

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2014年1月30日(30.01.2014)



(10) 国際公開番号

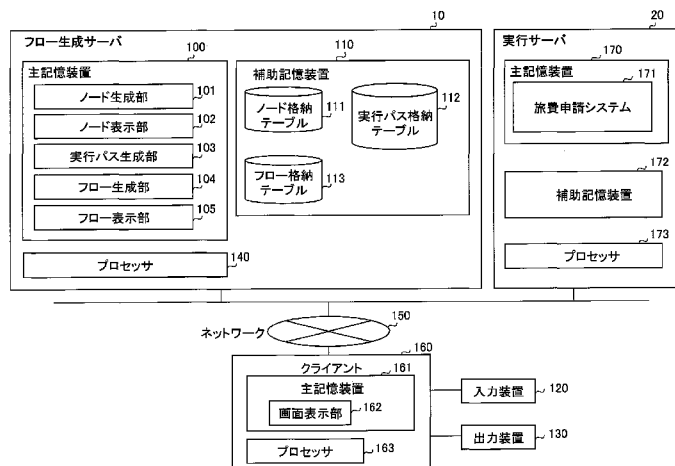
WO 2014/016884 A1

- (51) 国際特許分類:
G06Q 10/06 (2012.01)
- (21) 国際出願番号: PCT/JP2012/068565
- (22) 国際出願日: 2012年7月23日(23.07.2012)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (71) 出願人(米国を除く全ての指定国について): 株式会社日立製作所(HITACHI, LTD.) [JP/JP]; 〒1008280 東京都千代田区丸の内一丁目6番6号 Tokyo (JP).
- (72) 発明者; および
- (75) 発明者/出願人(米国についてのみ): 水藤 裕介(MITOU, Yusuke) [JP/JP]; 〒2448555 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所 I Tプラットフォーム事業本部内 Kanagawa (JP). 市川 和幸(ICHIKAWA, Kazuyuki) [JP/JP]; 〒2448555 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所 I Tプラットフォーム事業本部内 Kanagawa (JP).
- (74) 代理人: 藤井 正弘(FUJII, Masahiro); 〒1050001 東京都港区虎ノ門一丁目16番4号アーバン虎ノ門ビル 後藤特許事務所 Tokyo (JP).
- (81) 指定国(表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) 指定国(表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).
- 添付公開書類:
— 国際調査報告(条約第21条(3))

(54) Title: COMPUTER SYSTEM, DATA PROCESSING METHOD, AND RECORDING MEDIUM

(54) 発明の名称: 計算機システム、データ処理方法、及び、記録媒体

[図1]



100 Flow generation server
20 Execution server
100, 161, 170 Primary storage device
101 Node generation unit
102 Node display unit
103 Execution path generation unit
104 Flow generation unit
105 Flow display unit
110, 172 Supplementary storage device
111 Node storage table

112 Execution path storage table
113 Flow storage table
120 Input device
130 Output device
140, 163, 173 Processor
150 Network
160 Client
162 Screen display unit
171 Travel expense request system

(57) Abstract: A computer system comprises: a plurality of execution paths which include identifiers which denotes a plurality of executed processes, and which denotes a sequence wherein each process is executed; and a flow generation unit which generates a flowchart on the basis of each execution path. The flow generation unit: extracts, from the plurality of execution paths, identifiers of the processes which are executed after executing the same number of processes from initially executed processes; executes a first determination of whether the extracted identifiers of each process are the same; generates, if the extracted identifiers of each process are the same as a result of the first determination, one shared process symbol including the identifier of each process which is the same; generates, if at least one of the extracted identifiers of each process differs from the other identifiers as a result of the first determination, a plurality of unshared process symbols including each of the extracted process identifiers for which at least one is determined to be different in the first determination; and generates a flowchart which includes the generated shared process symbol and at least one of the generated

unshared process symbols.

(57) 要約:

[続葉有]



計算機システムは、実行された複数の処理を示す識別子を含み、かつ、各処理が実行された順番を示す、複数の実行パスと、各実行パスに基づいて、フローチャート生成するフロー生成部と、を有し、フロー生成部は、最初に実行された処理から同じ数の処理が実行された処理の識別子を、複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第1の判定を実行し、第1の判定の結果、抽出された各処理の識別子が同じである場合、同じである各処理の識別子を含む一つの共通処理記号を生成し、第1の判定の結果、抽出された各処理の識別子のうち少なくとも一つが他の識別子と異なる場合、第1の判定において少なくとも一つが異なると判定された、抽出された処理の識別子を各々含む複数の非共通処理記号を生成し、生成された共通処理記号、及び、生成された非共通処理記号の少なくとも一つを含むフローチャートを生成する。

明 細 書

発明の名称： 計算機システム、データ処理方法、及び、記録媒体
技術分野

[0001] 本発明は、計算機システムに関し、特に、フローチャートを生成する計算機システム、データ処理方法及びデータ処理プログラムを格納した非一次的記録媒体に関する。

背景技術

[0002] 近年では、業務における作業の効率化を意図し、フローチャートによって表現された業務手順を、従業者間で共有することが多い。フローチャートによって表現される業務手順には、属人化されがちな業務手順、又は、従業者が保持する知識及びノウハウ等が含まれる。

[0003] しかし、このようなフローチャートは、従業者、又は、従業者から委託された人によって作成されなければならない。人によってフローチャートが作成される結果、フローチャートを作成するためのコストが必要になり、また、作成されるフローチャートの品質が安定しないなどの問題が生じる。

[0004] このような問題を解決するため、従来から、フローチャートを自動で生成する技術が求められている。

[0005] 従来において、予め用意されたフローチャートについて、このフローチャートを構成する各プロセスの実行主体がコンピュータなのか人なのかを判定し、実行主体がコンピュータである場合、予め定義された「条件」に基づいて「部分フロー」を生成する方法が提案されている（例えば、特許文献1参照）。

先行技術文献

特許文献

[0006] 特許文献1：特開2007-122651号公報

発明の概要

発明が解決しようとする課題

[0007] しかし、特許文献１に記載された方法は、部分フローを生成するために必要な「予め用意されたフローチャート」を自動で生成することができない。つまり、特許文献１に記載された方法を用いた場合も、人が「予め用意されたフローチャート」を作成しなければならず、その結果、フローチャートを作成するためのコストが必要になる。

[0008] また、人が業務手順を表現するフローチャートを作成する場合、作成されたフローチャートが、実際には実行されない処理の流れ（経路）を示す場合がある。これは、計算機システムにおいて実際に実行される処理の流れとは独立して、フローチャートが人によって作成されることにより発生する。

[0009] このように実際には実行されない処理の流れは、フローチャート作成時には人によって気づかれにくく、従業員等によるフローチャートの利用時には、業務の流れをわかりにくくさせるという課題がある。

[0010] 本発明は、前述のような課題を解決するため、人の業務に沿ったフローチャートを自動的に生成するシステムの提供を目的とする。

課題を解決するための手段

[0011] 本発明の代表的な一形態によると、プロセッサ、及び、メモリを有する計算機システムであって、前記計算機システムは、実行された複数の処理を示す識別子を含み、かつ、前記各処理が実行された順番を示す、複数の実行パスと、前記各実行パスに基づいて、フローチャートを生成するフロー生成部と、を有し、前記フロー生成部は、最初に実行された前記処理から同じ数の前記処理が実行された前記処理の識別子を、前記複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第１の判定を実行し、前記第１の判定の結果、前記抽出された各処理の識別子が同じである場合、前記同じである各処理の識別子を含む一つの共通処理記号を生成し、前記第１の判定の結果、前記抽出された各処理の識別子のうち少なくとも一つが他の前記識別子と異なる場合、前記第１の判定において少なくとも一つが異なると判定された、前記抽出された処理の識別子を各々含む複数の非共通処理記号を生成し、前記生成された共通処理記号、及び、前記生成された非

共通処理記号の少なくとも一つを含む前記フローチャートを生成する。

発明の効果

[0012] 本発明の一実施形態によると、実際に実行された処理に沿った精度の高いフローチャートを作成することができる。

図面の簡単な説明

[0013] [図1]本実施例1の計算機システムを示すブロック図である。

[図2]本実施例1のノード格納テーブルを示す説明図である。

[図3]本実施例1の実行パスを生成する処理を示す説明図である。

[図4]本実施例1の実行パス格納テーブルを示す説明図である。

[図5]本実施例1の実行パスからフローチャートを生成する処理を示すフローチャートである。

[図6]本実施例1のフローチャートの生成処理によって生成されたフローチャートの表示例を示す説明図である。

[図7]本実施例2のフローチャートを生成するための処理を示すフローチャートである。

[図8]本実施例2のフローチャートの生成処理によって生成されたフローチャートの表示例を示す説明図である。

[図9]本実施例3の計算機システムを示すブロック図である。

[図10]本実施例3の計算機システムによる処理の概要を示す説明図である。

[図11]本実施例3のユーザによって作成されたフローチャートを示す説明図である。

[図12]本実施例3のフローチャートに含まれる処理の流れを示す説明図である。

[図13]本実施例3の分岐ノードの接続先に関する分岐情報を示す説明図である。

[図14]本実施例3のフローチャートの処理の流れの数を算出する処理を示すフローチャートである。

[図15]本実施例3のフローチャートの処理の数を算出する際のスタックの変

化を示す説明図である。

[図16]本実施例4の計算機システムを示すブロック図である。

[図17]本実施例4の実行パス生成部による処理を示す説明図である。

発明を実施するための形態

[0014] 以下、フロー自動生成方法およびプログラムの実施形態として、実施例をもとに図面で説明する。

実施例 1

[0015] 本実施例1の計算機システムは、ユーザが実際に行った業務システムへの作業に従って、フローチャートを生成する。

[0016] 図1は、本実施例1の計算機システムを示すブロック図である。

[0017] 本実施例1の計算機システムは、フロー生成サーバ10、実行サーバ20、ネットワーク150、及び、クライアント160を備える。

[0018] 実行サーバ20は、プログラムによって実装された業務システムを実行することによって、ネットワーク150を介してクライアント160に、サービスを提供する計算機システムである。実行サーバ20は、プロセッサ173、補助記憶装置172、及び、主記憶装置170を有する。

[0019] クライアント160は、実行サーバ20によって提供されるサービスを用いて、ユーザが作業をするための計算機システムである。クライアント160は、プロセッサ163及び主記憶装置161を有し、入力装置120及び出力装置130に接続される。

[0020] フロー生成サーバ10は、クライアント160からの入力に基づいて、ユーザの作業手順を示すフローチャートを生成する計算機システムである。フロー生成サーバ10は、プロセッサ140、主記憶装置100、及び、補助記憶装置110を有する。

[0021] ネットワーク150は、実行サーバ20、クライアント160、及び、フロー生成サーバ10を接続するためのネットワークである。ネットワーク150は、例えば、インターネット、LAN、又は、WAN等のネットワークでもよい。

- [0022] 実行サーバ20におけるプロセッサ173は、CPU等の演算装置であり、補助記憶装置172に格納されるデータ、及び、ソースプログラム等を主記憶装置170に読み出すことによって、プログラムを実行する。補助記憶装置172は、データ、及び、ソースプログラム等を保持する。主記憶装置170は、プロセッサが実行するプログラムを保持する記憶領域である。
- [0023] 主記憶装置170は、旅費申請システム171を保持する。旅費申請システム171は、プログラムによって実装された業務システムであり、実施例1においてユーザにサービスを提供する。以下において、旅費申請システム171が提供するサービスを、旅費申請業務と記載する。
- [0024] クライアント160におけるプロセッサ163は、CPU等の演算装置であり、主記憶装置161を用いてプログラムを実行する。主記憶装置161は、プロセッサが実行するプログラムを保持する。
- [0025] 主記憶装置161は、画面表示部162を保持する。画面表示部162は、実行サーバ20の旅費申請システム171の実行結果等を出力装置130に表示するためのプログラムである。また、画面表示部162は、入力装置120を介してユーザからの指示を受け付ける。そして、画面表示部162は、ネットワーク150を介して実行サーバ20に、ユーザからの指示を送信する。
- [0026] フロー生成サーバ10のプロセッサ140は、CPU等の演算装置である。プロセッサ140は、補助記憶装置110に格納されるデータ、及び、ソースプログラム等を主記憶装置100に読み出すことによって、プログラムを実行する。
- [0027] 補助記憶装置172には、ノード格納テーブル111、実行パス格納テーブル112、及び、フロー格納テーブル113が格納される。ノード格納テーブル111は、旅費申請システム171において実行される処理を示すノードを含む。
- [0028] 本実施例におけるノードとは、業務システムにおいて実行される処理を示すデータである。一つのノードは、一つの処理を示す。また、本実施例にお

ける処理とは、ユーザによる作業によって実行される処理を示す。このため、ユーザによる一つの作業は、一つのノードに対応する。

[0029] 実行パス格納テーブル１１２は、実行パスを含むテーブルである。本実施例における実行パスとは、実行されたノードと、ノードが実行された順番とを示す。

[0030] フロー格納テーブル１１３は、フロー生成サーバ１０において生成されたフローチャートを格納するためのテーブルである。

[0031] 主記憶装置１００は、ノード生成部１０１、ノード表示部１０２、実行パス生成部１０３、フロー生成部１０４、及び、フロー表示部１０５を保持する。

[0032] ノード生成部１０１は、業務システム（旅費申請システム１７１）からノードを生成するためのプログラムである。ノード表示部１０２は、ノード格納テーブル１１１に格納されるノードを、クライアント１６０に接続される出力装置１３０に表示させるためのプログラムである。

[0033] 実行パス生成部１０３は、実行されたノードを示す情報に基づいて、実行パスを生成するプログラムである。フロー生成部１０４は、実行パス格納テーブル１１２に格納される実行パスに基づいて、フローチャートを生成するプログラムである。フロー表示部１０５は、フロー格納テーブル１１３に格納されるフローチャートを、クライアント１６０に接続される出力装置１３０に表示させるためのプログラムである。

[0034] 図２は、本実施例１のノード格納テーブル１１１を示す説明図である。

[0035] ノード格納テーブル１１１は、ノード名１１１１、及び、ノードＩＤ１１１２を含む。ノード格納テーブル１１１に各エントリは、各ノードを示す。

[0036] ノード名１１１１は、各ノードを示す名称を含む。ノード名１１１１に含まれる名称は、ユーザが処理の内容を識別できる文字列である。ノードＩＤ１１１２は、各ノードを一意に示す識別子である。

[0037] 図２に示す１１１には、旅費申請システム１７１によって実行されるノードが含まれる。ノード生成部１０１は、旅費申請システム１７１に基づいて

、旅費申請システム１７１に基づいてノードを生成し、生成されたノードをノード格納テーブル１１１に格納する。

[0038] ノード生成部１０１は、ノードを生成するために、例えば、ユーザに処理を選択させる処理を示すソースプログラムを、旅費申請システム１７１から抽出してもよい。この場合、ノード生成部１０１は、抽出されたソースプログラムに含まれる、ユーザが選択する処理の名称（ユーザに表示される画面の名称等）を、ノード名１１１１に格納する。そして、ノード生成部１０１は、ノード名１１１１に値を格納されたエントリのノードＩＤ１１１２に、一意な識別子を割当ててゐる。

[0039] ノード表示部１０２は、クライアント１６０の画面表示部１６２に、ノード名１１１１が示すノード名を送信する。画面表示部１６２は、送信されたノード名を出力装置１３０に表示する。

[0040] ユーザは、出力装置１３０に表示されたノード名を、入力装置１２０によって選択する。例えば、ユーザは、旅費申請システム１７１を実行した際に、自らが行った作業の順番に従って、ノード名を選択する。

[0041] 画面表示部１６２は、ユーザによって選択されたノード名を受け付け、選択されたノード名を実行パス生成部１０３に送信する。

[0042] ここで、画面表示部１６２は、受信したノード名を旅費申請システム１７１に送信し、旅費申請システム１７１に処理を実行させてもよい。

[0043] 図３は、本実施例１の実行パスを生成する処理を示す説明図である。

[0044] 図３に示すノード選択結果４０は、ユーザに表示されたノード名と、ユーザが選択したノード名と、ユーザがノード名を選択した順番とを示す。ノード選択結果４０には、ノード４０ａ～ノード４０ｊが含まれる。

[0045] ノード４０ａ～ノード４０ｊは、画面表示部１６２によってユーザに表示されたノード名である。図３に示すノード選択結果４０は、ユーザが、ノード４０ａ、ノード４０ｂ、ノード４０ｄ、ノード４０ｅ、ノード４０ｆ、ノード４０ｇ、ノード４０ｈ、及び、ノード４０ｊの順番に、処理を選択したことを示す。

- [0046] 実行パス生成部１０３は、画面表示部１６２から送信された、ノード名とノード名が選択された順番とに基づいて、ノード選択結果４０を生成する。
- [0047] そして、実行パス生成部１０３は、ノード選択結果４０に基づいて、実行パス４１を生成する。実行パス４１は、選択されたノード名に対応するノードＩＤを要素に含む配列である。実行パス４１は、ノード選択結果４０が示すノード名が選択された順番にノードＩＤを有する。
- [0048] 実行パス生成部１０３は、ノード選択結果４０とノード格納テーブル１１１とを参照し、ノード選択結果４０のノード名をノード名１１１１に含むエントリのノードＩＤ１１１２を抽出する。これによって、実行パス生成部１０３は、ノード名に対応するノードＩＤを実行パス４１に格納する。
- [0049] なお、図３に示す実行パス４１には、ノードＩＤのみが含まれるが、実行パス４１の各要素には、ノードＩＤとノード名とが含まれてもよい。
- [0050] また、本実施例の実行パスは各要素の先頭から順に要素番号（図示されていない）を有する。本実施例の実行パスは、配列の先頭の要素に１の要素番号が割り当てられる。また、要素番号は、要素が一つ増えるに従い１ずつ増加する。
- [0051] 図４は、本実施例１の実行パス格納テーブル１１２を示す説明図である。
- [0052] 実行パス格納テーブル１１２には、実行パス生成部１０３によって生成された実行パスが、実行パスが示すノードが実行される業務システムごとに格納される。図４に示す実行パス格納テーブル１１２は、旅費申請システム１７１の実行パスを含む。図４における旅費申請システム１７１の実行パスは、実行パス４１及び実行パス４２である。
- [0053] 図５は、本実施例１の実行パスからフローチャートを生成する処理を示すフローチャートである。
- [0054] フロー生成部１０４は、定期的、又は、管理者等の指示を受信した際等のタイミングにおいて、図５に示す処理を開始する。フロー生成部１０４は、図５に示す処理において、最初に、実行パス格納テーブル１１２から実行パスをすべて取得する（６０１）。なお、実行パス格納テーブル１１２に複数

の業務システムに関する実行パスが格納される場合、フロー生成部１０４は、管理者等の指示に従い、フローチャートを生成する業務システムの実行パスをすべて取得する。

[0055] ステップ６０１の後、フロー生成部１０４は、主記憶装置１００にパラメータ*i*を保持する。そして、フロー生成部１０４は、パラメータ*i*に初期値として１を格納する（６０２）。パラメータ*i*は、実行パスにおける要素番号を示す。

[0056] ステップ６０２の後、フロー生成部１０４は、ステップ６０１において取得された各実行パスにおいて、パラメータ*i*が示す要素番号の要素を抽出する。そして抽出された要素を相互に比較する。そして、フロー生成部１０４は、比較の結果、パラメータ*i*が示す要素番号の要素のノードＩＤが各実行パスにおいてすべて一致するか否かを判定する（６０３）。

[0057] なお、フロー生成部１０４は、いずれかの実行パスが、パラメータ*i*が示す要素番号の要素を含まない場合も、パラメータ*i*が示す要素番号の要素のうち、ノードＩＤが異なる要素があると判定してよい。これによって、各実行パスに含まれる要素の数が異なる場合も、フロー生成部１０４は、図５に示す処理を実行可能である。

[0058] ステップ６０３における判定の結果、パラメータ*i*が示す要素番号の要素のノードＩＤが各実行パスにおいてすべて一致する場合、フロー生成部１０４は、パラメータ*i*が示す要素番号の要素と同じノード名及び同じノードＩＤのノード（共通ノード）を一つ生成する。ここで、フロー生成部１０４は、フローチャートにおいて共通ノードを表現するため、共通ノードを示す記号を生成する（６０４）。

[0059] フロー生成部１０４は、ステップ６０４において、前回実行されたステップ６０４において生成された共通ノードがある場合、前回生成された共通ノードの記号と新たに生成された共通ノードの記号とを、矢印によって接続する。ステップ６０４において用いられる共通ノードの記号を接続するための矢印は、処理の流れが前回生成された共通ノードから新たに生成された共通

ノードへ進むことを示す。

[0060] ステップ604の後、フロー生成部104は、パラメータ*i*に、1を加算し（605）、ステップ603に戻る。

[0061] ステップ603における判定の結果、各実行パスにおけるパラメータ*i*が示す要素番号の要素のうち、ノードIDが一つでも異なる要素がある場合、フロー生成部104は、分岐ノードを生成する。また、フロー生成部104は、フローチャートにおいて分岐ノードを表現するため、分岐ノードを示す記号を生成する（606）。

[0062] 本実施例における分岐ノードとは、共通ノードによって実行されていた処理の流れが、所定の条件によって、複数の異なる処理の流れに分岐することを示すノードである。また、分岐ノードは、ユーザによる判断、または、システムによる判定によって、処理がわかれることを示すノードである。

[0063] フロー生成部104は、ステップ606において、既に生成された共通ノードがある場合、最後に生成された共通ノードを示す記号と生成された分岐ノードを示す記号とを矢印によって接続する。ステップ606において用いられる共通ノードの記号と分岐ノードの記号とを接続する矢印は、処理の流れが共通ノードから分岐ノードへ進むことを示す。

[0064] ステップ606の後、フロー生成部104は、パラメータ*i*が示す値以上の要素番号の要素を、実行パス毎に抽出する。パラメータ*i*が示す値以上の要素番号の要素が、各実行パスに複数含まれる場合、フロー生成部104は、パラメータ*i*が示す値以上の要素番号の複数の要素を、実行パス毎に抽出する。

[0065] そして、フロー生成部104は、抽出された各実行パスの要素のノード名とノードIDのノード（非共通ノード）を、実行パス毎に生成する（607）。また、フロー生成部104は、非共通ノードを示す記号を生成する。ここで、実行パス毎に生成された複数の非共通ノードを、非共通ノード群と記載する。非共通ノードは、各実行パスが示す実行された順番によって、非共通ノード群に含まれる。

- [0066] フロー生成部１０４は、ステップ６０７において、ステップ６０６において生成された分岐ノードの記号と非共通ノード群の先頭の非共通ノードの記号とを矢印によって接続する。ステップ６０７において用いられる分岐ノードの記号と非共通ノードの記号とを接続する矢印は、処理の流れが分岐ノードから各非共通ノードに分岐することを示す。
- [0067] また、フロー生成部１０４は、ステップ６０７において、非共通ノード群に複数の非共通ノードが含まれる場合、実行された順を示す矢印によって、各非共通ノードの記号を接続する。
- [0068] 前述の処理によって、フロー生成部１０４は、共通ノードを示す記号、又は、非共通ノードを示す記号を含むフローチャートを生成できる。このため、フロー生成部１０４は、ステップ６０７において、生成されたフローチャートを、フロー格納テーブル１１３に格納する。
- [0069] 図６は、本実施例１のフローチャートの生成処理によって生成されたフローチャート５０の表示例を示す説明図である。
- [0070] 図６は、図５に示す処理によって生成されるフローチャート５０と、フローチャート５０を出力装置１３０に表示した場合の画面例とを示す。フローチャート５０には、共通ノード５１、分岐ノード５２、及び、非共通ノード５３が含まれる。共通ノード５１は、図５に示すステップ６０４において生成される共通ノードを示し、分岐ノード５２は、図５に示すステップ６０６において生成される分岐ノードを示す。
- [0071] また、非共通ノード５３は、図５に示すステップ６０７において生成される非共通ノードを示す。図６に示す非共通ノード５３は、二つの非共通ノード群を含む。
- [0072] フロー表示部１０５は、図６に示すフローチャート５０をクライアント１６０に送信し、画面表示部１６２を介して、出力装置１３０にフローチャート５０を表示させてもよい。また、画面表示部１６２は、旅費申請システム１７１による処理の実行結果とともに、図６に示すフローチャート５０を表示してもよい。

- [0073] ユーザは、フローチャート50を参照することによって、旅費申請システム171が提供する旅費申請業務において、作業手順の中のいずれの処理を実行しているかを把握することができる。この結果、ユーザは効率的に、かつ、正確に、業務を遂行することができる。
- [0074] なお、前述のフロー生成部104は、図6に示す記号等によって表現されたフローチャートを生成するが、ユーザが処理の流れを認識できれば、図6に示すフローチャート以外の記号等によって表現されたフローチャートを生成してもよい。例えば、フロー生成部104は、ユーザが処理の流れを認識できれば、矢印を直線によって表現してもよい。
- [0075] 実施例1によれば、計算機システムは、ユーザによって実際に行われた作業に従ってフローチャートを生成するため、実際に行われた処理の流れのみを含む精度の高いフローチャートを生成できる。このため、ユーザが、実施例1によって生成されたフローチャートを参照した場合、ユーザは正確に作業手順を把握することができる。
- [0076] また、ユーザが自らフローチャートを作成することはないため、実施例1の計算機システムは、フローチャートの作成のための労力及び時間を低減することができる。

実施例 2

- [0077] 実施例1によるフローチャートの生成処理は、フローチャートにおいて最初に出現する分岐を示す分岐ノードのみを生成した。実施例2によるフローチャートの生成処理は、最後に出現する分岐ノードを生成する。
- [0078] 実施例2における計算機システムの構成は、図1に示す実施例1の構成と同じである。実施例2と実施例1との相違点は、フロー生成部104によるフローチャートの生成処理のみである。
- [0079] 図7は、本実施例2のフローチャートを生成するための処理を示すフローチャートである。
- [0080] 図7に示す処理のうち、ステップ801～ステップ805は、図5に示すステップ601～ステップ605と同じ処理である。このため、以下におい

て、図5に示す処理と異なる処理のみについて説明する。

- [0081] ステップ803における判定の結果、各実行パスにおけるパラメータ*i*が示す要素番号の要素のうち、ノードIDが一つでも異なる要素がある場合、フロー生成部104は、各実行パスの最後の要素の要素番号*j*を取得する（806）。なお、要素番号*j*は、実行パスによって値が異なってもよい。
- [0082] ステップ806の後、フロー生成部104は、ステップ806において取得された要素番号*j*の要素を各実行パスから抽出し、抽出された複数の要素のノードIDを相互に比較する。そして、フロー生成部104は、比較の結果、各実行パスにおける要素番号*j*の要素のノードIDがすべて一致するか否かを判定する（807）。
- [0083] ステップ807において、要素番号*j*の要素のノードIDがすべて一致する場合、フロー生成部104は、各実行パスの要素番号*j*の要素と同じノードID及びノード名を含む共通ノードを一つ生成する。また、フロー生成部104は、共通ノードを示す記号を生成する（808）。
- [0084] フロー生成部104は、ステップ808において、前回実行されたステップ808において生成された共通ノードがある場合、前回生成された共通ノードの記号と新たに生成された共通ノードの記号とを、矢印によって接続する。ステップ808において用いられる共通ノードの記号を接続するための矢印は、処理の流れが新たに生成された共通ノードから前回生成された共通ノードへ進むことを示す。
- [0085] そして、フロー生成部104は、各実行パスの要素番号*j*から、各々1を減算する（809）。ステップ809の後、フロー生成部104は、ステップ807に戻る。
- [0086] ステップ807において、各実行パスの要素番号*j*の要素のノードIDが一つでも異なる場合、フロー生成部104は、分岐ノードを生成する。ここで、フロー生成部104は、分岐ノードを示す記号と合流点とを生成する（810）。本実施例における合流点とは、非共通ノードによって実行されていた複数の処理の流れが一つの共通した処理に継続することを示す記号であ

る。

- [0087] ステップ804において生成された共通ノードがある場合、ステップ810においてフロー生成部104は、ステップ804において最後に生成された共通ノードの記号と分岐ノードの記号とを矢印によって接続する。ここで用いられる共通ノードの記号と分岐ノードの記号とを接続する矢印は、共通ノードから分岐ノードへ処理の流れが進むことを示す。
- [0088] また、ステップ808において生成された共通ノードがある場合、ステップ810においてフロー生成部104は、ステップ808に最後に生成された共通ノードと合流点とを矢印によって接続する。ここで用いられる共通ノードの記号と合流点とを接続する矢印は、合流点から共通ノードへ処理の流れが進むことを示す。
- [0089] ステップ810の後、フロー生成部104は、各実行パスの要素番号jからパラメータiまでの要素を、実行パス毎に抽出する。そして、フロー生成部104は、抽出された各要素のノード名とノードIDとを含む非共通ノードを、実行パス毎に生成する。また、フロー生成部104は、非共通ノードを示す記号を生成する(811)。生成された各非共通ノードは、実施例1と同じく複数の非共通ノード群に含まれる。
- [0090] フロー生成部104は、ステップ811において、生成された各非共通ノード群の先頭の非共通ノードの記号と、ステップ810において生成された分岐ノードの記号とを、矢印によって接続する。ここで用いられる分岐ノードの記号と非共通ノードの記号とを接続する矢印は、処理の流れが分岐ノードから各非共通ノードへ分岐することを示す。
- [0091] また、フロー生成部104は、ステップ811において、生成された非共通ノード群の終端の非共通ノードの記号と合流点とを、矢印によって接続する。ここで用いられる非共通ノードの記号と合流点とを接続する矢印は、処理の流れが非共通ノードから合流点に進むことを示す。
- [0092] 前述の処理によって、フロー生成部104は、フローチャートを生成する。そして、ステップ811において、フロー生成部104は、生成されたフ

ローチャートをフロー格納テーブル 113 に格納する。

[0093] 図 8 は、本実施例 2 のフローチャートの生成処理によって生成されたフローチャート 60 の表示例を示す説明図である。

[0094] 図 8 は、図 7 に示す処理によって生成されるフローチャート 60 と、フローチャート 60 を表示した例を示す。フローチャート 60 には、共通ノード 61、分岐ノード 62、非共通ノード 63、合流点 64、及び、共通ノード 65 が含まれる。共通ノード 61 は、図 7 に示すステップ 804 において生成される共通ノードを示し、分岐ノード 62 は、図 7 に示すステップ 810 において生成される分岐ノードを示す。

[0095] また、非共通ノード 63 は、図 7 に示すステップ 811 において生成される非共通ノードを示す。非共通ノード 63 は、二つの非共通ノード群を含む。また、共通ノード 65 は、図 6 に示す非共通ノード 53 の終端の非共通ノードに相当する。

[0096] また、合流点 64 は、図 7 に示すステップ 810 において生成される合流点である。また、共通ノード 65 は、図 7 に示すステップ 808 において生成される共通ノードを示す。

[0097] フロー表示部 105 は、実施例 1 と同じく、フロー格納テーブル 113 に格納されるフローチャート 60 を、画面表示部 162 を介して出力装置 130 に表示してもよい。

[0098] なお、前述のフロー生成部 104 は、図 8 に示す記号等によって表現されたフローチャートを生成するが、ユーザが処理の流れを認識できれば、図 8 に示すフローチャート以外の記号等によって表現されたフローチャートを生成してもよい。例えば、合流点 64 は、結合された矢印によって表現されてもよい。

[0099] 実施例 2 によれば、実施例 1 において非共通ノードとしてフローチャートに含まれる処理も、共通ノードとしてフローチャートに含めることが可能である。これによって、実施例 2 の計算機システムは、ユーザが作業手順を把握しやすいフローチャートを生成できる。

実施例 3

- [0100] 実施例 3 の計算機システムは、あらかじめ与えられたフローチャートと、実行パスとに基づいて、再度フローチャートを生成する必要があるか否かを判定する。これによって、既にフローチャートが生成されている場合においても、ユーザが実際に行う作業を正確に表現できていないフローチャートを訂正することができる。
- [0101] 図 9 は、本実施例 3 の計算機システムを示すブロック図である。
- [0102] 図 9 に示す計算機システムは、実施例 3 の処理を行うために有する構成のみを有する。
- [0103] 実施例 3 の計算機システムは、フロー生成サーバ 10、ネットワーク 150、及び、クライアント 160 を少なくとも有する。
- [0104] 実施例 3 におけるクライアント 160 は、実施例 1 のクライアント 160 と同じ物理的な構成を有し、ユーザによる指示を受け付け、ユーザにフローチャートを表示するための計算機である。実施例 3 におけるネットワーク 150 も、実施例 1 におけるネットワーク 150 と同じである。
- [0105] フロー生成サーバ 10 は、実施例 1 と同じく、プロセッサ 140、主記憶装置 100、及び、補助記憶装置 110 を有する。実施例 3 の主記憶装置 100 は、テスト履歴記録部 201、フロー展開部 202、フロー比較部 203、フロー再生成部 204、及び、フロー表示部 105 を含む。
- [0106] テスト履歴記録部 201 は、テストにおいて実行された処理を示す実行パスを生成するプログラムである。フロー展開部 202 は、あらかじめ与えられたフローチャートを解析し、フローチャートに基づいて処理の流れ（経路）の数を算出するプログラムである。
- [0107] フロー比較部 203 は、フローチャートに基づいて算出された処理の流れの数と実行パスの数とが同じ数か否かを判定するプログラムである。
- [0108] フロー再生成部 204 は、実施例 1 及び実施例 2 のフロー生成部 104 と同じであり、実行パスに基づいてフローチャートを生成するプログラムである。フロー表示部 105 は、実施例 1 及び実施例 2 のフロー表示部 105 と

同じである。

[0109] 補助記憶装置 110 は、テスト履歴テーブル 211、及び、フロー格納テーブル 113 を含む。フロー格納テーブル 113 は、実施例 3 の処理によって生成されたフローチャートを保持するためのテーブルである。

[0110] テスト履歴テーブル 211 は、実施例 3 における実行パスを保持するためのテーブルである。テスト履歴テーブル 211 は、実施例 1 及び実施例 2 の実行パス格納テーブル 112 に相当する。

[0111] なお、フロー生成サーバ 10 が、実施例 1 又は実施例 2 と、実施例 3 との処理を行うことができ、この場合、フロー生成サーバ 10 は、図 9 に示すプログラム及びテーブルの他に、図 1 に示すプログラム及びテーブルを有する。また、図 9 に示す計算機システムは、実行サーバ 20 を有してもよい。

[0112] 図 10 は、本実施例 3 の計算機システムによる処理の概要を示す説明図である。

[0113] 以下に示す処理によって、実施例 3 の計算機システムは、業務 A（実施例 1 及び実施例 2 における旅費申請システム 171 に相当）に関するフローチャートを生成する。

[0114] 実施例 3 におけるユーザは、まず、テスト対象である業務 A のフローチャートを作成する（71）。ここで作成されるフローチャートは、クライアント 160 に入力される。そして、フローチャートに含まれる各記号には、クライアント 160 が有するプログラムによって、又は、ユーザによって、ノード名とノード ID とを含むノードが割当てられる。

[0115] そして、ユーザは、クライアント 160 及びネットワーク 150 等を介して、自らが作成したフローチャートをフロー生成サーバ 10 に送信する。フロー生成サーバ 10 は、ユーザによって作成されたフローチャートを受信する（72）。

[0116] また、ユーザは、ステップ 71 において作成されたフローチャートに基づいて、クライアント 160 を用いて、業務 A をテストする（73）。例えば、ステップ 73 において、ユーザは、作成されたフローチャートをクライア

ント 160 の出力装置 130 に表示させ、あらかじめユーザが保持していたテスト項目に従って、テストする必要がある処理の流れを示すフローチャートの各記号を選択する。クライアント 160 は、ユーザが選択した各記号が示すノードのノード名とノード ID とを、フロー生成サーバ 10 に送信する。

[0117] ステップ 73 の後、又は、ステップ 73 の実行中において、テスト履歴記録部 201 は、クライアント 160 から送信されたノード名とノード ID とに基づいて、実行パスを生成する。そして、テスト履歴記録部 201 は、生成された実行パスをテスト履歴テーブル 211 に格納する（74）。

[0118] 図 10 に示すステップ 74 において、テスト履歴記録部 201 は、 n 個（ n は、任意の正数）の実行パスを生成する。

[0119] 図 10 に示すテスト履歴テーブル 211 には、 $T_1 \sim T_n$ の実行パスが、テストの実行結果として格納される。 $T_1 \sim T_n$ の実行パスは、実施例 1 の実行パスと同じく、ノード ID 及びノード名が含まれる複数の配列を含む。

[0120] ステップ 74 の後、フロー展開部 202 は、ステップ 72 において受信したフローチャートを解析し、解析の結果、フローチャートが表現する処理の流れの数を算出する（75）。図 10 に示すステップ 75 において、フロー展開部 202 は、処理の流れの数 m を算出する。

[0121] ステップ 75 の後、フロー比較部 203 は、ステップ 74 において生成された実行パスの数 n と、ステップ 75 において算出された処理の流れの数 m とが同じ数か否かを判定する（76）。処理の流れの数 m と実行パスの数 n とが同じ数である場合、フロー再生成部 204 がステップ 72 において受信したフローチャートを変更する必要はない。このため、フロー比較部 203 は、図 10 に示す処理を終了する（78）。

[0122] 処理の流れの数 m よりも実行パスの数 n が少ない場合、ステップ 72 において受信したフローチャートには無駄な処理の流れが含まれる。このため、フロー再生成部 204 は、業務 A に関するフローチャートを、実施例 1 又は実施例 2 に示すフローチャートを生成する処理によって再度生成し直す（7

7)。

[0123] なお、ユーザは、自らが作成したフローチャートに基づいてテストを実行するため、処理の流れの数 m を実行パスの数 n が上回ることはない。

[0124] ステップ 77 において実行されるフローチャートを再生成する処理は、実施例 1 の図 5 に示す処理、又は、実施例 2 の図 7 に示す処理と同じである。フロー再生成部 204 は、図 5 又は図 7 に示す処理によって、実行パス格納テーブル 112 の代わりにテスト履歴テーブル 211 に格納された実行パスを用いて、フローチャートを再生成する。

[0125] ステップ 77 の後、実施例 3 のフロー表示部 105 は、再生成されたフローチャートをクライアント 160 の画面表示部 162 に送信し、画面表示部 162 は、再生成されたフローチャートを出力装置 130 に表示してもよい。また、実施例 3 に示す計算機システムが、実施例 1 又は実施例 2 の計算機システムと同じく実行サーバ 20 を有する場合、再生成されたフローチャートと、実行サーバ 20 において実行される業務 A との画面を同時に表示してもよい。

[0126] なお、前述の実施例 3 において、ステップ 75 において処理の流れを算出するフローチャートは、ユーザによって作成されたフローチャートであったが、実施例 1 又は実施例 2 におけるフローチャートを生成する処理によって、生成されたフローチャートでもよい。例えば、実施例 1 又は実施例 2 における処理によってフローチャートを生成してから時間が経過し、ユーザによる作業手順とフローチャートとに相違が発生した場合に、実施例 3 の計算機システムは、フローチャートを再生成してもよい。

[0127] この場合、フロー展開部 202 は、ステップ 75 においてフロー格納テーブル 113 に格納されていたフローチャートを取得し、取得されたフローチャートの処理の流れを算出してもよい。

[0128] 図 11 は、本実施例 3 のユーザによって作成されたフローチャート 80 を示す説明図である。

[0129] 図 11 に示すフローチャート 80 は、ステップ 72 においてフロー生成サ

ーバ10が受信する、ユーザによって作成されたフローチャートの具体例である。フロー展開部202は、フローチャート80に基づいて、処理の流れの数 m を算出する。

[0130] フローチャート80には、共通ノード81、分岐ノード82、非共通ノード83、共通ノード85、分岐ノード86、非共通ノード87、及び、終端ノード89が含まれる。

[0131] 本実施例における終端ノードとは、自ノードの処理が終了した後、実行される処理が無いノードである。

[0132] また、フローチャート80は、経路A～Fを含む。経路Aは、処理が分岐ノード82の一つ目の接続先（図11に示す左側の接続先）を進む経路である。経路Bは、処理が経路Aを進み、かつ、分岐ノード86の一つ目の接続先（図11に示す左側の接続先）を進む経路である。経路Cは、処理が経路Aを進み、かつ、分岐ノード86の二つ目の接続先（図11に示す右側の接続先）を進む経路である。

[0133] 経路Dは、処理が分岐ノード82の二つ目の接続先（図11に示す右側の接続先）を進み、かつ、分岐ノード86の一つ目の接続先を経由する経路である。経路Eは、処理が経路Dを進み、かつ、分岐ノード86の一つ目の接続先を進む経路である。経路Fは、処理が経路Dを進み、かつ、分岐ノード86の二つ目の接続先を進む経路である。

[0134] このため、フローチャート80に含まれる処理の流れは、経路B、C、E、Fによって表現される。

[0135] 図12は、本実施例3のフローチャート80に含まれる処理の流れを示す説明図である。

[0136] 図12に示す複数の経路B、C、E、Fは、図11に示すフローチャート80が示すすべての処理の流れを示す。

[0137] 図13は、本実施例3の分岐ノードの接続先に関する分岐情報90を示す説明図である。

[0138] 分岐情報90は、フローチャートに含まれる分岐ノードごとに、主記憶装

置 1 0 0 に保持される配列である。フロー展開部 2 0 2 は、主記憶装置 1 0 0 に、フローチャート 8 0 に含まれる分岐ノードの数と同じ数の分岐情報 9 0 を保持する。

[0139] 分岐情報 9 0 は、分岐 I D 9 1、及び、接続配列を含む。分岐 I D 9 1 は、フローチャート 8 0 に含まれる各分岐ノードに割り当てられる識別子を含む。

[0140] 接続配列は、分岐 I D 9 1 が示す分岐ノードから処理が進む経路を示す。図 1 3 に示す接続配列は、要素として、経路 1 (9 2)、経路 2 (9 3)、及び、N U L L 9 4 を含む。

[0141] 経路 1 (9 2) は、分岐ノードの一つ目の接続先を示す要素である。経路 2 (9 3) は、分岐ノードの二つ目の接続先を示す要素である。

[0142] 図 1 3 に示す分岐情報 9 0 は、分岐ノードが二つの接続先と接続される場合の分岐情報 9 0 を示す。分岐ノードが三つ以上の接続先と接続される場合、接続配列には、接続される接続先の数と同じ数の要素が少なくとも含まれる。

[0143] N U L L 9 4 は、N U L L 値を含み、接続配列の最後に格納される。フロー展開部 2 0 2 は、分岐情報 9 0 を用いることによって、フローチャートに含まれる処理の数を算出する。

[0144] 図 1 4 は、本実施例 3 のフローチャート 8 0 の処理の流れの数を算出する処理を示すフローチャートである。

[0145] 図 1 4 に示す処理は、図 1 0 に示すステップ 7 5 に対応する。

[0146] 図 1 4 に示す処理は、フローチャート 8 0 の先頭のノード（例えば、図 1 1 に示す「旅費申請システムにログイン」のノード）から終端ノード 8 9 までの経路の数を算出する処理である。すなわち、図 1 2 に示す経路 B、C、E、F の数を算出する処理である。フロー展開部 2 0 2 は、経路の数を算出することによって、フローチャート 8 0 が表現する処理の流れの数 m を算出する。

[0147] このため、フロー展開部 2 0 2 は、図 1 4 に示す処理において、フローチ

ャート 80 の先頭のノードから終端ノード 89 まで、すべての分岐ノードにおけるすべての分岐のパターンをたどり、先頭のノードから終端のノードまでの経路の数を算出する。

[0148] フロー展開部 202 は、図 10 に示すステップ 74 の後、ステップ 75 において図 14 に示す処理を開始する。

[0149] 図 14 に示す処理において、フロー展開部 202 は、まず、パラメータ m に、初期値として 0 を格納する (301)。パラメータ m は、主記憶装置 100 に保持される。パラメータ m は、フローチャート 80 の先頭のノードから終端のノードまでの経路の数、すなわち、フローチャート 80 の処理の流れの数を示す。

[0150] ステップ 301 の後、フロー展開部 202 は、フローチャート 80 の先頭のノードを取得する (302)。ステップ 302 の後、フロー展開部 202 は、取得されたノードが共通ノード又は非共通ノードであるか否かを判定する (303)。

[0151] ステップ 303 において、取得されたノードが共通ノード又は非共通ノードであると判定された場合、フロー展開部 202 は、フローチャート 80 の次のノードを取得する (304)。ステップ 304 の後、フロー展開部 202 は、ステップ 303 に戻る。

[0152] ステップ 303 において、取得されたノードが共通ノード又は非共通ノードではないと判定された場合、フロー展開部 202 は、取得されたノードが分岐ノードか否かを判定する (305)。取得されたノードが分岐ノードである場合、フロー展開部 202 は、分岐ノードである取得されたノードに関する分岐情報 90 を、主記憶装置 100 に保持されたスタックに積む (306)。

[0153] 主記憶装置 100 に保持されるスタックには、フロー展開部 202 がフローチャート 80 をたどる際に経由した分岐ノードに関する分岐情報 90 が、経由した順に蓄積される。ステップ 306 においてスタックに積まれる分岐情報 90 の、`null` 94 以外のすべての要素には、空か、又は、未確認で

あることを示す値が含まれる。

- [0154] ステップ306の後、フロー展開部202は、フローチャート80の次のノードを取得する(307)。ステップ307の後、フロー展開部202は、ステップ303に戻る。
- [0155] ステップ305の判定の結果、取得されたノードが分岐ノードではない場合、すなわち、取得されたノードが終端ノード89である場合、フロー展開部202は、ステップ308を実行する。
- [0156] ステップ308～ステップ315の処理は、フロー展開部202が先頭ノードから終端ノード89までフローチャート80をたどった場合に実行される処理である。このため、ステップ308において、フロー展開部202は、フローチャート80の処理の流れの数を示すパラメータmに1を加算する(308)。
- [0157] ステップ308の後、フロー展開部202は、主記憶装置100に保持されたスタックを参照し、スタックが空か否かを判定する。
- [0158] ステップ308の後、スタックが空であるとステップ309において判定された場合、空のスタックは、フロー展開部202がたどってきたフローチャート80の経路には、分岐ノードが一つも含まれないことを示す。このため、フロー展開部202がたどってきたフローチャート80の経路以外、他のフローチャート80の経路は存在しないため、フロー展開部202は、ステップ315を実行し、経路の数の算出処理を終了する。
- [0159] また、ステップ312の後、スタックが空であるとステップ309において判定された場合、空のスタックは、フロー展開部202がすべての分岐ノードに接続されるすべての経路のパターンをたどり終わったことを示す。このため、フロー展開部202は、ステップ315を実行し、経路の数の算出処理を終了する。
- [0160] ステップ309において、スタックは空ではなく、分岐情報90を含むと判定された場合、スタックは、フロー展開部202がたどるべき他の分岐ノード又は他の経路がまだ残っていることを示す。このため、フロー展開部2

02は、スタックの最上部から分岐情報90を取り出す(310)。スタックの最上部の分岐情報90は、最も遅くに積まれた分岐情報90である。

[0161] ステップ310の後、フロー展開部202は、ステップ310において取り出された分岐情報90の接続配列の要素の中で、ステップ308までにフロー展開部202がたどってきた分岐ノードの接続先を示す要素に、“×”を格納する(311)。

[0162] なお、接続配列の要素に“×”が格納されている場合、要素が示す接続先を含む経路は、パラメータmに既に含まれることを示す。このため、接続配列の要素に格納される値は、“×”以外のいずれの値でもよい。

[0163] ステップ311の後、フロー展開部202は、ステップ310において取り出された分岐情報90の接続配列の要素の中で、“×”又は“null”が格納されていない要素を一つ特定できるか否かを判定する(312)。これによって、フロー展開部202は、次にたどるべき経路があるか否かを判定する。

[0164] 取り出された分岐情報90の接続配列の要素の中で、“×”又は“null”が格納されていない要素がないと判定された場合、フロー展開部202は、ステップ309に戻る。ここで、フロー展開部202は、取り出された分岐情報90をスタックに戻さないため、フロー展開部202は、ステップ309に戻った後、スタックに残る分岐情報90が示す分岐ノードの経路の数を数えることができる。

[0165] 取り出された分岐情報90の接続配列の要素の中で、“×”又は“null”が格納されていない要素を一つ特定できると判定された場合、フロー展開部202は、取り出された分岐情報90をスタックに再度積む(313)。これによって、フロー展開部202は、取り出された分岐情報90が示す分岐ノードにおいて、まだたどられていない経路を数えることができる。

[0166] ステップ313の後、フロー展開部202は、ステップ312において特定された要素(すなわち、ステップ313において再度スタックに積まれた分岐情報90の、“×”又は“null”が格納されていない要素)が示す

分岐ノードの接続先に接続されるノード（非共通ノード）を取得する（314）。ステップ314の後、フロー展開部202は、ステップ303に戻り、取得されたノードが共通ノードか非共通ノードかを判定する。

[0167] ステップ309において、スタックが空であると判定された場合、フロー展開部202は、パラメータmの値をフロー比較部203に出力する（314）。そして、フロー展開部202は、ステップ314の後、図14に示す処理を終了する。

[0168] なお、分岐情報90の接続配列の要素が“×”以外の値を格納する場合、フロー展開部202は、接続配列の要素に、“確認中”、及び“確認済み”を示す値を格納してもよい。この場合、フロー展開部202は、ステップ306及びステップ313において、次にたどる接続先を示す要素に“確認中”の値を格納する。そして、フロー展開部202は、ステップ311において、“確認中”が格納された要素を、“確認済み”の値によって更新する。また、フロー展開部202は、ステップ312において、“確認済み”又は“null”が格納されていない要素を一つ特定できるか否かを判定する。これによって、フロー展開部202は、分岐情報90に、たどってきた接続先の情報を格納することができる。

[0169] 図15は、本実施例3のフローチャート80の処理の数を算出する際のスタックの変化を示す説明図である。

[0170] フロー展開部202が、図14に示す処理によって、図11に示すフローチャート80の処理の数を算出する場合の、スタックの変化を図15に示す。図15に示す各スタックの状態は、図11に示す経路A～Fに対応する。

[0171] フロー展開部202が、ステップ301～ステップ304によって、共通ノード81をたどった後、フロー展開部202は、ステップ305及びステップ306によって、分岐ノード82の分岐情報90（分岐ID91が“001”）を、スタックに積む（A）。

[0172] 分岐ノード82の分岐情報90をスタックに積んだ後、フロー展開部202は、ステップ307、ステップ303及びステップ304によって、分岐

ノード８２の一つ目の接続先（図１１に示す“日帰り”側）を経由する経路Ａと、共通ノード８５とをたどる。そして、フロー展開部２０２は、ステップ３０５及びステップ３０６によって、分岐ノード８６の分岐情報９０（分岐ＩＤ９１が“００２”）を、スタックに積む（Ｂ）。

[0173] 分岐ノード８６の分岐情報９０をスタックに積んだ後、フロー展開部２０２は、ステップ３０７、ステップ３０３及びステップ３０４によって、分岐ノード８６の一つ目の接続先（図１１に示す“領収書なし”側）を経由する経路Ｂをたどる。そして、フロー展開部２０２は、ステップ３０５において、取得されたノードが終端ノード８９であると判定し、ステップ３０８を実行する（パラメータ $m=1$ ）。

[0174] ステップ３０８の後、スタックは空ではないため（ステップ３０９）、ステップ３１０において、フロー展開部２０２は、分岐ノード８６の分岐情報９０をスタックから取り出す。そして、フロー展開部２０２は、分岐ノード８６の分岐情報９０のうち、分岐ノード８６の一つ目の接続先を示す接続配列の要素に“×”を格納する。

[0175] ステップ３１１の後、分岐ノード８６の分岐情報９０には、“×”又は“null”が格納されていない要素があるため（ステップ３１２）、フロー展開部２０２は、ステップ３１３において、分岐ノード８６の分岐情報９０をスタックに再度積む（Ｃ）。

[0176] そして、フロー展開部２０２は、ステップ３１４、ステップ３０３、及び、ステップ３０４によって、分岐ノード８６の二つ目の接続先（図１１に示す“領収書あり”側）を経由する経路Ｃをたどる。そして、フロー展開部２０２は、ステップ３０５において、取得されたノードが終端ノード８９であると判定した場合、ステップ３０８を実行する（パラメータ $m=2$ ）。

[0177] ステップ３０８の後、スタックは空ではないため（ステップ３０９）、フロー展開部２０２は、ステップ３１０及びステップ３１１によって、取り出された分岐ノード８６の分岐情報９０のうち、分岐ノード８６の二つ目の接続先の接続配列の要素に“×”を格納する。ここで、分岐ノード８６の分岐

情報 90 の接続配列には、すべて “×” 又は “n u l l” が格納される（ステップ 312）。

[0178] このため、フロー展開部 202 は、ステップ 309 に戻り、ステップ 311 において、分岐ノード 82 の分岐情報 90 を、スタックから取り出す。これは、分岐ノード 86 の分岐情報 90 は既に取り出された後であるため、分岐ノード 82 の分岐情報 90 が最も上部に積まれているためである。

[0179] フロー展開部 202 は、ステップ 311 において、分岐ノード 82 の分岐情報 90 のうち、分岐ノード 82 の一つ目の接続先を示す要素に “×” を格納する。

[0180] ステップ 311 の後、分岐ノード 82 の分岐情報 90 には、“×” 又は “n u l l” が格納されていない要素があるため（ステップ 312）、フロー展開部 202 は、ステップ 313 において、分岐ノード 82 の分岐情報 90 をスタックに再度積む（D）。

[0181] そして、フロー展開部 202 は、ステップ 314、ステップ 303、及び、ステップ 304 によって、分岐ノード 82 の二つ目の接続先（図 11 に示す “宿泊” 側）を経由する経路 D と、共通ノード 85 とをたどる。そして、ステップ 305 及びステップ 306 によって、分岐ノード 86 の分岐情報 90 を、スタックに積む（E）。

[0182] 分岐ノード 86 の分岐情報 90 をスタックに積んだ後、フロー展開部 202 は、ステップ 307、ステップ 303 及びステップ 304 によって、分岐ノード 86 の一つ目の接続先を経由する経路 E をたどる。そして、フロー展開部 202 は、ステップ 305 において、取得されたノードが終端ノード 89 であると判定し、ステップ 308 を実行する（パラメータ $m=3$ ）。

[0183] フロー展開部 202 は、ステップ 310 において分岐ノード 86 の分岐情報 90 をスタックから取り出す。そして、ステップ 311 において、分岐ノード 86 の一つ目の接続先の接続配列の要素に “×” を格納する。そして、フロー展開部 202 は、ステップ 313 において、分岐ノード 86 の分岐情報 90 をスタックに再度積む（F）。

[0184] そして、フロー展開部202は、ステップ314、ステップ303、及び、ステップ304によって、分岐ノード86の二つ目の接続先を経由する経路Fをたどる。そして、取得されたノードが終端ノード89である場合（ステップ305）、フロー展開部202は、ステップ308を実行する（パラメータ $m=4$ ）。

[0185] フロー展開部202は、ステップ310において、分岐ノード86の分岐情報90を取り出し、ステップ311において、分岐ノード86の二つ目の接続先の接続配列の要素に“×”を格納する。ここで、分岐ノード86の分岐情報90の接続配列には、すべて“×”又は“null”が格納される（ステップ312）。

[0186] このため、フロー展開部202は、ステップ309に戻り、ステップ310において、分岐ノード82の分岐情報90を取り出す。そして、ステップ311において、分岐ノード82の二つ目の接続先の接続配列の要素に“×”を格納する。ここで、分岐ノード82の分岐情報90の接続配列には、すべて“×”又は“null”が格納される（ステップ312）。

[0187] このため、分岐ノード82の分岐情報90はスタックに戻されない（G）。そして、フロー展開部202は、ステップ309において、スタックが空であると判定する。フロー展開部202は、ステップ309において、フローチャートに含まれる処理の流れの数として、パラメータ m の値“4”をフロー比較部203に出力する。

[0188] 以上によって、図10に示すステップ75が終了する。

[0189] 実施例3によれば、フローチャートに含まれる処理の流れの数を算出することによって、フローチャートを再生成する必要があるか否かを判定することができ、フローチャートの無駄な再生成処理をする必要がない。また、実際に行われるテストに基づいて生成された実行パスによって、ユーザが真に必要な処理のみを含むフローチャートを再生成できる。

実施例 4

[0190] 実施例1及び実施例2において、実行パス生成部103は、ノード格納テ

ーブル 1 1 1 に格納されるノードを用いて実行パスを生成した。実施例 4 における実行パス生成部 413 は、ユーザによる作業に基づいて取得されたログ情報から実行パスを動的に生成する。

[0191] 図 1 6 は、本実施例 4 の計算機システムを示すブロック図である。

[0192] 実施例 4 の計算機システムは、実施例 1 の計算機システムと同じく、フロー生成サーバ 1 0、実行サーバ 2 0、及び、クライアント 1 6 0 を備える。実施例 4 のクライアント 1 6 0 は、実施例 1 のクライアント 1 6 0 と同じである。

[0193] 実施例 4 のフロー生成サーバ 1 0 は、主記憶装置 1 0 0 にフロー生成部 1 0 4、及びフロー表示部 1 0 5 を有し、また、補助記憶装置 1 1 0 に実行パス格納テーブル 1 1 2 及びフロー格納テーブル 1 1 3 を有する点において、実施例 1 のフロー生成サーバ 1 0 と同じである。しかし、実施例 4 のフロー生成サーバ 1 0 が、ノード生成部 1 0 1、ノード表示部 1 0 2、及び、ノード格納テーブル 1 1 1 を有さず、実行パス生成部 4 1 3 を有する点において、実施例 4 のフロー生成サーバ 1 0 と実施例 1 のフロー生成サーバ 1 0 とは異なる。

[0194] 実施例 4 の実行サーバ 2 0 は、主記憶装置 1 7 0 に旅費申請システム 1 7 1 を有する点において、実施例 1 の実行サーバ 2 0 と同じである。しかし、実施例 4 の実行サーバ 2 0 が、補助記憶装置 1 7 2 にログ情報格納テーブル 1 7 4 を有する点において、実施例 4 の実行サーバ 2 0 と実施例 1 の実行サーバ 2 0 とは異なる。

[0195] 図 1 7 は、本実施例 4 の実行パス生成部 4 1 3 による処理を示す説明図である。

[0196] ユーザは、クライアント 1 6 0 の出力装置 1 3 0 に表示された、旅費申請システム 1 7 1 に関する操作画面に必要な情報を入力することによって、旅費申請システム 1 7 1 に指示を送信する。図 1 7 に示す画面遷移 4 0 1 は、ユーザに表示された画面の遷移を示す。

[0197] 図 1 7 に示す画面遷移 4 0 1 は、画面 4 0 2、画面 4 0 3、及び、画面 4

04を含み、画面402、画面403、画面404の順にユーザに表示されたことを示す。

[0198] 旅費申請システム171は、ユーザが旅費申請業務を開始してから終了するまで、入力装置120を介してユーザからの指示を受信する。そして、旅費申請システム171は、受信したユーザからの指示に従って、ユーザによっていずれの画面にどのような入力されたかを示すログ情報を、ログ情報格納テーブル174に記録する。

[0199] 図17に示すログ情報格納テーブル174は、ログ405、ログ406、及び、ログ407を含む。また、ログ情報格納テーブル174は、旅費申請業務を行ったユーザを示すユーザIDを含む。

[0200] ログ405は、画面402に対応するログ情報であり、ログ406は、画面403に対応するログ情報であり、ログ407は、画面404に対応するログ情報である。

[0201] 実施例4の実行パス生成部413は、ログ情報格納テーブル174に格納されたログ情報からノードを生成し、生成されたノードを含む実行パスを生成する。そして、実行パス生成部413は、生成された実行パスを実行パス格納テーブル112に格納する。

[0202] 例えば、実施例4の実行パス生成部413は、ログ情報格納テーブル174に含まれる、先頭に“画面情報XX（Xは、任意の正数）”を含むログを抽出する。そして、実行パス生成部413は、抽出されたログから、“：（コロン）”の後に続く文字列をさらに抽出し、抽出された文字列をノード名としてノードを生成する。また、実行パス生成部413は、生成されたノードにノードIDを割り当てる。

[0203] 図17に示す実行パスには、ノード408、ノード409、及びノード410が含まれる。ノード408は、ログ405に基づいて生成されたノードである。ノード409は、ログ406に基づいて生成されたノードである。ノード410は、ログ407に基づいて生成されたノードである。

[0204] ログ情報テーブル174には、ユーザが旅費申請システム171による旅

費申請業務を開始してから終了するまでの処理の流れが格納される。このため、実行パス生成部413は、ユーザが旅費申請業務を開始してから終了するまでの実行パスを生成でき、ユーザによる作業のすべてを実行パスとして取得することができる。

[0205] また、ログ情報テーブル174には、ユーザIDが格納されるため、実行パス生成部413は、ユーザごとの実行パスを生成できる。そして、フロー生成部104は、実行パス生成部413によって生成された実行パスに基づいてフローチャートを生成するため、各ユーザのためのフローチャートを生成することができる。

[0206] さらに、実施例4によれば、旅費申請システム171が提供する旅費申請業務に関してユーザが作業した場合、作業における操作結果に従って、実行パス生成部413が実行パスを生成するため、フロー生成サーバ10があらかじめノードを保持する必要がない。このため、フロー生成サーバ10が保持するデータ量を低減できる。

[0207] また、ユーザが作業している間も、実施例4の実行パス生成部413は実行パスを動的に生成するため、実行パスを速やかに生成できる。

[0208] なお、本発明は前述した実施例に限定されるものではなく、様々な変形例が含まれる。前述した実施例は本発明をわかりやすく説明するために詳細に説明したものであり、必ずしも全ての構成を備えるものに限定されるものではない。

[0209] また、実施例1～実施例4におけるフロー生成サーバ10は、プログラムによって実装される機能を、物理的な集積回路等の装置によって実装してもよい。また、本実施例において主記憶装置100が有する複数のプログラムは、一つのプログラムによって実装されてもよい。また、一つのプログラムは、複数のサブプログラムによって実装されてもよい。

[0210] また、本実施例の各機能を実現するためのプログラム及びテーブル等は、ハードディスク、又は、SSD (Solid State Drive) 等の記憶装置に格納されることが可能であり、さらに、ICカード、SDカー

ド、又は、DVD等の記録媒体に格納されることが可能である。

[0211] また、本実施例における補助記憶装置は、テーブルのフォーマットによってデータを保持する必要はなく、データを保持できればいかなる方法によってデータを保持してもよい。例えば、実行パス格納テーブル112は、実行パス格納記憶部112であってもよい。

[0212] また、本実施例の計算機システムは、フロー生成サーバ10、実行サーバ20、及び、クライアント160を有したが、例えば、フロー生成サーバ10、実行サーバ20、及び、クライアント160の機能を一つの計算機によって実装してもよい。

産業上の利用可能性

[0213] 業務システムにおけるユーザインタフェース等、ユーザが作業手順を把握する必要があるシステムに適用可能である。

請求の範囲

[請求項1]

プロセッサ、及び、メモリを有する計算機システムであって、
前記計算機システムは、

実行された複数の処理を示す識別子を含み、かつ、前記各処理が実行された順番を示す、複数の実行パスと、

前記各実行パスに基づいて、フローチャートを生成するフロー生成部と、を有し、

前記フロー生成部は、

最初に実行された前記処理から同じ数の前記処理が実行された前記処理の識別子を、前記複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第1の判定を実行し、

前記第1の判定の結果、前記抽出された各処理の識別子が同じである場合、前記同じである各処理の識別子を含む一つの共通処理記号を生成し、

前記第1の判定の結果、前記抽出された各処理の識別子のうち少なくとも一つが他の前記識別子と異なる場合、前記第1の判定において少なくとも一つが異なると判定された、前記抽出された処理の識別子を各々含む複数の非共通処理記号を生成し、

前記生成された共通処理記号、及び、前記生成された非共通処理記号の少なくとも一つを含む前記フローチャートを生成することを特徴とする計算機システム。

[請求項2]

請求項1に記載された計算機システムであって、

前記フロー生成部は、前記第1の判定の結果、前記抽出された各処理が異なる場合、

最後に実行される前記処理までに同じ数の前記処理が実行される前記各処理の識別子を、前記複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第2の判定を実行し、

前記第2の判定の結果、前記抽出された各処理の識別子が同じであ

る場合、前記同じである各処理の識別子を含む一つの共通処理記号を生成し、

前記第2の判定の結果、前記抽出された各処理の識別子のうち少なくとも一つが異なる場合、前記第2の判定において少なくとも一つが異なると判定された、前記抽出された処理の識別子を各々含む複数の非共通処理記号を生成することを特徴とする計算機システム。

[請求項3]

請求項1又は2に記載された計算機システムであって、

前記計算機システムは、

所定の前記フローチャートを入力される入力装置と、

前記ユーザによって実行された処理手順に基づいて、前記各実行パスを生成する記録部と、

前記入力装置に入力された前記所定のフローチャートが表現する処理の流れの数を算出するフロー展開部と、

前記フロー展開部によって算出された処理の流れの数と、前記記録部によって生成された前記各実行パスの数とを比較するフロー比較部と、を有し、

前記フロー生成部は、前記フロー比較部による比較の結果、前記算出された処理の流れの数よりも前記生成された各実行パスの数が少ない場合、前記記録部によって生成された各実行パスに基づいて、前記フローチャートを生成することを特徴とする計算機システム。

[請求項4]

請求項1又は2に記載された計算機システムであって、

前記計算機システムは、

ユーザに情報を表示する出力装置と、

前記フローチャートを、前記出力装置に表示するフロー表示部と、

前記フローチャートが示す処理を実行する業務システムと、を有し、

前記業務システムは、前記フローチャートが示す各処理の実行結果を、前記出力装置に表示し、

前記フロー表示部は、前記業務システムが前記ユーザに表示する各実行結果とともに、前記フローチャートを前記出力装置に表示することを特徴とする計算機システム。

[請求項5]

プロセッサ、及び、メモリを有する計算機システムによるデータ処理方法であって、

前記方法は、

前記プロセッサが、実行された複数の処理を示す識別子を含み、かつ、前記各処理が実行された順番を示す、複数の実行パスを、前記メモリに格納する格納手順と、

前記プロセッサが、前記各実行パスに基づいて、フローチャートを生成するフロー生成手順と、を含み、

前記フロー生成手順は、

前記プロセッサが、最初に実行された前記処理から同じ数の前記処理が実行された前記処理の識別子を、前記複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第1の判定を実行する手順と、

前記プロセッサが、前記第1の判定の結果、前記抽出された各処理の識別子が同じである場合、前記同じである各処理の識別子を含む一つの共通処理記号を生成する手順と、

前記プロセッサが、前記第1の判定の結果、前記抽出された各処理の識別子のうち少なくとも一つが他の前記識別子と異なる場合、前記第1の判定において少なくとも一つが異なると判定された、前記抽出された処理の識別子を各々含む複数の非共通処理記号を生成する手順と、

前記プロセッサが、前記生成された共通処理記号、及び、前記生成された非共通処理記号の少なくとも一つを含む前記フローチャートを生成する手順と、を含むことを特徴とするデータ処理方法。

[請求項6]

請求項5に記載されたデータ処理方法であって、

前記フロー生成手順は、前記第 1 の判定の結果、前記抽出された各処理が異なる場合、

前記プロセッサが、最後に実行される前記処理までに同じ数の前記処理が実行される前記各処理の識別子を、前記複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第 2 の判定を実行する手順と、

前記第 2 の判定の結果、前記抽出された各処理の識別子が同じである場合、前記同じである各処理の識別子を含む一つの共通処理記号を生成する手順と、

前記第 2 の判定の結果、前記抽出された各処理の識別子のうち少なくとも一つが異なる場合、前記第 2 の判定において少なくとも一つが異なると判定された、前記抽出された処理の識別子を各々含む複数の非共通処理記号を生成する手順と、を含むことを特徴とするデータ処理方法。

[請求項 7]

請求項 5 又は 6 に記載されたデータ処理方法であって、

前記計算機システムは、所定の前記フローチャートを入力される入力装置を、さらに有し、

前記方法は、

前記プロセッサが、前記ユーザによって実行された処理手順に基づいて、前記各実行パスを生成する記録手順と、

前記プロセッサが、前記入力装置に入力された前記所定のフローチャートが表現する処理の流れの数を算出するフロー展開手順と、

前記プロセッサが、前記フロー展開手順によって算出された処理の流れの数と、前記記録手順によって生成された前記各実行パスの数とを比較するフロー比較手順と、を有し、

前記フロー生成手順は、前記フロー比較手順による比較の結果、前記算出された処理の流れの数よりも前記生成された各実行パスの数が少ない場合、前記プロセッサが、前記記録手順によって生成された各

実行パスに基づいて、前記フローチャートを生成する手順を含むことを特徴とするデータ処理方法。

[請求項8]

請求項5又は6に記載されたデータ処理方法であって、

前記計算機システムは、ユーザに情報を表示する出力装置を、さらに有し、

前記方法は、

前記フローチャートを、前記出力装置に表示するフロー表示手順と

、

前記フローチャートが示す処理を実行する業務実行手順と、を有し

、

前記業務実行手順は、前記プロセッサが、前記フローチャートが示す各処理の実行結果を、前記出力装置に表示する手順を含み、

前記フロー表示手順は、前記プロセッサが、前記業務実行手順が前記ユーザに表示する各実行結果とともに、前記フローチャートを前記出力装置に表示する手順を含むことを特徴とするデータ処理方法。

[請求項9]

データ処理プログラムを格納する非一時的な記録媒体であって、

計算機システムは、プロセッサ、及び、前記データ処理プログラムを保持するメモリを、有し、

前記データ処理プログラムは、前記プロセッサに、

実行された複数の処理を示す識別子を含み、かつ、前記各処理が実行された順番を示す、複数の実行パスを、前記メモリに格納する格納手順と、

前記各実行パスに基づいて、フローチャートを生成するフロー生成手順と、を実行させ、

前記フロー生成手順は、

最初に実行された前記処理から同じ数の前記処理が実行された前記処理の識別子を、前記複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第1の判定を実行する手順と、

前記第 1 の判定の結果、前記抽出された各処理の識別子が同じである場合、前記同じである各処理の識別子を含む一つの共通処理記号を生成する手順と、

前記第 1 の判定の結果、前記抽出された各処理の識別子のうち少なくとも一つが他の前記識別子と異なる場合、前記第 1 の判定において少なくとも一つが異なると判定された、前記抽出された処理の識別子を各々含む複数の非共通処理記号を生成する手順と、

前記生成された共通処理記号、及び、前記生成された非共通処理記号の少なくとも一つを含む前記フローチャートを生成する手順と、を含むデータ処理プログラムを格納する非一時的な記録媒体。

[請求項10]

請求項 9 に記載された記録媒体であって、

前記データ処理プログラムの前記フロー生成手順は、前記第 1 の判定の結果、前記抽出された各処理が異なる場合、

最後に実行される前記処理までに同じ数の前記処理が実行される前記各処理の識別子を、前記複数の実行パスから抽出し、当該抽出された各処理の識別子が同じか否かを判定する第 2 の判定を実行する手順と、

前記第 2 の判定の結果、前記抽出された各処理の識別子が同じである場合、前記同じである各処理の識別子を含む一つの共通処理記号を生成する手順と、

前記第 2 の判定の結果、前記抽出された各処理の識別子のうち少なくとも一つが異なる場合、前記第 2 の判定において少なくとも一つが異なると判定された、前記抽出された処理の識別子を各々含む複数の非共通処理記号を生成する手順と、を含むデータ処理プログラムを格納する記録媒体。

[請求項11]

請求項 9 又は 10 に記載された記録媒体であって、

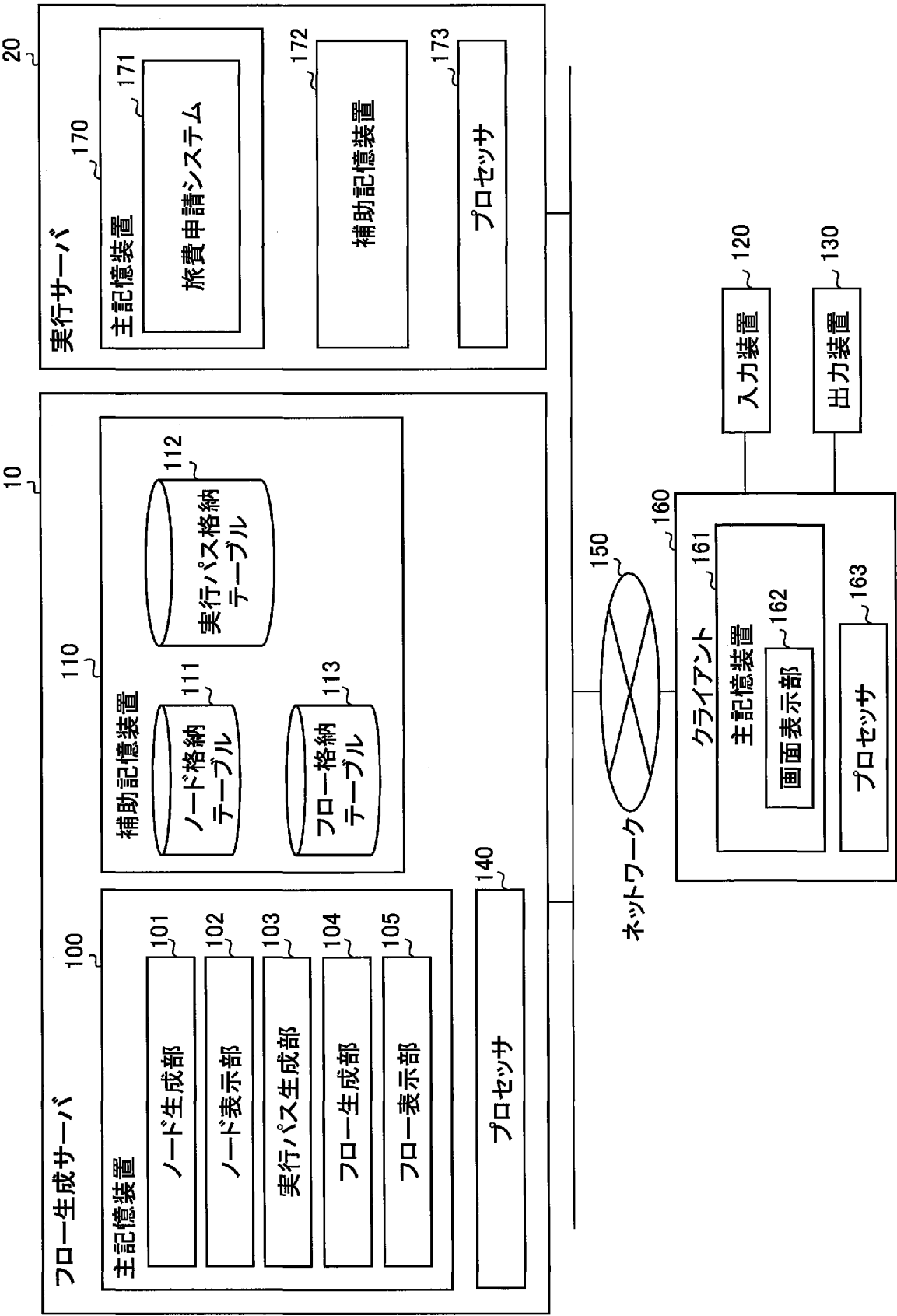
前記計算機システムは、所定の前記フローチャートを入力される入力装置を、さらに有し、

前記データ処理プログラムは、前記プロセッサに、
前記ユーザによって実行された処理手順に基づいて、前記各実行パスを生成する記録手順と、
前記入力装置に入力された前記所定のフローチャートが表現する処理の流れの数を算出するフロー展開手順と、
前記フロー展開手順によって算出された処理の流れの数と、前記記録手順によって生成された前記各実行パスの数とを比較するフロー比較手順と、を実行させ、
前記フロー生成手順は、前記フロー比較手順による比較の結果、前記算出された処理の流れの数よりも前記生成された各実行パスの数が少ない場合、前記記録手順によって生成された各実行パスに基づいて、前記フローチャートを生成する手順を含むデータ処理プログラムを格納する記録媒体。

[請求項12]

請求項9又は10に記載された記録媒体であって、
前記計算機システムは、ユーザに情報を表示する出力装置を、さらに有し、
前記データ処理プログラムは、前記プロセッサに、
前記フローチャートを、前記出力装置に表示するフロー表示手順と、
前記フローチャートが示す処理を実行する業務実行手順と、を実行させ、
前記業務実行手順は、前記フローチャートが示す各処理の実行結果を、前記出力装置に表示する手順を含み、
前記フロー表示手順は、前記業務実行手順が前記ユーザに表示する各実行結果とともに、前記フローチャートを前記出力装置に表示する手順を含むデータ処理プログラムを格納する記録媒体。

[図1]



[図2]

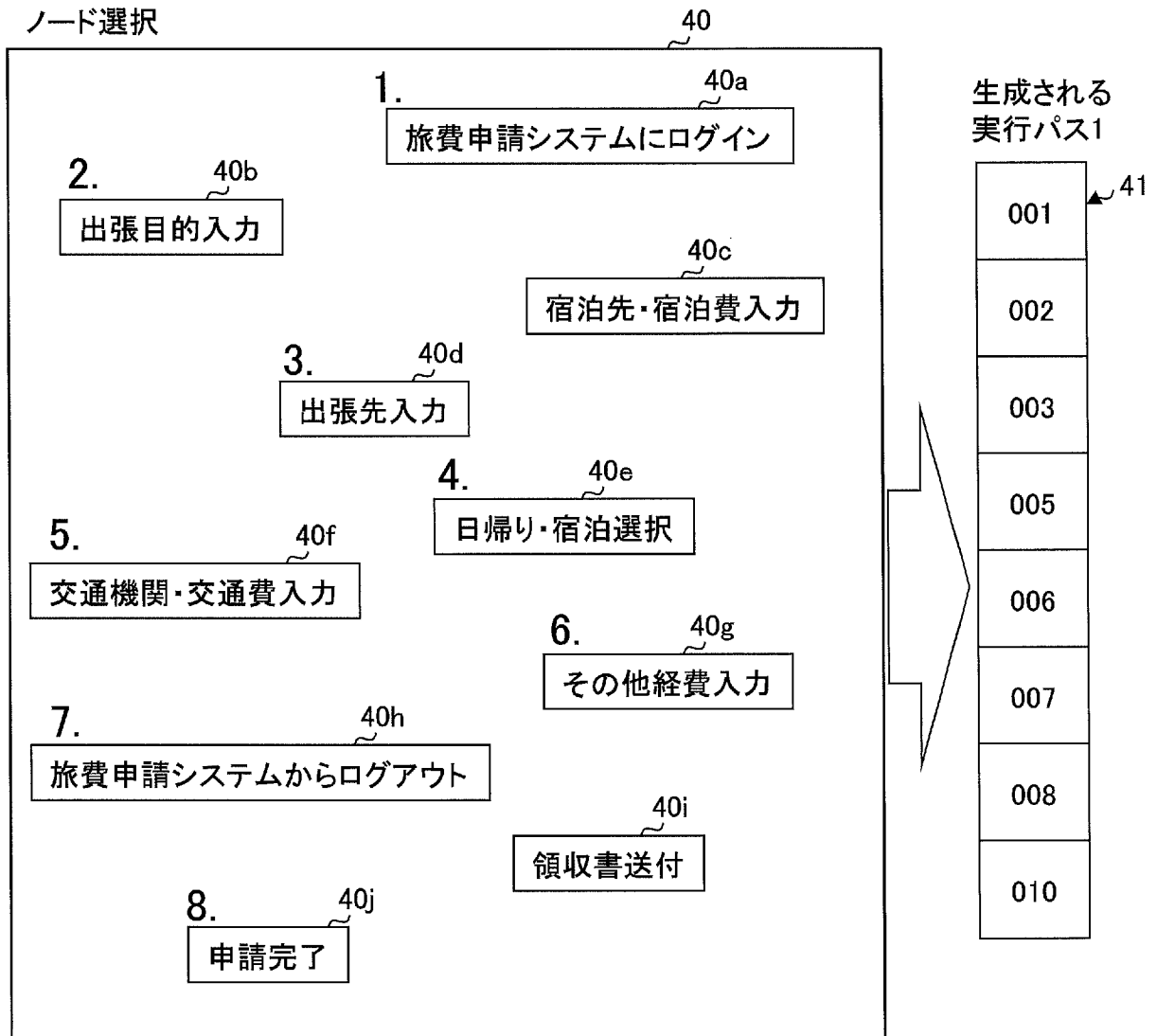
The diagram shows a table structure for a travel expense application business. A bracket labeled 1111 spans the first two columns, 'NodeName' and 'NodeID'. Another bracket labeled 1112 spans the 'NodeID' column. An arrow labeled 111 points to the top-right corner of the table.

旅費申請業務	
NodeName	NodeID
旅費申請システムにログイン	001
出張目的入力	002
出張先入力	003
宿泊先・宿泊費入力	004
日帰り・宿泊選択	005
交通機関・交通費入力	006
その他経費入力	007
旅費申請システムからログアウト	008
領収書送付	009
申請完了	010
...	...

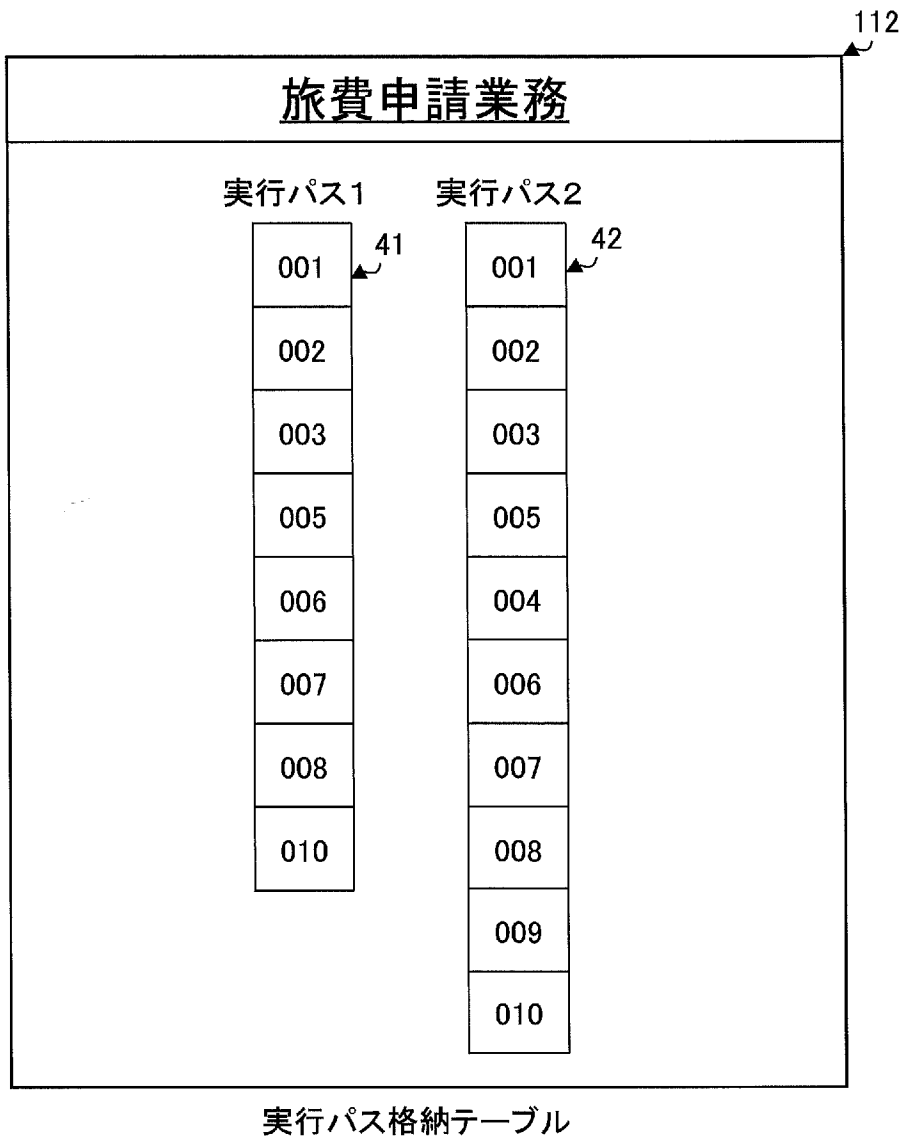
ノード格納テーブル

[図3]

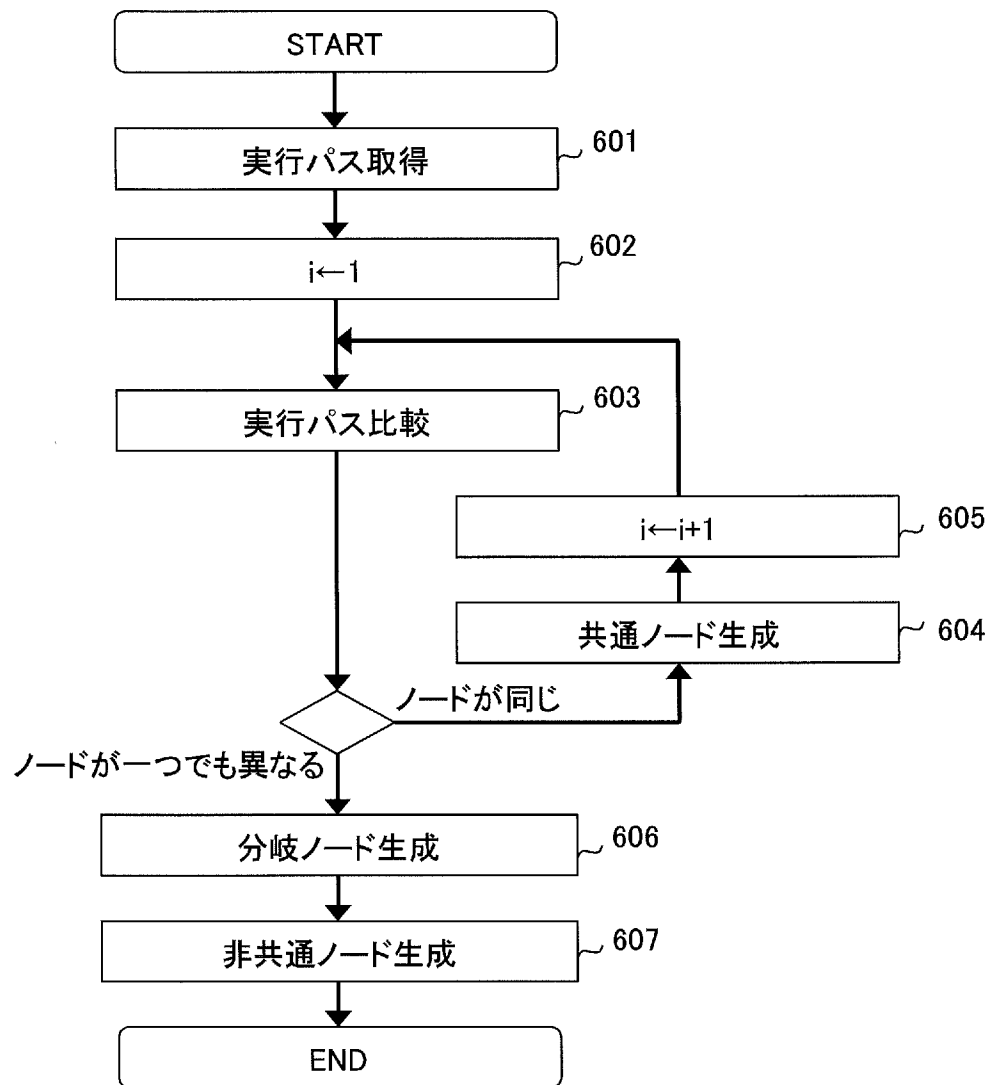
ノード選択



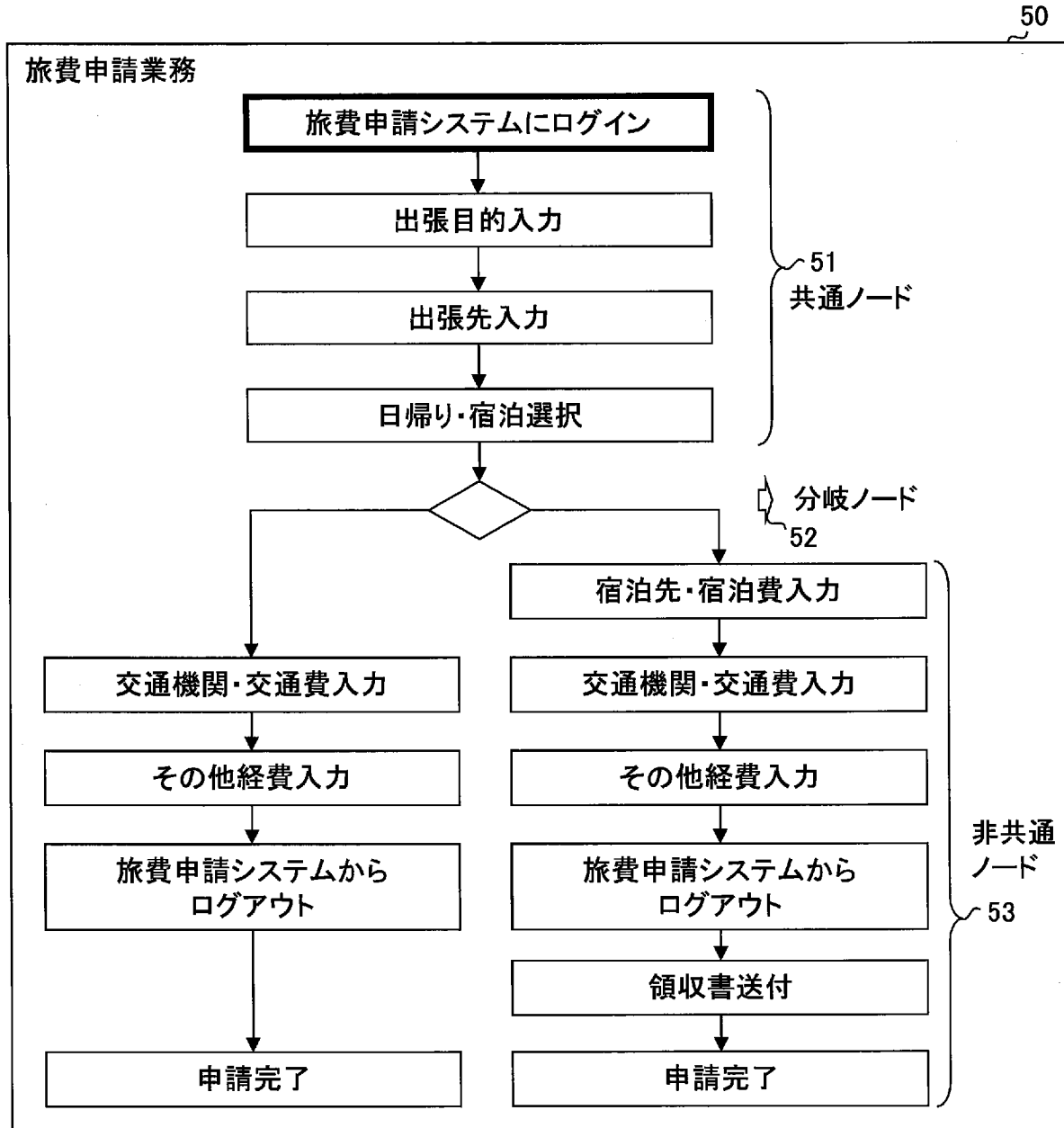
[図4]



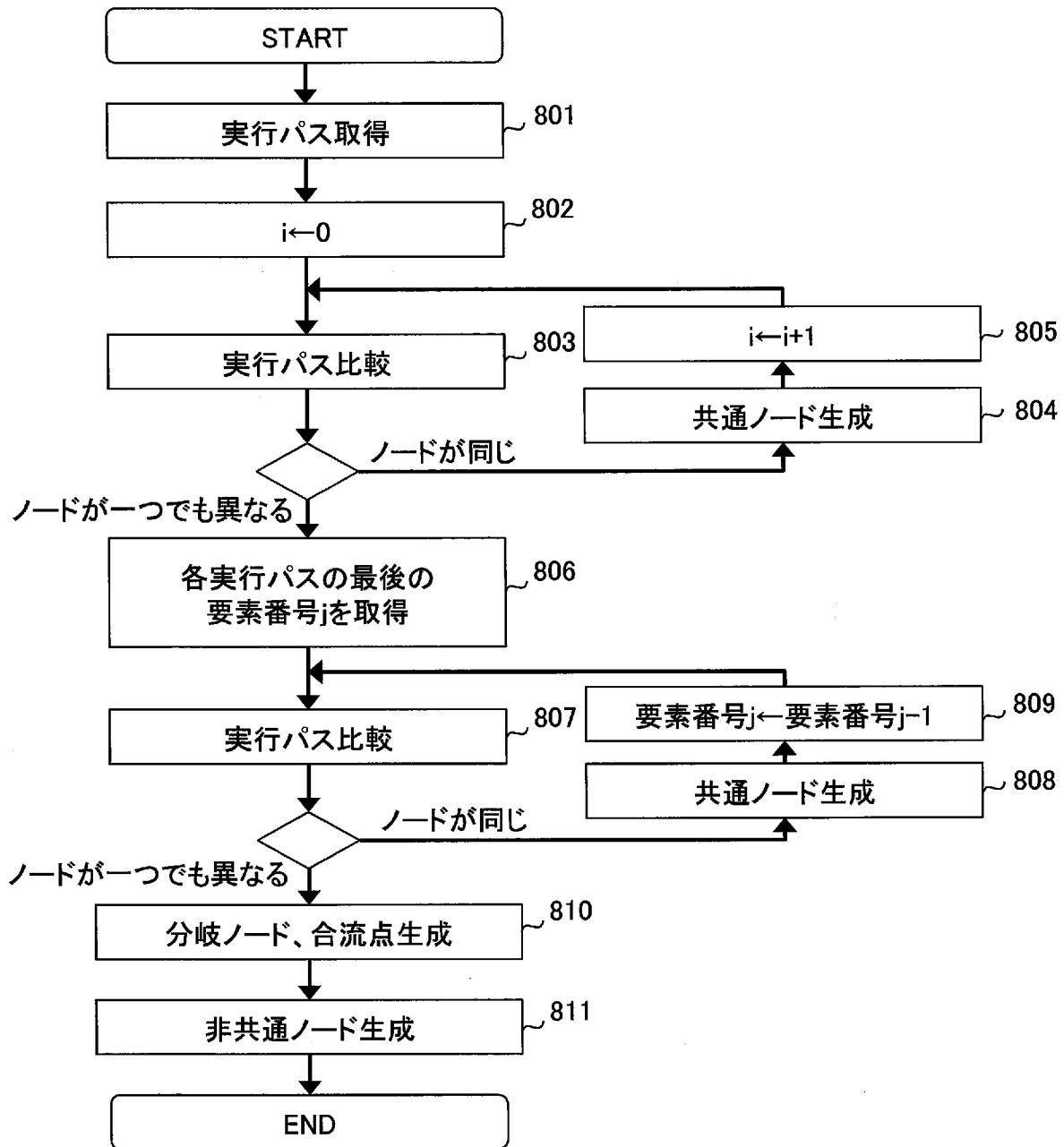
[図5]



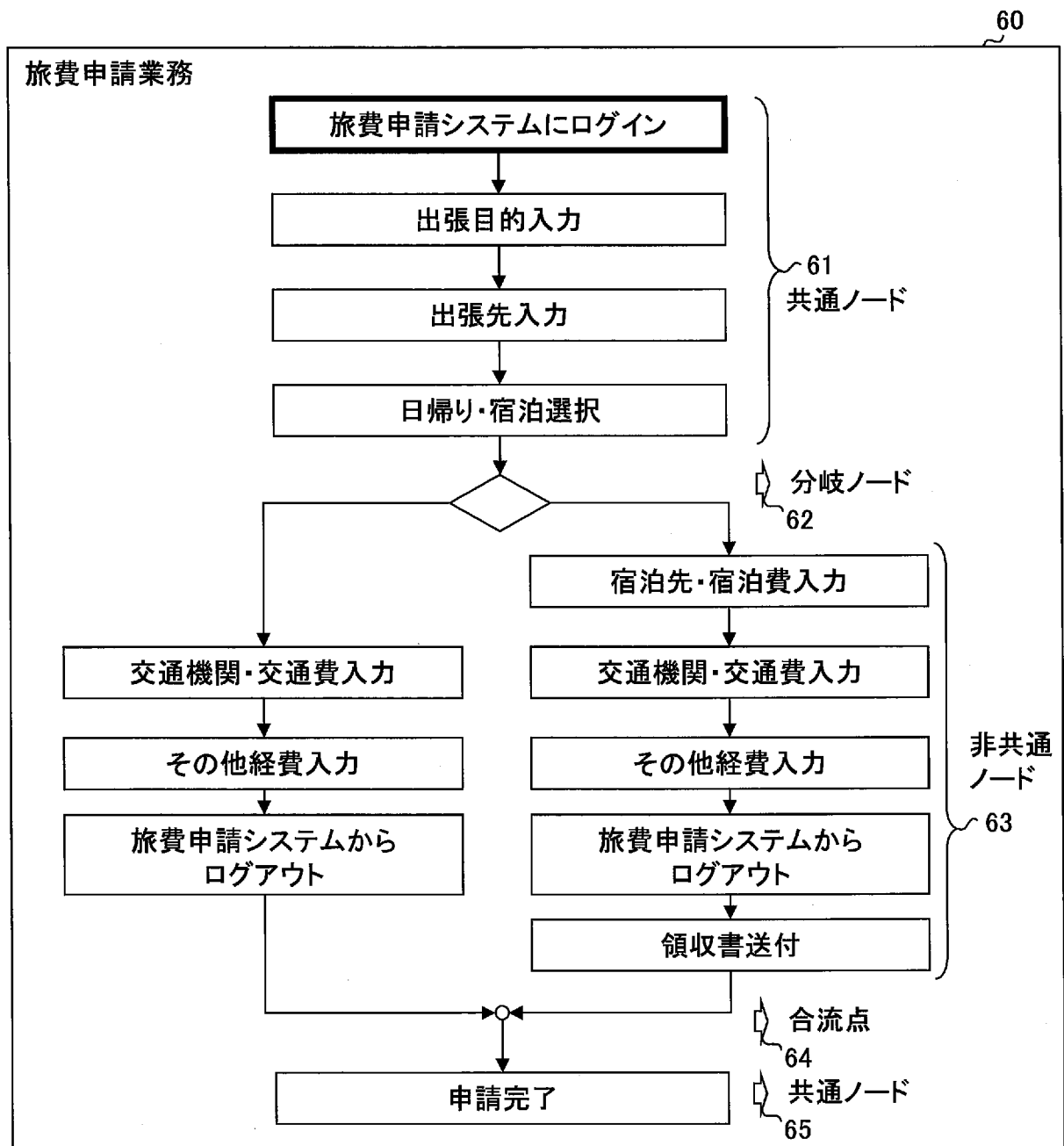
[図6]



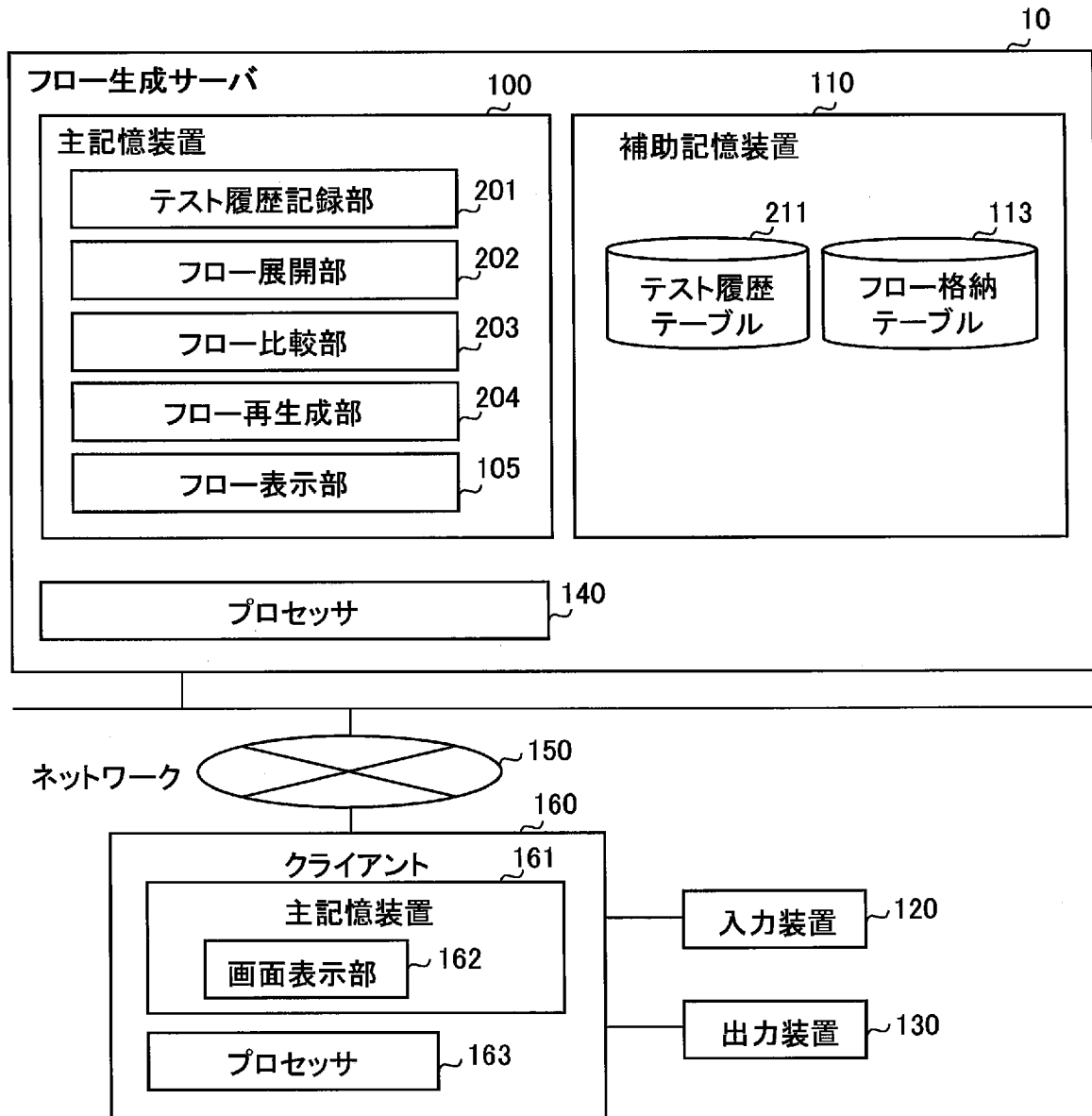
[図7]



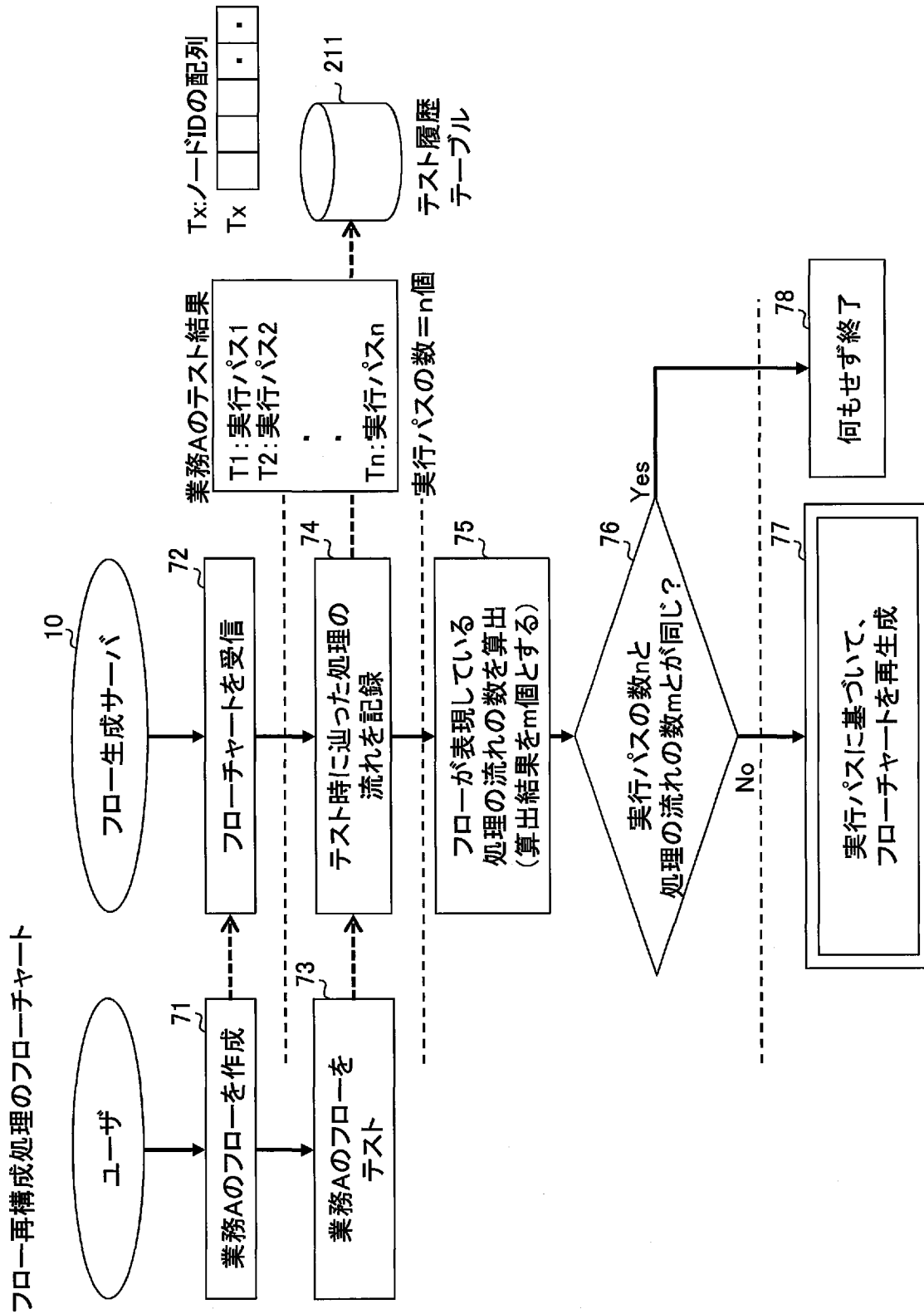
[図8]



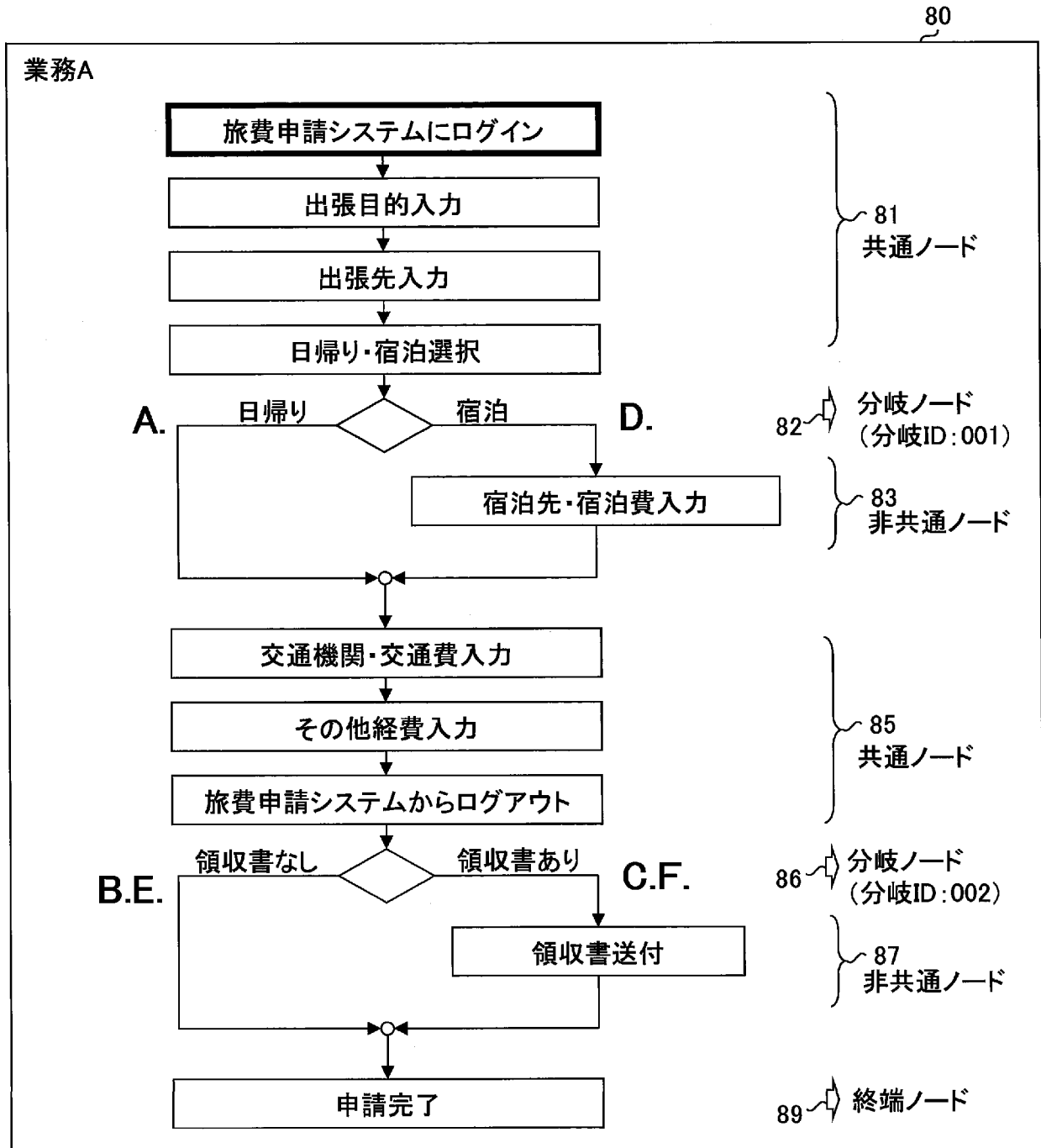
[図9]



[図10]



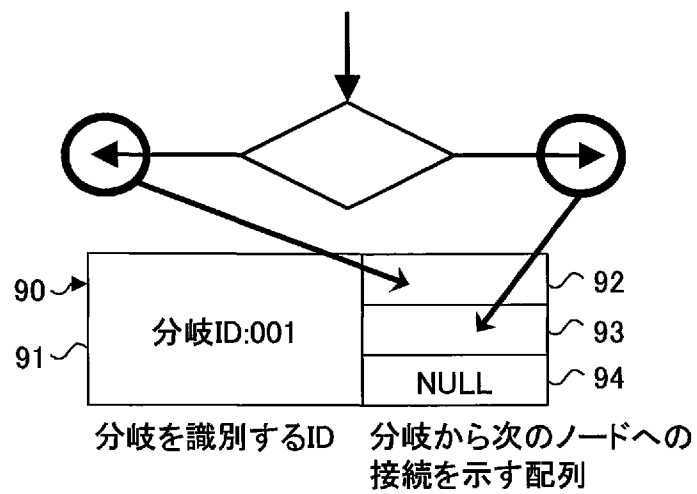
[図11]



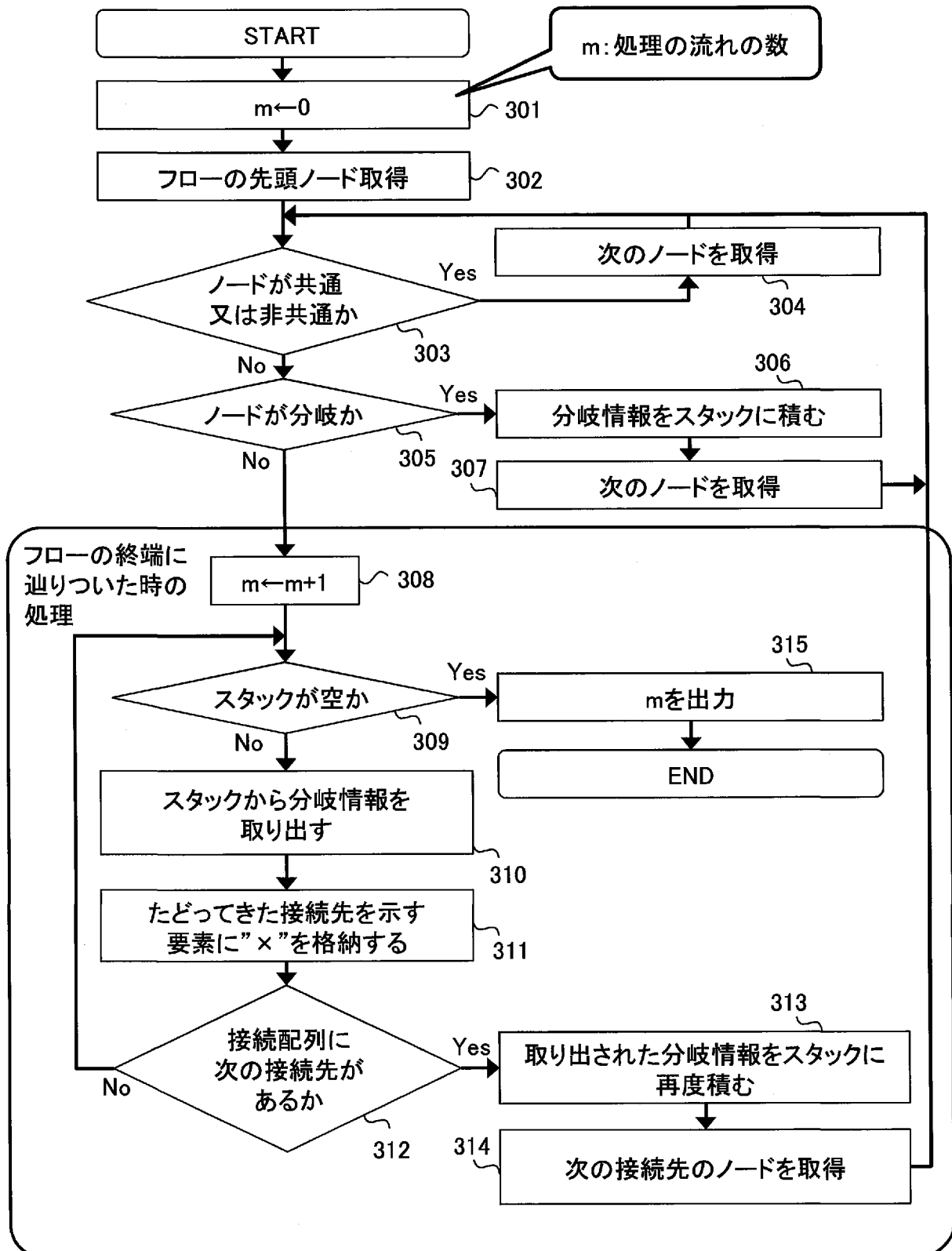
[図12]



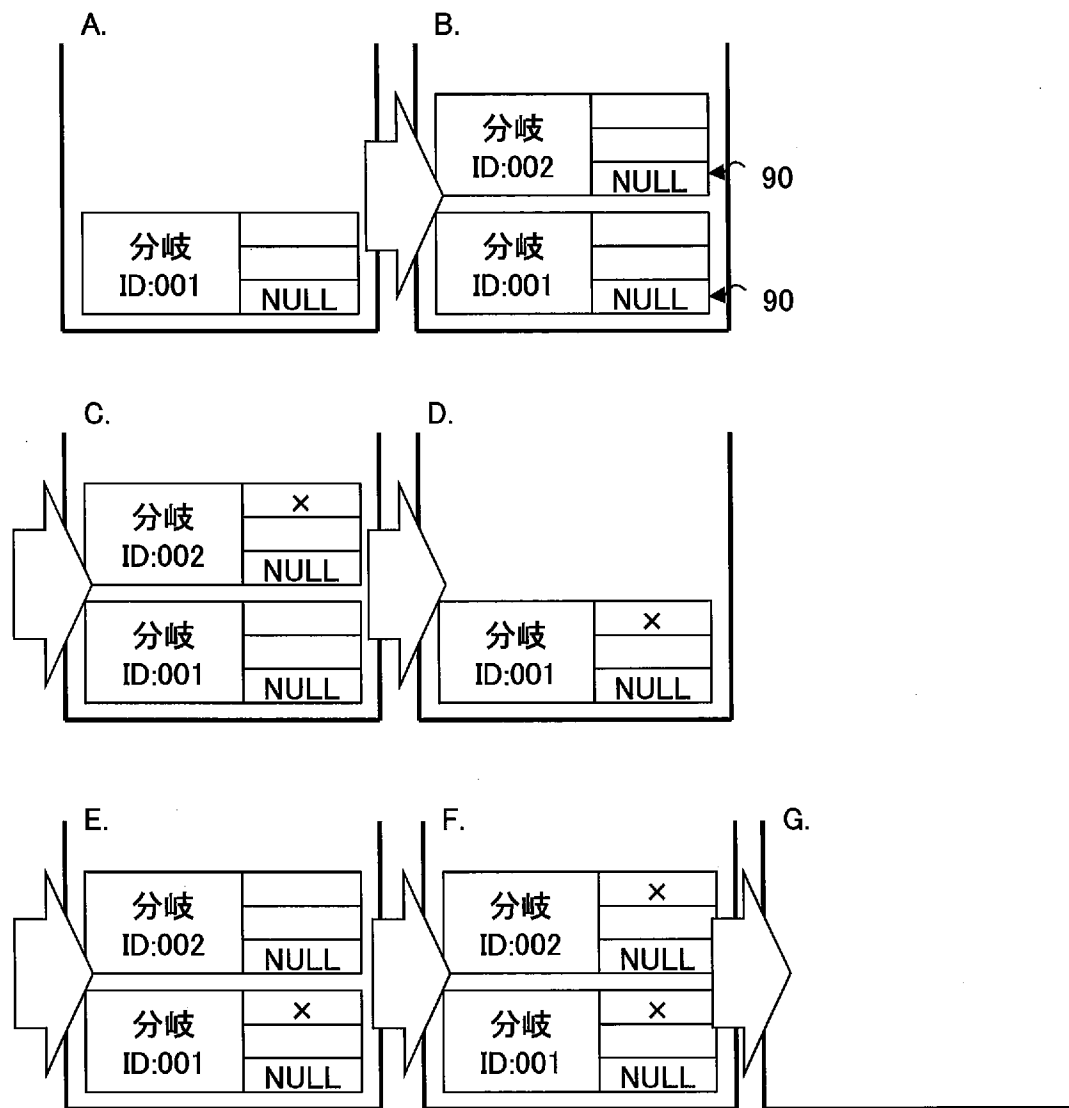
[図13]



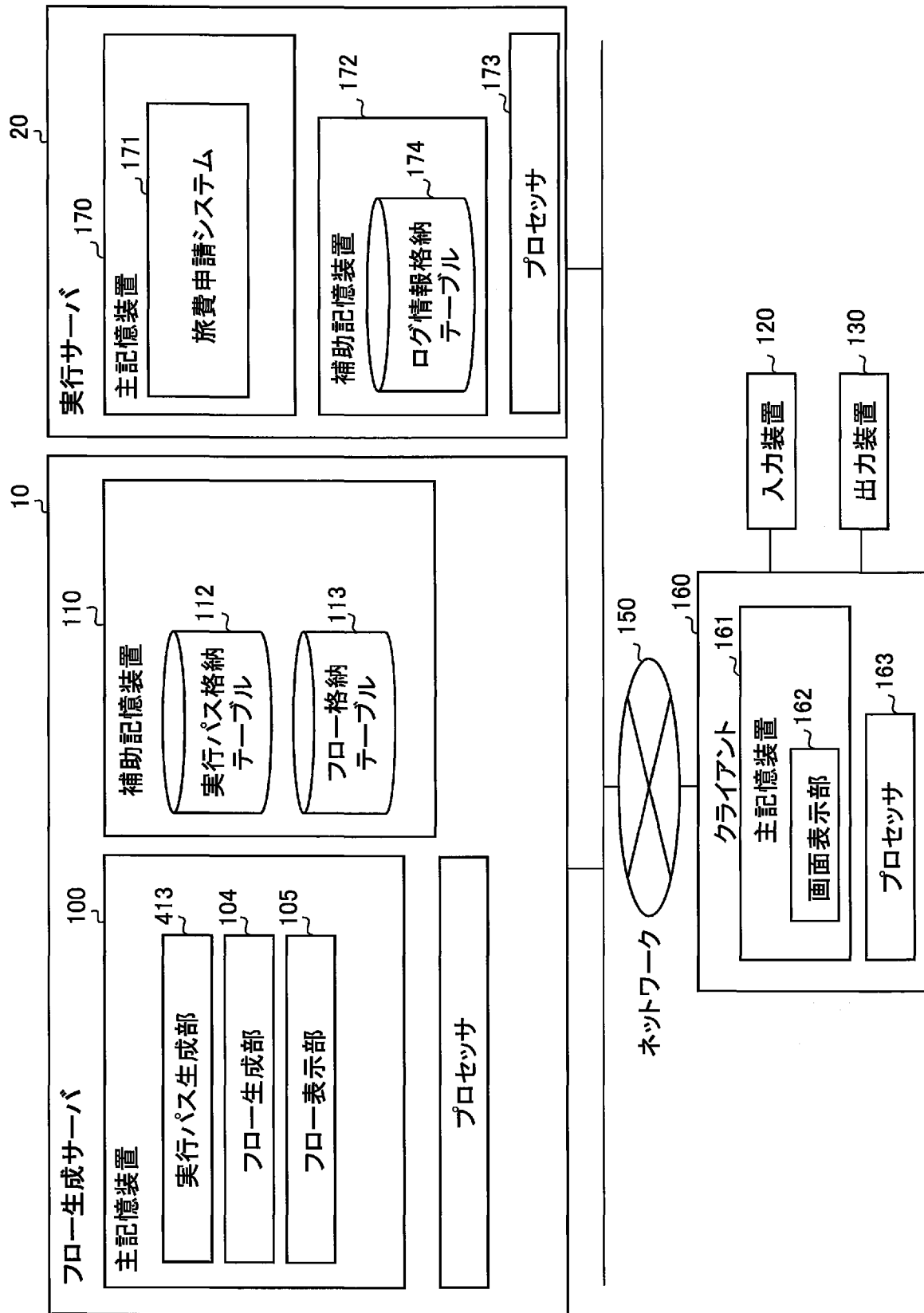
[図14]



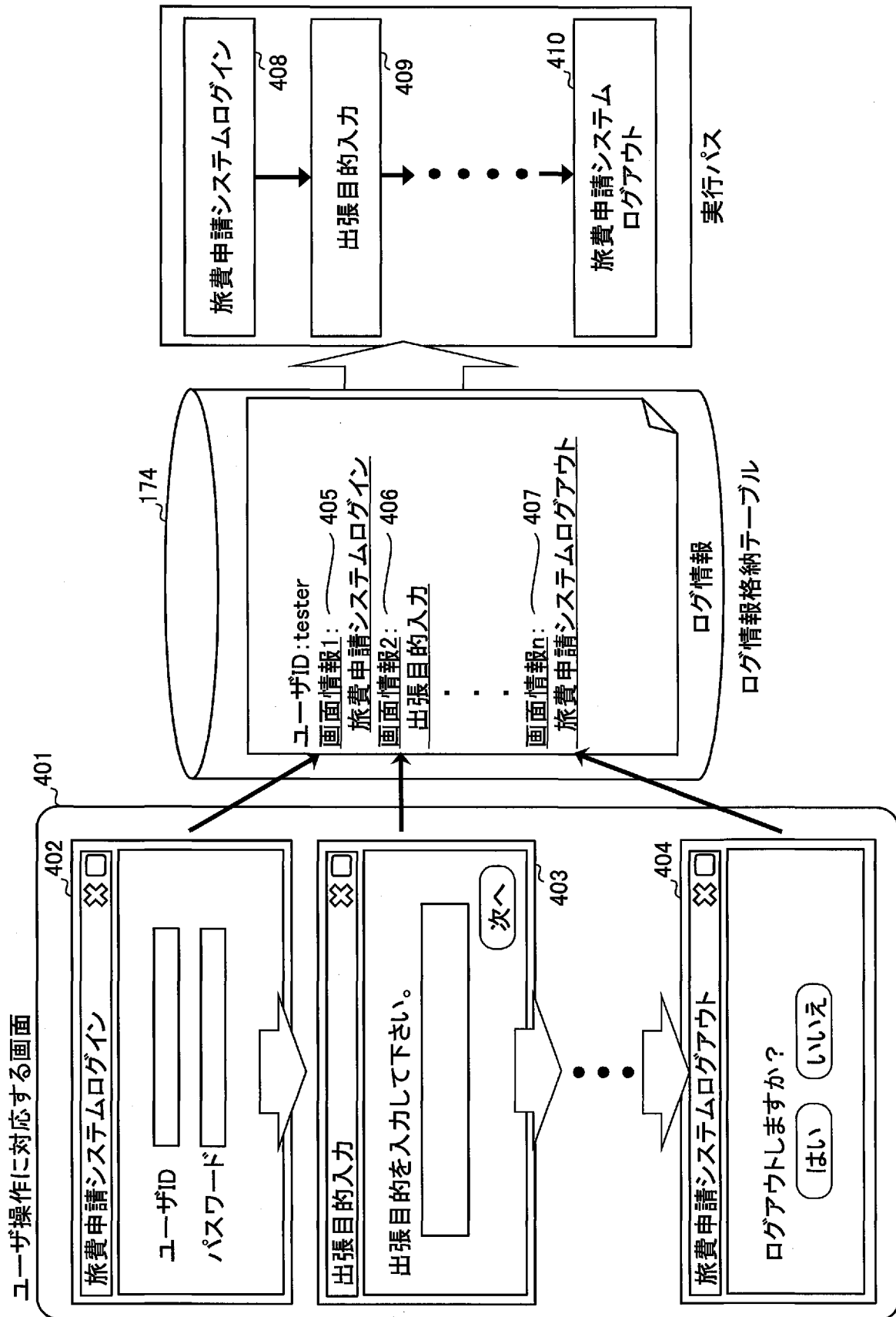
[図15]



[図16]



[図17]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2012/068565

A. CLASSIFICATION OF SUBJECT MATTER

G06Q10/06 (2012.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06Q10/06

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Jitsuyo Shinan Toroku Koho	1996-2012
Kokai Jitsuyo Shinan Koho	1971-2012	Toroku Jitsuyo Shinan Koho	1994-2012

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2008-293101 A (Hitachi INS Software, Ltd.), 04 December 2008 (04.12.2008), abstract; claim 2; paragraphs [0101], [0163] & US 2008/0294485 A1 & DE 102007046049 A1	1-12
A	JP 2008-310570 A (Hitachi, Ltd.), 25 December 2008 (25.12.2008), entire text; all drawings (Family: none)	1-12
A	JP 2010-224956 A (Toshiba Corp.), 07 October 2010 (07.10.2010), entire text; all drawings (Family: none)	1-12

☒ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search
27 September, 2012 (27.09.12)

Date of mailing of the international search report
09 October, 2012 (09.10.12)

Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2012/068565

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2004-206438 A (Mitsubishi Electric Corp.), 22 July 2004 (22.07.2004), entire text; all drawings (Family: none)	1-12
A	JP 2004-164449 A (Hitachi, Ltd.), 10 June 2004 (10.06.2004), entire text; all drawings (Family: none)	1-12
A	JP 2010-224955 A (Toshiba Corp.), 07 October 2010 (07.10.2010), entire text; all drawings (Family: none)	1-12
A	JP 2007-122651 A (Fujitsu Ltd.), 17 May 2007 (17.05.2007), entire text; all drawings & US 2007/0168928 A1	1-12
A	JP 2010-191709 A (Fuji Xerox Co., Ltd.), 02 September 2010 (02.09.2010), entire text; all drawings (Family: none)	1-12
A	JP 2006-302032 A (NTT Data Corp.), 02 November 2006 (02.11.2006), entire text; all drawings (Family: none)	1-12

A. 発明の属する分野の分類 (国際特許分類 (I P C))

Int.Cl. G06Q10/06 (2012.01) i

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (I P C))

Int.Cl. G06Q10/06

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1 9 2 2 - 1 9 9 6 年
日本国公開実用新案公報	1 9 7 1 - 2 0 1 2 年
日本国実用新案登録公報	1 9 9 6 - 2 0 1 2 年
日本国登録実用新案公報	1 9 9 4 - 2 0 1 2 年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
A	JP 2008-293101 A (日立アイ・エヌ・エス・ソフトウェア株式会社) 2008. 12. 04, 【要約】、【請求項 2】、段落【0 1 0 1】、【0 1 6 3】 & US 2008/0294485 A1 & DE 102007046049 A1	1 - 1 2
A	JP 2008-310570 A (株式会社日立製作所) 2008. 12. 25, 全文、全図 (ファミリーなし)	1 - 1 2
A	JP 2010-224956 A (株式会社東芝) 2010. 10. 07, 全文、全図 (ファミリーなし)	1 - 1 2
A	JP 2004-206438 A (三菱電機株式会社) 2004. 07. 22, 全文、全図 (ファミリーなし)	1 - 1 2

☒ C 欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの
「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの
「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)
「O」口頭による開示、使用、展示等に言及する文献
「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」特に関連のある文献であって、当該文献と他の 1 以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」同一パテントファミリー文献

国際調査を完了した日

2 7 . 0 9 . 2 0 1 2

国際調査報告の発送日

0 9 . 1 0 . 2 0 1 2

国際調査機関の名称及びあて先

日本国特許庁 (I S A / J P)

郵便番号 1 0 0 - 8 9 1 5

東京都千代田区霞が関三丁目 4 番 3 号

特許庁審査官 (権限のある職員)

山本 雅士

電話番号 0 3 - 3 5 8 1 - 1 1 0 1 内線 3 5 6 2

5 L

3 7 8 6

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
A	JP 2004-164449 A (株式会社日立製作所) 2004.06.10, 全文、全図 (ファミリーなし)	1 - 1 2
A	JP 2010-224955 A (株式会社東芝) 2010.10.07, 全文、全図 (ファミリーなし)	1 - 1 2
A	JP 2007-122651 A (富士通株式会社) 2007.05.17, 全文、全図 & US 2007/0168928 A1	1 - 1 2
A	JP 2010-191709 A (富士ゼロックス株式会社) 2010.09.02, 全文、全図 (ファミリーなし)	1 - 1 2
A	JP 2006-302032 A (株式会社エヌ・ティ・ティ・データ) 2006.11.02, 全文、全図 (ファミリーなし)	1 - 1 2