

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5800000号
(P5800000)

(45) 発行日 平成27年10月28日(2015.10.28)

(24) 登録日 平成27年9月4日(2015.9.4)

(51) Int.Cl. F I
G06F 17/22 (2006.01) G O 6 F 17/22 6 6 4
 G O 6 F 17/22 6 4 7

請求項の数 6 (全 18 頁)

(21) 出願番号	特願2013-207165 (P2013-207165)	(73) 特許権者	000002369
(22) 出願日	平成25年10月2日(2013.10.2)		セイコーエプソン株式会社
(62) 分割の表示	特願2008-269681 (P2008-269681) の分割		東京都新宿区西新宿2丁目4番1号
原出願日	平成20年10月20日(2008.10.20)	(74) 代理人	100094525
(65) 公開番号	特開2014-38643 (P2014-38643A)		弁理士 土井 健二
(43) 公開日	平成26年2月27日(2014.2.27)	(74) 代理人	100094514
審査請求日	平成25年10月31日(2013.10.31)		弁理士 林 恒徳
前置審査		(72) 発明者	中村 英雄
			長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内
		(72) 発明者	大野 省司
			長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

最終頁に続く

(54) 【発明の名称】 制御方法

(57) 【特許請求の範囲】

【請求項1】

端末で、アプリケーションの出力データ、印刷もしくは表示機能を有する複数のデバイスのうち出力対象となるデバイスの優先順位を指定する情報、および、前記出力対象となるデバイスに対応した情報変換を行う情報変換規則である出力用のスタイルシートを指定する情報、を含む第1文書を送信し、

制御装置で、前記第1文書を取得し、当該第1文書に含まれた前記優先順位に基づいた前記出力対象のデバイスについて、取得した前記第1文書を、前記第1文書で指定された前記出力用のスタイルシートに基づいて、前記出力対象となるデバイスのデバイス制御情報を含む第2文書に変換し、前記第2文書を送信し、

前記出力対象となるデバイスで、前記第2文書を取得し、前記第2文書に基づく印刷もしくは表示を行う

ことを特徴とする制御方法。

【請求項2】

前記制御装置で、前記第2文書を、前記出力対象となるデバイスに対応するデバイス制御サービスに出力する請求項1に記載の制御方法。

【請求項3】

前記出力対象となるデバイスに対応するデバイス制御サービスは、当該出力対象となるデバイスとは別体で存在するサービス、および当該対象となるデバイス内に存在するサービスを含む請求項2に記載の制御方法。

【請求項 4】

前記第 1 文書は、前記出力対象となるデバイスへの配信結果を前記アプリケーションに
入力するための入力用のスタイルシートを指定する情報をさらに含んでおり、

前記制御装置で、前記出力対象となるデバイスへの配信結果を含むマークアップ文書で
ある第 3 文書を、前記入力用のスタイルシートに基づいて、前記アプリケーション用のマ
ークアップ文書である第 4 文書に変換し、前記第 4 文書を前記端末に入力し、

前記アプリケーションは、前記第 4 文書を、取得する請求項 2 または 3 に記載の制御方
法。

【請求項 5】

前記制御装置で、複数の前記アプリケーションから、前記出力データを取得し、前記複
数のアプリケーションに対応した複数種類の前記入力用のスタイルシートを用いる請求項
4 に記載の制御方法。

10

【請求項 6】

前記複数のデバイスには、解釈可能な記述言語が異なる複数種類のデバイスが含まれ、
前記制御装置で、前記第 1 文書を、記述言語の異なる前記第 2 文書に変換するための複
数種類の前記出力用のスタイルシートを用いる請求項 1 ないし 5 のいずれか 1 項に記載の
制御方法。

【発明の詳細な説明】

【技術分野】

【0001】

20

本発明は、アプリケーションから取得した情報に基づいてデバイスを制御するデバイス
制御システム、デバイス制御システムのサービス実現方法およびそのプログラムに関する
ものである。

【背景技術】

【0002】

従来、コンピュータに接続されたプリンタ、ディスプレイ、カードリーダー等の複数種類
のデバイスを制御する場合、コンピュータにアプリケーション・プログラム（以下、単に
「アプリケーション」と称する。）をインストールして、各デバイスを制御する。この場
合、例えばこれら複数種類のデバイスに、一連の処理を実行させようとする、第 1 のデ
バイスに動作を指令してそれに対するイベント情報を受け取り、次に第 2 のデバイスに動
作を指令してそれに対するイベント情報を受け取る・・・といったデバイス毎の処理をア
プリケーションに書き込む必要があり、アプリケーションの作成に手間がかかる。そこで
、アプリケーションと複数のデバイスとの間に管理モジュールを介在させ、アプリケー
ションからは、デバイス名と処理内容（処理識別子）を管理モジュールに通知し、管理モ
ジュールは当該通知内容をテーブルに書き込んで、各デバイスに対し、一連の処理を実行さ
せるようにしたシステムが知られている（例えば、特許文献 1）。この構成によれば、ア
プリケーションに、一連の処理を実行させるための処理内容を書き込む必要が無くなるた
め、作成が容易となる。また、デバイスの増減があった場合でも、アプリケーションの変
更を最小限に留めることができる、などの利点がある。

30

【先行技術文献】

40

【特許文献】

【0003】

【特許文献 1】特開 2000 - 242594 号公報

【発明の概要】

【発明が解決しようとする課題】

【0004】

ところが、上記特許文献 1 に記載のシステムでは、プリンタに印刷する印刷内容、レイ
アウトまたは書式などの修正が必要となった場合、アプリケーションの変更が必要となる
。また、デバイスの入れ替えによって機種が変更となった場合も、その制御情報（コマ
ンド）を変更しなければならないため、アプリケーションの変更が必要となる。このよう

50

、上記特許文献1に記載のシステムは、処理内容の変更や機種の違いを、管理モジュールによって吸収できる構成とはなっていない。

【0005】

本発明は、上記の問題点に鑑み、処理内容の変更、並びにデバイスの種類や仕様の変更に伴うアプリケーションの変更を不要とすることができるデバイス制御システム、デバイス制御システムのサービス実現方法およびそのプログラムを提供することを目的とする。

【課題を解決するための手段】

【0006】

本発明のデバイス制御システムは、アプリケーションから情報を取得して、複数のデバイスを統合的に制御するデバイス制御システムであって、マークアップ文書の情報変換規則を示す複数のスタイルシートを記憶するスタイルシートデータベースと、アプリケーションの出力データ、並びに出力対象となる対象デバイスおよび出力用のスタイルシートを指定する情報を含むマークアップ文書である第1文書を取得する通信インターフェースと、通信インターフェースから取得した第1文書を、出力用のスタイルシートに基づいて、対象デバイスのデバイス制御情報を含むマークアップ文書である第2文書に変換するマークアップ文書変換部と、を備えたことを特徴とする。

10

【0007】

本発明のデバイス制御システムのサービス実現方法は、マークアップ文書の変換規則を示す複数のスタイルシートを記憶するスタイルシートデータベースを管理すると共に、アプリケーションから情報を取得して、複数のデバイスを統合制御するデバイス制御システムのサービス実現方法であって、デバイス制御システムが、アプリケーションの出力データ、並びに当該出力データの出力対象となる対象デバイスおよび出力用のスタイルシートを指定する情報を含むマークアップ文書である第1文書を取得するステップと、スタイルシートデータベースから出力用のスタイルシートを抽出し、当該出力用のスタイルシートに基づいて、第1文書を、対象デバイスのデバイス制御情報を含むマークアップ文書である第2文書に変換するステップと、第2文書を、対象デバイスに向けて出力するステップと、を実行することを特徴とする。

20

【0008】

これらの構成によれば、スタイルシートに基づいて、アプリケーションの出力データ(第1文書)を、出力対象(制御対象)となる対象デバイスのデバイス制御情報を含む第2文書に変換するため、処理内容が変更となった場合や、デバイスの入れ替えがあった場合(種類や仕様の異なるデバイスに入れ替えられた場合)でも、アプリケーションの変更を不要とすることができる。つまり、スタイルシートによって、処理内容の変更、並びにデバイスの種類や仕様の違いを吸収できるため、アプリケーションによる直接的なデバイス制御を不要とすることができる。これにより、アプリケーションの作成が容易になると共に、処理内容の変更やデバイスの入れ替えに伴うアプリケーションの変更処理に要する手間やコストを軽減できる。また、マークアップ文書の記述言語として、XML(Extensible Markup Language)が知られているが、これを用いることで、記述内容が機能を表すと共に特定の言語や文化に依存しないため設計が容易である、文書構造の曖昧さを排除できる、などの利点がある。

30

40

【0009】

なお、「第1文書」として、「出力データ」、「出力対象となる対象デバイスを指定する情報」および「出力用のスタイルシートを指定する情報」を、必ずしも同時に取得する必要はない。例えば、「出力データ」と、「出力対象となる対象デバイスを指定する情報」および「出力用のスタイルシートを指定する情報」とを、別のタイミングで取得し、両者を関連付ける情報に基づいて、第2文書への変換処理を行っても良い。

【0010】

上記に記載のデバイス制御システムにおいて、マークアップ文書変換部から取得した第2文書を、対象デバイスに対応するデバイス制御サービスに出力するデバイス別マークアップ文書入出力部をさらに備えたことを特徴とする。

50

【0011】

上記に記載のデバイス制御システムにおいて、対象デバイスに対応するデバイス制御サービスには、当該対象デバイスとは別体で存在するサービス、および当該対象デバイス内に存在するサービスが含まれていることを特徴とする。

【0012】

これらの構成によれば、デバイス別マークアップ文書入出力部により、「対象デバイスとは別体で存在するサービス」、および「対象デバイス内に存在するサービス（サービス化されたデバイス）」を呼び出すことが可能であり、言い換えれば、これらのデバイス制御サービスを同様に扱うことができる。これにより、例えば既存のデバイスドライバ（対象デバイスとは別体で存在するサービス）を利用する既存デバイスが、サービス化された新デバイスに入れ替えられたような場合でも、アプリケーションの変更を不要とすることができる。

10

【0013】

上記に記載のデバイス制御システムにおいて、第1文書は、対象デバイスへの配信結果をアプリケーションに入力するための入力用のスタイルシートを指定する情報をさらに含んでおり、デバイス別マークアップ文書入出力部は、各対象デバイスへの配信結果を含むマークアップ文書である第3文書を、マークアップ文書変換部に入力し、マークアップ文書変換部は、第3文書を、入力用のスタイルシートに基づいて、アプリケーション用のマークアップ文書である第4文書に変換し、通信インターフェースは、第4文書を、アプリケーションに入力することを特徴とする。

20

【0014】

上記に記載のデバイス制御システムにおいて、通信インターフェースは、複数のアプリケーションから、出力データを取得可能であり、スタイルシートデータベースは、複数のアプリケーションに対応した複数種類の入力用のスタイルシートを記憶していることを特徴とする。

【0015】

これらの構成によれば、アプリケーションは、デバイスへの配信結果を示す第4文書を取得することができる。また、デバイス制御システムが、複数のアプリケーションと提携している場合でも、アプリケーション毎に入力用のスタイルシートを指定できるため、各アプリケーションに応じた第4文書を入力することができる。

30

【0016】

なお、「配信結果」とは、「成否情報」、「補足情報」および「応答情報」を含む概念である。ここで、「補足情報」とは、デバイスのステータス等を示す情報を指し、「応答情報」とは、デバイスによる入力データを指す。

【0017】

上記に記載のデバイス制御システムにおいて、スタイルシートデータベースは、第1文書を、記述言語の異なる第2文書に変換するための複数種類の出力用のスタイルシートを記憶しており、複数のデバイスには、解釈可能な記述言語が異なる複数種類のデバイスが含まれていることを特徴とする。

【0018】

この構成によれば、1のデバイス制御システムで、解釈可能な記述言語が異なる複数種類のデバイスを制御することができる。これにより、例えばプリンタの機能をディスプレイに実現させる（印刷出力により情報提供していたが、これを画面表示によって実現できるようにした場合など）といったことも、アプリケーションの変更を必要とすることなくスタイルシートの書き換えだけで可能となり、使用環境に応じて、デバイスの入れ替えを柔軟に行うことができる。

40

【0019】

本発明のプログラムは、コンピュータに、上記に記載のデバイス制御システムのサービス実現方法における各ステップを実行させるためのものであることを特徴とする。

【0020】

50

このプログラムを用いることにより、処理内容の変更、並びにデバイスの種類や仕様の変更に伴うアプリケーションの変更を不要とすることができるデバイス制御システムのサービスを実現することができる。

【図面の簡単な説明】

【0021】

【図1】店内管理システムのシステム構成図である。

【図2】サーバーの機能ブロック図である。

【図3】「配信情報」および「配信結果」の説明図である。

【図4】店内管理システムの情報配信処理を示すフローチャートである。

【図5】店内管理システムの応答処理を示すフローチャートである。

【図6】XML文書1（出力データ）の具体例を示す図である。

【図7】XML文書1（配信情報）の具体例を示す図である。

【図8】出力スタイルシートおよびXML文書2の具体例を示す図である。

【図9】XML文書4（配信結果）の具体例を示す図である。

【図10】サーバーの代替処理を示すフローチャートである。

【図11】予備出力スタイルシートおよびXML文書2の具体例を示す図である。

【図12】出力スタイルシートおよびXML文書2の具体例を示す図である。

【図13】予備出力スタイルシートおよびXML文書2の具体例を示す図である。

【発明を実施するための形態】

【0022】

以下、本発明の一実施形態に係るデバイス制御システム、デバイス制御システムのサービス実現方法およびそのプログラムについて、添付図面を参照しながら詳細に説明する。本発明は、アプリケーションから取得した情報を、個々のデバイスに好適な情報に変換して配信することにより、各デバイスを統合的に制御することを特徴とするものである。そこで、本発明のデバイス制御システム等を、レストラン等の飲食店における店内管理システム（オーダリング端末から取得した注文情報に基づいて、キッチンプリンタや厨房機器等の各種デバイスを制御するシステム）に適用した場合を例示する。

【0023】

図1は、本実施形態に係る店内管理システムSYのシステム構成図である。店内管理システムSYは、顧客からの注文を入力するオーダリング端末10と、オーダーターミナルとして機能するコンピュータ20（以下、「PC」と称する。）と、統合デバイス制御サービス44が組み込まれたサーバー30（デバイス制御システム）と、キッチンプリンタ61や厨房機器62等のデバイス50と、から成る。また、PC20、サーバー30およびデバイス50は、イントラネット等のネットワークNTを介して接続されている。

【0024】

PC20は、CPU（Central Processing Unit）21、ROM（Read Only Memory）22、RAM（Random Access Memory）23、入力デバイス24、通信インターフェース25、ディスプレイ26およびHDD（Hard Disk Drive）27を備えている。通信インターフェース25は、オーダリング端末10と無線通信を行うためのものであり、本実施形態では主に注文情報の受信に用いられる。

【0025】

また、HDD27には、アプリケーション・プログラム41（以下、単に「アプリケーション」と称する。）と、スタイルシート生成プログラム42と、が記憶されている。アプリケーション41は、情報配信対象となるデバイス50（受信者）に対し、「送信者」として機能する。また、スタイルシート生成プログラム42は、サーバー30において、各デバイス50に対応した情報変換を行うための情報変換規則であるスタイルシートを生成するためのプログラムである。

【0026】

一方、サーバー30も、PC20と同様に、一般的なコンピュータの構成を有しており、CPU31、ROM32、RAM33、入力デバイス34、ディスプレイ35および記

10

20

30

40

50

憶装置 36 を備えている。記憶装置 36 には、統合デバイス制御サービス 44 と、デバイスサービス登録プログラム 46 と、が記憶されていると共に、データベース 45 を有している。当該データベース 45 には、上記のスタイルシートや、各デバイス 50 に対応するデバイス制御サービス 73, 75 (図 2 参照) に関する情報が登録される。

【 0 0 2 7 】

統合デバイス制御サービス 44 は、本実施形態の骨子を成す構成要素であり、PC 20 (アプリケーション 41) から取得した情報を、データベース 45 から読み出したスタイルシートを用いて、各デバイス 50 に好適な情報に変換し、各デバイス 50 に対応したデバイス制御サービス 73, 75 に出力する。また、デバイスサービス登録プログラム 46 は、ネットワーク NT 上に存在するデバイス制御サービス 73, 75 の存在を統合デバイス制御サービス 44 が認識できるように、予めデータベース 45 に登録しておくためのプログラムである。

10

【 0 0 2 8 】

一方、デバイス 50 としては、POS ターミナル 51、および当該 POS ターミナル 51 に接続されたキッチンプリンタ 61、厨房機器用のプロトコルコンバータ 52、および当該プロトコルコンバータ 52 に接続された厨房機器 62 の他、ネットワーク NT への接続機能を有する各種デバイス (キッチンディスプレイ 63、照明機器 64、空調機器 65、映像・音響機器 66、テーブルディスプレイ 67、クーポン発券機 68) を有している。なお、POS ターミナル 51 に接続されたキッチンプリンタ 61 や、プロトコルコンバータ 52 に接続された厨房機器 62 など、直接ネットワーク接続できないローカルデバイスの制御においては、POS ターミナル 51 やプロトコルコンバータ 52 が、アプリケーション 41 から配信された情報を受信する「受信者」として機能する。

20

【 0 0 2 9 】

このように、店内管理システム SY では、オーダリング端末 10 から出力された注文情報に応じて、上記のデバイス 50 を統合的に制御できるようになっている。具体的には、注文情報にグリル部門のオーダーが含まれる場合、厨房のグリル部門に設置されたキッチンプリンタ 61 に、調理指示伝票を発行させる。また、注文情報にドリンク部門のオーダーが含まれる場合、ドリンク部門に設置された別のキッチンプリンタ 61 に、調理指示伝票を発行させる。また、キッチンディスプレイ 63 には、注文された料理のレシピ情報を表示させ、照明機器 64 には、注文情報に応じた機器 (間接照明、スポットライトなど) を選択して注文情報に応じた照度で点灯させる。また、空調機器 65 には、注文情報に応じた温度や湿度に調整するための指令を出し、映像・音響機器 66 には、注文情報に応じた映像・音楽を注文情報に応じた画質・音質で映像・音声出力させる。さらに、オーダーした顧客が着席するテーブルのテーブルディスプレイ 67 には、注文内容、オーダーされた料理の原材料情報、カロリー、アレルギー情報等を表示させ、クーポン発券機 68 には、注文情報に応じたクーポン券を発行させる。このように、店内管理システム SY では、顧客の 1 回の注文に応じて、店内に配置された複数種類のデバイス 50 の制御が可能である。以下、その実現方法について、詳細に説明する。

30

【 0 0 3 0 】

図 2 は、サーバー 30 の具体的な機能を示すブロック図である。サーバー 30 は、上記の統合デバイス制御サービス 44 およびデータベース 45 の他、スタイルシート管理部 71、デバイス制御サービス管理部 72 および既存デバイス 50 a 用のデバイス制御サービス 73 を備えている。

40

【 0 0 3 1 】

スタイルシート管理部 71 は、スタイルシートの管理 (登録、削除、更新など) を行う。また、スタイルシートを簡単に生成できるようにするためのスタイルシート生成ツール (図示省略) を、PC 20 に提供する。スタイルシート生成ツールとしては、例えばキッチンプリンタ 61 でレシート印刷を行うために、XML POS 形式に変換するためのスタイルシートを対象にし、予め用意された項目を変更するだけで、フォーマットを変更できるようなツールを提供する。

50

【 0 0 3 2 】

デバイス制御サービス管理部 7 2 は、デバイスサービス登録プログラム 4 6 (図 1 参照) が実行されることによって機能するものであり、ネットワーク N T 上に存在するデバイス制御サービス 7 3 , 7 5 の管理 (登録、削除、更新など) を行う。なお、デバイス制御サービス管理部 7 2 は、プライベート U D D I (Universal Description, Discovery and Integration) を利用して、デバイス制御サービス 7 3 , 7 5 の検索や照会を行う。

【 0 0 3 3 】

デバイス制御サービス 7 3 は、既存デバイス 5 0 a を制御するためのサービスである。なお、既存デバイスドライバを利用する場合は、既存デバイスドライバ用コマンド変換モジュールおよび既存ドライバ A P I (いずれも図示省略) がさらに必要となる。なお、デバイス 5 0 によっては、デバイス制御サービス 7 5 を内蔵しているものなど、サーバー 3 0 内のデバイス制御サービス 7 3 を必要としないものも存在する。そのようなデバイス 5 0 (以下、「新デバイス 5 0 b」と称する。) については、統合デバイス制御サービス 4 4 と、直接情報の入出力を行う。また、デバイス制御サービス 7 5 は、W S D L (Web Services Description Language) によりサービスが定義されていれば、デバイス 5 0 内および P C 2 0 内のいずれに存在していても良い。

【 0 0 3 4 】

一方、データベース 4 5 は、スタイルシートを記憶するスタイルシートデータベース 8 1 と、デバイス制御サービス 7 3 , 7 5 に関する情報 (以下、「デバイス別サービス情報」と称する。) を記憶するデバイス別サービス情報データベース 8 2 と、を有している。

【 0 0 3 5 】

スタイルシートデータベース 8 1 は、P C 2 0 から取得したスタイルシート (スタイルシート管理部 7 1 によって登録されたスタイルシート) を記憶する。ここで、スタイルシートは、X S L (Extensible Stylesheet Language) で記述される。また、デバイス別サービス情報データベース 8 2 は、デバイス制御サービス管理部 7 2 によって登録されたデバイス別サービス情報を記憶する。ここで、デバイス別サービス情報は、W S D L で記述される。

【 0 0 3 6 】

統合デバイス制御サービス 4 4 は、通信インターフェース 8 4、X M L (Extensible Markup Language) 文書変換部 8 5 (マークアップ文書変換部)、およびデバイス別 X M L 文書入出力部 8 6 (デバイス別マークアップ文書入出力部) を有している。統合デバイス制御サービス 4 4 は、主に、P C 2 0 から情報を配信する情報配信処理、並びにデバイス 5 0 (既存デバイス 5 0 a および新デバイス 5 0 b) からの応答処理を実現する。なお、情報配信処理に際して、アプリケーション 4 1 またはその他の送信者から提供される情報を、以下「配信情報」と称する。当該「配信情報」は、アプリケーション 4 1 から提供される場合、アプリケーション 4 1 の出力データと同時に (出力データに添付されて) 取得可能である。また、応答処理に際して、デバイス 5 0 から入力される情報を、以下「配信結果」と称する。

【 0 0 3 7 】

例えば、情報配信処理を行う場合、通信インターフェース 8 4 は、アプリケーション 4 1 の出力データ (注文情報) と、その出力対象 (情報配信対象) となる対象デバイスや出力用のスタイルシート (以下、「出力スタイルシート」と称する。) に関する情報を含む配信情報と、を取得する。これらはいずれも X M L で記述されており、以下、「X M L 文書 1 (第 1 文書) 」と総称する。また、X M L 文書変換部 8 5 は、通信インターフェース 8 4 から取得した X M L 文書 1 を、スタイルシートデータベース 8 1 から読み出した出力スタイルシートに基づいて、対象デバイスのデバイス制御コマンド (デバイス制御情報) を含む X M L 文書 (以下、「X M L 文書 2 (第 2 文書) 」) と変換する。なお、X M L 文書変換部 8 5 は、X S L T (XSL Transformation) を利用して、X M L 文書を変換する。さらに、デバイス別 X M L 文書入出力部 8 6 は、X M L 文書変換部 8 5 から取得した X M L 文書 2 を、対象デバイスに対応するデバイス制御サービス 7 3 , 7 5 に出力

10

20

30

40

50

する。

【0038】

また、応答処理を行う場合、デバイス別XML文書入出力部86は、各デバイス50からの配信結果を含むXML文書(以下、「XML文書3(第3文書)」と称する。)を、XML文書変換部85に入力する。XML文書変換部85は、入力された当該XML文書3を、スタイルシートデータベース81から読み出した入力用のスタイルシート(以下、「入力スタイルシート」と称する。)に基づいて、アプリケーション41用のXML文書(以下、「XML文書4(第4文書)」と称する。)に変換する。さらに、通信インターフェース84は、XML文書変換部85から取得したXML文書4を、アプリケーション41に入力する。

10

【0039】

次に、図3を参照し、情報配信処理に用いられる「配信情報」、および応答処理に用いられる「配信結果」の詳細について説明する。なお、各情報要素に付加された大括弧は、多重度を表すものである。

【0040】

図3(a)に示すように、「配信情報」は、1以上の「受信者情報」を有している。また、「受信者情報」は、受信者となる対象デバイス(デバイス制御サービス73,75)を特定するための「宛先情報」と、当該対象デバイスに好適な情報に変換するための出力スタイルシートを指定する「出力スタイルシート情報」と、送信者となるアプリケーション41に好適な情報に変換するための入力スタイルシートを指定する「入力スタイルシート情報」と、配信エラーの判定基準となる「制限時間」と、正常に情報配信が行われなかった場合、代わりに情報を配信する受信者(以下、「予備受信者」と称する。)に関する「受信者情報(以下、「予備受信者情報」と称する。)」と、を有している。なお、当該「予備受信者情報」も、上記の「配信情報」が有する「受信者情報(以下、「主受信者情報」と称する。)」と同じ構成要素を含んでいる。つまり、「受信者情報」は階層構造となっており、その階層に従って優先順位が決定されている。したがって、「予備受信者情報」に基づく配信が正常に行われなかった場合、次に優先順位の低い「予備受信者情報」に基づいて配信を行う・・・といった代替処理が繰り返されることとなる。

20

【0041】

続いて、「配信結果」について説明する。図3(b)に示すように、「配信結果」は、配信を試行した1以上の受信者に関する「受信者情報(主受信者情報)」を有している。また、「受信者情報」は、受信者となる対象デバイスを特定するための「宛先情報」と、当該対象デバイスへの「送信結果」と、正常に情報配信が行われなかった場合、代わりに情報を配信する受信者(予備受信者)に関する「受信者情報(予備受信者情報)」と、を有している。また、「送信結果」は、送信の成否を示す「成否情報」と、対象デバイスの状態(ステータス)などを示す「補足情報」と、対象デバイスからの入力データである「応答情報」と、を有している。

30

【0042】

なお、「配信情報」と「配信結果」は、図3(c)に示すように、統合することが可能である。つまり、「配信情報」のデータ構造の中に「配信結果」を追加し、要求メッセージ(図7参照)のスキーマと応答メッセージ(図9参照)のスキーマとを、一つにまとめることが可能である。

40

【0043】

次に、図4および図5を参照し、情報配信処理および応答処理について説明する。図4は、店内管理システムSYの情報配信処理を示すフローチャートである。同図に示すように、まずPC20(アプリケーション41)は、情報配信処理の開始前に、出力スタイルシートおよび入力スタイルシートを生成し、サーバー30に送信しておく(S01)。サーバー30(スタイルシート管理部71)は、これらのスタイルシートを、スタイルシートデータベース81に登録する(S02)。

【0044】

50

一方、PC20は、オーダリング端末10からの注文情報の取得に伴い、出力データを生成し、配信情報と共に、XML文書1として、サーバー30(統合デバイス制御サービス44)に配信する(S03)。なお、配信情報は、出力データの配信前に、予め送信しておくことも可能である。サーバー30は、PC20からXML文書1を取得し(S04、通信インターフェース84)、予め登録されていた出力スタイルシートに基づいて、XML文書1をXML文書2に変換する(S05、XML文書変換部85)。なお、出力スタイルシートは、配信情報の受信者情報に含まれる「出力スタイルシート情報」によって指定されたものを用いる。これにより、受信者(デバイス50)毎に異なる出力スタイルシートを適用可能となっている。したがって、受信者がキッチンプリンタ61の場合は、XMLPOSで表されるXML文書2に変換し、受信者がキッチンディスプレイ63の場合は、XHTML等の表示用言語で表されるXML文書2に変換する、といった処理も可能である。

10

【0045】

また、サーバー30は、変換後のXML文書2を、各デバイス50のデバイス制御サービス73、75に出力する(S06、デバイス別XML文書入出力部86)。なお、出力先(対象デバイス)は、配信情報の受信者情報に含まれる「宛先情報」によって特定される。デバイス50(デバイス制御サービス73、75)は、サーバー30から出力されたXML文書2を取得する(S07)。

【0046】

続いて、図5のフローチャートを参照し、店内管理システムSYの応答処理について説明する。デバイス50は、S07で取得したXML文書2に対する配信結果として、成否情報、補足情報および応答情報を含むXML文書3を生成し、サーバー30に送信する(S11)。サーバー30は、当該XML文書3を取得し(S12、デバイス別XML文書入出力部86)、予め登録されていた入力スタイルシートに基づいて、XML文書3をXML文書4に変換する(S13、XML文書変換部85)。なお、入力スタイルシートは、配信情報の受信者情報に含まれる「入力スタイルシート情報」によって指定されたものを、スタイルシートデータベース81から読み出して用いる。これにより、サーバー30が複数のPC20(アプリケーション41)と提携している場合でも、送信者であるアプリケーション41毎に異なる入力スタイルシートを利用して、好適な情報に変換することができる。

20

30

【0047】

また、サーバー30は、XML文書2を複数のデバイス50に対して送信した場合、当該複数のデバイス50からの配信結果(XML文書3)を統合して、XML文書4に変換する。サーバー30は、変換後のXML文書4を、XML文書1(出力データを)送信したPC20に入力する(S14)。また、送信者となるPC20(アプリケーション41)は、サーバー30から入力された当該XML文書4を取得する(S15)。

【0048】

次に、図6ないし図9を参照し、「出力データ」、「配信情報」、「スタイルシート」および「配信結果」の具体例について説明する。図6は、XML文書1に相当する「出力データ」の一例を示す図である。ここでは、オーダリング端末10から、注文情報「生ビールを1つ、ヒレスステーキを1つ、シーザーサラダを1つ」を取得した場合に生成される出力データを示している。符号101に示すように、「出力データ」には、シーケンス番号、アイテムID、アイテム名(料理名)を示す情報が含まれている。なお、ドリンク部門のアイテムIDは100番台であり、グリル部門のアイテムIDは200番台であり、サラダ部門のアイテムIDは300番台と規定されているものとする。この他、「出力データ」には、レストラン名、注文日時、オーダリング端末10のID、オーダリング端末10の操作者コード、価格(単価、合計)、顧客データ(性別、年代)などの情報を含めても良い。

40

【0049】

図7は、XML文書1に相当する「配信情報」の一例を示す図である。同図の例では、

50

5台のキッチンプリンタ61についての「受信者情報」が記述されている。符号111に示すように、第1のデバイス要素では、「キッチンプリンタ1」が主受信者であり、その予備受信者が「キッチンプリンタ2」であり、さらにその予備受信者が「キッチンプリンタ3」となっている。したがって、最初に「キッチンプリンタ1」に送信を試み、配信エラーとなった場合は、「キッチンプリンタ2」に送信を試み、さらに配信エラーとなった場合は、「キッチンプリンタ3」に送信を試みることとなる。一方、符号112に示すように、第2のデバイス要素では、「キッチンプリンタ4」が主受信者であり、その予備受信者が「キッチンプリンタ5」となっている。したがって、最初に「キッチンプリンタ4」に送信を試み、配信エラーとなった場合は、「キッチンプリンタ5」に送信を試みることとなる。また、上記のとおり「配信情報」の中には、受信者毎に、「出力スタイルシート情報」および「制限時間」が記述されている。なお、同図の例では、「配信情報」に含めることが可能な「入力スタイルシート情報」についての記述は、省略している。

10

【0050】

図8(a)は、「出力スタイルシート」の一例を示す図である。ここでは、ドリンク部門のキッチンプリンタ61用に生成された出力スタイルシートを例示している。同図は、符号121に示すように、アイテムID(メニューID)が100番台のアイテムについて、注文内容(注文があった旨、並びにアイテム名および注文数)を印刷するためのスタイルシートを例示している。また、同図のスタイルシートにより、XML文書2は、キッチンプリンタ61に接続されたPOSターミナル51が解釈可能な印刷言語(XMLPOS)で記述される。

20

【0051】

図8(b)は、XML文書2の一例を示す図である。同図の例は、図8(a)の出力スタイルシートに基づいて情報変換が行われた場合のXML文書2を例示している。すなわち、符号122に示すように、キッチンプリンタ61に対し、注文内容の印刷指令を行う内容となっている。

【0052】

図9は、XML文書4に相当する「配信結果」の一例を示す図である。同図の例は、図7に示した「配信情報」に対応するものである。すなわち、符号131および符号132に示すように、以下の内容が記述されている。“主受信者「キッチンプリンタ1」および「キッチンプリンタ4」に対して送信を試み、「キッチンプリンタ1」はデバイス50が見つからずに動作が失敗し、その予備受信者である「キッチンプリンタ2」に送信を試みた。しかし、「キッチンプリンタ2」もタイムアウトとなり動作が失敗した。そこで、さらにその予備受信者である「キッチンプリンタ3」に送信を試み、正常に動作した。一方、「キッチンプリンタ4」については、予備受信者(「キッチンプリンタ5」)に送信を試みることなく、正常に動作した。”なお、「配信結果」の「送信結果」は、「デバイス要素」に相当する。また、「送信結果」の「成否情報」は、「レスポンスコード」要素に相当し、「補足情報」および「応答情報」は、「ビジネスエラー」要素やその子要素に相当する。

30

【0053】

次に、図10ないし図13を参照し、サーバー30の代替処理について具体的に説明する。上記のとおりサーバー30は、主受信者に対して正常に配信できなかった場合、予備受信者に対して情報配信を試みるものとしたが、同一の情報を送信すると、所望する目的が達成できなかつたり、主受信者と予備受信者とが異なる種類のデバイス50である場合など、正常に動作させることができなかつたりする場合がある。そこで、このような問題を解決するための具体的な処理内容について、以下説明する。

40

【0054】

図10は、サーバー30の代替処理を示すフローチャートである。サーバー30(統合デバイス制御サービス44)は、図5に示した応答処理によって、XML文書3を取得すると(S21、図5のS12に相当)、情報配信が成功したか否かを判別する(S22)。ここで、情報配信が正常に行われなかったと判別した場合は(S22:No)、「配信

50

情報」に「予備受信者情報」が含まれているか否かを判別する（S23）。「配信情報」に「予備受信者情報」が含まれている場合は（S23：Yes）、予備受信者用の出力スタイルシート（以下、「予備出力スタイルシート」と称する。）に基づいて、XML文書1をXML文書2に変換し（S24，XML文書変換部85）、当該XML文書2を予備受信者となる対象デバイスのデバイス制御サービス73，75に出力する（S25，デバイス別XML文書入出力部86）。なお、予備受信者への配信を行った場合は、さらにその情報配信が成功したか否かを判別し（S22）、情報配信が正常に行われなかったと判別した場合は（S22：No）、さらに優先順位の低い「予備受信者情報」が存在するかどうかを判別し（S23）、S24以降の処理を行う。

【0055】

一方、情報配信が正常に行われたと判別した場合（S22：Yes）、並びに「配信情報」に「予備受信者情報」が含まれていない場合は（S23：Yes）、予備受信者への配信を行うことなく、入力スタイルシートに基づいて、「配信結果」を示すXML文書3をXML文書4に変換し（S26，図5のS13に相当）、当該XML文書4を送信者であるアプリケーション41に入力する（S27，図5のS14に相当）。

【0056】

続いて、図11ないし図13を参照し、「予備出力スタイルシート」の具体例について説明する。図11（a）は、ドリンク部門から他部門（グリル部門またはサラダ部門）のキッチンプリンタ61へ出力するための出力スタイルシートを示したものであり、図8に示した「出力スタイルシート」を主受信者用の出力スタイルシート（以下、「主出力スタイルシート」と称する。）とした場合の「予備出力スタイルシート」となる。なお、図11ないし図13において、下線部は、図8に示した「出力スタイルシート」および「XML文書2」の記述と異なる部分を示している。したがって、下線部分の書換えのみで（アプリケーション41の変更を必要とすることなく）、代替処理用の「予備出力スタイルシート」の生成が可能であり、ひいては、本実施形態における代替処理を実現可能となっている。

【0057】

符号141に示すように、図11（a）の「予備出力スタイルシート」は、アイテムIDが100番台のアイテムについて、注文内容と共に、ドリンク部門（本来の配信先）に転送して欲しい旨を印刷するためのスタイルシートとなっている。つまり、ドリンク部門のキッチンプリンタ61がレシート切れ等で正常に動作できない場合、他部門のキッチンプリンタ61で、図11（b）に示すXML文書2（符号142参照）に基づく印刷を実行可能となっている。これにより、他部門のスタッフが、ドリンク部門に調理指令の印刷物を回すことができ、調理業務を支障なく遂行することができる。

【0058】

なお、図11（a）に示した「予備出力スタイルシート」は、他部門（グリル部門またはサラダ部門）のキッチンプリンタ61で共用することができる。つまり、出力スタイルシートは、必ずしもデバイス50毎に異なるものを用意しておく必要は無い。また、「予備出力スタイルシート」としては、上記のとおり「ドリンク部門に転送して欲しい旨」の追加情報を調理指示伝票の中に挿入するのではなく、用紙のヘッダ余白部や、送付状として付加した用紙に印刷するような内容を記述しても良い。

【0059】

一方、図12（a）は、ドリンク部門の厨房機器62（ドリンクベンダー）に対する動作を指令するための出力スタイルシートを示したものである。また、図13（a）は、ドリンク部門のキッチンディスプレイ63に対する動作を指令するための出力スタイルシートを示したものである。なお、図13の出力スタイルシートは、図12に示した「出力スタイルシート」を「主出力スタイルシート」とした場合の「予備出力スタイルシート」となる。

【0060】

符号151に示すように、図12（a）の出力スタイルシートは、ドリンクベンダーに

10

20

30

40

50

対し、アイテムIDが100番台のアイテム（ドリンク類）をグラス等に注ぐ動作を指令するためのスタイルシートとなっている。また、当該出力スタイルシートに基づいて、図12（b）に示すXML文書2（符号152参照）を生成可能となっている。なお、当該XML文書2は、プロトコルコンバータ52（図1参照）が解釈可能な言語で記述されている。

【0061】

また、符号161に示すように、図13（a）の出力スタイルシートは、ドリンク部門のキッチンディスプレイ63に対し、アイテムIDが100番台のアイテム（ドリンク類）の注文内容を表示させるためのスタイルシートとなっている。また、当該スタイルシートに基づいて、図13（b）に示すXML文書2（符号162参照）を生成可能となっている。なお、当該XML文書2は、キッチンディスプレイ63が解釈可能な言語（XHTML）で記述されている。

10

【0062】

このように（図12および図13に例示したように）、ドリンクベンダーが故障等により正常に動作できない場合、キッチンディスプレイ63を用いて、同じ目的を実現する（ドリンクの注文に対応する）ことができる。つまり、アプリケーション41の変更を全く必要とすることなく、「配信情報」に「受信者情報」を追加し、さらに下線部を書き換えた代替用のスタイルシートを生成するだけで、種類の異なるデバイス50で、同じ目的を実現することが可能となる。

【0063】

20

以上、説明したとおり、本実施形態の店内管理システムSYによれば、スタイルシートによって、処理内容の変更、並びにデバイス50の種類や仕様の違いを吸収できるため、アプリケーション41による直接的なデバイス制御を不要とすることができる。これにより、アプリケーション41の作成が容易になると共に、処理内容の変更やデバイス50の入れ替えに伴うアプリケーション41の変更を不要とすることができる。

【0064】

また、デバイス別XML文書入出力部86により、既存デバイス50aおよび新デバイス50bに対応した2種類のデバイス制御サービス73, 75を呼び出すことができるため、例えば既存デバイス50aが新デバイス50bに入れ替えられたような場合にも、アプリケーション41の変更を不要とすることができる。

30

【0065】

また、「出力データ」、「配信情報」および「配信結果」をXMLで記述することにより、テキストエディタ等で簡単に編集ができると共に、XSL以外の情報変換規則を新たに生み出す必要が無いなどの利点がある。また、デバイス別サービス情報（デバイス制御サービス73, 75に関する情報）をWSDLで記述することにより、Webサービスの実装が容易となる。また、印刷言語としてXMLPOSを用いることで、メーカーを問わず、30種類以上のPOSデバイスを制御することが可能となる。

【0066】

また、送信者であるアプリケーション41毎に入力スタイルシートを指定できるため、統合デバイス制御サービス44が、複数のアプリケーション41と提携している場合でも、各アプリケーション41に応じたXML文書4（配信結果）を取得することができる。

40

【0067】

また、代替処理により、主受信者情報に基づく情報配信が正常に行われなかった場合は、予備受信者情報に基づいて情報配信を行うため、主受信者と予備受信者の仕様が異なる場合でも、正常に動作させることができる。さらに、出力スタイルシートでは、XML文書2の記述言語を指定できるため、解釈可能な記述言語が異なる複数種類のデバイス50に対応することもできる。これにより、各店舗のデバイス環境に応じて、デバイス50の入れ替えを柔軟に行うことができる。

【0068】

なお、上記の実施形態では、店内管理システムSYを例示したが、他の環境でも、本実

50

施形態のサーバー 30 (統合デバイス制御サービス 44) を適用可能である。例えば、デバイス 50 として、電子メール送信装置、ページプリンタ、大判プリンタ、電子棚札、POS コンピュータを有する販促管理システム (いずれも図示省略) に、本実施形態のサーバー 30 を適用した場合、クリアランスセール情報の取得をトリガとして、顧客データベースからクリアランス商品に興味のある顧客を抽出して、セール情報を電子メール送信装置に送信させる。この場合、上得意客には、クーポン情報を添付し、第 1 のメールアドレスへの配信が失敗した場合は第 2 のメールアドレスへの配信を行う (代替処理) 。また、ページプリンタには、メールアドレスが登録されていない顧客用に、郵送用のダイレクトメールを印刷させ、大判プリンタには、セール情報の POP を印刷させる。さらに、電子棚札には、価格情報をセール価格に変更する指令を出し、POS コンピュータには、商品マスタに登録された商品価格の変更を指令する。

10

【 0069 】

また、上記の実施形態で示した同一デバイス間における代替処理を、POS ターミナル 51 間の転送に用いても良い。例えば、POS ターミナル 51 が複数設置されているスーパー等の売場において、任意の POS ターミナル 51 が故障した場合、予備受信者として指定された隣の POS ターミナル 51 のプリンタから、発行元として隣の POS ターミナル 51 の ID が付記されたレシートを発行することが可能である。

【 0070 】

また、上記の実施形態で示した異なる種類のデバイス間における代替処理を、クーポン発行に用いても良い。例えば、電子バリューリーダーライタやポイントカードリーダーライタ (電子クーポンを発行する装置) が故障した場合、予備受信者として指定されたクーポンプリンタ (紙のクーポンを発行する装置) でクーポンを発行することが可能である。その他、種々の環境で、統合デバイス制御サービス 44 を用いたサービスを実現可能である。

20

【 0071 】

また、上記の実施形態に示した、サーバー 30 や PC 20 の各構成要素をプログラムとして提供することが可能である。また、そのプログラムを各種記録媒体 (CD-ROM、フラッシュメモリ等) に格納して提供することも可能である。すなわち、コンピュータを、サーバー 30 や PC 20 の各構成要素として機能させるためのプログラム、およびそれを記録した記録媒体も、本発明の権利範囲に含まれるものである。その他、本発明の要旨を逸脱しない範囲で、適宜変更が可能である。

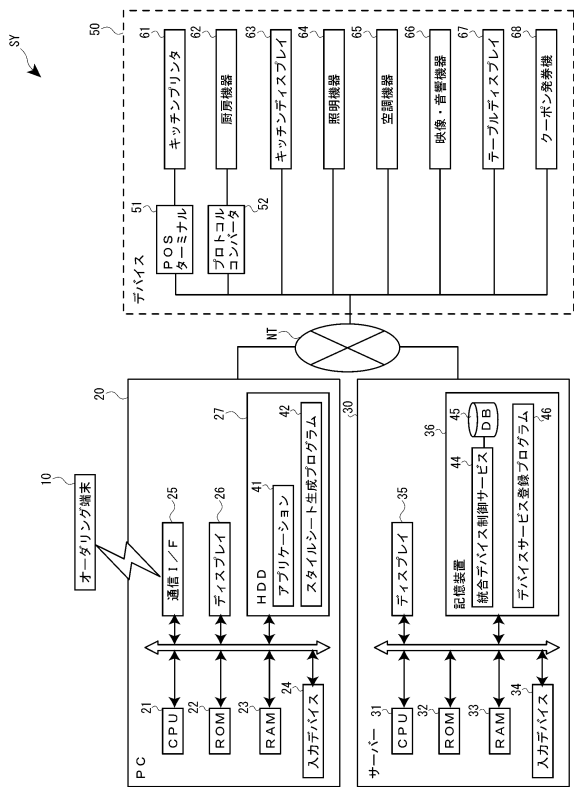
30

【 符号の説明 】

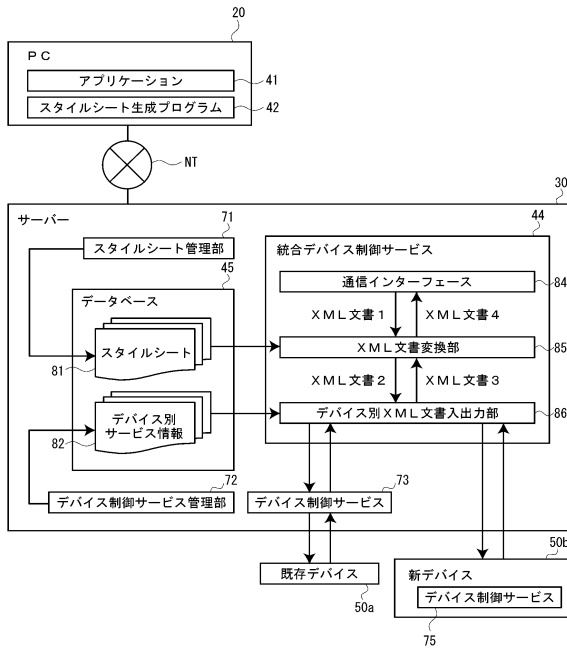
【 0072 】

10 ... オーダリング端末 20 ... PC 30 ... サーバ 41 ... アプリケーション 44 ... 統合デバイス制御サービス 50 ... デバイス 73 ... デバイス制御サービス (既存デバイス用) 75 ... デバイス制御サービス (新デバイス用) 81 ... スタイルシートデータベース 84 ... 通信インターフェース 85 ... XML 文書変換部 86 ... デバイス別 XML 文書入出力部 SY ... 店内管理システム

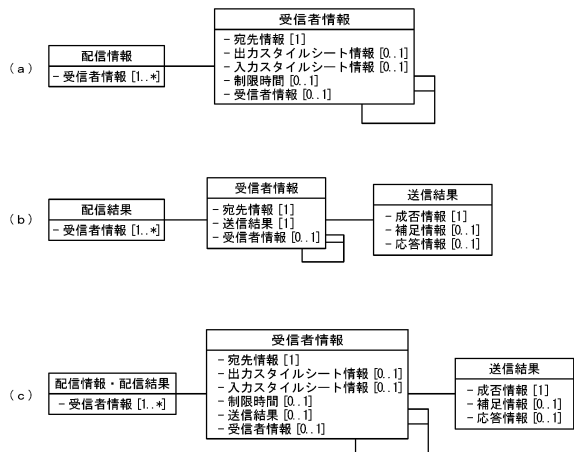
【図 1】



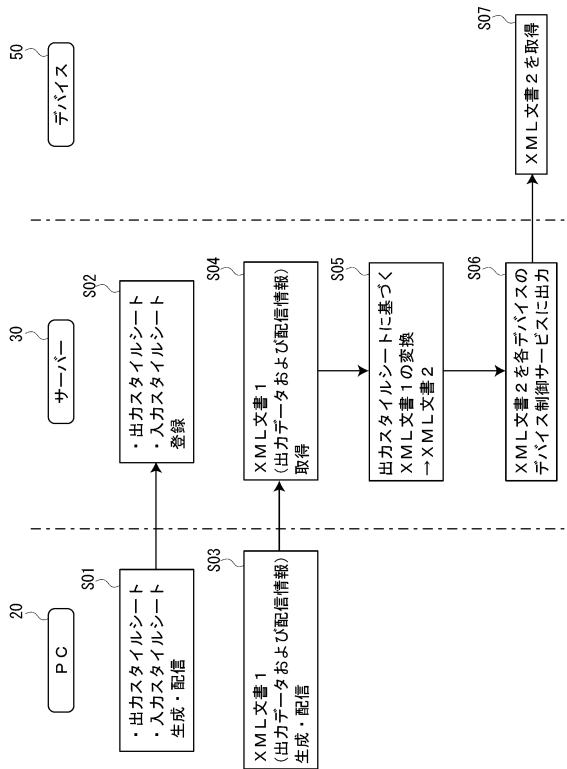
【図 2】



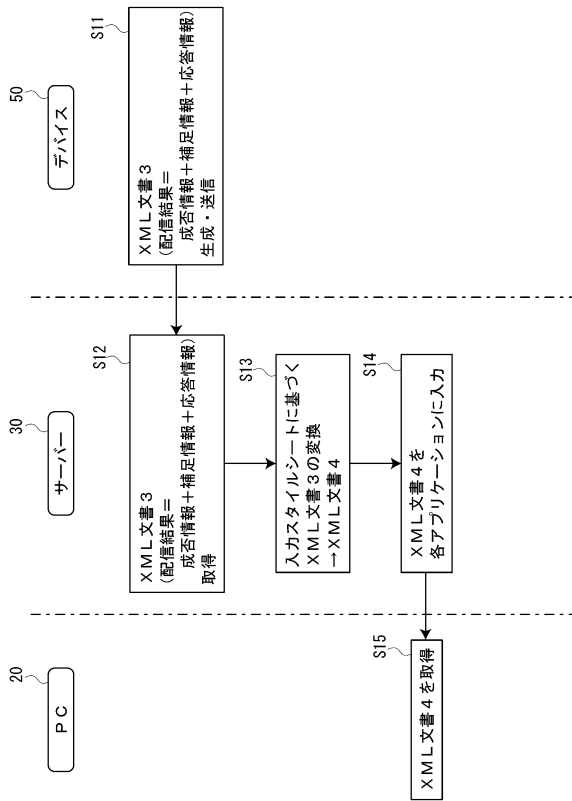
【図 3】



【図 4】



【図 5】



【図 6】

<XML文書1：出力データ>
注文情報（生ビールを1つ、ヒレステーキを1つ、シーザーサラダを1つ）

```

<?xml version="1.0" encoding="UTF-8" ?>
<Order>
  <Transaction>
    <CustomerOrderTransaction>
      <LineItem>
        <SequenceNumber>1</SequenceNumber>
        <Sale>
          <ItemID>101</ItemID>
          <Description>生ビール</Description>
          <Quantity>1</Quantity>
        </Sale>
      </LineItem>
      <LineItem>
        <SequenceNumber>2</SequenceNumber>
        <Sale>
          <ItemID>202</ItemID>
          <Description>ヒレステーキ</Description>
          <Quantity>1</Quantity>
        </Sale>
      </LineItem>
      <LineItem>
        <SequenceNumber>3</SequenceNumber>
        <Sale>
          <ItemID>303</ItemID>
          <Description>シーザーサラダ</Description>
          <Quantity>1</Quantity>
        </Sale>
      </LineItem>
    </CustomerOrderTransaction>
  </Transaction>
</Order>

```

【図 7】

<XML文書1：配信情報>

```

<?xml version="1.0" encoding="UTF-8" ?>
<Execute>
  <Device>
    <Name>KitchenPrinter1</Name>
    <Stylesheet>KitchenPrinter1_Meat.xml</Stylesheet>
    <Timeout>10000</Timeout>
    <Device>
      <Name>KitchenPrinter2</Name>
      <Stylesheet>KitchenPrinter2_Meat.xml</Stylesheet>
      <Timeout>10000</Timeout>
      <Device>
        <Name>KitchenPrinter3</Name>
        <Stylesheet>KitchenPrinter3_Meat.xml</Stylesheet>
        <Timeout>10000</Timeout>
      </Device>
    </Device>
  </Device>
  <Device>
    <Name>KitchenPrinter4</Name>
    <Stylesheet>KitchenPrinter4_Salad.xml</Stylesheet>
    <Timeout>10000</Timeout>
    <Device>
      <Name>KitchenPrinter5</Name>
      <Stylesheet>KitchenPrinter5_Salad.xml</Stylesheet>
      <Timeout>10000</Timeout>
    </Device>
  </Device>
</Execute>

```

【図 8】

<出カスタイルシート>
ドリンク部門のキッチンプリンタ用スタイルシート

```

<?xml version="1.0" encoding="UTF-8" ?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="xml" encoding="UTF-8" />
  <xsl:template match="/">
    <xsl:apply-templates select="./Sale"/>
  </xsl:template>
  <xsl:template match="Sale">
    <xsl:variable name="MenuID" select="./ItemID"/>
    <xsl:if test="$MenuID >= 100 and $MenuID <= 200">
      <Print>
        <Data>
          注文が入りました。
          <xsl:value-of select="./Description"/>を
          <xsl:value-of select="./Quantity"/>つお願いします。
        </Data>
      </Print>
    </xsl:if>
  </xsl:template>
</xsl:stylesheet>

```

<XML文書2>
ドリンク部門のキッチンプリンタへの出力

```

<?xml version="1.0" encoding="UTF-8" ?>
<Print>
  <Data>
    注文が入りました。生ビールを1つお願いします。
  </Data>
</Print>

```

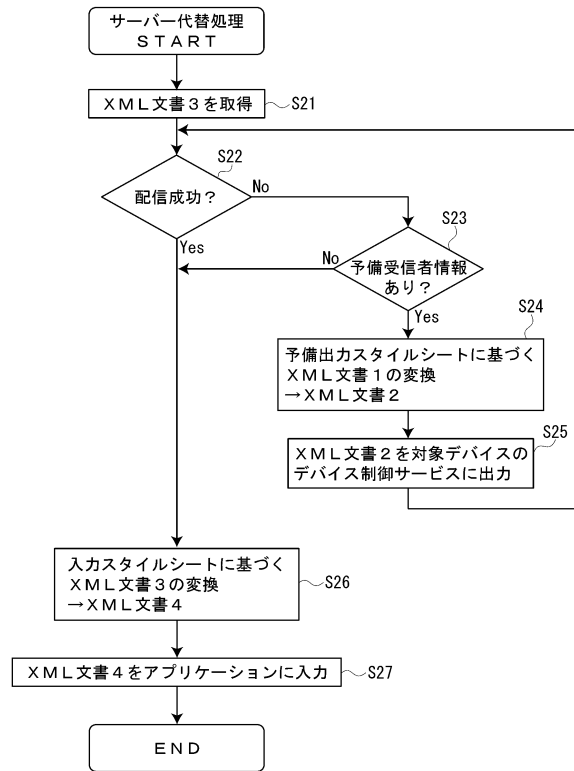
【図 9】

```

<XML文書 4 : 配信結果>
<?xml version="1.0" encoding="UTF-8"?>
<Execute>
131 <Device>
  <Name>KitchenPrinter1</Name>
  <Response ResponseCode="Rejected">
    <BusinessError Severity="Error">
      <Code>DeviceNotFound</Code>
      <Description>The device has not worked successfully.</Description>
    </BusinessError>
  </Response>
  <Device>
    <Name>KitchenPrinter2</Name>
    <Response ResponseCode="Rejected">
      <BusinessError Severity="Error">
        <Code>DeviceTimeout</Code>
        <Description>The device has not worked successfully.</Description>
      </BusinessError>
    </Response>
    <Device>
      <Name>KitchenPrinter3</Name>
      <Response ResponseCode="OK">
        <BusinessError Severity="Information">
          <Code>DeviceSuccess</Code>
          <Description>The device has worked successfully.</Description>
        </BusinessError>
      </Response>
    </Device>
  </Device>
132 <Device>
  <Name>KitchenPrinter4</Name>
  <Response ResponseCode="OK">
    <BusinessError Severity="Information">
      <Code>DeviceSuccess</Code>
      <Description>The device has worked successfully.</Description>
    </BusinessError>
  </Response>
</Device>
</Execute>

```

【図 10】



【図 11】

```

<予備出カスタイルシート (代替処理用)>
ドリンク部門から他部門のキッチンプリンタへ出力するためのスタイルシート
<?xml version="1.0" encoding="UTF-8"?>
<xml:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="xml" encoding="UTF-8"/>
  <xsl:template match="/">
    <xsl:apply-templates select="."/>
  </xsl:template>
141 <xsl:template match="Sale">
  <xsl:variable name="MenuID" select="."/>
  <xsl:if test="$MenuID >= 100 and $MenuID <= 200">
    <Print>
      <Data>
        注文が入りました。ドリンク部門へ転送してください!
        <xsl:value-of select="."/>を
        <xsl:value-of select="."/>つお願いします。
      </Data>
    </Print>
  </xsl:if>
</xsl:template>
</xsl:stylesheet>

```

```

<XML文書 2 (代替処理時)>
ドリンク部門のキッチンプリンタへの出力
142 <?xml version="1.0" encoding="UTF-8"?>
<Print>
  <Data>
    注文が入りました。ドリンク部門へ転送してください! 生ビールを1つお願いします。
  </Data>
</Print>

```

【図 12】

```

<出カスタイルシート>
ドリンク部門のドリンクベンダー用スタイルシート
<?xml version="1.0" encoding="UTF-8"?>
<xml:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="xml" encoding="UTF-8"/>
  <xsl:template match="/">
    <xsl:apply-templates select="."/>
  </xsl:template>
151 <xsl:template match="Sale">
  <xsl:variable name="MenuID" select="."/>
  <xsl:if test="$MenuID >= 100 and $MenuID <= 200">
    <Vend>
      <Pour>
        <xsl:value-of select="."/>を
        <xsl:value-of select="."/>つ
      </Pour>
    </Vend>
  </xsl:if>
</xsl:template>
</xsl:stylesheet>

```

```

<XML文書 2>
ドリンク部門のドリンクベンダーへの出力
152 <?xml version="1.0" encoding="UTF-8"?>
<Vend>
  <Pour>生ビールを1つ</Pour>
</Vend>

```


【 図 1 3 】

<予備出カスタイルシート (代替処理用)>
ドリンク部門のキッチンディスプレイ用スタイルシート

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html" encoding="UTF-8"/>

  <xsl:template match="/">
    <xsl:apply-templates select="//Sale"/>
  </xsl:template>

  <xsl:template match="Sale">
    <xsl:variable name="MenuID" select="//ItemID"/>
    <xsl:if test="$MenuID >= 100 and $MenuID <= 200">
      <html>
        <body>
          注文が入りました。
          <xsl:value-of select="//Description"/> を
          <xsl:value-of select="//Quantity"/> つお願いします。
        </body>
      </html>
    </xsl:if>
  </xsl:template>
</xsl:stylesheet>
```

161 (a)

<XML文書2 (代替処理時)>
ドリンク部門のキッチンディスプレイへの出力

```
<html>
  <body>注文が入りました。生ビールを1つお願いします。</body>
</html>
```

162 (b)

フロントページの続き

(72)発明者 古幡 整

長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

審査官 川 崎 博章

(56)参考文献 特開2010-097544(JP,A)

特開2007-210285(JP,A)

特開2007-207143(JP,A)

特表2005-507516(JP,A)

特開2003-216639(JP,A)

特表2002-539547(JP,A)

特開2002-032263(JP,A)

特開2005-038016(JP,A)

特開平07-219729(JP,A)

野呂 昌満,後藤 邦夫,ネットワークとマルチメディアトラック インターネットソフトウェア 初版,日本,共立出版株式会社,2002年 8月 1日,第1版ISBN: 4-320-02784-1, p.40-p.41

(58)調査した分野(Int.Cl., DB名)

G06F 17/20 - 17/28