

(11)特許出願公開番号

特開2010-20537

(P2010-20537A)

(43) 公開日 平成22年1月28日(2010.1.28)

(51) Int.Cl.

G06Q 10/00 (2006.01)

F I

G06F 17/60 162C

テーマコード (参考)

審査請求 未請求 請求項の数 6 O L (全 24 頁)

(21) 出願番号 特願2008-180361 (P2008-180361)

(22) 出願日 平成20年7月10日 (2008. 7. 10)

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(74) 代理人 100099759

弁理士 青木 篤

(74) 代理人 100119987

弁理士 伊坪 公一

(74) 代理人 100081330

弁理士 樋口 外治

(74) 代理人 100141254

弁理士 榎原 正巳

(74) 代理人 100113826

弁理士 倉地 保幸

[最終頁に続く](#)

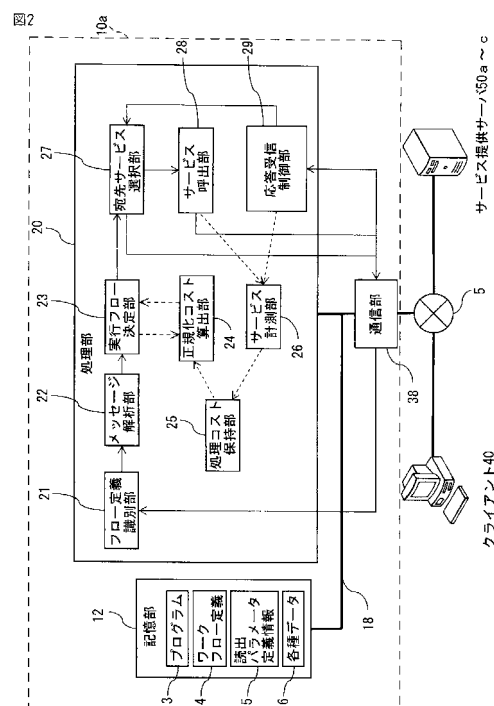
(54) 【発明の名称】 ワークフロー実行制御プログラム、ワークフロー実行制御装置、及びワークフロー実行制御方法

(57) 【要約】

【課題】ワークフロー処理において、補償トランザクションを最適化するように個々の処理の実行順は決定されていない。

【解決手段】1台以上のサービス提供サーバ50により提供される複数のサービスを、順次呼び出すワークフロー実行制御プログラムであって、1台以上のサービス提供サーバに接続されるコンピュータを、サービス毎の処理結果を用いて、サービス毎の処理コスト及びエラーの発生確率を算出するサービス計測部26と、1つのサービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化される正規化コストを、サービスの各々について算出する正規化コスト算出部24と、算出された正規化コストに従って、サービスの実行順序を決定する実行フロー決定部23と、サービス提供サーバに対して、実行順序に従ってサービスを順次呼び出すサービス呼出部28として機能させる。

【選択図】図2



【特許請求の範囲】**【請求項 1】**

1 台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御プログラムであって、前記 1 台以上のサービス提供サーバに接続されるコンピュータを、

前記サービス毎の処理結果を用いて、前記サービス毎の処理コスト及びエラーの発生確率を算出するサービス計測部と、

1 つの前記サービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化される正規化コストを、前記サービスの各々について算出する正規化コスト算出部と、

前記算出された正規化コストに従って、前記サービスの実行順序を決定する実行フロー決定部と、

前記サービス提供サーバに対して、前記実行順序に従って前記サービスを順次呼び出すサービス呼出部として機能させることを特徴とするワークフロー実行制御プログラム。

【請求項 2】

前記正規化コスト算出部は、前記複数のサービスの依存関係を定義するワークフロー定義を用いて、前記正規化コストを算出する請求項 1 に記載のワークフロー実行制御プログラム。

【請求項 3】

前記正規化コスト算出部は、前記サービスの依存関係を定義するワークフロー定義を用いて、依存関係を有するサービスのグループ単位に前記正規化コストの計算を行い、

前記実行フロー決定部は、前記グループ単位の正規化コストに従って、グループ単位の実行順序を決定する請求項 1 又は 2 に記載のワークフロー実行制御プログラム。

【請求項 4】

前記サービス計測部は、前記各サービスを実行するサービス提供サーバから前記各サービスを実行するために要した処理時間をさらに取得して、該処理時間に基づいて前記サービス毎の処理コストを算出する請求項 1 ～ 3 のいずれか 1 項に記載のワークフロー実行制御プログラム。

【請求項 5】

1 台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御装置であって、

前記サービス毎にエラーの発生確率及び処理コストを蓄積する処理コスト保持部と、

前記サービス毎の処理結果を用いて、前記サービス毎の処理コスト及びエラーの発生確率を算出するサービス計測部と、

1 つの前記サービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化される正規化コストを、前記サービスの各々について算出する正規化コスト算出部と、

前記算出された正規化コストに従って、前記サービスの実行順序を決定する実行フロー決定部と、

前記サービス提供サーバに対して、前記実行順序に従って前記サービスを順次呼び出すサービス呼出部と、

を有することを特徴とするワークフロー実行制御装置。

【請求項 6】

1 台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御方法であって、

前記サービス毎の処理結果を用いて、前記サービス毎の処理コスト及びエラーの発生確率を算出するステップと、

前記サービス毎の処理結果を用いて前記サービス毎の処理コスト及びエラーの発生確率を算出するステップと、

前記算出された正規化コストに従って、前記サービスの実行順序を決定するステップと

10

20

30

40

50

、
前記サービス提供サーバに対して、前記実行順序に従って前記サービスを順次呼び出すステップと、

を有することを特徴とするワークフロー実行制御方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ワークフローの制御に関する。

【背景技術】

【0002】

10

コンピュータを用いてワークフローを処理する場合、トランザクションとともに、ワークフローに規定された複数の処理手順を実行する必要がある。ワークフローとは、一般に、複数の処理手順を含む手続きを言う。トランザクションとは、複数のアプリケーションに跨る処理において、各アプリケーションの処理結果の整合性を保つための処理を言う。

従来、ワークフロー処理を行うために、B P E L (Business Process Execution Language) 等に従って、ワークフローを規定し、且つそのワークフローに基づいてサービスリクエストがサービスを呼び出す方式が用いられている。

【0003】

また、ワークフローを実行して、ワークフローに含まれる複数のアプリケーションサービス（以下、「サービス」と称す）を実行するために、複数サービスに対してアクセスをロックするようなトランザクションを行うと、各サービスが管理するリソースのロック期間が長くなる場合がある。そのため、各サービスを独立に実行し、あるサービスの処理ができなかった場合（以下「エラー」と称す）、他サービスに補償トランザクション（キャンセル処理 / ロールバック処理等）を実行させるようなトランザクション（「ロング・ランニング・トランザクション」と称される）がある。この補償トランザクションは、例えば、あるサービスの応答結果が特定の条件に合致した時に、他の実行済みサービスに対してキャンセル処理の依頼を行うようにして実行される。

20

【0004】

なお、ワークフロー処理において、補償トランザクションを行うためのフロー制御プログラムが提案されている。このプログラムは、各処理の処理時間が長い処理、又は、エラー発生確率が低い処理の実行順序を最後に配置するようにワークフローを定義する。しかし、個々の処理に対して実行順序を決めるが、補償トランザクションを最適化するように個々の処理の実行順序は決定されていない。

30

【0005】

【特許文献1】特開2006-195892号広報

【発明の開示】

【発明が解決しようとする課題】

【0006】

本出願は、1つのサービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化される正規化コストに従って、複数のサービスの実行順序を規定することにより、補償トランザクションを最適化したワークフロー処理を提供することを目的とする。

40

【課題を解決するための手段】

【0007】

上記課題を解決するために、1台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御プログラムが提供される。

上記ワークフロー実行制御プログラムは、上記1台以上のサービス提供サーバに接続されるコンピュータを、上記サービス毎の処理結果を用いて上記サービス毎の処理コスト及びエラーの発生確率を算出するサービス計測部と、1つの上記サービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化される正規化

50

コストを、上記サービスの各々について算出する正規化コスト算出部と、上記算出された正規化コストに従って、上記サービスの実行順序を決定する実行フロー決定部と、上記サービス提供サーバに対して、上記実行順序に従って上記サービスを順次呼び出すサービス呼出部として機能させる。

【発明の効果】

【0008】

開示するワークフロー実行制御プログラムは、エラー発生時のサービス実行及び補償トランザクションの実行コストを削減したワークフロー処理を提供することができる。

【発明を実施するための最良の形態】

【0009】

10

以下、図面を参照して、実施形態を説明する。

図1を用いて、ワークフロー実行制御装置の接続構成の概要について説明する。インターネット、LAN等のネットワーク5を介して、ワークフロー実行制御装置10a、クライアントコンピュータ40（以下、「クライアント40」と称する）、サービス提供サーバ50a～cが互いにアクセス可能に接続されている。

【0010】

〔第1の実施形態〕

図2を用いて、第1の実施形態に係るワークフロー実行制御装置10aについて説明する。

ワークフロー実行制御装置10aは、クライアント40から受信したリクエストメッセージに従って、1台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御を実施する。

20

ワークフロー実行制御装置10aは、単一又は複数のCPU（Central Processing Unit）及びメモリ（SRAM、DRAMなど）を含む処理部20、不揮発性半導体メモリ、ハードディスク装置、光ディスク装置等の記憶部12、及びNIC（Network Interface Card）等の通信部38を有する。ワークフロー実行制御装置10aは、HTTP等の既知のプロトコルを用いて、ネットワーク5を介してクライアント40及び/又はサービス提供サーバ50a～cへ各種データの送受信を行えるように構成される。

【0011】

また、記憶部12には、少なくともワークフロー実行制御装置10aにおける動作を制御するためのプログラム13、処理対象となる複数のサービスの実行順序及び/又は依存関係を規定した情報であるワークフロー定義14、読出パラメータ定義情報15、制御部20で実行される処理に利用される各種データ16が格納される。プログラム13には、オペレーティングシステム、及び、アプリケーション等が含まれる。

30

【0012】

処理部20は、記憶部12に格納されたプログラム13を実行することにより、フロー定義識別部21、メッセージ解析部22、実行フロー決定部23、正規化コスト算出部24、サービス計測部26、宛先サービス選択部27、サービス呼出部28、及び応答受信制御部29として機能する。

また、処理部20のメモリ又は記憶部12は、処理コスト保持部25として機能する。

40

【0013】

フロー定義識別部21は、クライアント40から通信部38を介して受信したリクエストメッセージの内容を解析して、リクエストメッセージに対応するワークフロー定義14を記憶部12から読み出す。ここで、リクエストメッセージとは、サービスに対する処理の要求仕様を規定したメッセージのことを言う。

メッセージ解析部22は、記憶部12から読出パラメータ定義情報15を読み出して、読出パラメータ定義情報15により指示されるリクエストメッセージ中の場所からパラメータ（例えば、予約先ホテル名）を読み出し、実行フロー決定部23に伝える。読出パラメータ定義情報15は、例えば、「XPath（XMLパス言語）」で規定し得る。

【0014】

50

実行フロー決定部 2 3 は、フロー実行順を決める対象のサービス群を、ワークフロー定義 1 4 から識別し、正規化コスト算出部 2 4 にそのサービス群に対するコスト計算を依頼する。また、実行フロー決定部 2 3 は、正規化コスト算出部 2 4 のコスト計算結果を受け取り、コスト降順にサービスの実行順序を決定する。実行フロー決定部 2 3 は、決定したフロー実行順序を宛先サービス選択部 2 7 に通知する。

正規化コスト算出部 2 4 は、実行フロー決定部 2 3 から指定されたサービス群およびメッセージ解析部 2 2 が読み出したパラメータをもとに、処理コスト保持部 2 5 から各サービスの処理時間、キャンセル処理時間、及びエラー発生確率を読み出す。正規化コスト算出部 2 4 は、読み出したデータを用いて、各サービスの正規化コストを計算し、実行フロー決定部 2 3 に渡す。

10

【 0 0 1 5 】

処理コスト保持部 2 5 は、サービスごと及び / 又はメッセージ解析部 2 2 の抽出したパラメータ (情報単位) ごとに、サービスの処理時間及びキャンセル処理時間である処理コスト、及びエラー発生確率を保持する。

サービス計測部 2 6 は、サービスごと及び / 又はメッセージ解析部 2 2 の抽出したパラメータごとに、サービス呼出部 2 8 からリクエスト送信時刻及び応答受信制御部 2 9 から応答時刻を取得して、サービスの処理時間及びキャンセル処理時間である処理コストを算出する。そして、サービス計測部 2 6 は、処理コストを処理コスト保持部 2 5 に保存する。また、サービス計測部 2 6 は、応答受信制御部 2 9 からサービス提供サーバ 5 0 からサービスの処理結果を取得して、処理結果が「OK」か「エラー」かを識別する。サービス計測部 2 6 は、処理結果がエラーの場合はエラー発生確率を再計算して、上記処理コストと合わせて処理コスト保持部 2 5 にエラー発生確率を記録する。

20

宛先サービス選択部 2 7 は、実行フロー決定部 2 3 が決定したフロー実行順序に従って、呼び出すサービスを決定する。また、宛先サービス選択部 2 7 は、実行中の各ワークフローの状態管理、エラー発生時のキャンセル用の処理呼び出し判断を行う。

サービス呼出部 2 8 は、フロー実行順序に従って決定されたサービスの実行を要求するサービス実行リクエストを、通信部 3 8 を介して決定されたサービスを提供するサービス提供サーバ 5 0 に送信してサービス提供サーバ 5 0 のサービスを読み出す。さらに、サービス呼出部 2 8 は、サービス実行リクエスト送信時の時刻であるリクエスト送信時刻をサービス計測部 2 6 に伝える。

30

応答受信制御部 2 9 は、通信部 3 8 を介して処理結果をサービス提供サーバ 5 0 から受信し、宛先サービス選択部 2 7 に渡す。応答受信制御部 2 9 は、処理結果を用いて、処理結果受信時の応答時刻をサービス計測部 2 6 に伝える。

【 0 0 1 6 】

クライアント 4 0 は、ワークフロー実行制御装置 1 0 a にリクエストメッセージを送信することにより、サービス提供サーバ 5 0 a ~ c が提供し得る A S P サービスの利用を要求する。

クライアント 4 0 は、液晶ディスプレイ等の表示部、キーボード及びマウス等の操作部、C P U、メモリ等を含む処理部、ハードディスク等の記憶部、及び N I C 等の通信部を有し、ネットワーク 5 を利用してワークフロー実行制御装置 1 0 a 及び / 又はクライアント 4 0 へ各種データの送受信を行えるように構成されている。

40

クライアント 4 0 には、Web サイトの Web ページをアクセスするための Web ブラウザがインストールされている。また、クライアント 4 0 は、パーソナルコンピュータ、ブラウザフォン、又は、P D A (Personal Digital Assistance) であっても良い。なお、図 1 では、クライアント 4 0 は、1 台で示されているが、この数に限定されるものではなく、複数台であっても良い。

【 0 0 1 7 】

サービス提供サーバ 5 0 a ~ c は、A S P サービスを提供するサーバであっても良い。サービス提供サーバ 5 0 a ~ c の各々は、C P U、メモリ等を含む処理部、ハードディスク等の記憶部、及び N I C 等の通信部を有する。サービス提供サーバ 5 0 a ~ c は、記憶

50

部に格納されたプログラムを処理部で実行することで、任意の A S P サービスであるサービス S 1 ~ S 3 をそれぞれクライアント 4 0 に提供する。また、サービス提供サーバ 5 0 a ~ c は、ネットワーク 1 を利用してワークフロー実行制御装置 1 0 a 及びクライアント 4 0 へ各種データの送受信を行えるように構成されている。

なお、図 1 では、サービス提供サーバは、3 台で示されているが、その台数に制限されない。また、1 台のサービス提供サーバが複数のサービスを提供することで、サービス提供サーバの台数を減少させても良い。

【0018】

図 3 及び図 4 を用いて、ワークフロー実行制御の処理フローの一例について説明する。

なお、図 3 及び図 4 では、クライアント 4 0、ワークフロー実行制御装置 1 0 a、及びサービス提供サーバ 5 0 は、それぞれ、A S P サービスの要求処理、ワークフロー実行制御、及びサービス処理を開始する。図 3 及び図 4 では、各装置における処理のフローは、実線で示され、各装置間を跨るデータの送受信は点線で示される。

【0019】

図 3 及び図 4 を用いて、クライアント 4 0 による A S P サービスの要求処理フローの一例について説明する。

まず、クライアント 4 0 は、サービス提供サーバ 5 0 a ~ c が提供し得る A S P サービスを利用するために、A S P サービスの要求処理を開始して、ワークフロー実行制御装置 1 0 a にリクエストメッセージを送信する（ステップ S 1 0 1）。クライアント 4 0 は、ワークフロー実行制御装置 1 0 a から、リクエストメッセージに含まれたワークフローが適切に実行されたか、又は、どのサービスがエラーであったかを示すメッセージ応答を受信して（ステップ S 1 1 6）、A S P サービスの要求処理を終了する。

【0020】

図 3 及び図 4 を用いて、ワークフロー実行制御装置 1 0 a によるワークフロー実行制御の処理フローの一例について説明する。

ワークフロー実行制御装置 1 0 a は、クライアント 4 0 から受信したリクエストメッセージに従ってワークフロー実行制御の処理を開始する。まず、ワークフロー実行制御装置 1 0 a の通信部 3 8 でクライアント 4 0 から送信されたリクエストメッセージを受信する（ステップ S 1 0 2）。なお、ステップ S 1 0 2 では、ワークフロー実行制御装置 1 0 a の通信部 3 8 は、サービス提供サーバ 5 0 a ~ c に送られたリクエスト送信に対する処理結果（O K 結果又はエラー結果）も受信する（後述のステップ S 1 1 4）。

【0021】

図 5 に、ステップ S 1 0 1 で送信されるリクエストメッセージの内容の一例を示す。リクエストメッセージは、サービスに対する処理の要求を規定したメッセージである。図示されるように、リクエストメッセージ 9 0 1 は、X M L フォーマットで定義され得る。リクエストメッセージ 9 0 1 は、9 0 2 に示す < name > タグ内に、リクエストメッセージに含まれるワークフローの名称である「wf 1」が示される。リクエストメッセージ 9 0 1 は、サービス提供サーバ 5 0 で実行されるサービス S 1、S 2、S 3 に対するリクエスト内容 9 1 1、9 2 1、9 3 1 を有する。また、リクエスト内容 9 1 1、9 2 1、9 3 1 は、9 1 2、9 2 2、9 3 2 に示す < name > タグ内のパラメータとして、リクエスト単位又はサービス単位となる予約リソースをそれぞれ含み、さらに、9 1 3、9 2 3、9 3 3 で示す部分にリクエスト仕様情報をそれぞれ含む。

【0022】

例えば、リクエスト内容 9 1 1 では、予約リソース（「リクエスト情報単位」とも称する）9 1 2 は「Hotel 1」であり、リクエスト仕様情報 9 1 3 はサービスに対する予約リソースを示す仕様（日付、部屋の種類、人数）である。リクエスト内容 9 2 1 では、予約リソース 9 2 2 は「Train 1」であり、リクエスト仕様情報 9 2 3 はサービスに対する予約リソースを示す仕様（日付、出発駅、到着駅）である。リクエスト内容 9 3 1 は、予約リソース 9 3 2 は「Airplane 1」であり、リクエスト仕様情報 9 3 3 はサービスに対する予約リソースを示す仕様（日付、出発空港、到着空港）である。

【 0 0 2 3 】

ステップ S 1 0 3 では、ワークフロー実行制御装置 1 0 a は、ステップ S 1 0 2 でクライアント 4 0 又はサービス提供サーバ 5 0 a ~ c から受信したメッセージか、クライアント 4 0 からのリクエストメッセージか、サービス提供サーバ 5 0 からの処理結果かを判断する（ステップ S 1 0 3）。受信したメッセージがリクエストメッセージの場合は、ステップ S 1 0 4 に進む。受信したメッセージがサービス提供サーバ 5 0 a ~ c からの処理結果の場合は、ステップ S 1 0 5 に進む。

ステップ S 1 0 4 では、フロー定義識別部 2 1 は、リクエストメッセージ 9 0 1 のワークフローの名称「w f 1」を読み出して、このワークフロー名称と関係付けられるワークフロー定義を読み出して、実行フロー決定処理を実行する（ステップ S 1 5 1 ~ S 1 5 5）。

10

【 0 0 2 4 】

図 6 を用いて、実行フロー決定処理のフローの一例を説明する。まず、メッセージ解析部 2 2 は、ステップ S 1 0 4 で読み出したワークフロー定義 1 4 を参照して、ワークフロー定義 1 4 の中にサービスの並列実行サービスの有無を判断する。ここで「並列実行サービス」とは、実行順を制約するサービス間の依存関係が何も無いサービスを言う。

【 0 0 2 5 】

図 7 は、ワークフロー定義 1 4 により規定された複数のサービスの実行順序及び依存関係の一例を示す図である。図示されるように、ワークフロー定義 1 4 には、サービス S 1、S 2、S 3 がそれぞれ並列に配置され、サービス間において実行処理の前処理や後処理等を規定する依存関係を有しないように規定される。つまり、ワークフロー定義 1 4 には、サービス S 1、S 2、S 3 が並列実行サービスであるとして規定される。ワークフローの処理が開始され、各サービスにリクエストを送信すると、各サービスは結果（OK 又はエラー）を受領し、エラーの場合は終了し、OK の場合はキャンセル要求を待つ。各サービスは、キャンセル要求を受信しなければサービスリクエスト処理を終了してワークフロー処理を終了する。各サービスは、キャンセル要求を受信すれば、キャンセル処理を実行し、さらにサービスリクエスト処理を終了してワークフロー処理を終了する。なお、図示されるワークフロー定義は、B P E L 等で記述することができる。

20

【 0 0 2 6 】

並列実行サービスがあればステップ S 1 5 2 に進み、並列実行サービスがなければ実行フロー決定処理を終了して、ステップ S 1 0 8 に進む。

30

ステップ S 1 5 2 では、メッセージ解析部 2 2 は、予約リソース 9 1 2、9 2 2、9 3 2 を読み出す。なお、リクエストメッセージ 9 0 1 から予約リソース 9 1 2、9 2 2、9 3 2 を呼び出すためには、図 8 に示す読出パラメータ定義情報 1 5 の X P a t h を利用することができる。また、図 8 に示されるように、ワークフロー定義 1 4 に規定されたサービス S 1、S 2、S 3 は、読出パラメータ定義情報 1 5 内の X p a t h に含まれる「hotel」、「train」、「airplane」というサービスにそれぞれ関係付けられる。

【 0 0 2 7 】

実行フロー決定部 2 3 は、正規化コスト算出部 2 4 に対してサービス S 1、S 2、S 3 に対する正規化コストの計算を依頼する。正規化コスト算出部 2 4 は、処理コスト保持部 2 5 から、正規化コスト算出に必要なサービス S 1、S 2、S 3 の計測データ（サービス毎の処理時間、キャンセル処理時間、エラー発生確率）を呼び出す（ステップ S 1 5 3）。図 9 に、計測データの一例を示す。処理コスト保持部 2 5 は、図示されるように予約リソース毎に計測データ（サービス毎の処理時間、キャンセル処理時間、エラー発生確率）を保持する。また、これらの計測データはサービス計測部 2 6 によって適宜修正が加えられる（後述）。

40

【 0 0 2 8 】

正規化コスト算出部 2 4 は、正規化コストの計算を行う（ステップ S 1 5 4）。正規化コストとは、1 つの前記サービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化されるデータであって、サービスにエラーが発生し

50

た場合の障害の大きさを定量的に評価するための指標である。さらに、処理コストは、本実施例では、サービスの1つのエラーにより生じる他のサービスのキャンセル処理時間及び他のサービスの実行に要した処理時間として規定される。したがって、この正規化コストは、最後に実行されるサービスがエラーにより実行できなくなった場合に、当該エラーサービス以外の実行済みサービスのキャンセル処理時間と、実行済みサービスの処理時間と、エラー発生確率から計算することができる。この場合、正規化コストは、以下の式により規定することができる。

(正規化コスト) = (既に実行された他のサービスのキャンセル処理時間 + 既に実行された他のサービスの処理時間) × (エラーの発生確率)・・・(式1)

【0029】

図9は、サービス計測部26が管理する計測データの一例を示す。図9に示される計測データを用いて、正規化コスト算出部24により式1を用いて計算されるサービスS1(Hotel1)、S2(Train1)、S3(Airplane1)のそれぞれの正規化コストは、以下のように計算される。

S1(Hotel1)の正規化コスト = (S2(Train1)のキャンセル処理時間 + S2(Train1)の処理時間 + S3(Airplane1)のキャンセル処理時間 + S3(Airplane1)の処理時間) × S2(Train1)のエラー発生確率 = (50 + 100) × 0.2 = 30

S2(Train1)の正規化コスト = (S1(Hotel1)のキャンセル処理時間 + S1(Hotel1)の処理時間 + S3(Airplane1)のキャンセル処理時間 + S3(Airplane1)の処理時間) × S1(Hotel1)のエラー発生確率 = (100 + 100) × 0.1 = 20

S3(Airplane1)の正規化コスト = (S1(Hotel1)のキャンセル処理時間 + S1(Hotel1)の処理時間 + S2(Train1)のキャンセル処理時間 + S2(Train1)の処理時間) × S3(Airplane1)のエラー発生確率 = (100 + 50) × 0.03 = 4.5

【0030】

実行フロー決定部23は、計算した正規化コストをもとに、コスト降順に実行フローを確定する(ステップS155)。上記した例においては、実行フロー順は、S1、S2、S3の順になる。

このように、本実施例では、正規化コストに従って、複数のサービスの実行順序を規定することにより、補償トランザクションを最適化したワークフロー処理を提供することができる。これにより、ワークフロー実行制御装置は、サービス提供サーバのエラー発生時の処理コストを軽減することができる。また、このような補償トランザクションの最適化並びに処理コストの軽減により、ワークフロー実行制御装置は、サービス提供サーバのCPUの処理負荷を低減することができる。

また、正規化コストのみならずワークフロー定義に規定される従属関係によってサービス間の処理実行順を反映することができるため、実質的なサービスの実行手順も柔軟に規定できる。

次に、ステップS151に戻り並列実行サービスがあれば、ステップS152～S155を繰り返し、並列実行サービスが無ければ図4のステップS108に進む。

【0031】

ステップS105では、サービス提供サーバ50a～cから提供された処理結果に対して、その処理結果がエラー応答(キャンセル処理の処理結果を含む)か、OK応答か否かを判断する。エラー応答の場合は、ステップS106に進み、OK応答の場合はステップS107に進む。

ステップS106では、サービス計測部26は、エラー応答によるエラー発生確率を更新し、また、キャンセル処理が実行された場合はキャンセル処理時間を更新する。具体例としては、応答受信制御部29は、サービス提供サーバから受け取った処理結果に含まれるサービスS1の「Hotel1」呼び出し結果を得た時刻(例:2008.5.24 10:00:00.30)、処理結果(エラー)をサービス計測部26に伝える。サービス計測部26は、処理コスト保持部25のサービスS1の当該エントリ「Hotel1」の平均処理時間と、エラー発生確率を更新する。例えば、過去10回の処理時間が50ms、今回が30msの場合、(50

10

20

30

40

50

$\times 10 + 30) / 11 = 48 \text{ ms}$ に更新される。またエラー発生確率は過去10回で0.2であったとすると、 $(10 \times 0.2 + 1) / 11 = 0.27$ に更新される。宛先サービス選択部27は、リクエスト処理がエラーであったため、ワークフローにしたがって処理を終了する。

ステップS107では、実行フロー決定部23が、後述されるステップS109で更新済みの最新の実行順序を読み出す。

【0032】

ステップS108では、宛先サービス選択部27は、実行フロー決定部23から受領したサービスの実行順に従ってリクエスト対象となる宛先サービスを特定する。

実行フロー決定部23は、特定された宛先サービスによって、実行ワークフローの状態を遷移する(ステップS109)。

【0033】

図10を用いて、実行ワークフローの状態遷移図の一例を説明する。正規化コスト算出部24により、サービス実行順はS1、S2、S3となったため、ステップS201、S203、S205に示すように、サービスS1、S2、S3に対するサービス実行リクエスト送信が行われる。

ステップS202に示すように、サービスS1リクエストに対する処理結果がOKであれば、ステップS202の処理結果の判定は「Yes」となり、サービスS2のリクエスト送信(ステップS203)に対する処理結果の判定処理(ステップS204)に進む。サービスS1リクエストに対する処理結果がエラーであれば、ステップS202の処理結果の判定は「No」となり、ステップS209に進む。ステップS209では、サービスS1リクエストの処理結果が「エラー結果」であることを含むメッセージ応答をクライアント40へ送信する。

ステップS204では、サービスS2リクエストに対する処理結果がOKであれば、ステップS204の処理結果の判定は「Yes」となり、サービスS3のリクエスト送信(ステップS205)に対する処理結果の判定処理(ステップS206)に進む。サービスS2リクエストに対する処理結果がエラーであれば、ステップS203の処理結果の判定は「No」となり、ステップS208に進んでサービスS1のリクエストのキャンセル処理を実行する。ステップS209では、サービスS2リクエストの処理結果が「エラー結果」であり、サービスS1リクエストに対してはキャンセル処理を実行されたことを示すメッセージ応答をクライアント40へ送信する。

ステップS206では、サービスS3リクエストに対する処理結果がOKであれば、ステップS206の処理結果の判定は「Yes」となり、ステップS209に進む。この場合、サービスS1、S2、S3リクエストの全ての処理結果がOKであることを示すメッセージ応答をクライアント40へ送信する。

ステップS206では、サービスS3リクエストに対する処理結果がエラーであれば、ステップS206の処理結果の判定は「No」となる。この場合、ステップS207に進んでサービスS2のリクエストのキャンセル処理を実行し、さらに、ステップS208に進んでサービスS1のリクエストのキャンセル処理を実行する。ステップS209では、サービスS3リクエストの処理結果が「エラー結果」であり、サービスS1、S2リクエストに対してはキャンセル処理を実行されたことを含むメッセージ応答を送信する。

上記ステップS202、S204、S206は、サービス提供サーバからの処理結果メッセージの受信(ステップS102)をイベントとして実行フロー決定部23により行われて、実行フロー状態として管理される。この実行フロー状態は、上述したようにステップS107で、読み込まれる。

【0034】

実行フロー決定部23は、実行フロー状態を監視して、全サービスについて呼び出し処理(リクエスト送信処理)が行われたか否かを判断する(ステップS110)。全サービスについて呼び出し処理が行われた場合は、ステップS115に進み、全サービスについて呼び出し処理が行われなかった場合は、ステップS111に進む。

ステップ S 1 1 1 では、サービス呼出部 2 8 が、サービス S 1、S 2、S 3 のいずれかに対するサービス実行リクエストを送信する。なお、このサービス実行リクエストには、サービス S 1、S 2、S 3 のいずれかに対応するリクエスト仕様情報 9 1 3、9 2 3、9 3 3 が含まれる。

【 0 0 3 5 】

ステップ S 1 1 5 では、ステップ S 2 0 9 の処理と同様に、ワークフロー実行制御装置 1 0 a からクライアント 4 0 に対して、リクエストメッセージに含まれたワークフローが適切に実行されたか、又は、どのサービスがエラーであったかを示すメッセージ応答を送信する。クライアント 4 0 は、ワークフロー実行制御装置 1 0 a からのメッセージ応答を受信する（ステップ S 1 1 6 ）。

10

ステップ S 1 1 7 では、サービス呼出部 2 8 は、リクエスト送信時刻（例えば、2008.5.24 10:00:00.00）をサービス計測部 2 6 に伝えて、ワークフロー実行制御装置 1 0 a によるワークフロー実行制御の処理フローを終了する。

【 0 0 3 6 】

図 3 及び図 4 を用いて、サービス提供サーバ 5 0 によるサービス処理の処理フローの一例について説明する。

サービス提供サーバ 5 0 a ~ c は、ワークフロー実行制御装置 1 0 a から送信されるサービス実行リクエストに従って、サービス処理を開始する。

まず、サービス提供サーバ 5 0 a ~ c は、ワークフロー実行制御装置 1 0 a からサービス実行リクエストを受信し（ステップ S 1 1 2 ）、サービス処理をサービス実行リクエストに従って実行する（ステップ S 1 1 3 ）。次に、サービス提供サーバ 5 0 a ~ c は、サービス処理の処理結果（OK 処理又はエラー処理）をワークフロー実行制御装置 1 0 a に送信して（ステップ S 1 1 4 ）、サービス処理を終了する。

20

【 0 0 3 7 】

[第 2 の実施形態]

図 1 1 を用いて、第 2 の実施形態に係るワークフロー実行制御装置 1 0 b について説明する。

ワークフロー実行制御装置 1 0 b は、第 1 の実施形態に係るワークフロー実行制御装置 1 0 a と同じ構成を有するとともに、さらに、正規化するコスト定義を設定し且つ保持するために使用されるコスト定義を保持するためのコスト定義保持部 3 0 を有する。

30

処理部 2 0 内のメモリ又は記憶部 1 2 は、コスト定義保持部 3 0 として機能することで、ワークフロー実行制御装置 1 0 b はコスト定義を保持する。例えば、コスト定義には、リソース予約の場合、リソースのキャンセル発生期限（以下、「キャンセル期限」と称す）と、キャンセルコスト、リクエストメッセージ中の予約日を示す場所が X P a t h で定義される（後述）。なお、キャンセルコストは、サービス提供サーバ運営者とキャンセル時支払コストの取り決め等に従い、ワークフロー実行制御装置 1 0 b の管理者が図示しない入力部を介して手動で入力することができる。

また、メッセージ解析部 2 2 は、コスト定義保持部 3 0 に含まれる定義にしたがって、リクエストメッセージ中のコスト計算に必要なパラメータ（例：リソースの予約日）を読み出し、実行フロー決定部 2 3 に伝えることができる。さらに、正規化コスト算出部 2 4 は、コスト定義保持部 3 0 が保持するコスト定義に従って、エラー発生時の処理コストを算出する。なお、本実施形態における処理コストは、サービスの 1 つのエラーにより生じる他のサービスのキャンセルコストである。

40

また、正規化コスト算出部 2 4 は、処理コスト保持部 2 5 から各サービスのエラー発生確率を読み出し、各サービスの正規化コストを算出し、実行フロー決定部 2 3 に渡す。

【 0 0 3 8 】

図 1 2 を用いて、ワークフロー実行制御の処理フローの一例について説明する。

図 1 2 に示すステップ S 1 0 1 ~ 1 0 9 は、図 3 に示すステップ S 1 0 1 ~ S 1 0 9 と同じであるため、説明を省略する。実行フロー決定処理（S 1 5 1 ~ S 1 6 4）が、第 1 の実施形態と異なる。

50

【 0 0 3 9 】

図 1 3 を用いて、実行フロー決定処理のフローの一例を説明する。

図 1 3 に示すステップ S 1 5 1 及び S 1 5 2 は、図 6 に示すステップ S 1 5 1 及び S 1 5 2 と同じであるため、説明を省略する。

ステップ S 1 6 1 では、メッセージ解析部 2 2 は、コスト定義保持部 3 0 のコスト定義を読み出す。図 1 4 を用いて、キャンセル時に発生するコストの効果を評価するためのコスト定義の一例を説明する。図示されるように、「キャンセル期限」、「キャンセルコスト」、リクエストメッセージ中の予約日を示す場所が X P a t h で定義される。本実施例では、予約先リソースの時刻情報、現在時刻（例：2008-5-24 10:00:00）から、「キャンセル期限」が決められる。

10

メッセージ解析部 2 2 は、受信したリクエストメッセージ中の予約時刻情報を読み出す（ステップ S 1 6 2）。読み出し結果は、例えば、Hotel1=2008-5-29, Train1=2008-5-29 10:00:00, Airplane1=2008-5-29 14:00:00となる。

【 0 0 4 0 】

正規化コスト算出部 2 4 は、各サービス S 1 (Hotel1)、S 2 (Train1)、S 3 (Airplane1) に対する正規化コストの計算を行う。

なお、上記のように、処理コストは、サービスの 1 つのエラーにより生じる他のサービスのキャンセルコストとして規定される。したがって、この正規化コストは、最後に実行されるサービスがエラーにより実行できなくなった場合に、当該エラーサービス以外の実行済みサービスのキャンセルコストと、エラー発生確率から計算する。この場合、正規化コストは、以下の式により規定する。

20

（正規化コスト）＝（既に実行された他のサービスのキャンセルコスト）×（エラーの発生確率）・・・（式 2）

【 0 0 4 1 】

各サービス S 1 の正規化コストは、他のサービス S 2、S 3 の処理コストの和に、S 1 のエラー発生確率を掛けた値とする。本実施例では、S 2 および S 3 のみキャンセルコストが発生する為、各サービスの正規化コストは、 $S 1 = (20000 + 5000) \times 0.2 = 5000$ 、 $S 2 = (0 + 20000) \times 0.1 = 2000$ 、 $S 3 = (0 + 5000) \times 0.03 = 150$ 、となる。

【 0 0 4 2 】

30

実行フロー決定部 2 3 は、上記計算した正規化コストをもとに、コスト降順にフローを確定する。本実施例では、S 1、S 2、S 3 の順に実行すると決定する（ステップ S 1 6 4）。

次に、ステップ S 1 5 1 に戻り並列実行サービスがあれば、ステップ S 1 6 2 ～ S 1 5 6 を繰り返し、並列実行サービスが無ければ図 1 2 のステップ S 1 0 8 に進む。

このように、本実施例では、「キャンセル期限」、「キャンセルコスト」等のコスト定義を用いて算出した正規化コストに従って、複数のサービスの実行順序を規定することにより、補償トランザクションを最適化したワークフロー処理を提供することができる。これにより、ワークフロー実行制御装置は、サービス提供サーバのエラー発生時の処理コストを軽減する。

40

【 0 0 4 3 】

[第 3 の実施形態]

第 3 の実施形態に係るワークフロー実行制御装置 1 0 c は、第 1 の実施形態に係るワークフロー実行制御装置 1 0 a と同じ構成を有する。

本実施例では、サービスをグループ化したワークフロー定義により、サービス実行順を決定する。図 1 5 を用いて、ワークフロー定義 1 4 の一例を説明する。図示のように、ワークフロー定義 1 4 では、グループ G 1、G 2、G 3 は互いに独立な関係を有するが、グループ G 1 内のサービス S 1 及び S 4 が従属的な関係をもって定義される。また、実施例として、サービス S 4 は、サービス S 1 の処理結果をメール通知するサービスとし、サービス S 4 はサービス S 1 の直後に実行されるものとする。なお、図示されるワークフロー

50

定義は、B P E L 等で記述することができる。

【 0 0 4 4 】

また、正規化コスト算出部 2 4 及び実行フロー決定部 2 3 は、第 1 の実施形態と比して下記に示す機能をさらに有する。

正規化コスト算出部 2 4 は、複数サービスがグループ化されて定義されていた場合、グループ化されたサービス群を 1 つのサービスとして正規化コストを計算し、実行フロー決定部に渡す。

実行フロー決定部 2 3 は、正規化コスト算出部 2 4 のコスト計算結果を受け取り、グループ間の実行順を決定する。決定した最終的なサービスの実行順を、宛先サービス選択部 2 7 に通知する。

【 0 0 4 5 】

なお、本処理は、第 1 の実施形態と比して実行フロー決定処理が異なるので、実行フロー決定処理を、図 1 6 を用いて説明する。また、図 1 5 に示すステップ S 1 5 1 及び S 1 5 2 は、図 6 に示すステップ S 1 5 1 及び S 1 5 2 と同じであるため、説明を省略する。

【 0 0 4 6 】

ステップ S 1 7 1 では、メッセージ解析部 2 2 は、受信したリクエストメッセージ中の予約リソース Hotel1, Train1, Airplane1 の情報を読み出す。なお、サービス S 4 については読み出す場所の定義がないため実行されない。

【 0 0 4 7 】

実行フロー決定部 2 3 は、w f 1 で入れかえ可能な処理として定義されている、グループ G 1 (サービス S 1 , S 4)、グループ G 2 (サービス S 2)、グループ G 3 (サービス S 3) に対して正規化コスト算出部 2 4 に正規化コストを計算する為の依頼をする。

正規化コスト算出部 2 4 は、各グループ G 1、G 2、G 3 に対する正規化コストの計算を行う (ステップ S 1 7 2)。本実施例では、各々の実行時間、キャンセル処理時間、N G 確率がそれぞれ図 9 のような関係になっているものとする。例えばグループ G 2 の正規化コストは、他のサービス S 1 , S 3 , S 4 の処理コストの和に、S 2 のエラー発生確率を掛けた値とする。この場合、各サービスの正規化コストは、 $G 1 = (5 0 + 1 0 0) \times (0 . 2 + 0 . 1) = 4 . 5$ 、 $G 2 = (1 0 0 + 1 0 0 + 1 . 5) \times 0 . 1 = 2 1 . 5$ 、 $G 3 = (1 0 0 + 5 0 + 1 5) = 4 . 9 5$ となる。

実行フロー決定部 2 3 は、上記計算した正規化コストをもとに、コスト降順にフローを確定する。本実施例では、G 1 , G 2 , G 3 (S 1 , S 4 , S 2 , S 3) の順に実行すると決定する (ステップ S 1 7 3)。

【 0 0 4 8 】

このように、本実施例では、正規化コストに従って、グループ化によりサービスの依存関係を考慮してサービスの実行順序を規定することもできる。これにより、複数サービス間の呼び出し順に制約がある状況下でも、ワークフロー実行制御装置は、補償トランザクションを最適化したワークフロー処理の提供、サービス提供サーバのエラー発生時の処理コストを軽減することができる。また、このような補償トランザクションの最適化並びに処理コストの軽減により、ワークフロー実行制御装置は、サービス提供サーバの C P U の処理負荷を低減することができる。

【 0 0 4 9 】

[第 4 の実施形態]

図 1 7 を用いて、第 4 の実施形態に係るワークフロー実行制御装置 1 0 d について説明する。

ワークフロー実行制御装置 1 0 d は、第 1 の実施形態に係るワークフロー実行制御装置 1 0 a と同じ構成を有する。

また、サービス計測部 2 6 は、第 1 の実施形態と比して、各サービスの処理結果を受け取る際に、該メッセージ内に含まれる処理時間情報を処理コストとして処理コスト保持部に記録する機能をさらに有する。

また、サービス提供サーバ 5 0 a ~ c 処理部は、記憶部に格納されたプログラムを実行

10

20

30

40

50

することで、計測エージェント部 5 1 として機能する。計測エージェント部 5 1 は、ワークフロー実行制御装置 1 0 d からリクエストメッセージを受信後、サービスの実行開始から実行完了までのサービス提供プログラム 5 2 を実行する CPU 実行時間を計測する。サービス提供サーバ 5 0 a ~ c は、処理結果に加えて上記 CPU 実行時間を、ワークフロー実行制御装置 1 0 d に送信する。

【 0 0 5 0 】

図 1 8 を用いて、サービス提供サーバの処理の一例を説明する。

サービス呼出部 2 8 は、サービス S 1 の「Hotel1」予約の呼び出しを行う為に、リクエストをサービス S 1 のサービス提供サーバ 5 0 に送付する。また、予約先のリソース (Hotel1)、XPath の定義 (/request/hotel/name/text()) をサービス計測部 2 6 に伝える。サービス提供サーバ 5 0 は上記サービス実行リクエストを待つ (ステップ S 1 8 1)。サービス提供サーバ 5 0 は、サービス実行リクエストを受け取ると、サービスを実行し (ステップ S 1 8 2)、計測エージェント部 6 1 が、サービス提供プログラムの処理の実行開始から実行完了までの CPU 実行時間を計測する (ステップ S 1 8 3)。本実施例では、3 0 m s の実行時間がかかったものとする。

【 0 0 5 1 】

サービス提供サーバ 5 0 は処理結果に加えて上記 CPU 実行時間を、ワークフロー実行制御装置 1 0 d に対して応答する (ステップ S 1 8 4)。

ワークフロー実行制御装置 1 0 d の応答受信制御部 2 9 は、サービス提供サーバ 5 0 の CPU 実行時間並びに処理結果を受け取り、例えば、サービス S 1 の処理に要する CPU 実行時間 (3 0 m s) と、処理結果 (エラー) をサービス計測部 2 6 に伝える。サービス計測部 2 6 は、上記実行時間 (3 0 m s) を用いて処理コストを算出し、処理コスト保持部 2 5 のサービス S 1 の該当エントリを更新する。

【 0 0 5 2 】

以上より、サービス提供サーバ 5 0 側のプログラムの実行時間を計ることができるため、例えばサービス実行前、実行後にサービス提供サーバ内で一時的に処理待ちキュー等の待ち時間が入った場合でも正確にコスト計測ができるようになる。

【 0 0 5 3 】

[第 5 の実施形態]

図 1 9 を用いて、第 5 の実施形態に係るワークフロー実行制御装置 1 0 e について説明する。

ワークフロー実行制御装置 1 0 e は、第 1 の実施形態に係るワークフロー実行制御装置 1 0 a と同じ構成を有する。また、ワークフロー実行制御装置 1 0 e は、ネットワーク 5 を介してワークフロー管理サーバ 6 0 に接続される。

また、フロー定義識別部 2 1 は、第 1 の実施形態と比して、クライアント 4 0 から受信したリクエストメッセージの内容を解析して、実行するワークフローを識別する機能をさらに有する。さらに、フロー定義識別部 2 1 は、識別したワークフローをワークフロー管理サーバ 6 0 に対して通信部 3 8 を介して問い合わせるためのリクエストを送信する。

また、ワークフロー管理サーバ 6 0 は、ワークフロー定義を一元管理するためのサーバである。ワークフロー管理サーバ 6 0 は、フロー定義識別部 2 1 からのリクエストに対して、指定されたワークフロー定義の応答を行うことができる。

【 0 0 5 4 】

以下、クライアントからのリクエストメッセージ受信時の動作説明を行う。クライアント 4 0 が、ワークフロー実行制御装置 1 0 e 宛にリクエストメッセージを送信する。リクエストの内容は、図 3 のステップ S 1 0 1 で送信されるメッセージと同様の内容とする。ワークフロー実行制御装置 1 0 e がリクエストを受信すると、フロー定義識別部 2 1 がワークフローの識別を行い、リクエストメッセージに指定された「w f 1」のワークフローを実行すると判断する。フロー定義識別部 2 1 は、ワークフロー管理サーバ 6 0 宛に「w f 1」のワークフローを取得するためのリクエストを送信し、ワークフロー管理サーバ 6 0 は、「w f 1」のワークフロー定義をワークフロー実行制御装置 1 0 e に対して応答する

。フロー定義識別部 21 は、受信したワークフロー定義をリクエストメッセージに対するワークフロー定義として適用する。以下、図 3 のステップ S103 と同様の処理を実行するため、説明を省略する。

なお、ワークフロー実行制御装置 10e は、上記ワークフロー管理サーバ 60 から取得したワークフロー定義は、一定期間キャッシュとして保持し、再利用しても良い。

【0055】

このように、複数のワークフロー実行制御装置 10 を並列に配置した場合に、各サーバに対してワークフローの定義を配信することなく、ワークフロー管理サーバ 60 で一元管理が可能となり、ワークフロー定義変更時のメンテナンスコストを削減できる。

【0056】

10

[第 6 の実施形態]

図 20 を用いて、第 6 の実施形態に係るクライアント 40a について説明する。

クライアント 40a は、プログラム配信サーバ 70 にネットワーク接続されている。第 1 ~ 5 の実施形態で説明したワークフロー実行制御装置の機能は、ワークフロー制御プログラムとしてクライアント 40a 内に配置し、クライアント 40a 内ブラウザ上でワークフロー制御プログラムを実行する。

ワークフロー制御プログラムは、別途プログラム配信サーバ 70 上に設置されており、例えばクライアント上で動作する Java (登録商標) プログラムとして配信される。

【0057】

20

以下、クライアントにおけるワークフロー制御プログラムの取得方法について説明する。クライアント 40a 上のブラウザからプログラム配信サーバ 70 上にリクエストを送信し、プログラム配信サーバ 70 上のワークフロー制御プログラムを取得する。ワークフロー制御プログラムに対して、ブラウザからサービス実行リクエストを受信する。この場合、同一マシン内のデータ渡しであるため、XML のメッセージ形式ではなく、プログラム呼び出しの引数等でワークフロー名や予約リソース名を指定しても良い。

リクエスト受信後の処理は、実施形態 1 と同様である為説明を省略する。

なお、ワークフロー制御プログラムは、リクエスト発行前に事前にプログラム配信サーバ 70 経由で取得しても良い。

【0058】

30

このように、ワークフロー実行制御装置を設置することなく、各クライアント側で複数サービスの呼び出しを行い、かつそのサービスのエラー発生時の処理コストを削減することが可能となる。また、このような補償トランザクションの最適化並びに処理コストの軽減により、クライアントは、サービス提供サーバの CPU の処理負荷を低減することができる。

【0059】

[第 7 の実施形態]

第 7 の実施形態に係るワークフロー実行制御装置 10f を説明する。クライアント 40 からのリクエストメッセージが、バイナリメッセージであった場合に対応する為に、メッセージ解析部 22 は、以下の機能を有しても良い。

メッセージ解析部 22 は、あらかじめ設定しておいた、メッセージの特定場所を読み込む定義 (バイト位置とサイズ) に基づいて、メッセージ中の指定された場所からパラメータ (例: 予約先ホテル名) を読み出し、実行フロー決定部 23 に伝える。

40

【0060】

図 21 に、リクエストメッセージの電文フォーマットの一例を示す。実施形態 1 のサービス実行順の決定処理 (ステップ S151 ~ S155) において、リクエストメッセージが例えば図 17 のような形態 (伝えられる情報は実施形態 1 と同じ内容) であるとする。そうすると、ステップ S152 のサービス呼出において、メッセージ解析部 22 は、受信したリクエストメッセージ中の予約リソース Hotel1, Train1, Airplane1 の情報を読み出す。

図 22 に、読出パラメータ定義情報の一例を示す。リクエストメッセージ中の予約リソ

50

ース情報を読み出すとき、場所を指定する方法として、図 2 2 に示されるように、例えば、メッセージ先頭からのバイト位置、その場所からのバイト数（サイズ）による指定を行っても良い。このようにすることで、ワークフロー実行制御装置に届くメッセージがバイナリデータであった場合でも、ワークフローの順序制御は可能である。

【 0 0 6 1 】

〔 第 8 の実施形態 〕

第 8 の実施形態に係るワークフロー実行制御装置 1 0 g を説明する。クライアント 4 0 からのリクエストメッセージが、例えば C S V (Comma Separated Values) 形式のような、表形式のメッセージであった場合に対応する必要がある。その場合、メッセージ解析部 2 2 は、あらかじめ設定しておいた、メッセージの特定場所を読み込む定義（行と列の位置）に基づいて、メッセージ中の指定された場所からパラメータ（例：予約先ホテル名）を読み出して、実行フロー決定部に伝える。

10

図 2 3 にリクエストメッセージの一例を示す。リクエストメッセージが、例えば、図 2 3 のような形態（伝えられる情報は実施形態 1 と同じ内容）であった場合について示す。

実施形態 1 のサービス実行順の決定処理（ステップ S 1 5 1 ～ S 1 5 5 ）において、ステップ S 1 5 2 において、メッセージ解析部 2 2 は、受信したリクエストメッセージ中の予約リソース「Hotel1」、「Train1」、「Airplane1」の情報を読み出す。

図 2 4 に、読出パラメータ定義情報の一例を示す。リクエストメッセージ中の予約リソース情報を読み出すとき、場所を指定する方法として、図 2 4 に示されるように、メッセージの行および列の位置による指定を行っても良い。このようにすることで、ワークフロー実行制御装置 1 0 に届くメッセージが C S V 等の表形式のデータであった場合でも、ワークフローの順序制御が可能になる。

20

【 0 0 6 2 】

以上述べた実施態様は、以下の付記の通りである。

（付記 1）

1 台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御プログラムであって、前記 1 台以上のサービス提供サーバに接続されるコンピュータを、

前記サービス毎の処理結果を用いて、前記サービス毎の処理コスト及びエラーの発生確率を算出するサービス計測部と、

30

1 つの前記サービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて、正規化される正規化コストを、前記サービスの各々について算出する正規化コスト算出部と、

前記算出された正規化コストに従って、前記サービスの実行順序を決定する実行フロー決定部と、

前記サービス提供サーバに対して、前記実行順序に従って前記サービスを順次呼び出すサービス呼出部として機能させることを特徴とするワークフロー実行制御プログラム。

（付記 2）

前記正規化コスト算出部は、前記複数のサービスの依存関係を定義するワークフロー定義を用いて、前記正規化コストを算出する付記 1 に記載のワークフロー実行制御プログラム。

40

（付記 3）

前記正規化コスト算出部は、前記サービスの依存関係を定義するワークフロー定義を用いて、依存関係を有するサービスのグループ単位に前記正規化コストの計算を行い、

前記実行フロー決定部は、前記グループ単位の正規化コストに従って、グループ単位の実行順序を決定する付記 1 又は 2 に記載のワークフロー実行制御プログラム。

（付記 4）

前記サービス計測部は、前記各サービスを実行するサービス提供サーバから前記各サービスを実行するために要した処理時間をさらに取得して、該処理時間に基づいて前記サービス毎の処理コストを算出する付記 1 ～ 3 のいずれかに記載のワークフロー実行制御プロ

50

グラム。

(付記 5)

前記コンピュータを、リクエストメッセージ中の特定の場所から情報単位を抜き出すメッセージ解析部としてさらに機能させ、

前記正規化コスト算出部は、前記抜き出した情報単位毎に正規化コストを算出し、

前記サービス計測部は、前記情報単位毎に前記処理コストと前記エラー発生確率を算出する付記 1 ~ 4 のいずれかに記載のワークフロー実行制御プログラム。

(付記 6)

前記処理コストは、前記サービスの 1 つのエラーにより生じる他のサービスのキャンセル処理時間及び該他のサービスの実行に要した処理時間である付記 1 ~ 5 のいずれかに記載のワークフロー実行制御プログラム。

(付記 7)

前記処理コストは、前記サービスの 1 つのエラーにより生じる他のサービスのキャンセルコストである付記 1 ~ 6 のいずれか 1 項に記載のワークフロー実行制御プログラム。

(付記 8)

1 台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御装置であって、

前記サービス毎にエラーの発生確率及び処理コストを蓄積する処理コスト保持部と、

前記サービス毎の処理結果を用いて、前記サービス毎の処理コスト及びエラーの発生確率を算出するサービス計測部と、

1 つの前記サービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化される正規化コストを、前記サービスの各々について算出する正規化コスト算出部と、

前記算出された正規化コストに従って、前記サービスの実行順序を決定する実行フロー決定部と、

前記サービス提供サーバに対して、前記実行順序に従って前記サービスを順次呼び出すサービス呼出部と、

を有することを特徴とするワークフロー実行制御装置。

(付記 9)

前記正規化コスト算出部は、前記複数のサービスの依存関係を定義するワークフロー定義を用いて、正規化コストを算出する付記 8 に記載のワークフロー実行制御装置。

(付記 10)

前記正規化コスト算出部は、前記サービスの依存関係を定義するワークフロー定義を用いて、依存関係を有するサービスのグループ単位に正規化コストの計算を行い、

前記実行フロー決定部は、前記グループ単位の正規化コストに従って、グループ単位の実行順を決定する付記 8 又は 9 に記載のワークフロー実行制御装置。

(付記 11)

前記サービス計測部は、前記各サービスを実行するサービス提供サーバから前記各サービスを実行するために要した処理時間をさらに取得して、該処理時間に基づいて前記サービス毎の処理コストを算出する付記 8 ~ 10 のいずれかに記載のワークフロー実行制御装置。

(付記 12)

リクエストメッセージ中の特定の場所から情報単位を抜き出すメッセージ解析部をさらに有し、

前記正規化コスト算出部は、前記抜き出した情報単位毎に正規化コストを算出し、

前記サービス計測部は、前記情報単位毎に前記処理コストと前記エラー発生確率を算出し、

前記処理コスト保持部は、前記情報単位毎に前記エラー発生確率と前記処理コストを保持する付記 8 ~ 11 のいずれかに記載のワークフロー実行制御装置。

(付記 13)

10

20

30

40

50

前記処理コストは、前記サービスの１つのエラーにより生じる他のサービスのキャンセル処理時間及び該他のサービスの実行に要した処理時間である付記８～１２のいずれかに記載のワークフロー実行制御装置。

（付記１４）

前記処理コストは、前記サービスの１つのエラーにより生じる他のサービスのキャンセルコストである付記８～１３のいずれかに記載のワークフロー実行制御装置。

（付記１５）

１台以上のサービス提供サーバにより提供される複数のサービスを、順次呼び出すワークフロー実行制御方法であって、

前記サービス提供サーバにおけるサービスの処理結果を取得して、前記サービス毎の処理コスト及びエラーの発生確率を算出するステップと、

１つの前記サービスのエラーにより生じる他のサービスの処理コストと、当該エラーの発生確率とを用いて正規化される正規化コストを、前記サービスの各々について算出するステップと、

前記算出された正規化コストに従って、前記サービスの実行順序を決定するステップと、

前記サービス提供サーバに対して、前記実行順序に従って前記サービスを順次呼び出すステップと、

を有することを特徴とするワークフロー実行制御方法。

【図面の簡単な説明】

【００６３】

【図１】ワークフロー実行制御装置の接続構成の概要を示す図である。

【図２】ワークフロー実行制御装置の一例を示す図である。

【図３】ワークフロー実行制御の処理フローの一例を示す図である。

【図４】ワークフロー実行制御の処理フローの一例を示す図である。

【図５】リクエストメッセージの内容の一例を示す図である。

【図６】実行フロー決定処理のフローの一例を示す図である。

【図７】実行ワークフロー定義の一例を示す図である。

【図８】読出パラメータ定義情報の一例を示す図である。

【図９】計測データの一例を示す図である。

【図１０】実行ワークフローの状態遷移の一例を示す図である。

【図１１】ワークフロー実行制御装置の一例を示す図である。

【図１２】ワークフロー実行制御の処理フローの一例を示す図である。

【図１３】実行フロー決定処理のフローの一例を示す図である。

【図１４】読出パラメータ定義情報の一例を示す図である。

【図１５】実行ワークフロー定義の一例を示す図である。

【図１６】実行フロー決定処理のフローの一例を示す図である。

【図１７】ワークフロー実行制御装置の一例を示す図である。

【図１８】サービス提供サーバの処理の一例を示す図である。

【図１９】ワークフロー実行制御装置の一例を示す図である。

【図２０】クライアントの一例を示す図である。

【図２１】リクエストメッセージの電文フォーマットの一例を示す図である。

【図２２】読出パラメータ定義情報の一例を示す図である。

【図２３】リクエストメッセージの内容の一例を示す図である。

【図２４】読出パラメータ定義情報の一例を示す図である。

【符号の説明】

【００６４】

１０、１０ａ～ｇ ワークフロー実行制御装置

２１ フロー定義識別部

２２ メッセージ解析部

10

20

30

40

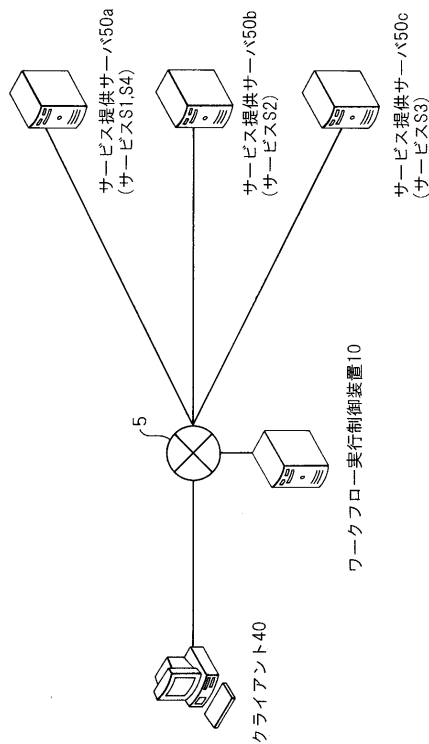
50

- 2 3 実行フロー決定部
- 2 4 正規化コスト算出部
- 2 5 処理コスト保持部
- 2 6 サービス計測部
- 2 7 宛先サービス選択部
- 2 8 サービス呼出部
- 2 9 応答受信制御部
- 4 0 クライアント
- 5 0 a ~ c サービス提供サーバ
- 6 0 ワークフロー管理サーバ
- 7 0 プログラム配信サーバ

10

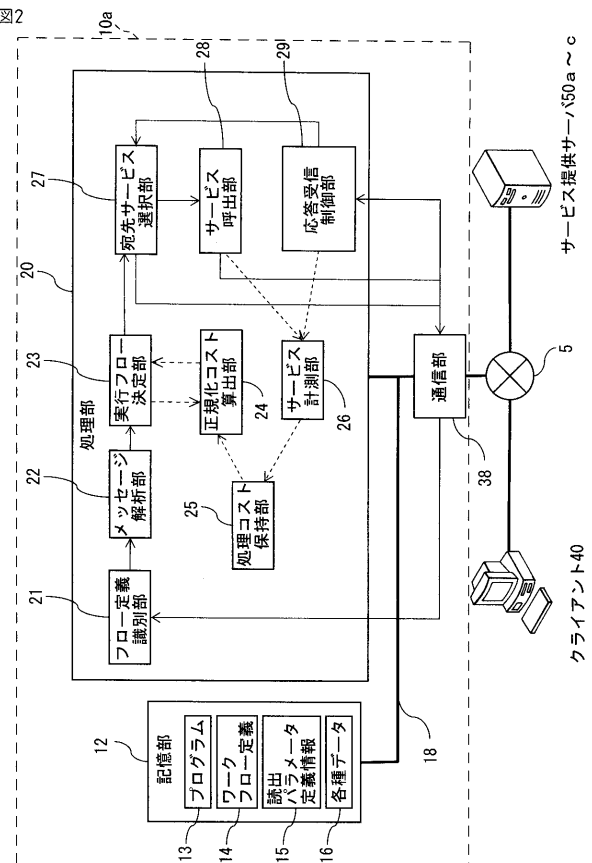
【図 1】

図1



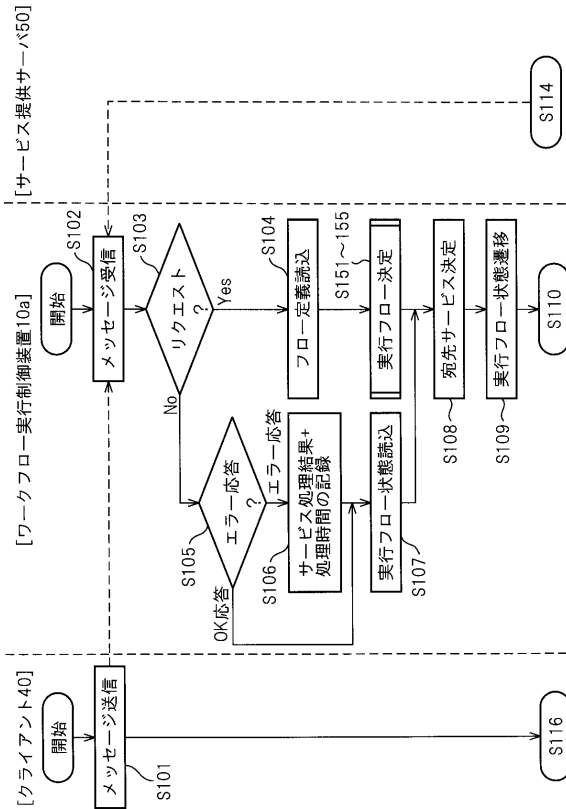
【図 2】

図2



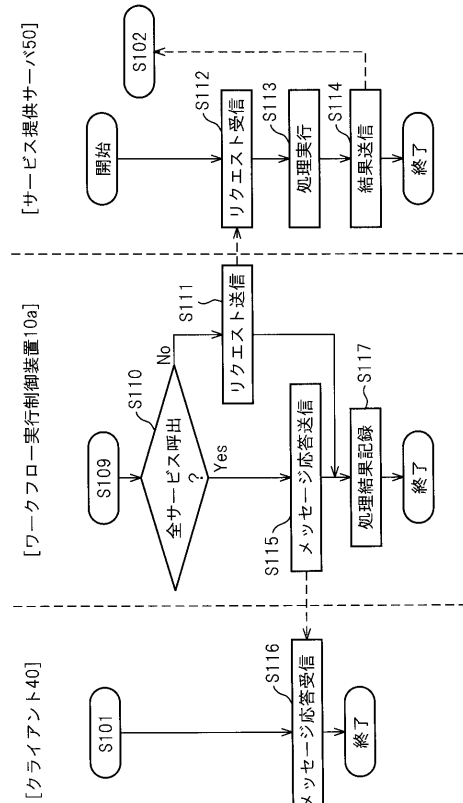
【図 3】

図3



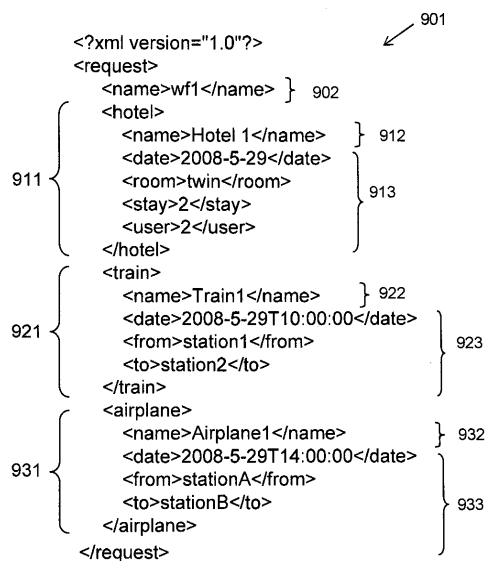
【図 4】

図4



【図 5】

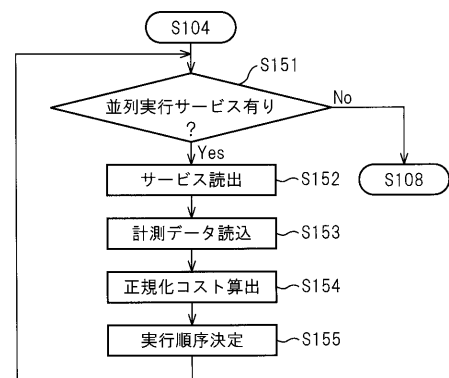
図5



【図 6】

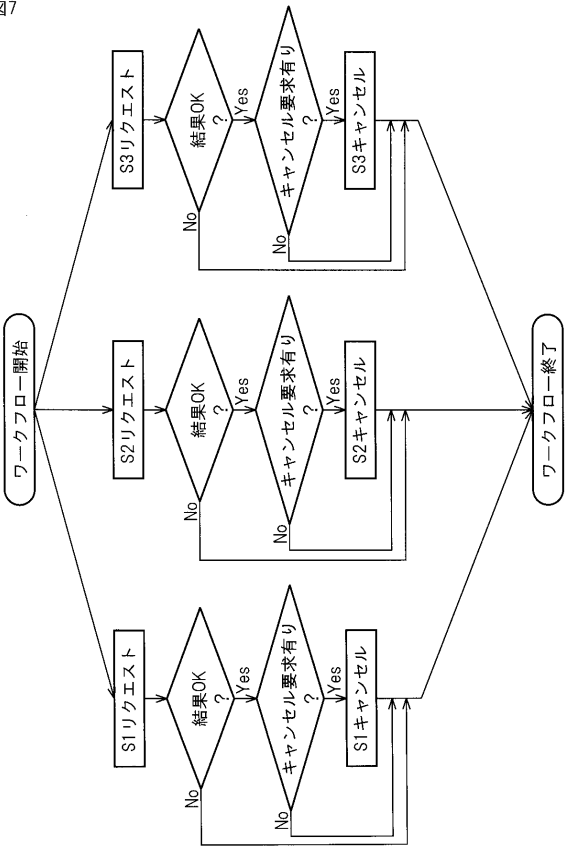
図6

[サービス提供サーバ50]



【図 7】

図7



【図 8】

図8

ワークフロー名	サービス	識別子(XPath)
wf1	S1	/request/hotel/name/text()
wf1	S2	/request/train/name/text()
wf1	S3	/request/airplane/name/text()

【図 9】

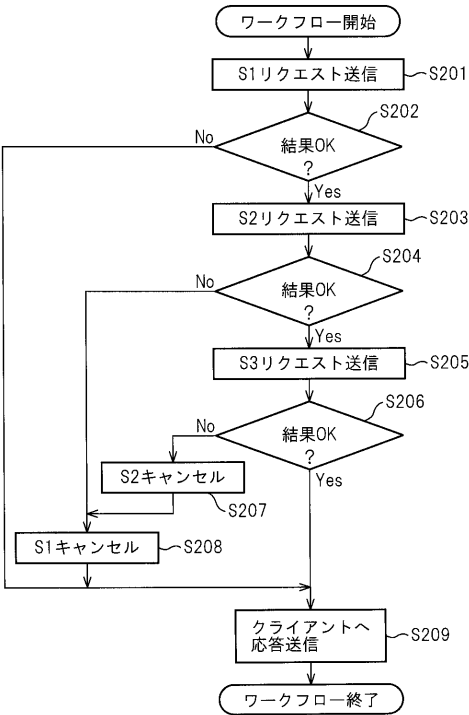
図9

サービス	定義	情報単位	実行処理時間	キャンセル処理時間	エラー確率
S1	-	-	65ms	40ms	0.15
S1	/request/hotel/name/text()	Hotel1	60ms	40ms	0.2
S1	/request/hotel/name/text()	Hotel2	70ms	40ms	0.01
S2	-	-	15ms	35ms	0.01
S2	/request/train/name/text()	Train1	15ms	35ms	0.1
S3	-	-	55ms	45ms	0.01
S3	/request/airplane/name/text()	Airplane1	55ms	45ms	0.03
S4	-	-	10ms	5ms	0.1

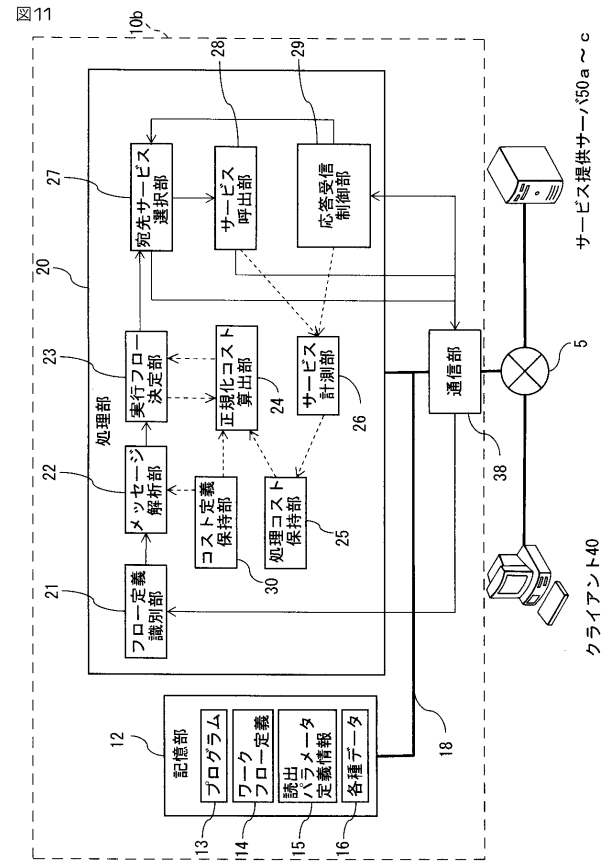
【図 10】

図10

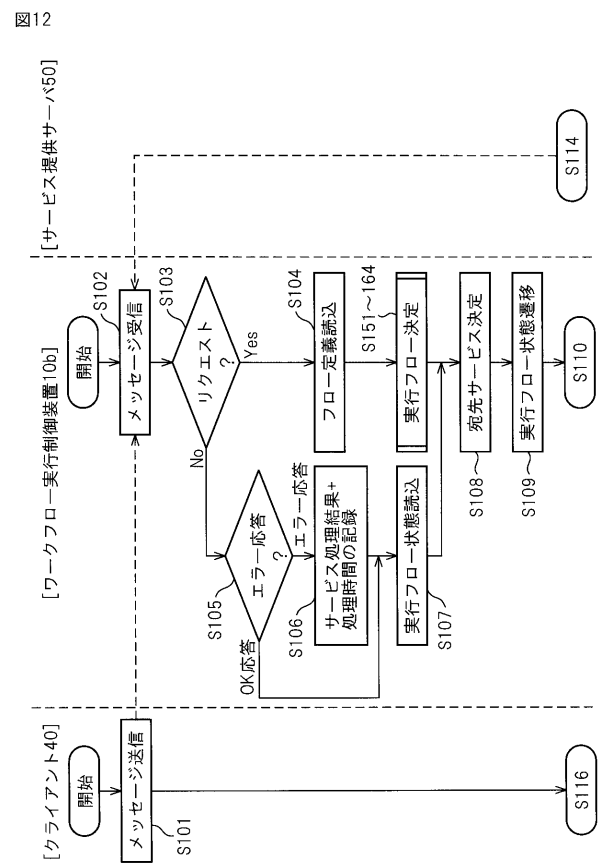
[ワークフロー実行制御装置10a]



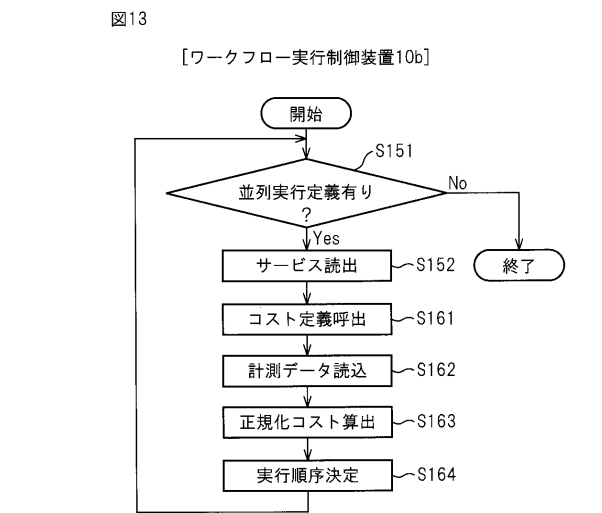
【図 1 1】



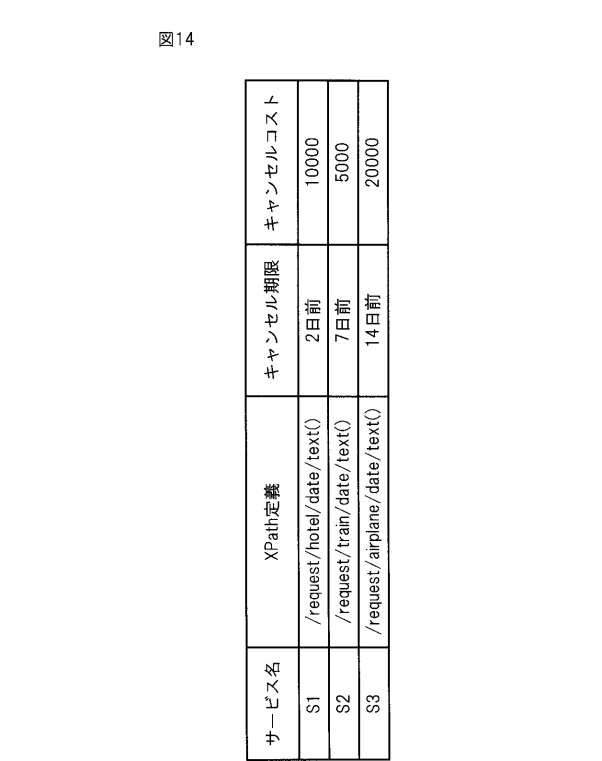
【図 1 2】



【図 1 3】

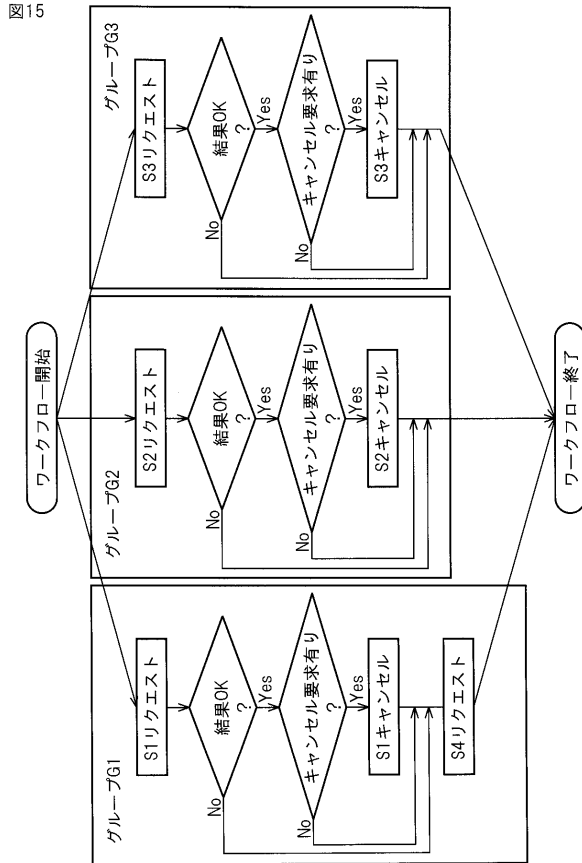


【図 1 4】

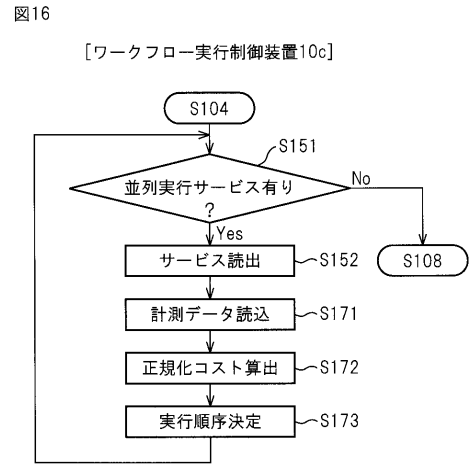


サービス名	XPath定義	キャンセル期限	キャンセルコスト
S1	/request/hotel/date/text()	2日前	10000
S2	/request/train/date/text()	7日前	5000
S3	/request/airplane/date/text()	14日前	20000

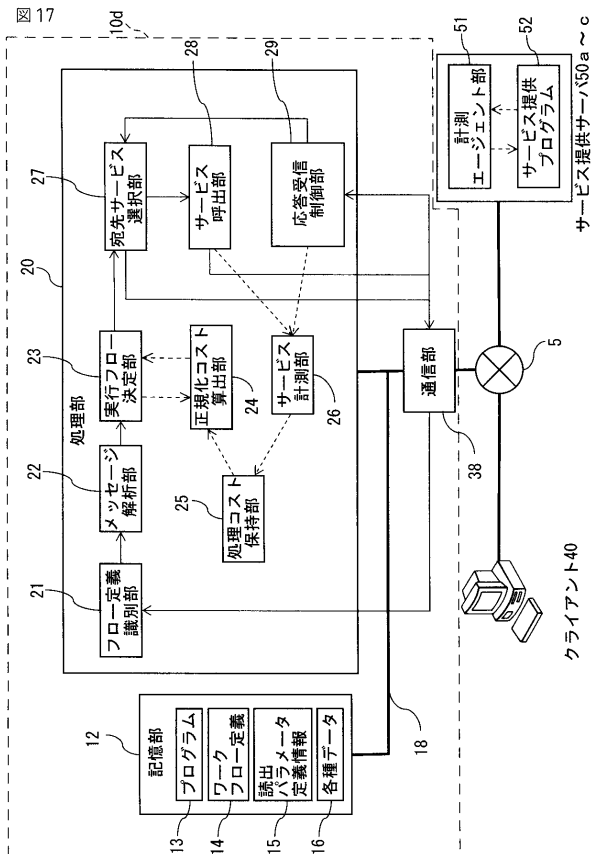
【図 15】



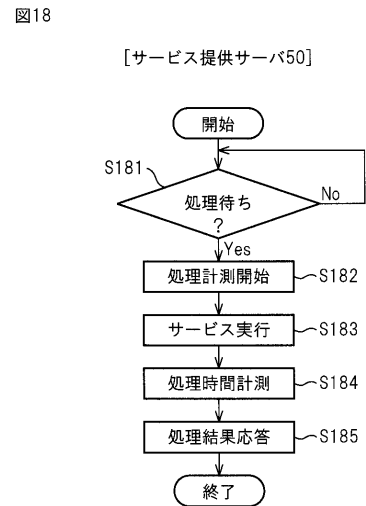
【図 16】



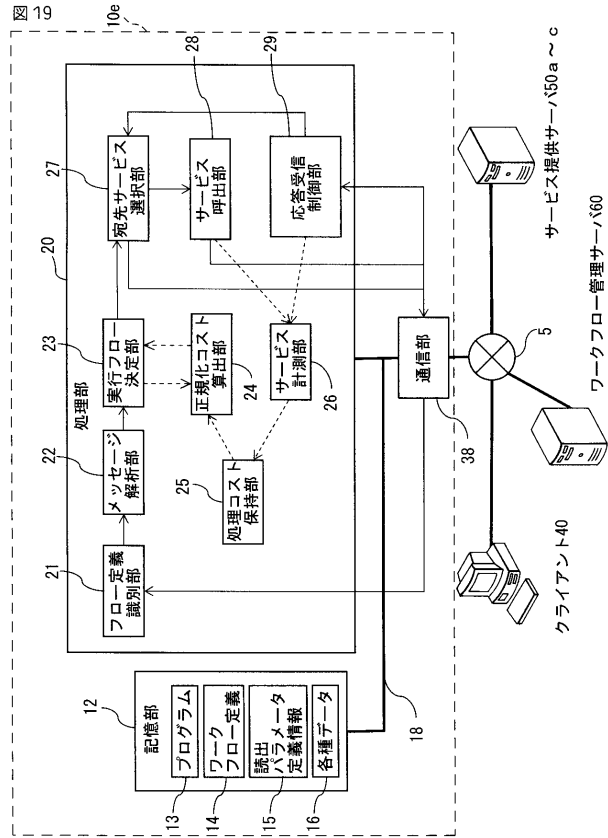
【図 17】



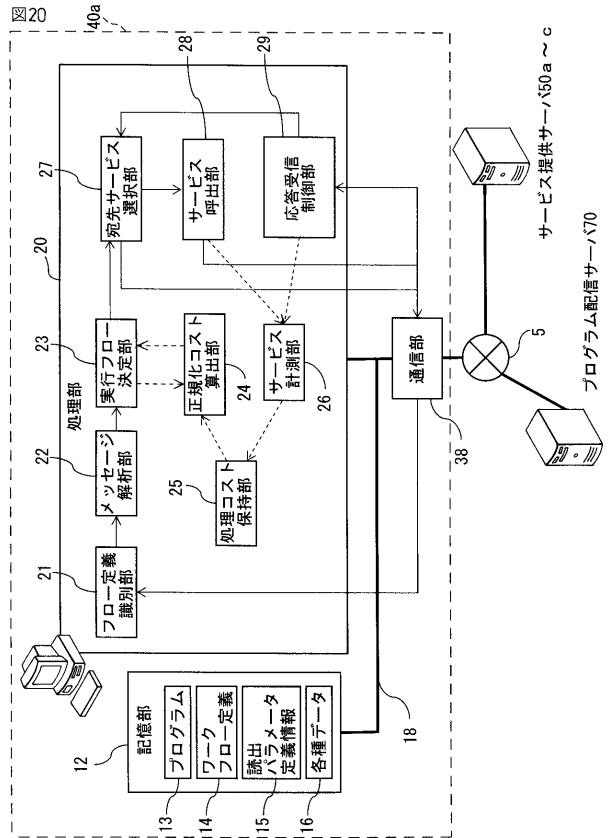
【図 18】



【図 19】

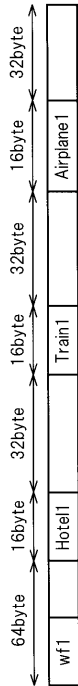


【図 20】



【図 21】

図21



【図 22】

図22

ワークフロー名	サービス	バイト位置	サイズ
wf1	S1	64	16
wf1	S2	112	16
wf1	S3	160	16

【図 23】

図 23

wf1
hotel, Hotel1, 2008-5-29, twin, 2, 2
train, Train1, 2008-5-29T10:00:00, station1, station2
airplane, Airplane1, 2008-5-29T14:00:00, stationA, stationB

【図 24】

図24

ワークフロー名	サービス	行	列
wf1	S1	2	1
wf1	S2	3	1
wf1	S3	4	1

フロントページの続き

(74)代理人 100114177

弁理士 小林 龍

(72)発明者 的場 一峰

神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内