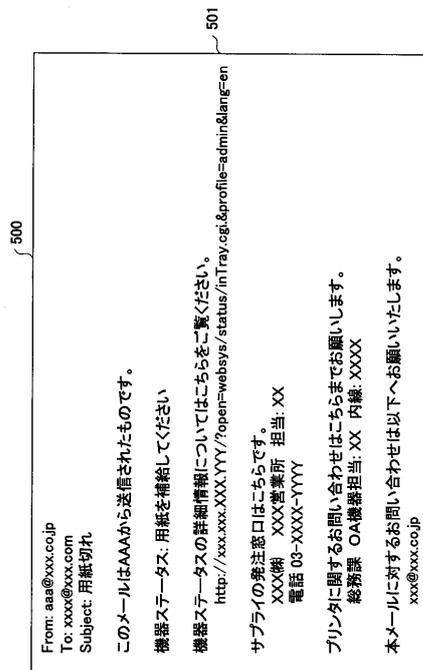
【図11】

イベント・URL対応テーブルの構成例を示す図

イベント	URL
用紙切れ	websys/status/inTray.cgi
トナー残りわずか	websys/status/toner.cgi
IPアドレス競合	websys/network/TCP/IP.cgi
FAX受信	webfax/faxlist.cig

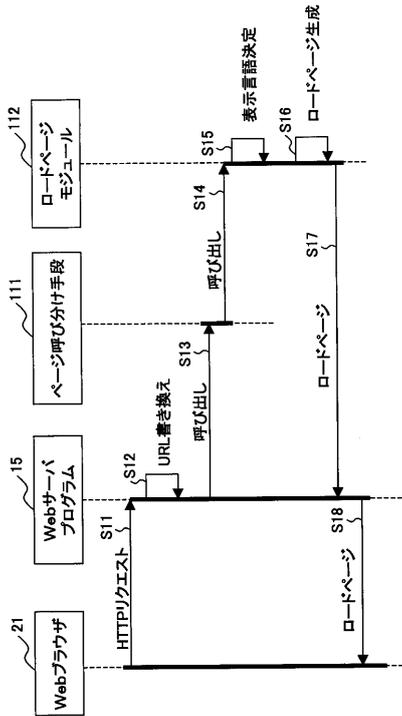
【図12】

用紙切れを通知するイベント通知メールの例を示す図



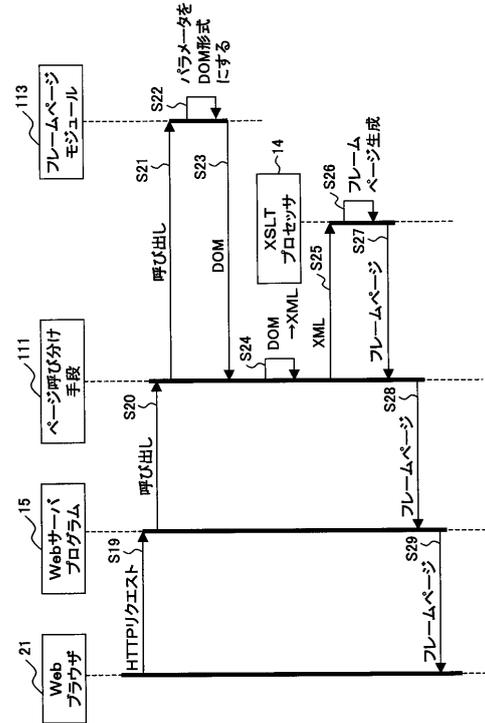
【図13】

画像処理装置による詳細情報の提供処理を説明するためのシーケンス図



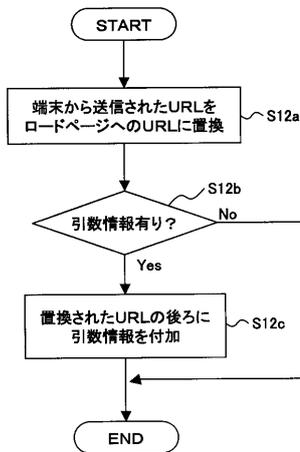
【図14】

画像処理装置による詳細情報の提供処理を説明するためのシーケンス図



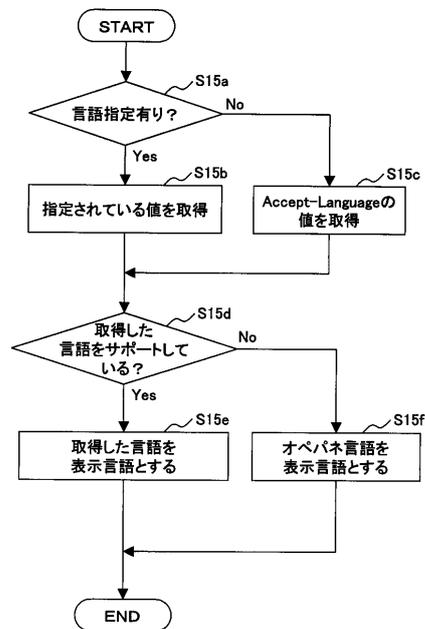
【図15】

WebサーバプログラムによるURLの置換処理を説明するためのフローチャート



【図16】

ロードページモジュールによる表示言語の選択処理を説明するための処理



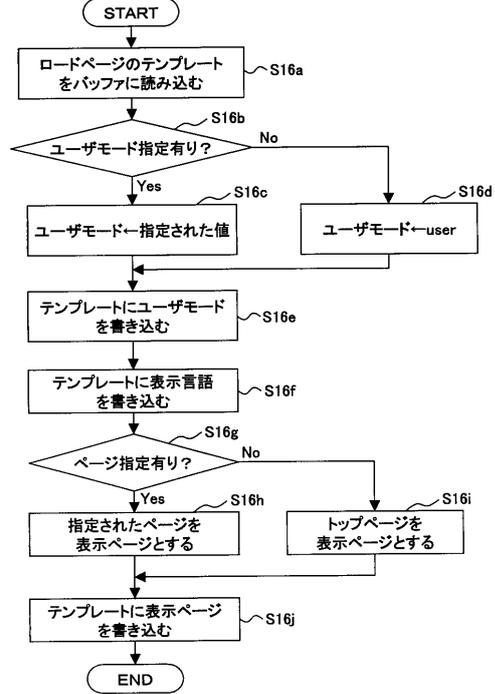
【図17】

言語情報管理テーブルの構成例を示す図

サポート言語	オペバネ言語
ja	1
en	

【図18】

ロードページモジュールによるロードページの生成処理を説明するためのフローチャート



【図19】

ロードページのテンプレートの定義例を示す図

```

    150
    <html>
    <head>
    <title>LoadPage</title>
    <script language="javascript">
    153
    function jumpToMainFrame() {
    self.document.cookie="cookieOnOffChecker=on: path=";
    self.location="/web/_PROFILE/_LANG_CODE/_websys/mainFrame.cgi?_JUMP_TARGET=";
    152
    }
    // -->
    </script>
    </head>
    <body onLoad="jumpToMainFrame()">
    151
    </body>
    </html>
  
```

【図20】

完成したロードページの定義例を示す図

```

    181
    <html>
    <head>
    <title>LoadPage</title>
    <script language="javascript">
    182
    function jumpToMainFrame() {
    self.document.cookie="cookieOnOffChecker=on: path=";
    self.location="/web/admin/en/websys/mainFrame.cgi/open?websys/status/inTray.cgi&profile=admin&lang=en";
    181
    }
    // -->
    </script>
    </head>
    <body onLoad="jumpToMainFrame()">
    </body>
    </html>
  
```

【図 2 1】

XML形式に変換された引数情報の例を示す図

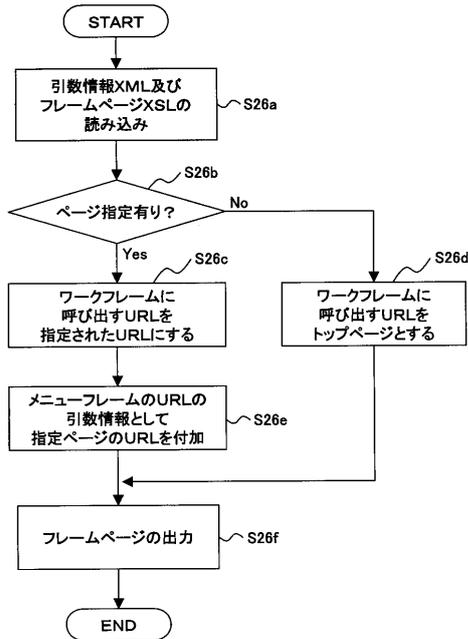
```

<?xml version="1.0" encoding="UTF-8" ?>
<Response>
  161 <returnValue>SUCCESS</returnValue>
  162 <lang>en</lang>
  163 <profile>admin</profile>
  <open>websys/status/inTray.cgi</open>
</Response>

```

【図 2 2】

XSLTプロセッサによるフレームページの生成処理を説明するためのフローチャート



【図 2 3】

フレームページXSLの定義例を示す図

```

<?xml version="1.0" encoding="UTF-8" ?>
<xsl:stylesheet version="1.0" .....>
  .....
  <frameset rows="40,*" .....>
  171 <frame noresize="noresize" src="header.cgi" name="header" ..... />
  <frameset cols="160,*" .....>
  <frame name="menu" .....>
  <xsl:choose>
  1721 <xsl:when test="open!=''">
  <xsl:attribute name="src">
  <xsl:value-of select="concat('menu.cgi?open=',open)"/>
  </xsl:attribute>
  </xsl:when>
  1722 <xsl:otherwise>
  <xsl:attribute name="src">
  <xsl:value-of select="'menu.cgi'"/>
  </xsl:attribute>
  </xsl:otherwise>
  </xsl:choose>
  </frame>
  <frame name="work" .....>
  <xsl:choose>
  1731 <xsl:when test="open!=''">
  <xsl:attribute name="src">
  <xsl:value-of select="concat('..',open)"/>
  </xsl:attribute>
  </xsl:when>
  1732 <xsl:otherwise>
  <xsl:attribute name="src">
  <xsl:value-of select="'topPage.cgi'"/>
  </xsl:attribute>
  </xsl:otherwise>
  </xsl:choose>
  </frame>
  </frameset>
  .....
</xsl:stylesheet>

```

【図 2 4】

生成されたフレームページの定義例を示す図

```

<!DOCTYPE html PUBLIC "....."
<html lang="en">
<head>
  .....
  </head>
  251 <frameset rows="40,*" .....>
  <frame noresize src="header.cgi" name="header" .....>
  252 <frameset cols="160,*" .....>
  <frame name="menu" src="menu.cgi?open=websys/status/inTray.cgi" .....>
  253 <frame name="work" src="..../websys/status/inTray.cgi" .....>
  </frameset>
  </frameset>
</html>

```



---

フロントページの続き

(51)Int.Cl.

F I

G 0 6 F 17/30 1 1 0 F

(56)参考文献 特開2003-050765(JP,A)

繰上 敬子,Webデザイン・テクニック(サイトプランニング編),DREAMWEAVER  
& FIREWORKS Webデザイン・テクニック for Windows,日本,(株)  
毎日コミュニケーションズ,2001年 6月19日,127頁

(58)調査した分野(Int.Cl.,DB名)

G 0 6 F 13/00

G 0 6 F 3/14

G 0 6 F 12/00

G 0 6 F 17/30