

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7285283号
(P7285283)

(45)発行日 令和5年6月1日(2023.6.1)

(24)登録日 令和5年5月24日(2023.5.24)

(51)国際特許分類 F I
G 0 6 F 8/30 (2018.01) G 0 6 F 8/30

請求項の数 9 (全23頁)

(21)出願番号	特願2021-98052(P2021-98052)	(73)特許権者	000005108 株式会社日立製作所 東京都千代田区丸の内一丁目6番6号
(22)出願日	令和3年6月11日(2021.6.11)	(74)代理人	110002365 弁理士法人サンネクスト国際特許事務所
(65)公開番号	特開2022-189463(P2022-189463 A)	(72)発明者	野田 昌太郎 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
(43)公開日	令和4年12月22日(2022.12.22)	(72)発明者	大越 淳平 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
審査請求日	令和4年3月2日(2022.3.2)	(72)発明者	櫻山 俊彦 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
		(72)発明者	馬場 恒彦

最終頁に続く

(54)【発明の名称】 分散トランザクション管理方法、及びシステム

(57)【特許請求の範囲】

【請求項1】

分散トランザクションにおいて呼び出される複数の参加システムそれぞれの呼び出し順番を規定した情報である分散トランザクションフロー、及び当該複数の参加システムの各々について当該参加システムの特性であるサービス特性を表す情報であるサービス特性情報の入力を受け付ける入力部と、

呼び出し先の参加システムのサービス特性の制約条件を表す情報をパターン毎に含んだパターン情報を基に、前記サービス特性情報が表す参加システム毎に、当該参加システムのサービス特性に対応した制約条件を満たすパターンを適用パターンとして特定する適用パターン判定部と、

前記複数の参加システムの各々についてパターン適用済呼び出しプログラムを生成し、当該複数の参加システムの各々のパターン適用済呼び出しプログラムと、前記分散トランザクションフローが表す呼び出し順序とを基に、当該呼び出し順序で実行される複数のパターン適用済呼び出しプログラムを含んだ分散トランザクションプログラムを生成する分散トランザクションプログラム生成部とを備え、

前記複数の参加システムの各々について、前記パターン適用済呼び出しプログラムは、それぞれパターンが記述された複数のパターンプログラムのうち、当該参加システムについて前記特定された適用パターンのパターンプログラムが、当該参加システムを呼び出し先とした呼び出しプログラムに適用されたプログラムである、

分散トランザクション管理システム。

【請求項 2】

請求項 1 に記載の分散トランザクション管理システムにおいて、
前記分散トランザクションプログラムを実行する分散トランザクション実行部、
を備える分散トランザクション管理システム。

【請求項 3】

請求項 1 に記載の分散トランザクション管理システムにおいて、
前記パターン情報が、パターン毎に、パターン間の依存関係を表す情報を含み、
前記適用パターン判定部が、前記サービス特性情報が表す参加システム毎に、当該参加
システムの適用パターンを、当該参加システムのサービス特性に対応した制約条件を表す
情報の他に、当該参加システムに対応した、パターン間の依存関係を表す情報を基に、特
定する、
分散トランザクション管理システム。

10

【請求項 4】

請求項 3 に記載の分散トランザクション管理システムにおいて、
各パターンについて、パターン間の依存関係を表す情報は、パターンの階層関係を表す
情報である依存パターン（階層）を含み、
前記適用パターン判定部が、前記サービス特性情報が表す参加システム毎に、当該参加
システムの適用パターンとして、当該参加システムのサービス特性に対応した制約条件を
満たし、且つ、当該参加システムについて既に特定済の適用パターンの階層が当該参加シ
ステムに対応した依存パターン（階層）が表す階層を満たすパターンを特定する、
分散トランザクション管理システム。

20

【請求項 5】

請求項 3 に記載の分散トランザクション管理システムにおいて、
各パターンについて、パターン間の依存関係を表す情報は、下記（a）及び（b）のう
ちの少なくとも一つを含み、

（a）当該パターンが適用パターンとされる参加システムの呼び出し順番の直前の呼
び出し順番の参加システムについての適用パターンを表す情報である依存パターン（前）、

（b）当該パターンが適用パターンとされる参加システムの呼び出し順番の直後の呼
び出し順番の参加システムについての適用パターンを表す情報である依存パターン（後）、

30

前記適用パターン判定部が、前記サービス特性情報が表す参加システム毎に、当該参加
システムの適用パターンとして、当該参加システムのサービス特性に対応した制約条件を
満たし、且つ、下記（A）及び（B）のうちの一つを満たすパターンを特定す
る、

（A）当該参加システムの呼び出し順番の直前の呼び出し順番の参加システムについ
ての適用パターンが、当該参加システムに対応した依存パターン（前）が表す適用パター
ンに該当する、

（B）当該参加システムの呼び出し順番の直後の呼び出し順番の参加システムについ
ての適用パターンが、当該参加システムに対応した依存パターン（後）が表す適用パター
ンに該当する、

40

分散トランザクション管理システム。

【請求項 6】

請求項 1 に記載の分散トランザクション管理システムにおいて、
前記分散トランザクションプログラム生成部は、前記分散トランザクションプログラムの
構造が一意に定まらない場合、所定のルール又はユーザによる入力を基に、分散トラン
ザクションプログラムの構造を決定する、
分散トランザクション管理システム。

【請求項 7】

請求項 6 に記載の分散トランザクション管理システムにおいて、
前記パターン情報が、パターン毎に、当該パターンの適用がオプションか否かを表す情

50

報を含み、

前記分散トランザクションプログラムの構造が一意に定まらない場合とは、特定された少なくとも一つの適用パターンがオプションである場合である、
分散トランザクション管理システム。

【請求項 8】

請求項 6 に記載の分散トランザクション管理システムにおいて、

適用がオプションである少なくとも一つのパターンについて、当該パターンの適用に関するルールに従う処理のためのパターンルールプログラムが関連付けられており、前記特定された適用パターンに適用がオプションの適用パターンがあり、当該適用パターンについて、関連付けられたパターンルールプログラムがある場合、前記分散トランザクションプログラム生成部は、前記分散トランザクションプログラムの構造を一意に定めるために当該パターンルールプログラムを実行する、
分散トランザクション管理システム。

10

【請求項 9】

コンピュータが、分散トランザクションフロー及びサービス特性情報の入力を受け付け、

前記分散トランザクションフローは、分散トランザクションにおいて呼び出される複数の参加システムそれぞれの呼び出し順番を規定した情報であり、

当該複数の参加システムの各々は、当該分散トランザクションに参加するシステムであり、

前記サービス特性情報は、当該複数の参加システムの各々について当該参加システムの特性であるサービス特性を表す情報であり、

20

コンピュータが、呼び出し先の参加システムのサービス特性の制約条件を表す情報をパターン毎に含んだパターン情報を基に、前記サービス特性情報が表す参加システム毎に、当該参加システムのサービス特性に対応した制約条件を満たすパターンを適用パターンとして特定し、

コンピュータが、前記複数の参加システムの各々についてパターン適用済呼び出しプログラムを生成し、当該複数の参加システムの各々のパターン適用済呼び出しプログラムと、前記分散トランザクションフローが表す呼び出し順序とを基に、当該呼び出し順序で実行される複数のパターン適用済呼び出しプログラムを含んだ分散トランザクションプログラムを生成し、

30

前記複数の参加システムの各々について、前記パターン適用済呼び出しプログラムは、それぞれパターンが記述された複数のパターンプログラムのうち、当該参加システムについて前記特定された適用パターンのパターンプログラムが、当該参加システムを呼び出し先とした呼び出しプログラムに適用されたプログラムである、
分散トランザクション管理方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、概して、分散トランザクションの管理に関する。

【背景技術】

40

【0002】

近年、クラウドネイティブやアジャイル開発の潮流を受け、マイクロサービスアーキテクチャに注目が集まっている。マイクロサービスアーキテクチャは、システムを複数の小さなサービス（マイクロサービス）の集合体で構成されるシステムの開発手法の一つで、アジリティやスケラビリティの向上に寄与する。

【0003】

マイクロサービスアーキテクチャにおいては、複数のマイクロサービスが協調して動作するため、サービス間におけるデータの整合性維持は重要な問題となる。ここでデータが整合している（データの整合性が保たれている）とは、サービスによって実現されるアプリケーションにおいてデータの不整合が生じていない状態を指し、例えば、銀行の口座間

50

送金において口座 A から口座 B に送金がされる場合、口座 A から出金された金額が過不足無く口座 B に入金されている状態を指す。一方、口座 A からのみ出金が行われている状態や口座 B に二重に入金が行われている状態はデータの不整合が生じた典型例である。

【 0 0 0 4 】

データの整合性維持には、分散トランザクションと呼ばれる手法が用いられる。分散トランザクションの代表的な仕様の一つとして X A が挙げられる。X A においては、2 相コミットと呼ばれる合意プロトコルを用いることにより、サービス間でデータの整合性を維持することが可能となる。一方、マイクロサービスアーキテクチャにおいては、外部サービスと連携する必要があるなどの理由から、特定の合意プロトコル（例えば、X A における 2 相コミット）を前提とできない場合もある。このため、各サービスの A P I (A p p l i c a t i o n P r o g r a m m i n g I n t e r f a c e) に応じた要求の逐次処理により分散トランザクションを実現する手法 (S A G A) も台頭してきている。

10

【 0 0 0 5 】

マイクロサービスアーキテクチャにより開発されたシステムに分散トランザクション（特に、S A G A を含んだ分散トランザクション）を実装する場合、システム障害とそれ起因するデータ不整合に対して適切な設計パターン（以下、パターン）の組み合わせが必要となる。例えば、分散トランザクション管理システム（分散トランザクションの実行を管理するサービス）による参加システム（分散トランザクションに参加するサービス）の呼び出しが通信障害（又はその他の理由）により失敗した場合、データ不整合を防止するために、参加システムのサービス特性（参加システムの種類（例えば、参加システムの仕様、機能（例えば実装されているアプリケーション））に応じたなんらかのパターンの適用が必要となる。例えば、参加システムが処理の冪等性（複数回の同一要求に対し、同一状態に内部状態の遷移が保障され、また同一の結果を返す性質）を持つ場合には、呼び出し処理を再実行するパターンを、当該参加システムを呼び出す呼び出しプログラムへ適用することで、データ不整合を防止することができる。

20

【 0 0 0 6 】

しかし、サービス特性とパターンは多数存在する。これらの中から複数のサービス特性それぞれに応じて適切なパターンを選択し、それら選択されたパターンを組み合わせるには、高度な知識と多大な工数を要するという問題がある。

【 0 0 0 7 】

パターンを選択する技術として、例えば、特許文献 1 に開示の技術がある。特許文献 1 では、分散トランザクションの実行時にトランザクションの長さに応じて適用するパターンを切り替える技術が記載されている。

30

【 0 0 0 8 】

また、組み合わせたパターンの実装に関わる技術として、例えば、特許文献 2 に開示の技術がある。特許文献 2 では、テンプレートと部品リストからプログラムを生成する技術が記載されている。

【先行技術文献】

【特許文献】

【 0 0 0 9 】

【文献】米国特許出願公開第 2 0 1 6 / 0 2 1 0 3 2 2 号明細書
特開 2 0 1 3 - 2 2 2 4 3 9 号公報

40

【発明の概要】

【発明が解決しようとする課題】

【 0 0 1 0 】

特許文献 1 と特許文献 2 に記載の技術に基づいて、分散トランザクションに関わるパターンが適用されたプログラムを動的に切り替え可能な構成が得られると考えられる。具体的には、特許文献 2 に記載の技術を利用して、パターンが適用されたプログラムを自動生成し、特許文献 1 に記載の技術を利用して、パターンが適用されたプログラムを切り替えることが考えられる。

50

【 0 0 1 1 】

しかし、これらの特許文献 1 及び 2 に記載の技術に基づいてプログラムの生成及び切替えを可能にする分散トランザクション管理システム（以下、管理システム）を構築しようとすると、実装工数が大きくなるという問題が発生する。すなわち、参加システムのサービス特性に応じたパターンの適用ができず、参加システムの変更によりサービス特性が変化するたびに、管理システムの改変が必要となるという問題が発生する。

【 0 0 1 2 】

本発明の目的は、参加システムのサービス特性が変化しても、管理システムの実装を改変することなく、適切なパターンを組み合わせた分散トランザクションを実行することにある。

【課題を解決するための手段】

【 0 0 1 3 】

分散トランザクション管理システムが、分散トランザクションフロー及びサービス特性情報の入力を受け付ける。分散トランザクションフローは、分散トランザクションにおいて呼び出される複数の参加システムそれぞれの呼び出し順番を規定した情報である。当該複数の参加システムの各々は、当該分散トランザクションに参加するシステムである。サービス特性情報は、当該複数の参加システムの各々について当該参加システムの特性であるサービス特性を表す情報である。分散トランザクション管理システムが、呼び出し先の参加システムのサービス特性の制約条件を表す情報をパターン毎に含んだパターン情報を基に、サービス特性情報が表す参加システム毎に、当該参加システムのサービス特性に対応した制約条件を満たすパターンを適用パターンとして特定する。分散トランザクション管理システムが、複数の参加システムの各々についてパターン適用済呼び出しプログラムを生成し、当該複数の参加システムの各々のパターン適用済呼び出しプログラムと、前記分散トランザクションフローが表す呼び出し順序とを基に、当該呼び出し順序で実行される複数のパターン適用済呼び出しプログラムを含んだ分散トランザクションプログラムを生成する。複数の参加システムの各々について、パターン適用済呼び出しプログラムは、それぞれパターンが記述された複数のパターンプログラムのうち、当該参加システムについて上記特定された適用パターンのパターンプログラムが、当該参加システムを呼び出し先とした呼び出しプログラムに適用されたプログラムである。

【発明の効果】

【 0 0 1 4 】

本発明によれば、参加システムのサービス特性が変化しても、管理システムの実装を改変することなく、適切なパターンを組み合わせた分散トランザクションを実行することができる。

【図面の簡単な説明】

【 0 0 1 5 】

【図 1】実施形態に係るソフトウェア構成を示す図である。

【図 2】実施形態に係るハードウェア構成を示す図である。

【図 3】実施形態に係る分散トランザクションに係る処理を示すシーケンス図である。

【図 4】実施形態に係るパターンプログラムテーブル、及びパターンプログラムを示す図である。

【図 5】実施形態に係るパターンテーブルを示す図である。

【図 6】実施形態に係る呼び出しプログラムテーブル、及び呼び出しプログラムを示す図である。

【図 7】実施形態に係るサービス特性テーブルを示す図である。

【図 8】実施形態に係る分散トランザクション編集画面を示す図である。

【図 9】実施形態に係るパターン適用済参加システム呼び出しプログラムの生成を説明する図である。

【図 10】実施形態に係る分散トランザクションフローを示す図である。

【図 11】実施形態に係る分散トランザクションプログラム生成処理を示すフロー図であ

10

20

30

40

50

る。

【図12】図11のS1103を示すフロー図である。

【図13】図11のS1104を示すフロー図である。

【図14】本実施形態に係るパターンルールテーブル、及びパターンルールプログラムを示す図である。

【発明を実施するための形態】

【0016】

実施形態について、図面を参照して説明する。なお、以下に説明する実施形態は特許請求の範囲に係る発明を限定するものではなく、また、実施形態の中で説明されている諸要素、及びその組み合わせの全ては、発明の解決手段に必須であるとは限らない。

10

【0017】

以下の説明では、「インターフェース装置」は、一つ以上のインターフェースデバイスでよい。当該一つ以上のインターフェースデバイスは、下記のうちの少なくとも一つでよい。

・一つ以上のI/O(Input/Output)インターフェースデバイス。I/O(Input/Output)インターフェースデバイスは、I/Oデバイスと遠隔の表示用計算機とのうちの少なくとも一つに対するインターフェースデバイスである。表示用計算機に対するI/Oインターフェースデバイスは、通信インターフェースデバイスでよい。少なくとも一つのI/Oデバイスは、ユーザインターフェースデバイス、例えば、キーボード及びポインティングデバイスのような入力デバイスと、表示デバイスのような出力デバイスとのうちのいずれでもよい。

20

・一つ以上の通信インターフェースデバイス。一つ以上の通信インターフェースデバイスは、一つ以上の同種の通信インターフェースデバイス(例えば一つ以上のNIC(Network Interface Card))であってもよいし二つ以上の異種の通信インターフェースデバイス(例えばNICとHBA(Host Bus Adapter))であってもよい。

【0018】

また、以下の説明では、「メモリ」は、一つ以上の記憶デバイスの一例である一つ以上のメモリデバイスであり、典型的には主記憶デバイスでよい。メモリにおける少なくとも一つのメモリデバイスは、揮発性メモリデバイスであってもよいし不揮発性メモリデバイスであってもよい。

30

【0019】

また、以下の説明では、「永続記憶装置」は、一つ以上の記憶デバイスの一例である一つ以上の永続記憶デバイスでよい。永続記憶デバイスは、典型的には、不揮発性の記憶デバイス(例えば補助記憶デバイス)でよく、具体的には、例えば、HDD(Hard Disk Drive)、SSD(Solid State Drive)、NVME(Non-Volatile Memory Express)ドライブ、又は、SCM(Storage Class Memory)でよい。

【0020】

また、以下の説明では、「記憶装置」は、メモリと永続記憶装置の少なくともメモリでよい。

【0021】

40

また、以下の説明では、「プロセッサ」は、一つ以上のプロセッサデバイスでよい。少なくとも一つのプロセッサデバイスは、典型的には、CPU(Central Processing Unit)のようなマイクロプロセッサデバイスでよいが、GPU(Graphics Processing Unit)のような他種のプロセッサデバイスでもよい。少なくとも一つのプロセッサデバイスは、シングルコアでもよいしマルチコアでもよい。少なくとも一つのプロセッサデバイスは、プロセッサコアでもよい。少なくとも一つのプロセッサデバイスは、処理の一部又は全部を行うハードウェア記述言語によりゲートアレイの集合体である回路(例えばFPGA(Field-Programmable Gate Array)、CPLD(Complex Programmable Logic Device)又はASIC(Application Specific Integrated Circuit))といった広義のプロセッサデバイスでもよい。

50

【 0 0 2 2 】

また、以下の説明では、「 $\times \times \times$ テーブル」といった表現にて、入力に対して出力が得られる情報を説明することがあるが、当該情報は、どのような構造のデータでもよいし（例えば、構造化データでもよいし非構造化データでもよいし）、入力に対する出力を発生するニューラルネットワーク、遺伝的アルゴリズムやランダムフォレストに代表されるような学習モデルでもよい。従って、「 $\times \times \times$ テーブル」を「 $\times \times \times$ 情報」と言うことができる。また、以下の説明において、各テーブルの構成は一例であり、一つのテーブルは、二つ以上のテーブルに分割されてもよいし、二つ以上のテーブルの全部又は一部が一つのテーブルであってもよい。

【 0 0 2 3 】

また、以下の説明では、「プログラム」を主語として処理を説明する場合があるが、プログラムは、プロセッサによって実行されることで、定められた処理を、適宜に記憶装置及び/又はインターフェース装置を用いながら行うため、処理の主語が、プロセッサ（或いは、そのプロセッサを有する装置又はシステム）とされてもよい。プログラムは、プログラムソースから計算機のような装置にインストールされてもよい。プログラムソースは、例えば、プログラム配布サーバ又は計算機が読み取り可能な記録媒体（例えば非一時的な記録媒体）であってもよい。また、以下の説明において、二つ以上のプログラムが一つのプログラムとして実現されてもよいし、一つのプログラムが二つ以上のプログラムとして実現されてもよい。

【 0 0 2 4 】

また、以下の説明では、同種の要素を区別しないで説明する場合には、参照符号のうちの共通符号を使用し、同種の要素を区別して説明する場合には、参照符号を使用することがある。

【 0 0 2 5 】

また、以下の説明では、要素（例えば、参加システムやパターン）の識別情報の例として、名称又はIDが採用されるが、名称やIDに代えて又は加えて他種の情報が採用されてもよい。

【 0 0 2 6 】

以下、本発明の一実施形態を説明する。

【 0 0 2 7 】

本実施形態における情報処理システムのソフトウェア構成を、図1を参照して説明する。図1は、本実施形態の情報処理システムにおけるソフトウェア構成を説明するための機能ブロック図である。

【 0 0 2 8 】

図1に示す情報処理システム1（以下、単に「本システム1」と記述する場合がある）では、参加システムのサービス特性が変化しても、分散トランザクション管理システム100（以下、単に「管理システム100」と記述する場合がある）の実装を改変することなく、適切なパターンの組み合わせた分散トランザクションを、データ整合性維持を実現しつつ実行することができる。「参加システム」は、分散トランザクションに参加するサービス（システム）である。

【 0 0 2 9 】

図1に示すように、本システム1は、本システムで実行される分散トランザクションを管理する管理システム100と、複数の参加システム110（110A、110B、110C、...）とを備える。なお、図1では、管理システム100及び参加システム110を異なる機能ブロックとして記載しているが、これらのシステム100及び110のいずれか2つ以上、あるいはシステム100及び110の少なくとも一部は、1つの計算機として構成されていてもよい。また、システム100及び110の少なくとも一つは、物理的な計算機システム（一つ以上の物理的な計算機）でもよいし、物理的な計算機システムに基づく論理的な計算機システム（例えば、仮想計算機システム又はクラウドコンピューティングサービスシステム）でもよい。

10

20

30

40

50

【 0 0 3 0 】

また、図 1 では、詳細を後述する各機能部 (5 1、1 0 1、1 0 2、1 0 3、1 0 4、1 0 5、1 1 1、1 1 2)、各記憶領域 (1 0 6、1 0 7、1 0 8、1 0 9)、及びデータベース 1 1 3 を異なる機能ブロックで記載しているが、ハードウェアとして異なるものであることを要求するものではない。したがって、各機能部 (5 1、1 0 1、1 0 2、1 0 3、1 0 4、1 0 5、1 1 1、1 1 2) の動作は 1 つ以上のハードウェアで行われてもよく、各記憶領域 (1 0 6、1 0 7、1 0 8、1 0 9)、及びデータベース 1 1 3 は、ハードディスクドライブ等の 1 つ以上の記憶デバイスで実現されていてもよい。

【 0 0 3 1 】

管理システム 1 0 0 は、入力部 5 1 と、適用パターン判定部 1 0 1 と、パターン管理部 1 0 2 と、分散トランザクションプログラム生成部 1 0 4 と、呼び出しプログラム管理部 1 0 3 と、分散トランザクション実行部 1 0 5 と、パターンテーブル記憶領域 1 0 6 と、パターンプログラム記憶領域 1 0 8 と、呼び出しプログラム記憶領域 1 0 7 と、パターンルール記憶領域 1 0 9 とを備える。

10

【 0 0 3 2 】

参加システム 1 1 0 について、参加システム 1 1 0 A (参加システム 1) を例に説明する。参加システム 1 1 0 A は、トランザクション実行部 1 1 1 と、データベース管理部 1 1 2 と、データベース 1 1 3 とを備える。他の参加システム 1 1 0 B、1 1 0 C、... も、参加システム 1 1 0 A と同様の構成を取ることができる。

【 0 0 3 3 】

図 2 は、本システム 1 におけるハードウェア構成を示すブロック図である。以下、図 2 を参照して、本システム 1 におけるハードウェア構成を説明する。

20

【 0 0 3 4 】

管理システム 1 0 0 及び参加システム 1 1 0 の各々は、サーバ装置、パーソナルコンピュータ又はワークステーションなどの一般的な情報処理装置 (計算機) により実現することができる。すなわち、図 2 に示す通り、これらのシステムのハードウェア構成は、プロセッサの一例である CPU (2 0 1、2 1 1)、メモリの一例である主メモリ (2 0 2、2 1 2)、永続記憶装置の一例であるストレージ (2 0 3、2 1 3)、インターフェース装置の一例である NIC (2 0 5、2 1 5)、入力デバイスの一例であるキーボード (2 0 6、2 1 6) 及びマウス (2 0 7、2 1 7)、出力デバイスの一例であるディスプレイ (2 0 8、2 1 8)、及びこれらを接続するバス (2 0 4、2 1 4) を備えた構成とすることができる。そして、各システム (1 0 0、1 1 0) はネットワーク 2 3 0 を経由して相互に接続されている。管理システム 1 0 0 の NIC 2 0 5 に各参加システム 1 1 0 が通信可能に接続される。

30

【 0 0 3 5 】

以下は、各システム (1 0 0、1 1 0) のハードウェア構成について、管理システム 1 0 0 を代表例として説明する。他のシステム (1 1 0) のハードウェア構成要素は、基本的に管理システム 1 0 0 と同様の構成とすることができる。

【 0 0 3 6 】

CPU 2 0 1 は、プログラムを実行することにより管理システム 1 0 0 が備える各機能部を制御する。具体的には、例えば、CPU 2 0 1 は、必要なプログラムを主メモリ 2 0 2 に読み込み、当該プログラムを実行することで、各機能部を実現する。

40

【 0 0 3 7 】

NIC 2 0 5 は、ネットワーク 2 3 0 と接続するためのインターフェースデバイスである。

【 0 0 3 8 】

主メモリ 2 0 2 は、通常の RAM (R a n d o m A c c e s s M e m o r y) などの揮発性メモリで構成され、CPU 2 0 1 が実行するプログラムや参照するデータが記憶される。

【 0 0 3 9 】

50

ストレージ 203 は、HDD (Hard Disk Drive) や SSD (Solid State Drive) などの情報を記憶する記憶デバイスである。

【0040】

キーボード 206、及びマウス 207 は、ユーザ 300 がデータやコマンドを入力するための入力デバイスである。

【0041】

ディスプレイ 208 は、例えば、液晶ディスプレイモニタなどから構成され、必要な画面や各処理の結果を表示する表示デバイスである。

【0042】

本実施形態において、特に言及のない場合、ユーザ 300 (図 3 参照) から各システム (100、110) への入力、及び各システム (100、110) からユーザ 300 への情報の表示は、これらキーボード (206、216)、マウス (207、217)、及びディスプレイ (208、218) によって集約的に行うものとする。

【0043】

図 3 は、本システム 1 の動作の概要を示すシーケンス図である。本システム 1 の意図するところは、分散トランザクションフローに代えて又は加えて、参加システム 110 のサービス特性が変化した場合においても、管理システム 100 の実装を改変することなく、適切なパターンを適用した分散トランザクションプログラムの実行により、データの整合性の維持を可能とするところにある。以下、図 3 のシーケンス図に沿って、本実施形態を説明する。

【0044】

本実施形態の初期状態では、各機能部 (101、102、103、104、105) の動作に必要なプログラムがストレージ 203 に格納されている。パターンプログラム記憶領域 108 には、図 4 に示すパターンプログラムテーブル 401 とそれに紐付いているパターンプログラム (パターンプログラム (Retry) 411 がその一例) とが格納されている。パターンテーブル記憶領域 106 には、図 5 に示すパターンテーブル 501 が格納されている。呼び出しプログラム記憶領域 107 には、図 6 に示す参加システムの呼び出しプログラムテーブル 601 とそれに紐付いている呼び出しプログラム (参加システム 2 が呼び出し先である呼び出しプログラム 611 がその一例) が格納されている。パターンルール記憶領域 109 には、図 14 に示すパターンルールテーブル 1401 とそれに紐付いているパターンルールプログラム (パターンルールプログラム (rule001) 1411 はその一例) が格納されている。

【0045】

ユーザ 300 は、例えばユーザ端末 (ユーザ 300 の情報処理端末) を用いて、分散トランザクション実行プログラム生成に係る情報 A301 として、図 7 に示すサービス特性テーブル 701 と図 10 に示す分散トランザクションフロー 1001 とを管理システム 100 に入力する。管理システム 100 の入力部 51 が、フロー 1001 及びテーブル 701 の入力を受け付ける。管理システム 100 は、後述の処理により、これらの入力 (フロー 1001 及びテーブル 701) に基づき分散トランザクションフロー 1001 に対応した分散トランザクションプログラムを生成する (S311)。

【0046】

図 7 は、サービス特性テーブル 701 を示す図である。

【0047】

サービス特性テーブル 701 は、参加システム 110 毎に行を有し、「サービス」列 702、及び「サービス特性」列 703 の 2 つの列を有する。

【0048】

「サービス」列 702 には、管理システム 100 から呼び出される参加システム 110 (例えば参加システム 110 の名称) が列挙されている。図 7 が示す例によれば、参加システムとして 5 つの参加システムが列挙されている。

【0049】

10

20

30

40

50

「サービス特性」列 703 には、各参加システムが有するサービス特性（例えば、参加システムの仕様、機能（例えば実装されているアプリケーション）の名称）が記載されている。例えば、「参加システム 1」は、「SAGA（非同期）」というサービス特性を備える。「サービス特性」列 703 には、参加システムのサービス特性に加え、管理システム 100 の特性が記述されてもよい。

【0050】

図 10 は、分散トランザクションフロー 1001 を示す図である。

【0051】

分散トランザクションフロー 1001 は、本システム 1 で実行される分散トランザクションにおいて呼び出される複数の参加システム 110 それぞれの呼び出しの順番（参加システムの呼び出し順序）を規定した情報（データ）である。本実施形態においては、「参加システム 1」から「参加システム 5」までの合計 5 つのシステムがシーケンシャルに呼ばれることが記述されている。分散トランザクションフロー 1001 は、例えば DAG（Directed Acyclic Graph）で表現される。ノード 1002 は、参加システム 110 の呼び出しを表し、エッジ（結線）1003 は、呼び出しの順序を表す。分散トランザクションフロー 1001 では、一つの参加システム 110 によるトランザクション実行が終了した場合に二つ以上の参加システム 110 を並行して呼び出すことが規定されていてもよいし、二つ以上の参加システム 110 の各々がトランザクション実行を終了した場合に一つの参加システム 110 を呼び出すことが規定されていてもよい。

【0052】

分散トランザクションプログラム生成処理（図 3 の S311）を、図 11 を用いて説明する。また、生成される分散トランザクションプログラムとして、図 8 を適宜参照する。

【0053】

図 11 は、分散トランザクションプログラム生成処理（図 3 の S311）を示すフロー図である。以下、本フロー図に従って分散トランザクションプログラム生成処理を説明する。

【0054】

図 8 は、分散トランザクションプログラム生成処理によって生成された分散トランザクションプログラム 803 をユーザが編集するために用いられる分散トランザクション編集画面 801 を示す図である。分散トランザクション編集画面 801 は、例えば管理システム 100 の分散トランザクションプログラム生成部 104 により提供される。編集画面 801 におけるフロー修正画面 802 に表示された分散トランザクションプログラム 803 を適宜参照して説明する。

【0055】

管理システム 100 の分散トランザクションプログラム生成部 104 は、S1101 にて、分散トランザクションフロー 1001 を参照し、トランザクションの順序を決定する。具体的には、分散トランザクションプログラム生成部 104 は、図 10 に示した分散トランザクションフロー 1001 を参照し、参加システム 110 の呼び出し（ノード 1002）をエッジ（結線）1003 に沿って迎えることにより、参加システム 1 から参加システム 5 までのシステム呼び出しが順に実行されることを読み取り、図 8 の参加システム 1 呼び出し 811 から参加システム 5 呼び出し 812 までのシステム呼び出し順序を確定する。なお、フロー修正画面 802 に表示の分散トランザクションプログラム 803 は、実行時には画面の上から下に向かって（参加システム 1 呼び出し 811 から参加システム 5 呼び出し 812 に向かって）実行される。

【0056】

管理システム 100 の適用パターン判定部 101 は、S1102 にて、サービス特性テーブル 701 とパターンテーブル 501 を参照し、分散トランザクションプログラム 803 の生成のための適用パターンを決定する。この処理をサービス特性テーブル 701 及びパターンテーブル 501 を用いて説明する。

【0057】

10

20

30

40

50

図5は、パターンテーブル501を示す図である。本実施形態においては、パターンテーブル501は、パターン（典型的には、パターン化された処理ロジックの名称）毎に行を有し、「パターン」列511、「依存パターン（前）」列512、「依存パターン（後）」列513、「依存パターン（階層）」列514、「呼び出し元制約」列515、「呼び出し先制約」列516、及び「オプション」列517の7列から構成される。例えば、「パターン」列511、「依存パターン（前）」列512、「依存パターン（後）」列513、「依存パターン（階層）」列514にはそれぞれパターンの名称が記載される。各列の利用方法については、以降の処理の説明において適宜説明する。

【0058】

S1102の処理において、管理システム100の適用パターン判定部101は、サービス特性テーブル701、及びパターンテーブル501を参照し、分散トランザクションプログラム803中の各システム呼び出し（例えば、参加システム1呼び出し811）に適用可能なパターンを特定する。さらに、分散トランザクションプログラム生成部104は、適用パターン判定部101によって特定されたパターンを表す情報を適用パターン判定部101より受け取り、当該パターンの適用処理を実施する。この処理を、参加システム2が呼び出し先である呼び出し813へのパターン適用を例に説明する。

【0059】

管理システム100の適用パターン判定部101は、サービス特性テーブル701を参照し、参加システム2が「SAGA（同期）」、「TCC」、「冪等（多重実行防止あり）」の3つのサービス特性を有することを読み取る。続いて、管理システム100の適用パターン判定部101は、パターンテーブル501の「呼び出し先制約」列516を参照し、読み取られたサービス特性「SAGA（同期）」、「TCC」及び「冪等（多重実行防止あり）」にそれぞれ対応する（一致する）パターン「SAGA（同期）」、「TCC」、「Retry（多重実行なし）」を特定する。「呼び出し先制約」は、呼び出し先としての参加システムのサービス特性の要件である制約条件を意味する。なお、パターン「SAGA（同期）」は、各参加システムのローカルトランザクションの逐次実行により分散トランザクションを実現するパターンである。パターン「TCC」は、Try、Confirm、及びCancelの3種の命令に対応したAPI（Application Programming Interface）に応じたシステム呼び出しを行うパターンである。パターン「Retry（多重実行なし）」は、システム呼び出しにおいて冪等性を考慮せずにリトライを行うパターンである。パターンテーブル501に記載の他のパターンも前述の様なサービス特性に応じた、データ整合性に寄与するパターンとなっている。「呼び出し元制約」列515は、各パターンを適用する際に管理システム100が備えるべきサービス特性が記述されるが、本実施形態においては、管理システム100は「呼び出し元制約」列515のすべての特性を備えている。このため、管理システム100は、パターンテーブル501に記載のすべてのパターンを、呼び出し元制約を除く他の条件（呼び出し先制約や後述する依存パターン）を満足すれば、各参加システムの呼び出しに適用することができる（言い換えれば、例えば、管理システム100が「呼び出し元制約」列515のすべての特性を備えていることが前提であれば、「呼び出し元制約」列515は無くてよい）。なお、管理システム100のサービス特性は、実施形態に応じて、サービス特性テーブル701に含めてもよいし、サービス特性を有さない場合は当該パターンをパターンテーブル501に記載しなくてもよい。「呼び出し先制約」列516の「-」は、特定のサービス特性によらず適用可能なことを示す特別な記号である。したがって、「SAGA（同期）」、「TCC」、「Retry（多重実行なし）」のパターンの他、「XA（排他制御）」、「多重受信メッセージ破棄（同期）」、「多重受信メッセージ破棄（非同期）」も、適用されるパターンの候補となるが、当該パターンを適用するためにすでに前後の参加システムへのパターン適用を示す「依存パターン（前）」、又は「依存パターン（後）」にパターンの記載がある（すなわち、これらのパターン適用のためには、すでに何らかのパターンが分散トランザクションプログラムへ適用されている必要がある）ため、S1102では、これらのパターンの適用はなされない。

10

20

30

40

50

【 0 0 6 0 】

さらに、管理システム 1 0 0 の適用パターン判定部 1 0 1 は、特定された「S A G A (同期)」、「T C C」、及び「R e t r y (多重実行なし)」の各パターンについて、「依存パターン(階層)」列 5 1 4 を参照し、依存関係において矛盾が生じないように適用パターンの構造を決定する。具体的には、例えば、管理システム 1 0 0 の適用パターン判定部 1 0 1 は、「S A G A (同期)」の行を参照し、依存パターンがない(「依存パターン(階層)」列 5 1 4 において、「-」は依存パターンなしに当該パターンを適用できることを示す特別な記号)ことを確認し、パターン「S A G A (同期)」を参加システム 2 に適用することを決定する。次に、管理システム 1 0 0 の適用パターン判定部 1 0 1 は、「T C C」の行を参照し、依存パターンが「S A G A (同期)」であることを特定し、「S A G A (同期)」の下位の適用パターンとして、「T C C」を参加システム 2 に適用することを決定する。最後に、管理システム 1 0 0 の適用パターン判定部 1 0 1 は、「R e t r y (多重実行なし)」の行を参照し、依存パターンが「S A G A (同期)」、「S A G A (非同期)」、及び「T C C」であることを特定する。本実施形態においては、依存パターンが複数ある場合は、どれか 1 つ以上が適用されていれば当該パターンを適用することができる。ただし、複数パターンが適用されている場合、これら全てに階層関係の矛盾が無いようにパターンが適用されていなければならない。よって、本実施形態においては、「T C C」の下位の適用パターンとして、「R e t r y (多重実行なし)」が適用されることが決定される。なお、「依存パターン(階層)」にパターンが記載されているにも関わらず、記載されたパターンが適用されていない場合は、当該行のパターンを適用することはできない。例えば、「T C C」は、「S A G A (同期)」パターンが適用されていない場合には、適用することができない。

10

20

【 0 0 6 1 】

参加システム 2 呼び出し 8 1 3 へのパターン適用で説明した処理を他の参加システムへも実施することにより、図 8 記載の分散トランザクションプログラム 8 0 3 において、「X A (排他制御)」、及び「多重受信メッセージ破棄(非同期)」を除いた、すなわち、それぞれの参加システム呼び出しプログラムに対し、「X A」、「S A G A (非同期)」、「S A G A (同期)」、「T C C」、「多重受信メッセージ破棄(非同期)」、「R e t r y (多重実行なし)」、「R e t r y (多重実行あり)」の各パターンが、適宜、適用された分散トランザクションプログラム 8 0 3 の生成が決定される。

30

【 0 0 6 2 】

管理システム 1 0 0 の適用パターン判定部 1 0 1 は、S 1 1 0 3 において、さらにパターンテーブル 5 0 1 を参照し、前後の適用パターンからさらに適用するパターンを決定する。S 1 1 0 3 の処理を、図 1 2 を用いて説明する。

【 0 0 6 3 】

図 1 2 は、図 1 1 の S 1 1 0 3 (管理システム 1 0 0 が実行するさらなる適用パターンの探索、及び適用処理)を示すフロー図である。

【 0 0 6 4 】

管理システム 1 0 0 の適用パターン判定部 1 0 1 は、S 1 2 0 1 において、分散トランザクションプログラム 8 0 3 を走査し、各参加システムの呼び出しプログラム(例えば、参加システム 2 呼び出し 8 1 3)の前後の呼び出しプログラムに適用されたパターンを取得し、さらにパターンテーブル 5 0 1 を参照することで、前後の適用パターンから各参加システムの呼び出しに適用可能なパターンを決定する。S 1 2 0 1 の処理によって適用されるパターンを、参加システム 2 呼び出し 8 1 3 を例に取り説明する。参加システム 2 呼び出し 8 1 3 には、例えば図 8 に示すように、すでに「S A G A (同期)」8 2 1、「T C C」8 2 2、「R e t r y (多重実行なし)」8 2 3 といったパターンが適用されている。S 1 1 0 2 で説明した様に、「X A (排他制御)」、「多重受信メッセージ破棄(同期)」、「多重受信メッセージ破棄(非同期)」が候補の適用パターンとして抽出される。

40

【 0 0 6 5 】

管理システム 1 0 0 の適用パターン判定部 1 0 1 は、S 1 2 0 2 において、適用可能な

50

パターンはあるか、すなわち、S 1 2 0 1において抽出されたパターンのうち、パターンテーブル5 0 1の「依存パターン（前）」列5 1 2、「依存パターン（後）」列5 1 3の各列を参照し、抽出されたパターンの適用可否を判定する。S 1 2 0 2の処理を「参加システム3呼び出し」8 1 5を例に取り説明する。適用パターン判定部1 0 1は、「参加システム4呼び出し」8 1 4（参加システム3呼び出しの順番の1つ後の順番である参加システム4呼び出し）に「X A」8 2 4が適用されていることから、パターンテーブル5 0 1の「依存パターン（後）」列5 1 3に記述された「X A」を含んだ行に対応の「X A（排他制御）」を、「参加システム3呼び出し」8 1 5に適用可能と判定する（S 1 2 0 3以降のループ処理により、最終的に「参加システム1呼び出し」8 1 1、及び「参加システム2呼び出し」8 1 3に対しても「X A（排他制御）」が適用可能と判定される）。また、適用パターン判定部1 0 1は、「参加システム1呼び出し」8 1 1（参加システム2呼び出しの順番の1つ前の順番である参加システム1呼び出し）に適用された「S A G A（非同期）」8 2 6を参照し、「参加システム2呼び出し」8 1 3に、パターンテーブル5 0 1の「依存パターン（前）」列5 1 3に記述された「S A G A（非同期）」を含んだ行に対応の「多重受信メッセージ破棄（非同期）」8 2 7を適用可能と判定する。

10

【0 0 6 6】

なお、適用パターン判定部1 0 1は、パターンテーブル5 0 1の「X A（排他制御）」は適用が任意であることを示す「オプション」列5 1 7が「Y e s」となっていることを読み取り、当該パターンは適用かつ適用は任意と判定する。パターン「X A（排他制御）」とは、「依存パターン（後）」列5 1 3の「X A」パターンが適用される参加システムのデータベースに対し、事前に排他制御と呼ばれるデータの読み書きに関わるロックをかけるパターンである。これにより、「X A（排他制御）」が適用された参加システムの分散トランザクション実行時に、「X A」が適用された参加システムの分散トランザクションに係るデータの書き換えを防止する効果がある。例えば、オプションが「Y e s」のパターンは、適用可能なパターンの数が一定数未満の場合に適用されてよい（言い換えれば、適用可能なパターンの数が一定数以上の場合に適用されないでよい）。

20

【0 0 6 7】

適用パターン判定部1 0 1は、適用可能なパターンが1つ以上あればS 1 2 0 3に処理を進め、適用可能なパターンがなければフローを終了する。

【0 0 6 8】

管理システム1 0 0の適用パターン判定部1 0 1は、S 1 2 0 3において、S 1 2 0 2において適用可能と判定されたパターンの分散トランザクションプログラム8 0 3への適用を決定し、適用は任意とされたパターンについては、適用任意の決定を行い、次のS 1 2 0 1に処理を移す。

30

【0 0 6 9】

S 1 2 0 1、S 1 2 0 2、及びS 1 2 0 3を適用可能なパターンがなくなるまで繰り返す。

【0 0 7 0】

管理システム1 0 0の分散トランザクションプログラム生成部1 0 4は、図1 1のS 1 1 0 4において、分散トランザクションプログラム8 0 3のパターン構造を決定する。S 1 1 0 4の処理を、図1 3を用いて説明する。図1 3は、図1 1のS 1 1 0 4（パターン構造の決定処理）を示すフロー図である。

40

【0 0 7 1】

管理システム1 0 0の分散トランザクションプログラム生成部1 0 4は、S 1 3 0 1において、パターン構造が一意かどうかを判断する。本実施形態においては、「X A（排他制御）」8 2 5の適用がオプションとなっており、一意に定まらないため処理がS 1 3 0 2に移る。

【0 0 7 2】

管理システム1 0 0の分散トランザクションプログラム生成部1 0 4は、S 1 3 0 2において、パターン構造に関わるパターンルールは定義済みかを確認する。この処理を、図

50

14を用いて説明する。

【0073】

図14は、管理システム100によって管理されるパターンルールを示している。パターンルールは、管理システム100のパターン管理部102によって管理され、パターンルールテーブル1401とパターンルールテーブル1401と紐付けて管理されているパターンルールプログラム(パターンルールプログラム(rule001)1411はその一例)からなる。パターンルールテーブル1401は、適用がオプションであるパターン毎に行を有し、「パターン」列1402及び「プログラム」列1403を有する。本実施形態においては、適用が一意に定まっていない「XA(排他制御)」825について、「パターン」列1402に「XA(排他制御)」があり「プログラム」列1403に「rule001」があり、「rule001」に、それに対応するパターンルールプログラム(rule001)1411が定義(関連付け)されている。このため、「XA(排他制御)」825についてパターンルールは定義済みとしてされる。この場合、処理がS1303に移る。

10

【0074】

管理システム100の分散トランザクションプログラム生成部104は、S1303において、パターン管理部102からのパターンルールの取得、及び適用を実施する。図14を参照した例によれば、「XA(排他制御)」に対応したパターンルール「rule001」が取得及び適用される。パターンルール「rule001」には、「常に適用」というパターンルールプログラム(rule001)1411が紐付けて管理されており、適用可能なシステム呼び出し全てに当該パターンが適用される。したがって、本実施形態においては、参加システム1呼び出し811、参加システム2呼び出し813、及び参加システム3呼び出し815に対し、当該パターン(「XA(排他制御)」825)が適用される(図8参照)。

20

【0075】

管理システム100の分散トランザクションプログラム生成部104は、S1304において、パターン構造は一意かを判断し、一意であれば処理を終了し、一意でなければS1305に処理を移す。本実施形態においては、図14のパターンルールにより、すべてのパターンが一意に定まるため、処理が終了する。

【0076】

図14に示したパターンルールが定義されておらず、処理がS1305に進んだ場合の処理を、図8を用いて説明する。

30

【0077】

図8が示す分散トランザクション編集画面801は、パターン構造が一意に定まらなかった場合において、ユーザ300が分散トランザクションを編集し、パターン構造を一意に定めるために用いられる画面である。分散トランザクション編集画面801は、フロー修正画面802と決定ボタン804からなる。ユーザ300は、パターン構造が一意に定まっていない「XA(排他制御)」825の適用範囲をフロー修正画面802で確認しつつ、マウス207の操作により当該パターンの適用範囲を矢印82の範囲で定めることができる。ユーザ300は、編集が終了した後、決定ボタン804をマウス207でクリックすることにより編集結果を確定させる。なお、これらのユーザ300による操作は、分散トランザクションプログラム生成部104に入力され、S1305の処理結果として分散トランザクションプログラム803に反映される。

40

【0078】

管理システム100の分散トランザクションプログラム生成部104は、S1105において、図11のS1101からS1104までによって適用が決定されたパターンを、各参加システムのシステム呼び出しプログラム(811、812、813、814、815)に適用し、実行される分散トランザクションプログラム803を生成する。

【0079】

参加システムの呼び出しプログラム(811、812、813、814、815)への

50

パターンプログラムの適用を、図 4、図 6、及び図 9 を用いて説明する。

【 0 0 8 0 】

図 4 は、管理システム 1 0 0 のパターン管理部 1 0 2 によって管理されるパターンプログラムテーブル 4 0 1 及び当該テーブルに紐付けて管理されるパターンプログラム（パターンプログラム（R e t r y）4 1 1 はその一例）を示す図である。パターンプログラムテーブル 4 0 1 は、パターンプログラム毎に行を有し、「パターン」列 4 0 2、及び「プログラム」列 4 0 3 の 2 列から構成されている。「パターン」列 4 0 2 には、適用パターン判定部 1 0 1 によって適用可否が判定されるパターンを含むパターン（例えばパターンの名称）の一覧が記載されている。「プログラム」列 4 0 3 には「パターン」列 4 0 2 で定義されたパターンに対応したプログラムの ID が管理されている。パターンプログラム（R e t r y）4 1 1 は、「R e t r y」パターンに対応したパターンプログラムの一例である。「パターンプログラム」とは、処理のパターン（典型的にはパターン化された処理ロジック）が記述されたプログラムである。

10

【 0 0 8 1 】

図 6 は、管理システム 1 0 0 の呼び出しプログラム管理部 1 0 3 によって管理される、各参加システムの呼び出しプログラムを管理する呼び出しプログラムテーブル 6 0 1 と当該テーブルに紐づけて管理されている呼び出しプログラム（呼び出しプログラム（参加システム 2）6 1 1 はその一例）を示す図である。呼び出しプログラムテーブル 6 0 1 は、参加システム毎（呼び出しプログラム毎）に行を有し、「参加システム」列 6 0 2 と「プログラム」列 6 0 3 の 2 つの列から構成される。「参加システム」列 6 0 2 には、分散トランザクションで呼び出される参加システム（例えば参加システムの名称）が列挙されている。また、「プログラム」列 6 0 3 には、各参加システムに対応した呼び出しプログラムの ID が記載されている。呼び出しプログラム（参加システム 2）6 1 1 は、参加システム 2 の呼び出しプログラムに対応した呼び出しプログラムの一例である。

20

【 0 0 8 2 】

図 9 は、呼び出しプログラムに対し、パターンが適用される一例を示した図である。

【 0 0 8 3 】

管理システム 1 0 0 の分散トランザクションプログラム生成部 1 0 4 は、図 1 1 の S 1 1 0 5 において、適用パターン判定部 1 0 1 から適用パターンに関わる判定結果を受け取る。分散トランザクションプログラム生成部 1 0 4 は、パターン管理部 1 0 2 から適用パターンに対応するパターンプログラム（図 9 の例ではパターンプログラム（R e t r y）4 1 1）を取得し、呼び出しプログラム管理部 1 0 3 から各システム呼び出しに対応したプログラム（図 9 の例では呼び出しプログラム（参加システム 2）6 1 1）を取得する。分散トランザクションプログラム生成部 1 0 4 は、取得された呼び出しプログラムに対応した参加システムに適用のパターンに対応したパターンプログラムに、当該呼び出しプログラムを適用する（例えば埋め込む）ことで、パターン適用済み呼び出しプログラム（図 9 の例では R e t r y パターンを適用済みの参加システム 2 呼び出しプログラム 9 0 3）を生成する。

30

【 0 0 8 4 】

参加システム 2 呼び出しプログラム 6 1 1 へのパターンプログラム（R e t r y）4 1 1 の適用を例に、パターン適用済み呼び出しプログラムの生成を説明する。まず、パターンプログラム（R e t r y）4 1 1 には、「@ R e t r y」というアノテーションと呼び出しプログラムを挿入する「m e t h o d（）」が関数として定義されている。アノテーションは、当該関数にアノテーション記載の処理、若しくは機能をさらに付与することを示す、複数のプログラミング言語に採用されている仕様の一つである。分散トランザクションプログラム生成部 1 0 4 は、パターンプログラム（R e t r y）4 1 1 の「m e t h o d（）」で定義された関数部分に、呼び出しプログラム（参加システム 2）6 1 1 を挿入することにより、R e t r y パターン適用済み参加システム 2 呼び出しプログラム 9 0 3 を生成することができる。なお、パターンプログラムや参加システム呼び出しプログラムの記述、及びパターン適用済み参加システム呼び出しプログラムの生成は、実施形態に

40

50

応じて適切な記述や方法を用いることができる。例えば、分散トランザクションにおいて標準的な仕様であるXAや、当該仕様に関わる実装を用いてもよい。

【0085】

図11のS1105では、分散トランザクションに参加する参加システム毎（分散トランザクションフロー1001又はサービス特性テーブル701から特定される参加システム毎に）、パターン適用済み呼び出しプログラムが生成される。参加システム毎のパターン適用済み呼び出しプログラムと、分散トランザクションフロー1001が表す参加システム順（呼び出し順）とを基に、当該参加システム順に実行されるパターン適用済み呼び出しプログラムを含んだ分散トランザクションプログラム803が生成される。

【0086】

以上の図11記載の一連のフローにより、分散トランザクションプログラム803が生成され、図3のS311が終了する。

【0087】

図3を参照して、分散トランザクションに係る操作（A302）、トランザクション要求A303、トランザクション結果A304、及び結果の応答（A305）について説明する。

【0088】

本実施形態においては、ユーザ300は管理システム100で生成された分散トランザクションプログラム803の実行を、管理システム100に分散トランザクションに係る操作A302として要求することで、分散トランザクションを実行させることができる。管理システム100の分散トランザクション実行部105は、ユーザ300からの要求を受け取り、分散トランザクションに関わる複数（又は一つ）の参加システム110それぞれに順序通りにトランザクション要求A303を発行することにより、分散トランザクションプログラム803を実行する。各参加システム、例えば参加システム1は、管理システム100からのトランザクション要求A303を受け付け、トランザクション実行部111、データベース管理部112、及びデータベース113の連携により参加システム1におけるトランザクションを実行し、結果をトランザクション結果A304として管理システム100に返す。各参加システムにおけるトランザクションは、実施形態に応じて、例えばXAに代表される仕様に基づいてもよいし、特定のプロトコルや参加システムが提供するAPIに応じたSAGAの実行により実現されていてもよい。

【0089】

ユーザ300は、分散トランザクションに係る操作の結果の応答（A305）を管理システム100から受け取り、分散トランザクションに係る一連の処理を終える。

【0090】

かくして、本実施形態の情報処理システム1では、分散トランザクションフロー、及び参加システムのサービス特性が所与とできない環境下においても、分散トランザクション管理システム100の実装を改変することなく、適切なパターンの組み合わせによりデータ整合性維持に要する工数を減じることができる。

【0091】

以上の説明は、例えば下記のように総括することができる。下記の総括は、上述の実施形態の説明の補足又は実施形態の変形例を含んでよい。

【0092】

分散トランザクション管理部は、入力部51と、適用パターン判定部101と、分散トランザクションプログラム生成部104とを有する。入力部51は、分散トランザクションフロー1001及びサービス特性テーブル701の入力を受け付ける。分散トランザクションフロー1001は、分散トランザクションにおいて呼び出される複数の参加システム110それぞれの呼び出し順番を規定した情報である。複数の参加システム110の各々は、当該分散トランザクションに参加するシステムである。サービス特性テーブル701は、複数の参加システム110の各々について当該参加システムの特性であるサービス特性を表すテーブルである。適用パターン判定部101は、呼び出し先の参加システム1

10

20

30

40

50

10のサービス特性の制約条件を表す情報をパターン毎に含んだパターンテーブル501を基に、サービス特性テーブル701が表す参加システム110毎に、当該参加システム110のサービス特性に対応した制約条件を満たすパターンを適用パターンとして特定する。分散トランザクションプログラム生成部104は、複数の参加システムの各々についてパターン適用済呼び出しプログラムを生成し、当該複数の参加システム110の各々のパターン適用済呼び出しプログラムと、分散トランザクションフロー1001が表す呼び出し順序とを基に、当該呼び出し順序で実行される複数のパターン適用済呼び出しプログラムを含んだ分散トランザクションプログラム803を生成する。複数の参加システム110の各々について、パターン適用済呼び出しプログラムは、それぞれパターンが記述された複数のパターンプログラムのうち、当該参加システム110について上記特定された適用パターンのパターンプログラムが、当該参加システム110を呼び出し先とした呼び出しプログラムに適用されたプログラムである。

10

【0093】

これにより、参加システム110のサービス特性が変化しても、変化後のサービス特性は入力されるサービス特性テーブル701に記述されているため、参加システム110について、変化後のサービス特性に適したパターンが特定され、そのパターンのパターンプログラムが呼び出しプログラムに適用されたパターン適用済呼び出しプログラムが自動生成され、そのパターン適用済呼び出しプログラムを含む分散トランザクションプログラム803が自動生成される。結果として、管理システムの実装を改変することなく、適切なパターンを組み合わせた分散トランザクションを実行することができる。

20

【0094】

分散トランザクション管理システム100が、分散トランザクションプログラム803を実行する分散トランザクション実行部105を備えてよい。これにより、分散トランザクションプログラム803の生成後に分散トランザクションプログラム803を分散トランザクション管理システム100が実行することが可能である。

【0095】

パターンテーブル501が、パターン毎に、パターン間の依存関係を表す情報を含んでよい。パターン間の依存関係は、パターンの階層関係や、前後（例えば、直前又は直後）のパターンとの関係でよい。適用パターン判定部101が、サービス特性テーブル701が表す参加システム110毎に、当該参加システム110の適用パターンを、当該参加システム110のサービス特性に対応した制約条件を表す情報の他に、当該参加システム110に対応した、パターン間の依存関係を表す情報を基に、特定してよい。これにより、パターン間の依存関係に矛盾しない適切なパターンを適用パターンとして特定することができる。

30

【0096】

各パターンについて、パターン間の依存関係を表す情報は、パターンの階層関係を表す情報である依存パターン（階層）を含んでよい。適用パターン判定部101が、サービス特性テーブル701が表す参加システム110毎に、当該参加システム110の適用パターンとして、当該参加システム110のサービス特性に対応した制約条件を満たし、且つ、当該参加システム110について既に特定済の適用パターンの階層が当該参加システム110に対応した依存パターン（階層）が表す階層を満たすパターンを特定してよい。これにより、パターンの階層関係に矛盾しない適切なパターンを適用パターンとして特定することができる。

40

【0097】

各パターンについて、パターン間の依存関係を表す情報は、下記（a）及び（b）のうちの少なくとも一つ、

（a）当該パターンが適用パターンとされる参加システム110の呼び出し順番の直前の呼び出し順番の参加システム110についての適用パターンを表す情報である依存パターン（前）、

（b）当該パターンが適用パターンとされる参加システム110の呼び出し順番の直後の

50

呼び出し順番の参加システム 110 についての適用パターンを表す情報である依存パターン（後）、

を含んでよい。適用パターン判定部 101 が、サービス特性テーブル 701 が表す参加システム 110 毎に、当該参加システム 110 の適用パターンとして、当該参加システム 110 のサービス特性に対応した制約条件を満たし、且つ、下記（A）及び（B）のうちの少なくとも一つを満たすパターンを特定してよい。これにより、呼び出し順番が直前又は直後の順番である参加システム 110 に適用のパターンに矛盾しない適切なパターンを適用パターンとして特定することができる。

（A）当該参加システム 110 の呼び出し順番の直前の呼び出し順番の参加システム 110 についての適用パターンが、当該参加システム 110 に対応した依存パターン（前）が表す適用パターンに該当する。

10

（B）当該参加システム 110 の呼び出し順番の直後の呼び出し順番の参加システム 110 についての適用パターンが、当該参加システム 110 に対応した依存パターン（後）が表す適用パターンに該当する。

【0098】

分散トランザクションプログラム生成部 104 は、分散トランザクションプログラム 803 の構造が一意に定まらない場合、所定のルール又はユーザによる入力を基に、分散トランザクションプログラム 803 の構造を決定してよい。これにより、分散トランザクションプログラム 803 の構造を適切に定めることができる。例えば、分散トランザクションプログラム生成部 104 は、分散トランザクションプログラム 803 の構造が一意に定まらない場合、所定のルールを基に分散トランザクションプログラム 803 の構造を決定し、それでもその構造が一意に定まらない場合に、ユーザによる入力を基に、分散トランザクションプログラム 803 の構造を一意に決定してよい。

20

【0099】

パターンテーブル 501 が、パターン毎に、当該パターンの適用がオプションか否かを表す情報を含んでよい。分散トランザクションプログラム 803 の構造が一意に定まらない場合とは、特定された少なくとも一つの適用パターンがオプションである場合でよい。このように、パターンの適用をオプションにすることに伴い生じる、分散トランザクションプログラム 803 の構造が一意に定まらないという課題を、上述のようにルール又はユーザ入力により解決することができる。

30

【0100】

適用がオプションである少なくとも一つのパターンについて、当該パターンの適用に関するルールに従う処理のためのパターンルールプログラムが関連付けられており、特定された適用パターンに適用がオプションの適用パターンがあり、当該適用パターンについて、関連付けられたパターンルールプログラムがある場合、分散トランザクションプログラム生成部 104 は、分散トランザクションプログラム 803 の構造を一意に定めるために当該パターンルールプログラムを実行してよい。このようにして、分散トランザクションプログラム 803 の構造が一意に定まらない場合には該当のルールプログラムを用いて自動的に分散トランザクションプログラム 803 の構造が一意に定めることが期待できる。

【0101】

40

以上、一実施形態を説明したが、これは本発明の説明のための例示であって、本発明の範囲をこの実施形態にのみ限定する趣旨ではない。本発明は、他の種々の形態でも実施することが可能である。

【0102】

例えば、図 11 において S1103 及び S1104 が無く（例えば、適用パターンが常に一義的に決まる場合（例えば、「オプション」列 517 が無い場合））、S1102 の次に S1105 が行われてもよい。また、例えば、図 11 の S1103 及び S1104 のうち S1103 が無く S1104 は存在してもよい。

【符号の説明】

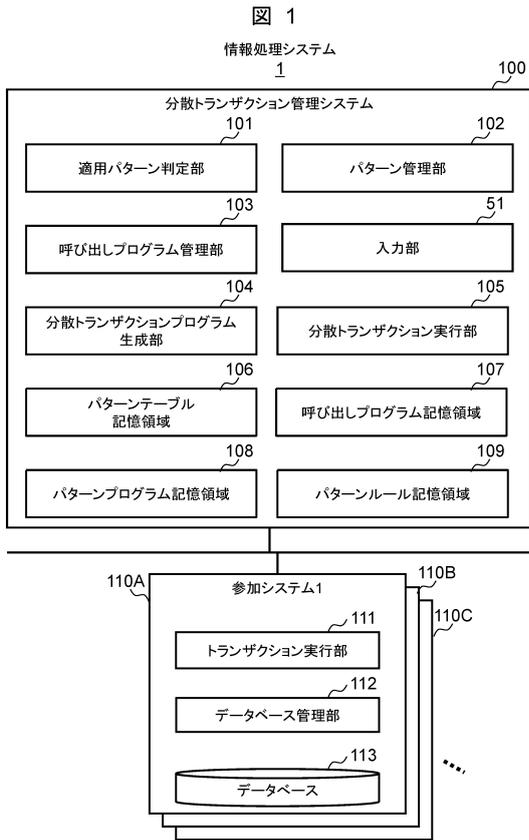
【0103】

50

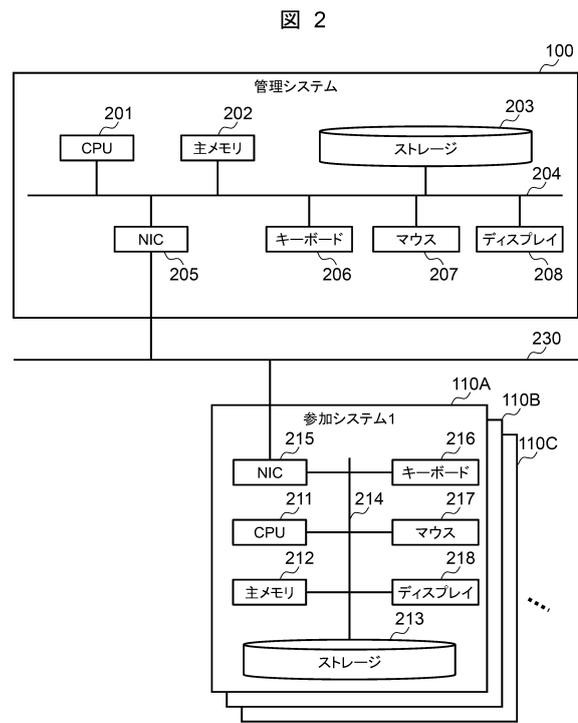
100 : 分散トランザクション管理システム

【図面】

【図 1】



【図 2】



10

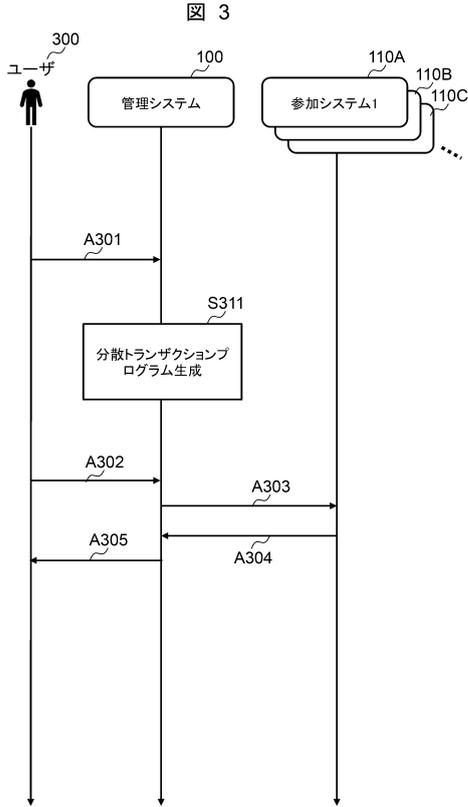
20

30

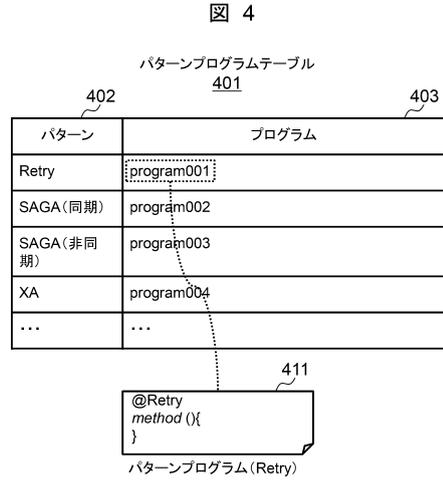
40

50

【 図 3 】



【 図 4 】



10

20

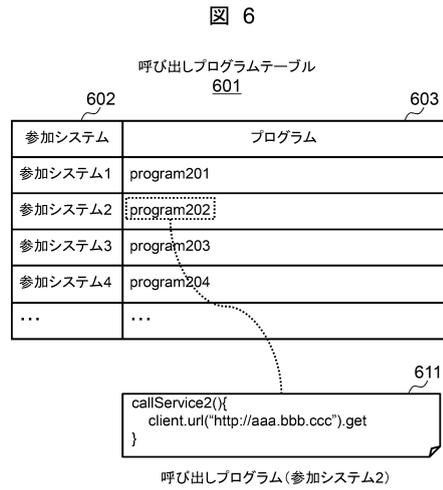
【 図 5 】

図 5

パターンテーブル 501

511	512	513	514	515	516	517
パターン	依存パターン (前)	依存パターン (後)	依存パターン (前後)	呼び出し元制約	呼び出し先制約	オプション
XA	-	-	XA	XA	XA	-
XA(排他制御)	-	XA, XA(排他制御)	-	-	-	Yes
SAGA(同期)	-	-	SAGA(同期)	SAGA(同期)	SAGA(同期)	-
SAGA(非同期)	-	-	SAGA(非同期)	SAGA(非同期)	SAGA(非同期)	-
TCC	-	-	TCC	TCC	TCC	-
多重戻りメッセージ 返答(同期)	-	-	SAGA(同期)	SAGA(同期)、 TCC	-	-
多重戻りメッセージ 返答(非同期)	-	-	SAGA(非同期)	SAGA(非同期)、 TCC	-	-
Retry (多重実行なし)	-	-	SAGA(同期)、 SAGA(非同期)、 多重戻りメッセージ 返答(同期)、 TCC	Retry (多重実行なし)	異常 (多重実行防止なし)	...

【 図 6 】



30

40

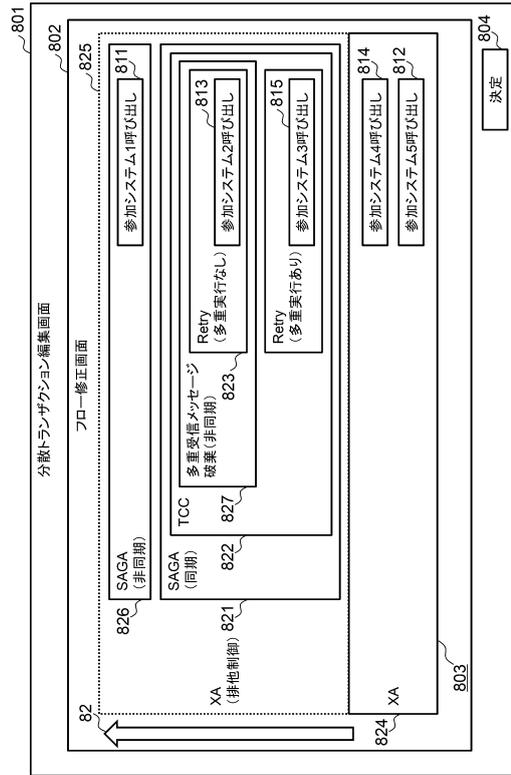
50

【 図 7 】

図 7
サービス特性テーブル
701

サービス	サービス特性
参加システム1	SAGA(非同期)
参加システム2	SAGA(同期), TCC, 幂等(多重実行防止あり)
参加システム3	SAGA(同期), TCC, 幂等(多重実行防止なし)
参加システム4	XA
参加システム5	XA

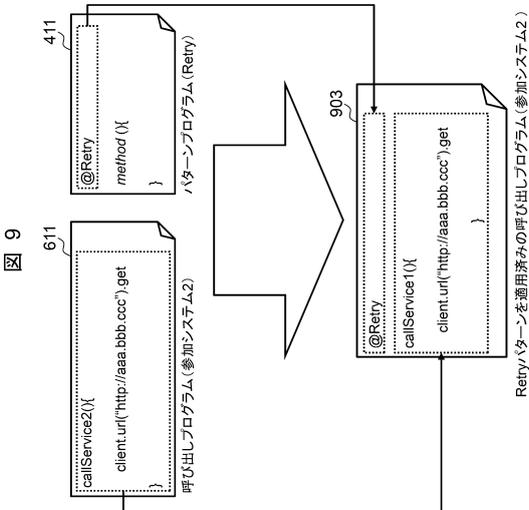
【 図 8 】



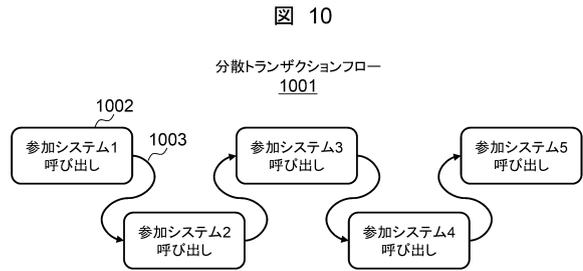
10

20

【 図 9 】



【 図 10 】



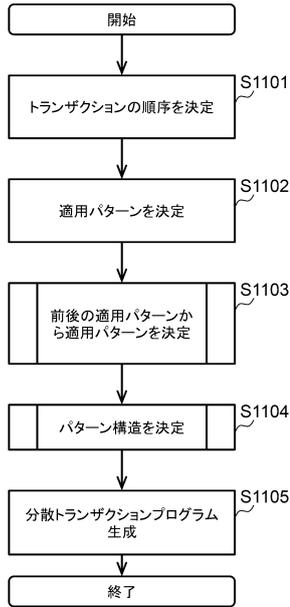
30

40

50

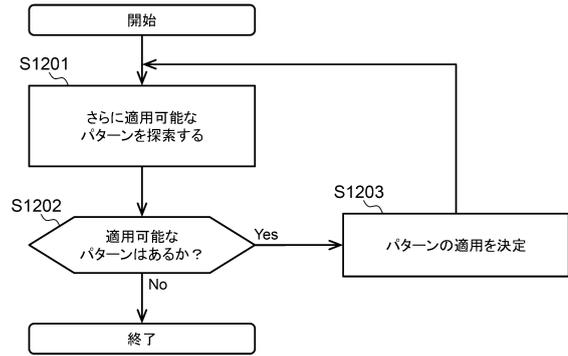
【図 1 1】

図 11



【図 1 2】

図 12

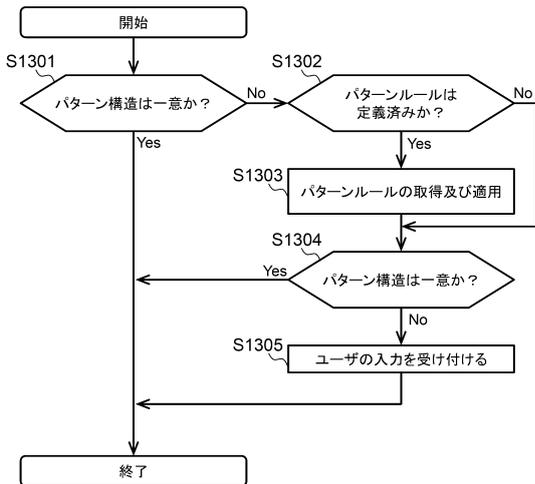


10

20

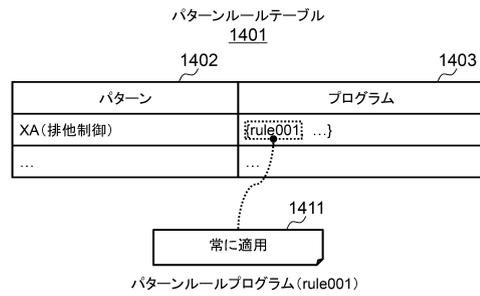
【図 1 3】

図 13



【図 1 4】

図 14



30

40

50

フロントページの続き

東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内

審査官 北川 純次

- (56)参考文献 特開2004-272912(JP,A)
特開平8-286918(JP,A)
特開平9-223004(JP,A)
米国特許出願公開第2019/0220331(US,A1)
- (58)調査した分野 (Int.Cl., DB名)
G06F 8/30
G06F 9/44