

(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
2012年8月16日(16.08.2012)



(10) 国際公開番号
WO 2012/108015 A1

- (51) 国際特許分類:
G06F 12/00 (2006.01)
- (21) 国際出願番号: PCT/JP2011/052758
- (22) 国際出願日: 2011年2月9日(09.02.2011)
- (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): 富士通株式会社(FUJITSU LIMITED) [JP/JP]; 〒2118588 神奈川県川崎市中原区上小田中4丁目1番1号 Kanagawa (JP).
- (72) 発明者; および
- (75) 発明者/出願人(米国についてのみ): 荒川 尊弘 (ARAKAWA, Takahiro) [JP/JP]; 〒2118588 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 Kanagawa (JP). 松田 隆(MATSUDA, Takashi) [JP/JP]; 〒2118588 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 Kanagawa (JP).
- (74) 代理人: 真田 有, 外(SANADA, Tamotsu et al.); 〒1800004 東京都武蔵野市吉祥寺本町1丁目10番31号 NOF吉祥寺本町ビル5階 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, RO, RS, RU, 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, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, MD, 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).

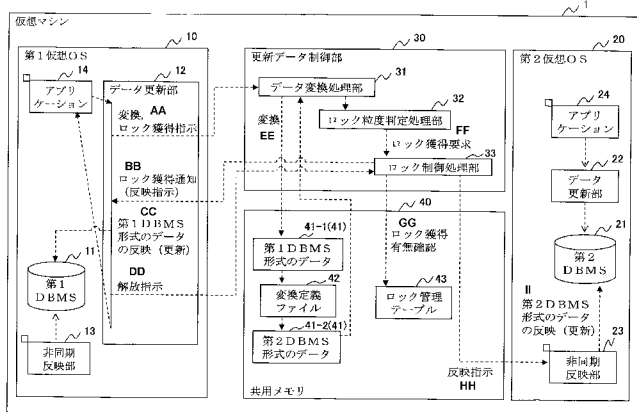
添付公開書類:

— 国際調査報告(条約第21条(3))

(54) Title: DATA SYNCHRONIZATION METHOD, DATA SYNCHRONIZATION PROGRAM, AND DATA SYNCHRONIZATION CONTROL DEVICE

(54) 発明の名称: データ同期方法、データ同期プログラム、及びデータ同期制御装置

[図2]



- 1 Virtual machine
10 First virtual OS
11 First DBMS
12 Data update unit
13, 23 Asynchronous reflection unit
14, 24 Application
20 Second virtual OS
21 Second DBMS
22 Data update unit
30 Update data controller
31 Data conversion processor
32 Lock acquisition request processor
33 Lock control processor
40 Shared memory

- 41-1(41) First DBMS format data
41-2(41) Second DBMS format data
42 Conversion definition file
43 Lock management table
AA Convert, lock acquisition command
BB Lock acquisition notification (reflection command)
CC Reflect first DBMS format data (update)
DD Release command
EE Conversion
FF Lock acquisition request
GG Lock acquisition presence/absence check
HH Reflection command
II Reflect second DBMS format data (update)

(57) Abstract: The present invention comprises obtaining first update data (41-1) relating to an update process performed on a first database (11), converting the first update data (41-1) to second update data (41-2) corresponding to the format of a second database (21) on the basis of conversion definition information (42) set in advance, determining a lock range on the basis of the first update data (41-1) and pre-update data not yet subjected to the update process, locking the first and second databases (11, 21) on the basis of the determined lock range, and updating the first and second databases (11, 21) using the first and second update data (41-1, 41-2), respectively.

(57) 要約: 第1のデータベース(11)に対して行なわれる更新処理にかかる第1更新データ(41-1)を取得し、予め設定された変換定義情報(42)に基づいて、前記第1更新データ(41-1)を第2のデータベース(21)の形式に対応した第2更新データ(41-2)に変換し、前記第1更新データ(41-1)と、前記更新処理が行なわれる前の更新前データとに基づいて、ロック範囲を決定し、決定されたロック範囲に基づいて、前記第1及び第2のデータベース(11, 21)をロックし、前記第1及び第2更新データ(41-1, 41-2)を用いて、前記第1及び第2のデータベース(11, 21)をそれぞれ更新する。

明 細 書

発明の名称：

データ同期方法、データ同期プログラム、及びデータ同期制御装置

技術分野

[0001] 本件は、データ同期方法、データ同期プログラム、及びデータ同期制御装置に関する。

背景技術

[0002] 基幹系システムや各種サービスを提供するシステム等、1以上の情報処理装置により構成される情報処理システムでは、扱われる情報の収集、管理のために、状況に応じて種々のデータベースが用いられている。

データベースとしては、例えば、Oracle（登録商標）、SQL Server（登録商標）、MySQL、及びPostgreSQL等のSQLベースのデータベースや、Symfoware（登録商標）、Power RW+、及びPervasive SQL（登録商標）等のRead/Writeインタフェースのデータベース等が挙げられる。

[0003] システム内で2以上のデータベースを用いる場合に使用されるこれらのデータベースは、データベース間の同期等の観点から、構造（種類）が同一のものに統一されることが理想である。しかしながら、例えば以下の状況下等においては、互いに異なる種類のデータベースであり、互いに異なるデータ構造を持つデータベースが混在して使用される場合がある。

[0004] （a）業務統合などで、異なる業務で用いていた互いに異なる構造を持つデータベースをそれぞれ引き続き使用し、データのための整合性を合わせて運用する場合。

（b）構造（種類）が同一のデータベースへの将来的な統合を控え、過渡的な対処として、互いに異なる構造を持つデータベースを並行して使い続ける場合。

（c）使用しているデータベースのライセンス料が高く、新たに並行して

導入するデータベースとして、使用しているものとは構造（種類）が異なる安価なデータベースを採用する場合。

[0005] （d）データベースが持つ機能に差があり、１種類のデータベースに統合できない場合。

上記（a）～（d）に示すような場合、互いに異なる構造を持つデータベースのデータを同期させるために、例えば、図１７～図１９に示す第１～第３のいずれかの手法を用いることができる。

[0006] 以下の例においては、仮想マシン１００において、第１DBMS（Database Management System）１１１を備える第１仮想OS（Operating System）１１０と、第２DBMS１２１を備える第２仮想OS１２０とが実行される場合の、第１DBMS１１１と第２DBMS１２１とのデータを同期する例を示す。

図１７は、第１仮想OS１１０のアプリケーション１１３が第２仮想OS１２０のデータベースにアクセスしてデータの同期を行なう第１手法を示す図であり、図１８は、データベースが備えるトリガー機能により、データを同期する第２手法を示す図である。また、図１９は、データベースが備えるレプリケーション機能により、データを同期する第３手法を示す図である。

[0007] ここで、図１７～図１９に示すように、仮想マシン１００は、情報処理システム上で動作し、第１仮想OS１１０及び第２仮想OS１２０を実行する。また、仮想マシン１００は、第１仮想OS１１０及び第２仮想OS１２０に対する、情報処理システムのCPU（Central Processing Unit）やメモリ等のリソースの割り当て等の管理を行なう。

仮想マシン１００としてはハイパーバイザ等が挙げられる。

[0008] 第１仮想OS１１０は、第１DBMS１１１と、通信部１１２とを備え、第２仮想OS１２０は、第２DBMS１２１と、通信部１２２とを備える。第１DBMS１１１、第２DBMS１２１は、それぞれ第１仮想OS１１０、第２仮想OS１２０で管理されるデータベースである。本例においては、第１DBMS１１１と、第２DBMS１２１とは、互いに異なる種類のデー

データベースであり、互いに異なるデータ構造を持つ。

[0009] また、第1仮想OS 110と第2仮想OS 120とは、通信部112及び122により、ネットワークを介して相互に通信可能である。

以下、図17～図19を参照しながら、アプリケーション113又は114により第1DBMS 111が更新される場合の、第1DBMS 111と第2DBMS 121とのデータ同期の手法について説明する。

[0010] 第1手法では、図17に示す例では、互いに異なる構造を持つデータベースのデータを自動的に同期させるため、データベースにアクセスする全てのプログラムに対して変更が加えられた場合を示す。

例えば、第1仮想OS 110上で実行されるアプリケーション113が、第1DBMS 111のみにアクセスするものであった場合、第1DBMS 111及び第2DBMS 121の双方にアクセスするように、アプリケーション113に変更が加えられる。

[0011] アプリケーション113に対する変更により、図17に示す(1)～(4)の順序で、アプリケーション113による第2DBMS 121へのデータの同期が行なわれる。

すなわち、アプリケーション113は、第1DBMS 111に変更を加えるとともに(1)、当該変更にかかるデータを通信部112に出力する(2)。通信部112は、アプリケーション113から入力されたデータをネットワーク経由で第2仮想OS 120の通信部122に送信する(3)。最後に、通信部122は、通信部112から受信したデータを、第2DBMS 121に反映(更新)し(4)、仮想マシン100におけるデータの同期処理が完了する。

[0012] なお、図17に示す(2)において通信部112に入力されるデータは、アプリケーション113によって、第1DBMS 111の形式に対応したデータから第2DBMS 121の形式に対応したデータへ変換されたものである。

第2手法においては、図18に例示するように、互いに異なる構造を持つ

データベースのデータを同期させるため、データベースが備えるトリガー機能を用いて、相手先データベースにデータを整合させて同期させる。

[0013] トリガー機能とは、データベースに対する何らかのアクション、例えば更新操作などをトリガーとして、所定の処理が実行される機能である。

仮想マシン100では、このトリガー機能115が用いられ、図18に示す(1)～(10)の順序で、第2DBMS121へのデータの同期が行なわれる。

すなわち、アプリケーション114が第1DBMS111に変更を加えると(1)、第1DBMS111はトリガー機能115を起動する(2)。そして、トリガー機能115は、当該変更にかかるデータをデータ変換部116に入力する(3)。データ変換部(4)は、第1DBMS111の形式に対応したデータをネットワーク通信用のデータへ変換し、変換したデータを通信部112へ出力する(4)。

[0014] 次いで、通信部112は、入力されたデータをネットワーク経由で第2仮想OS120の通信部122に送信し(5)、通信部122は、通信部112から受信したデータを、データ変換部126へ出力する(6)。そして、データ変換部126は、入力されたネットワーク通信用のデータを第2DBMS121の形式に対応したデータへ変換し、変換したデータを第2DBMS121に反映(更新)する(7)。

[0015] 第2DBMS121の更新が完了すると、通信部122は、更新完了の通知をネットワーク経由で第1仮想OS110の通信部112に送信する(8)。更新完了の通知は、通信部112及びトリガー機能115を経由して(9)、アプリケーション114に通知され(10)、仮想マシン100におけるデータの同期処理が完了する。

第3手法においては、図19に例示するように、互いに異なる構造を持つデータベースのデータを同期させるため、データベースが備えるレプリケーション機能を用いて、データを同期させる。

[0016] レプリケーション機能とは、1箇所のデータベースで行なわれた更新デー

タや操作を蓄積して記憶しておき、一定時間毎、又は更新量が一定量貯まった場合等の何らかのタイミングで、蓄積された更新データ等を別サイトへ送信し、別サイトのデータベースへ更新させる機能である。

仮想マシン１００では、このレプリケーション機能により、図１９に示す（１）～（６）の順序で、第２DBMS１２１へのデータの同期が行なわれる。

[0017] すなわち、アプリケーション１１４が第１DBMS１１１に変更を加えると（１）、第１DBMS１１１は、当該更新にかかる更新データや操作を蓄積し（２）、自身のデータベースの更新が完了した旨をアプリケーション１１４に通知する（３）。次いで、一定時間毎、又は更新量が一定量貯まった場合等の何らかのタイミングで、データ同期部１１８は、蓄積した更新データ１１７を通信部１１２に出力する（４）。

[0018] そして、通信部１１２は、入力された更新データ１１７をネットワーク経由で第２仮想OS１２０の通信部１２２に送信する（５）。最後に、通信部１２２は、通信部１１２から受信したデータを、データ同期部１２８による変換処理等を介して、第２DBMS１２１に反映（更新）し（６）、仮想マシン１００におけるデータの同期処理が完了する。

ここで、図１９における符号１１７は、蓄積した更新データを示す。また、データ同期部１１８及び１２８は、上述したレプリケーション機能の実行処理を行なう。

[0019] また、上述した第２及び第３手法では、通信部１１２及び１２２間の通信は、ネットワーク経由で行なわれる場合が多い。従って、第１仮想OS１１０と第２仮想OS１２０とは、互いに異なる情報処理装置のリソースを使用して実行されても良い。

なお、互いに同一の構造を持つデータベースのデータを同期させる技術としては、例えば、２つの電子計算機からアクセス可能な共用メモリに、それぞれのDBMS用の共用バッファを設ける技術等が知られている。

先行技術文献

特許文献

[0020] 特許文献1：特開平5－313985号公報

特許文献2：特開平8－95841号公報

特許文献3：特開平11－306061号公報

特許文献4：特開2009－15534号公報

発明の概要

発明が解決しようとする課題

[0021] 上述した第1～第3のデータ同期手法では、それぞれ以下のような問題がある。

(i) 第1手法では、データベースにアクセスする全てのプログラムに対して変更を行なうため、プログラムの変更に多くの工数が発生する。またプログラムの保守性も低下し、以後の保守工数増加の要因となる。

(ii) 第2手法では、データ更新時、更新データがネットワーク経由で更新先のデータベースに反映される場合が多い。ネットワーク経由による更新先のデータベース、例えば第2DBMS121へのデータの同期には、更新元データベース、例えば第1DBMS111に対するデータバスを通じたデータの更新に比べて時間がかかる。従って、双方のデータベースにおけるデータのロック区間にアクセスすることができない時間、つまりロック区間がロックされる時間が長くなり、データベースの共用度が低下する。

[0022] なお、データベースのロックとは、データベースのデータを更新する場合に、他のアプリケーション等から更新対象の範囲へのアクセスを禁止するために行なう排他制御であり、ロック区間とは、行(Row)等、更新対象の範囲である。

(iii) 第3手法では、DBMSの提供元や種類が異なると、レプリケーション機能に差異が生じ、データベース間のデータ同期に、レプリケーション機能を使用することができない場合がある。また、データの同期は、更新元のデータベース(例えば第1DBMS111)への更新から遅延して更新先のデータベース(例えば第2DBMS121)への更新が行なわれるため、

元のデータベースへのデータ更新後、システム使用者が更新先のデータベースを参照しても、更新後のデータが反映されていない場合がある。

[0023] (iv) 第2及び第3手法では、トリガー機能及びレプリケーション機能は、いずれもネットワークを経由して更新先のデータベースとデータ同期を行なうため、データ漏洩のリスクが高い。さらに、ネットワークが一時的に使用できない場合、データの同期を行なうことができず、またネットワーク復旧後にデータを同期させようとする、ネットワークへのアクセスが集中し、ネットワーク負荷が非常に高くなるという問題がある。

[0024] (v) また、データの同期に際し、例えば以下の(v-i)及び(v-ii)に示す問題により、よりデータベースの共用度を向上させる技術が望まれる。

(v-i) 更新元のデータベースで更新されるデータについて、互いに異なる構造を持つ更新先のデータベースとデータ同期を行なう場合、これら複数のデータベースのロック区間を統一しようとする、1つのデータベースにロック区間を設定する場合に比べて、ロック区間が長くなってしまう。

[0025] 例えば、更新元のデータベースのロック区間がテーブル内の1行分であった場合、更新先のデータベースに設定するロック区間は、更新元のデータベースのロック区間の全ての列(Column)が含まれるように設定される。更新元のデータベースと更新先のデータベースとは互いに異なる構造を持つため、更新元のデータベースのロック区間に含まれる項目が、更新先のデータベースの複数のテーブルに分散している場合がある。このとき、更新先のデータベースでは、複数のテーブル内の更新対象の行をロック区間に設定するため、ロック区間が長くなってしまう。

[0026] (v-ii) セキュリティ向上のため、データベースのテーブルを直接参照せず、必要な列のみをビュー(View)によって参照することがある。ビューによってあるテーブル内の列を参照する場合、例えば他のアプリケーション等により、ビューで参照していない列に対する更新が発生すると、更新が発生した行がロックされ、ビュー経由のアクセスが行えないという問題が

ある。

[0027] 上述の点に鑑み、本件の目的の１つは、互いに異なる種類のデータベース間でデータを同期させる際に、同期させるデータをデータベース間でより簡易に変換することである。

また、本件の目的の１つは、互いに異なる種類のデータベース間でデータを同期させる際に、データベースのロックによるアプリケーションへの影響を抑えることである。

さらに、本件の目的の１つは、互いに異なる種類のデータベース間でデータを同期させる際の、データ漏洩のリスクを低減することである。

[0028] また、本件の目的の１つは、互いに異なる種類のデータベース間でデータを同期させる際の、データベース間のタイムラグを低減することである。

なお、前記目的に限らず、後述する発明を実施するための形態に示す各構成により導かれる作用効果であって、従来技術によつては得られない作用効果を奏することも本発明の他の目的の１つとして位置付けることができる。

課題を解決するための手段

[0029] 本件のデータ同期方法は、第１のデータベースに対して行なわれる更新処理にかかる第１更新データを取得するステップと、予め設定された変換定義情報に基づいて、前記第１更新データを第２のデータベースの形式に対応した第２更新データに変換するステップと、前記第１更新データと、前記更新処理が行なわれる前の更新前データとに基づいて、ロック範囲を決定するステップと、決定されたロック範囲に基づいて、前記第１及び第２のデータベースをロックするステップと、前記第１及び第２更新データを用いて、前記第１及び第２のデータベースをそれぞれ更新するステップと、をそなえるものである。

[0030] また、本件のデータ同期プログラムは、第１のデータベースに対して行なわれる更新処理にかかる第１更新データを取得し、予め設定された変換定義情報に基づいて、前記第１更新データを第２のデータベースの形式に対応し

た第2更新データに変換し、前記第1更新データと、前記更新処理が行なわれる前の更新前データとに基づいて、ロック範囲を決定し、決定されたロック範囲に基づいて、前記第1及び第2のデータベースをロックし、前記第1及び第2更新データを用いて、前記第1及び第2のデータベースをそれぞれ更新する、処理をコンピュータに実行させるものである。

[0031] さらに、本件のデータ同期制御装置は、第1のデータベースに対して行なわれる更新処理にかかる第1更新データを取得して、予め設定された変換定義情報に基づいて、前記第1更新データを第2のデータベースの形式に対応した第2更新データに変換するデータ変換部と、前記第1更新データと、前記更新処理が行なわれる前の更新前データとに基づいて、ロック範囲を決定するロック範囲決定部と、ロック範囲決定部により決定されたロック範囲に基づいて、前記第1及び第2のデータベースをロックするとともに、前記第1及び第2のデータベースに対して、それぞれ前記第1更新データ及び前記データ変換部により変換された前記第2更新データを用いた更新を指示するロック制御部とをそなえるものである。

発明の効果

[0032] 開示の技術によれば、互いに異なる種類のデータベース間でデータを同期させる際に、同期させるデータをデータベース間でより簡易に変換することができる。

また、互いに異なる種類のデータベース間でデータを同期させる際に、データベースのロックによるアプリケーションへの影響を抑えることができる。

さらに、互いに異なる種類のデータベース間でデータを同期させる際の、データ漏洩のリスクを低減することができる。

[0033] また、互いに異なる種類のデータベース間でデータを同期させる際の、データベース間のタイムラグを低減することができる。

図面の簡単な説明

[0034] [図1]本実施形態の一例としての仮想マシンの構成例を示す図である。

[図2]本実施形態の一例としての仮想マシンによるデータ同期の処理を説明するための図である。

[図3]本実施形態の一例としての変換定義ファイルの構成例を示す図である。

[図4]本実施形態の一例としてのデータベースにおける数値データの内部表現形式を示す図である。

[図5]本実施形態の一例としてのデータ変換処理部によるデータ変換処理を説明するための図である。

[図6]本実施形態の一例としてのロック管理テーブルの構成例を示す図である。

[図7]本実施形態の一例としてのロック管理テーブルにおける列パターンを説明するための図である。

[図8]本実施形態の一例としての仮想マシンによるデータ同期の処理を説明するためのシーケンス図である。

[図9]本実施形態の一例としてのデータ更新部の処理を説明するためのフローチャートである。

[図10]本実施形態の一例としての更新データ制御部のデータ変換及びロック獲得の処理を説明するためのフローチャートである。

[図11]本実施形態の一例としてのデータ変換処理部のデータ変換の処理を説明するためのフローチャートである。

[図12]本実施形態の一例としてのロック粒度判定処理部のロック範囲決定の処理を説明するためのフローチャートである。

[図13]本実施形態の一例としてのロック制御処理部によるロック獲得の処理を説明するためのフローチャートである。

[図14]本実施形態の一例としての更新データ制御部のロック解放の処理を説明するためのフローチャートである。

[図15]本実施形態の一例としてのロック制御処理部によるロック解放の処理を説明するためのフローチャートである。

[図16]本実施形態の一例としての非同期反映部の処理を説明するためのフロ

ーチャートである。

[図17]従来のデータ同期手法一例としての第1手法を示す図である。

[図18]従来のデータ同期手法一例としての第2手法を示す図である。

[図19]従来のデータ同期手法一例としての第3手法を示す図である。

発明を実施するための形態

[0035] 以下、図面を参照して本発明の実施の形態を説明する。

〔1〕一実施形態

〔1-1〕一実施形態の構成

図1は、本実施形態の一例としての仮想マシン1の構成例を示す図である。

仮想マシン1は、実際のハードウェア構成によらない仮想的なコンピュータであり、仮想マシン1上で複数の仮想OSを実行するとともに、複数の仮想OSに対する各種リソースの割り当て等の管理を行なうものである。仮想マシン1としては、ハイパーバイザ等が挙げられる。

[0036] なお、本実施形態においては、仮想マシン1が、1以上の情報処理装置により構成される情報処理システム上で動作し、情報処理システムのCPUやメモリ等のリソースを使用する場合について例示する。

また、本実施形態においては、仮想マシン1は、第1仮想OS10、第2仮想OS20、及び更新データ制御部30を備え、これらを実行するとともに、情報処理システムのメモリの一部を共有メモリ40として使用する。さらに、仮想マシン1は、第1仮想OS10、第2仮想OS20及び更新データ制御部30に対して、情報処理システムのリソースの割り当て等の管理を行なう。

[0037] 第1仮想OS10及び第2仮想OS20は、それぞれ仮想マシン1が提供するカーネルにより、情報処理装置のハードウェアや、個々のメモリにアクセスすることができる。

第1仮想OS10は、第1DBMS11と、データ更新部12と、非同期反映部13と、アプリケーション14とを備える。また、第2仮想OS20

は、第2 DBMS 21と、データ更新部22と、非同期反映部23と、アプリケーション24とを備える。

[0038] 第1 DBMS 11及び第2 DBMS 21は、それぞれ第1仮想OS 10及び第2仮想OS 20で管理されるデータベースである。本実施形態においては、第1 DBMS 11と、第2 DBMS 21とは、互いに異なる種類のデータベースであり、互いに異なるデータ構造を持つ。なお、第1 DBMS 11及び第2 DBMS 21は、データを行及び列のマトリクス状に管理する。データベース内のデータが更新される際には、更新対象の行が指定され、指定された行内の更新対象の領域が更新される。

[0039] また、仮想マシン1は、第1仮想OS 10、第2仮想OS 20でそれぞれ管理される第1 DBMS 11、第2 DBMS 21について、後述する手法によりデータの同期を行なう。

図2は、本実施形態の一例としての仮想マシン1によるデータ同期の処理を説明するための図である。図2に示す例では、仮想マシン1による第1 DBMS 11及び第2 DBMS 12間のデータ同期の処理を、第1仮想OS 10のアプリケーション14が第1 DBMS 11に対する更新を行なう場合を例に挙げて説明する。

[0040] アプリケーション14は、第1仮想OS 10上で動作する任意のアプリケーションであり、第1 DBMS 11に対してデータの変更や参照等のアクセスを行なう。

また、アプリケーション14は、第1 DBMS 11に対してデータの変更、追加、削除等の更新を行なう場合、データ更新部12に対して更新処理要求（更新要求）を出力する。

[0041] データ更新部12は、アプリケーション14からデータベースの更新処理要求を入力されると、第1 DBMS 11に対するデータの更新を行なう。つまり、データ更新部12は、第1 DBMS 11の更新時にアプリケーション14から呼び出されるものであり、アプリケーション14は、データ更新部12を経由して第1 DBMS 11を更新する。

ここで、データ更新部 12 は、第 1 DBMS 11 と第 2 DBMS 21 との間でデータの同期を行なうために、第 1 DBMS 11 に対する更新を、第 2 DBMS 21 が更新されるタイミングと同時に行なう。

[0042] 具体的には、データ更新部 12 は、アプリケーション 14 からデータベースの更新処理要求を入力されると、アプリケーション 14 からアクセス可能なメモリ等から、更新前データ及び更新データ（第 1 更新データ）41-1 を取得する。

ここで、更新前データは、第 1 DBMS 11 に対して行なわれる更新処理にかかる更新前のデータである。また、更新データ 41-1 は、第 1 DBMS に対して行なわれる更新処理にかかる更新後のデータ（更新後データ）であり、第 1 DBMS 11 のデータ構造に対応した形式（第 1 DBMS 形式）のデータである。

[0043] そして、データ更新部 12 は、取得した更新前データ及び更新データ 41-1 から更新差分データを生成するとともに、更新差分データ及び更新データ 41-1 を、後述する変換指示及びロック獲得指示とともに更新データ制御部 30 に出力する。

データ更新部 12 による更新差分データの生成は、例えば、更新前データと更新後データ 41-1 とを比較し、更新される箇所（例えば更新される列）の情報を抽出することにより行なわれる。生成される更新差分データには、更新される箇所を示す情報が含まれる。なお、更新差分データには、更新される箇所の更新データが含まれてもよい。

[0044] なお、このデータ更新部 12 により出力される更新差分データは、データそのものでも良く、また、これらのデータが格納された場所のアドレス等、更新差分データを特定するための情報であっても良い。以下、これらをまとめて、単に「更新差分データ」という。

また、同様に、データ更新部 12 により出力される更新データ 41-1 は、データそのものでも良く、また、これらのデータが格納された場所のアドレス等、更新データ 41-1 を特定するための情報であっても良い。以下、

これらをまとめて、単に「更新データ 4 1 - 1」という。

[0045] つまり、データ更新部 1 2 は、更新差分データ及び更新データ 4 1 - 1 をデータ更新部 1 2 に出力する代わりに、若しくは加えて、例えば後述する共有メモリ 4 0 上に更新差分データ及び更新データ 4 1 - 1 を格納し、これらのデータを格納したアドレスを出力しても良い。

以下、更新データ制御部 3 0 に対して変換指示及びロック獲得指示を出力したデータ更新部 1 2 又は 2 2 を、要求元（指示元）のデータ更新部 1 2 又は 2 2、又は単に要求元（指示元）という。

[0046] 更新データ制御部 3 0 では、要求元の更新データ 4 1 - 1 が、要求元以外のデータ同期対象である第 2 DBMS 2 1 の形式に対応した第 2 DBMS 形式の更新データ（第 2 更新データ） 4 1 - 2 に変換される。また、更新データ制御部 3 0 では、後述の如くデータベースのロック範囲が決定され、第 1 DBMS 1 1 及び第 2 DBMS 2 1 がロックされ、要求元であるデータ更新部 1 2 へロック獲得通知（反映指示）が出力される。

[0047] データ更新部 1 2 は、第 1 DBMS 1 1 及び第 2 DBMS 2 1 のロック獲得通知（反映指示）が更新データ制御部 3 0 から入力されると、第 1 DBMS 1 1 に対して更新を行なう。また、データ更新部 1 2 は、更新データ制御部 3 0 に対して、ロックした第 1 DBMS 1 1 及び第 2 DBMS 2 1 のロック解放指示を出力する。

上述の如く、データ同期対象のデータベースである第 1 DBMS 1 1 及び第 2 DBMS 2 1 のロック獲得指示及びロック解放指示は、データ同期の要求元、つまり図 2 に示す例では仮想 OS 1 0 のデータ更新部 1 2 によって発行される。

[0048] 非同期反映部 1 3 は、第 2 仮想 OS 2 0 等の他の仮想 OS が要求元となったデータ同期が発生した場合に、第 1 DBMS 形式の更新データ 4 1 - 1 を第 1 DBMS 1 1 に対して反映（更新）する。

つまり、非同期反映部 1 3 は、他の仮想 OS からのデータベースの更新処理要求によって、反映指示と他の仮想 OS の DBMS 形式の更新データから

第1 DBMS形式に変換された更新データ41-1とが、更新データ制御部30から入力された場合に動作する。また、非同期反映部13による上述した処理は、データ更新部12やアプリケーション14とは非同期で行なわれる。

[0049] なお、非同期反映部13は、共有メモリ40上に格納された更新データ41-1のアドレスを、更新データ制御部30から入力されても良い。この場合、非同期反映部13は、入力されたアドレスにアクセスすることにより、更新データ41-1を取得する。

以下、更新データを示す符号としては、更新データ41-1及び41-2のうちの1つを特定する必要があるときには符号41-1、41-2を用いるが、任意の更新データを指すときには符号41を用いる。なお、更新データ41には、上述の如く、データ更新部12又は22から出力されたもの、すなわち要求元の更新データと、更新データ制御部30によって変換されたもの、すなわち要求元以外の同期対象のDBMS形式に変換された更新データとが含まれる。

[0050] なお、図2に示す例では、第1仮想OS10が要求元となったデータ同期が発生しているため、非同期反映部13による処理は行なわれない。

データ更新部22、非同期更新部23、及びアプリケーション24は、上述したデータ更新部12、非同期更新部13、及びアプリケーション14とそれぞれ同様の機能を備える。

[0051] すなわち、アプリケーション24は、第2仮想OS20上で動作する任意のアプリケーションであり、第2 DBMS 21に対してデータの変更や参照等のアクセスを行なう。

また、アプリケーション24は、第2 DBMS 21に対してデータの変更、追加、削除等の更新を行なう場合、データ更新部12に対して更新処理要求（更新要求）を出力する。

[0052] データ更新部22は、データ更新部12と同様に、アプリケーション24から第2 DBMS 21の更新処理要求を入力されると、更新処理にかかる更

新前データ及び更新データ 4 1 - 2 を取得する。そして、データ更新部 2 2 は、更新前データ及び更新データ 4 1 - 2 から更新差分データを生成し、更新データ 4 1 - 2 及び更新差分データを、変換指示及びロック獲得指示とともに更新データ制御部 3 0 に出力する。

[0053] また、データ更新部 2 2 は、更新データ処理部 3 0 から反映指示を入力されると、第 2 DBMS 2 1 に対して更新を行なうとともに、更新データ制御部 3 0 に対して、ロック解放指示を出力する。

非同期反映部 2 3 は、他の仮想 OS（図 2 に示す例では第 1 仮想 OS 1 0）からのデータベースの更新処理要求によって、反映指示と他の仮想 OS の DBMS 形式の更新データから第 2 DBMS 形式に変換された更新データ 4 1 - 2 とが、更新データ制御部 3 0 から入力された場合に動作する。

[0054] すなわち、図 2 に示す例においては、データ同期の要求元である第 1 仮想 OS 1 0 とは異なる、第 2 仮想 OS 2 0 の非同期反映部 2 3 は、更新データ制御部 3 0 からの反映指示に基づいて、更新データ 4 1 - 2 を第 2 DBMS 2 1 へ反映（更新）する。

つまり、第 2 仮想 OS 2 0 の非同期反映部 2 3 は、他の仮想 OS（第 1 仮想 OS 1 0）が要求元となったデータ同期の際に動作する。このとき、非同期反映部 2 3 は、第 2 DBMS 形式の更新データ 4 1 - 2 及び反映指示を更新データ制御部 3 0 から入力されると、更新データ 4 1 - 2 を第 2 DBMS 2 1 へ反映する。

[0055] なお、上述の如く、非同期反映部 2 3 は、共有メモリ 4 0 上に格納された更新データ 4 1 - 2 のアドレスを、更新データ制御部 3 0 から入力されても良く、この場合、入力されたアドレスにアクセスすることにより、更新データ 4 1 - 2 を取得する。

また、非同期反映部 2 3 に対する反映指示は、データ更新部 1 2 に対する反映指示と同時若しくはほぼ同時に更新データ制御部 3 0 から出力される。

[0056] なお、第 2 仮想 OS 2 0 のアプリケーション 2 4 が第 2 DBMS 2 1 に対する更新を行なう際には、図 2 を用いて上述した説明における、第 1 仮想 OS

S 1 0と第2仮想OS 2 0とを入れ替えることで、仮想マシン1によるデータ同期の処理を実現することができる。

なお、上述の如く、データ同期の要求元以外の仮想OSにおけるアプリケーション及びデータ更新部では、要求元によるデータ同期に伴う処理は行なわれず、指示待ちの状態となる。すなわち、図2に示す例においては、データ同期の要求元である第1仮想OS 1 0とは異なる、第2仮想OS 2 0のアプリケーション2 4及びデータ更新部2 2では、第1仮想OS 1 0によるデータ同期に伴う処理は行なわれず、指示待ちの状態となる。

[0057] 一方、上述の如く、データ同期の要求元の仮想OSにおける非同期反映部では、要求元によるデータ同期に伴う処理は行なわれず、指示待ちの状態となる。すなわち、図2に示す例においては、第1仮想OS 1 0が要求元となったデータ同期が発生しているため、非同期反映部1 3による処理は行なわれず、指示待ちの状態となる。

更新データ制御部3 0は、仮想マシン1に配置され、複数の仮想OS、すなわち第1仮想OS 1 0及び第2仮想OS 2 0のデータ更新部1 2及び2 2からアクセスされる。更新データ制御部3 0は、データ更新部1 2又は2 2から入力された要求に対する処理を行ない、結果をデータ更新部1 2及び2 2に通知する。

[0058] 更新データ制御部3 0は、データ変換処理部3 1と、ロック粒度判定処理部3 2と、ロック制御処理部3 3とを備える。

データ変換処理部3 1は、要求元のデータ更新部1 2又は2 2から変換指示と更新データ4 1とを入力されると、事前に定義された変換定義ファイル4 2に基づいて、更新データ4 1のデータ変換を行なう。

[0059] つまり、データ変換処理部3 1は、予め設定された変換定義ファイル4 2に基づいて、要求元のDBMSの形式に対応する更新データ4 1を、要求元のDBMS以外の全てのDBMSに対して、各形式に対応した更新データ4 1に変換する処理を行なう。

図2に示す例においては、データ変換処理部3 1は、第1DBMS 1 1の

形式に対応する第1更新データ41-1を、第2DBMS21の形式に対応した第2更新データ（変換後データ）41-2に変換する処理を行なう。

[0060] また、データ変換処理部31は、要求元のデータ更新部12又は22により入力された更新データ41と、データ変換処理部31によって変換された更新データ41とを共有メモリ40に格納する。なお、上述の如く、要求元のデータ更新部12又は22により要求元の更新データ41が共有メモリ40に格納される場合には、データ変換処理部31は、変換した更新データ41のみを共有メモリ40に格納しても良い。

[0061] なお、データ変換処理部31による変換結果は、変換指示の要求元のデータ更新部12又は22へ通知される。

また、データ変換処理部31は、データ変換が完了すると、更新データ41に基づいて、データベース内のテーブルにおける更新データ41が含まれる行を示す後述するロックキーを生成する。

[0062] ロック粒度判定処理部32は、更新対象のデータベースの領域をロックするロック粒度（ロック範囲）を決定する。

なお、ロック粒度（ロック範囲）とは、データベースのデータが更新される際に排他制御が行なわれる範囲である。本実施形態においては、ロック範囲は、ロック粒度判定処理部32により、第1DBMS11及び第2DBMS21におけるテーブルの列を最小単位として決定される。つまり、ロック範囲は、更新処理にかかる更新対象の行の中で、更新が行なわれる1以上の列を全て含む範囲である。

[0063] 具体的には、ロック粒度判定処理部32は、要求元の更新前データと更新データ41とに基づいて、ロック範囲を決定する。つまり、ロック粒度判定処理部32は、更新前データ及び更新後データ41に基づいて生成された更新差分データから、更新処理によって更新される範囲を判定し、当該範囲をロック範囲に決定する。

図2に示す例においては、ロック粒度判定処理部32によるロック範囲の決定は、更新前データ及び更新後データ41-1に基づいて生成された更新

差分データに基づいて行なわれる。

[0064] 上述の如く、更新差分データには、更新される箇所（列）を示す情報が含まれるため、ロック粒度判定処理部 32 は、更新される箇所を示す情報に基づいて、第 1 DBMS 11 及び第 2 DBMS 21 において更新される列（項目）を判定する。

そして、ロック粒度判定処理部 32 は、更新されると判定した列（項目）を全て含む範囲をロック範囲に決定する。

[0065] ここで、ロック粒度判定処理部 32 により生成されるロック範囲は、例えば、図 7 に示すような列パターンで表される。

列パターンとは、後述するロック制御処理部 33 によりロックされる対象の列を“1”、非対象の列を“0”としたビット列である。

例えば、図 7 に示すように、データベース内の 1 行に 8 列存在し、ロック粒度判定処理部 32 により 1 列目、3 列目、8 列目がロック範囲に決定された場合、列パターンは“10100001”となる。

[0066] 図 7 に示すような列パターンで表されるロック範囲について、更新データ制御部 30 によるロックが行なわれることにより、第 1 DBMS 11 及び第 2 DBMS 21 では、更新対象の行の中で、更新データ 41 により更新される列についてのみ排他制御が行なわれる。

すなわち、従来は行単位で排他制御が行なわれていたが、ロック粒度判定処理部 32 による列を最小単位としたロック範囲の決定により、更新対象の行及びロック範囲内の 1 列により特定される領域単位で排他制御を行なうことができる。従って、更新対象の行の中で更新されない範囲の列については、排他制御が行なわれずに済み、データベースの共用度の向上が実現できる。

[0067] 以下、更新対象の行及びロック範囲内の 1 列により特定される領域を、ブロックという。すなわち、本実施形態においては、データベースにおける 1 行×1 列のブロック単位でロック範囲が決定され、1 以上のブロックにより、排他制御が行なわれる領域を特定することができる。

ロック制御処理部 33 は、要求元のデータ更新部 12 又は 22 により入力されたロック獲得指示と、ロック粒度判定処理部 32 によって決定されたロック粒度とに基づいて、データ同期対象である第 1 DBMS 11 及び第 2 DBMS 21 をロックする。

[0068] 具体的には、ロック制御処理部 33 は、データ更新部 12 又は 22 から入力されたロック獲得指示及び更新データ 41 と、ロック粒度判定処理部 32 によって決定されたロック範囲とに基づいて、後述するロック管理テーブル 43 における、更新対象のロック範囲のロック状態を確認する。

ロック管理テーブル 43 には、データベース内のテーブルにおける更新データ 41 が含まれる行を示すロックキーが格納されている。ロック制御処理部 33 は、ロック管理テーブル 43 に格納されたロックキーとロック範囲とに基づいて、ロック粒度判定処理部 32 により決定された更新対象のロック範囲がロック可能であるか、又はロック済みであるか否かを確認する。

[0069] なお、ロック制御処理部 33 による、更新対象のロック範囲のロック状態の具体的な確認処理については、後述する。

ロック範囲が他のスレッドによってロックされていない場合は、ロック制御処理部 33 は、当該ロック範囲でロックを行ない、要求元のデータ更新部 12 又は 22 に対して、ロック獲得通知、つまり更新データ 41 の反映指示を行なう。また、ロック制御処理部 33 は、要求元とは異なる仮想 OS のデータ更新部 12 又は 22 に対して、データ変換処理部 31 によって変換された更新データ 41 の反映指示を行なう。

[0070] なお、ロック制御処理部 33 によるロック範囲のロックは、ロック管理テーブル 43 に対して、更新処理にかかるロック範囲がロック獲得状態であることを示すロック獲得情報を設定することにより行なわれる。ロック獲得情報には、更新対象のロックキーとロック範囲とが含まれる。

一方、ロック範囲が他のスレッドによってロックされている場合、すなわちロック範囲がロック済み資源であった場合、ロック制御処理部 33 は、ロックされているロック範囲が解放されるまで待ち合せを行なう。この待ち合

わせは、ロック制御処理部 33 が、ロック管理テーブル 43 に対して、更新処理にかかるロック範囲がロック待ち状態であることを示す待ち情報を、待ち合わせ中の要求として設定（登録）することにより行なわれる。

[0071] また、ロック制御処理部 33 は、要求元のデータ更新部 12 又は 22 からロックの解放指示を入力されると、ロック管理テーブル 43 における該当するロック範囲のロックを解放する。

なお、ロック制御処理部 33 によるロック解放に伴い、待ち合わせ中の要求についてロック範囲のロックが獲得可能となった場合、当該待ち合わせ中の要求は、要求元で予め設定された優先度に従って待ち状態が解除され、再度ロック獲得指示として処理される。具体的には、複数の待ち合わせ中の要求についてロック範囲のロックが獲得可能となった場合、例えば、最も高い優先度が設定された待ち合わせ中の要求の待ち状態が解除され、再度ロック獲得指示として処理される。

[0072] なお、待ち合わせ中の要求の優先度が同一である場合は、待ち状態の登録が行なわれた先後関係など、仮想マシン 1 により決定される所定の条件等に応じて、いずれかの要求の待ち状態が解除される。

待ち状態が解除され、再度ロック獲得指示として処理される待ち合わせ中の要求にかかる更新対象のロック範囲は、上述したロック制御処理部 33 によってロックされる。

[0073] なお、上述したデータ変換処理部 31 及びロック粒度判定処理部 32 によるそれぞれの処理は、いずれかの処理が先に、つまりシーケンシャルに行なわれても良いし、それぞれの処理が並行して行なわれても良い。いずれの場合でも、ロック制御処理部 33 による処理は、データ変換処理部 31 及びロック粒度判定処理部 32 によるそれぞれの処理が完了した後に実行される。

[0074] また、ここまで、要求元のデータ更新部 12 又は 22 が更新差分データを生成するものとして説明したが、これに限定されるものではなく、更新差分データの生成は、ロック粒度判定処理部 32 によって行なわれても良い。この場合、ロック粒度判定処理部 32 は、要求元のデータ更新部 12 又は 22

から更新前データ及び更新データ 4 1 を入力されて、更新差分データを生成する。

[0075] また、図 2 に示す例では、更新データ 4 1 及び変換指示、更新差分データ、及びロック獲得指示がいずれもデータ変換処理部 3 1 に入力されているが、これに限定されるものではない。例えば、変換指示がデータ変換処理部 3 1 に、ロック獲得指示がロック制御処理部 3 3 に、それぞれ入力されても良い。その際、データ変換処理部 3 1 には、変換指示とともに変更データ 4 1 が入力される。また、ロック粒度判定処理部 3 2 には、更新差分データ、又はロック粒度判定処理部 3 2 で更新差分データが生成される場合には変換前データ及び変換データ 4 1 が入力される。

[0076] さらに、この更新データ制御部 3 0 により扱われる更新差分データ、更新データ 4 1、ロック範囲等のデータは、データそのものでも良く、また、これらのデータが格納された場所（例えば共有メモリ 4 0）のアドレス等、これらのデータを特定するための情報であっても良い。

また、上述の如く、更新データ制御部 3 0 により、更新データ 4 1 の反映指示が入力されると、データ更新部 1 2 及び非同期反映部 2 3、又はデータ更新部 2 2 及び非同期反映部 1 3 は、対応する第 1 DBMS 1 1 及び第 2 DBMS 2 1 に対して、更新データ 4 1 の反映（更新）を行なう。

[0077] このとき、データ更新部 1 2 及び非同期反映部 2 3、又はデータ更新部 2 2 及び非同期反映部 1 3 は、アプリケーション 1 4 又は 2 4 からの更新要求に応じた処理により更新データ 4 1 の反映を行なう。更新データ 4 1 の反映処理の態様としては、例えば、対応する第 1 DBMS 1 1 及び第 2 DBMS 2 1 内の更新処理にかかるロックキーで特定される行に対する、更新データ 4 1 による変更、追加又は削除が挙げられる。

[0078] 以下、第 1 DBMS 1 1 及び第 2 DBMS 2 1 に対して更新データ 4 1 の反映を行なうデータ更新部 1 2 及び非同期反映部 2 3、又はデータ更新部 2 2 及び非同期反映部 1 3 を、まとめて更新部という。

更新データ 4 1 の変更では、更新部は、第 1 DBMS 1 1 及び第 2 DBM

S 2 1における更新対象の領域、つまり、更新対象の行及びロック範囲の列により特定される領域（１以上のブロック）に対して、ロック範囲に対応する更新データ 4 1 内のデータを、例えば上書きする。

[0079] 更新データ 4 1 の追加では、更新部は、第 1 DBMS 1 1 及び第 2 DBMS 2 1 における更新対象の領域、つまりブロックに対して、更新データ 4 1 を追加する。なお、更新データ 4 1 の追加により新たな行が追加される場合には、ロック粒度判定処理部 3 2 によるロック範囲の決定及びロック制御処理部 3 3 によるロック獲得処理は、行なわないこととしても良い。

[0080] 更新データ 4 1 に基づく削除では、更新部は、第 1 DBMS 1 1 及び第 2 DBMS 2 1 における更新対象の領域、つまりブロックに格納されたデータの削除を行なう。なお、この削除に用いられる更新データ 4 1 には、更新処理により削除される対象の領域、つまりロック範囲に決定される領域に、例えば“N u l l”、若しくは削除対象であることを示す任意の文字列等が設定される。

[0081] また、この削除において、更新対象の行全体を削除する場合、ロック範囲には、更新対象の行の中の全ての列が含まれる。つまり、更新対象の行にかかる更新データ 4 1 には、全ての列に、例えば“N u l l”、若しくは削除対象であることを示す任意の文字列等が設定される。或いは、更新対象の行にかかる更新データ 4 1 には、更新対象の行にかかる全ての列が削除対象であることを示す情報が設定されても良い。そして、ロック粒度判定処理部 3 2 により、更新データ 4 1（更新差分データ）に基づいて、全ての列がロック範囲に決定される。

[0082] なお、上述した更新データ 4 1 の反映処理の態様は、一例であり、例えば置換や移動等、既知の種々の処理を含むことができる。更新部は、例えば置換や移動等の処理については、上述した変更、追加及び削除の処理のうちの 1 以上の組み合わせにより行なっても良く、DBMS が備える機能を用いて行なっても良い。

共有メモリ 4 0 は、第 1 仮想 OS 1 0 及び第 2 仮想 OS 2 0 から同時にア

クセス可能なメモリである。

[0083] なお、共有メモリ40としては、RAM (Random Access Memory) 等が挙げられるが、RAM等に替えて、HDD (Hard Disk Drive) やSSD (Solid State Drive) 等の記憶装置を用いても良い。

共有メモリ40は、第1DBMS11の形式に対応した更新データ41-1及び第2DBMS21の形式に対応した更新データ41-2をそれぞれ格納する。また、共有メモリ40は、変換定義ファイル(変換定義情報)42と、ロック管理テーブル(ロック管理情報)43とを格納する。

[0084] 本実施形態においては、更新データ制御部30は、共有メモリ40上に格納されたデータや変換定義ファイル、ロック管理テーブル43等を用いて、上述した処理を行なう。

以下、変換定義ファイル42及びロック管理テーブル43の構成について説明する。

〔1-1-1〕変換定義ファイルの構成

図3は、本実施形態の一例としての変換定義ファイル42の構成例を示す図である。また、図4は、本実施形態の一例としてのデータベースにおける数値データの内部表現形式を示す図であり、数値の“123.45”を、文字列表現(UTF-16)、文字列表現(Shift-JIS)、及びCOBOL表現で表したものである。

[0085] 図5は、本実施形態の一例としてのデータ変換処理部31によるデータ変換処理を説明するための図である。なお、図5に示す例では、更新対象である更新データ41として、1列目に文字データ、2列目に数値データ、3列目に混在データを含むものとする。

図3に示すように、変換定義ファイル42には、更新データ制御部30にアクセスしているOSを示すOS識別子と、データベースの種類を示すDB識別子(DB種別)と、各DB識別子に対応したデータ特性とが含まれる。

[0086] ここで、図3に示す例では、データ特性には、文字データのコード系、後方空白の扱い、数値データの内部表現形式、並びに文字データと数値データ

とが含まれる混在データの列名及び混在データの扱いを示す情報、などが含まれる。

文字データのコード系は、データベースで用いられるUTF-16やShift-JISなどのコード系を示す。

- [0087] 後方空白の扱いは、データベースにおける文字列の操作時に、文字列長が格納先の列長に満たない場合、不足する部分について、文字のコード系で指定されたコードによる空白を設定するか否かを示す。

数値データの内部表現形式は、図4に示すように、文字列表現やCOBOL表現などのデータベース内部での数値データの表現形式を示す。

- [0088] なお、COBOL表現で表されたデータには、小数点の位置は含まれず、最終バイトの上位4ビットには、“c”（正）又は“d”（負）の符号が設定される。

混在データの列名及び取り扱いは、図3に示す“ROW03”のような列名、及び混在データを文字データ又は数値データのいずれのデータとして取り扱うかを示す。なお、列名は、混在データの列番号など列を特定することができるものであれば良い。また、データベースに複数のテーブルが含まれる場合には、混在データの列名とともに、当該列が存在するテーブル名が設定されても良い。

- [0089] データ変換処理部31は、変換定義ファイル42を参照し、変換対象、つまり要求元の更新データ41にかかるOS識別子及びDB識別子に基づいて、変換先、つまり要求元以外のデータベースの各形式に対応する変換方法を決定する。

例えば、図3及び図5に示すように、データ変換処理部31は、変換定義ファイル42を参照し、変換対象、つまり要求元のOS識別子“Win001”における更新データ41に基づいて、変換先のOS識別子“Win002”に対応する形式に変換するための変換方法を決定する。なお、変換方法の決定とは、要求元の更新データ41を変換先の形式に変換するための変換プログラム等の決定をいう。

[0090] そして、データ変換処理部 31 は、決定した変換方法に従って変換を行う。

具体的には、データ変換処理部 31 は、図 5 に示すように、要求元の更新データ 41 のうちの 1 列目の文字データを UTF-16 から Shift-JIS に変換し、2 列目の数値データを文字列表現 (UTF-16) から COBOL 表現に変換し、3 列目の混在データを文字データから数値データに変換する。なお、3 列目の混在データの変換は、変換定義ファイル 42 において指定されたデータ属性に変換されるため、例えば文字データであれば、図 3 に示す文字データのコード系に基づいて変換され、数値データであれば、内部表現形式に基づいて変換される。

[0091] [1-1-2] ロック管理テーブルの構成

図 6 は、本実施形態の一例としてのロック管理テーブル 43 の構成例を示す図であり、図 7 は、本実施形態の一例としてのロック管理テーブル 43 における列パターンを説明するための図である。

図 6 に示す例においては、仮想マシン 1 によりデータ同期される各 DBMS 内のテーブルは、互いに同じスキーマ名、テーブル名を持ち、同じロックキーでテーブル内の行を一意に決定することができるものとする。

[0092] なお、上述の如く、本実施形態においては、第 1 DBMS 11 と、第 2 DBMS 21 とは、互いに異なる構造を持つデータベースである。この場合、例えば、第 1 DBMS 11 及び第 2 DBMS 21 におけるスキーマ名、テーブル名、ロックキー等の対応関係を示すテーブル (情報) を、共有メモリ 40 に備えることができる。具体的には、この対応関係を示すテーブルには、第 1 DBMS 11 のロックキーと第 2 DBMS 21 のロックキーとの間の対応関係、及び第 1 更新データ 41-1 のロック範囲と第 2 更新データ 41-2 のロック範囲との間の対応関係などの情報が格納される。

[0093] この対応関係を示すテーブルにより、更新データ制御部 30 から参照する際の第 1 DBMS 11 及び第 2 DBMS 21 内のテーブル等の名称を、同じものとすることができる。

上述のように、ロック制御処理部 33 は、ロック管理テーブル 43 により、同期対象の全ての DBMS、つまり本実施形態においては第 1 DBMS 11 及び第 2 DBMS 21 に対するデータベースのロックを制御することができる。

[0094] ロック管理テーブル 43 には、第 1 DBMS 11 及び第 2 DBMS 21 のロック状態が設定される。

図 6 に示すように、ロック管理テーブル 43 には、スキーマ名テーブルと、テーブル名テーブルと、ロックキーテーブルと、獲得列パターンテーブルと、獲得スレッド ID テーブルと、待ちスレッド ID テーブルと、待ち列パターンテーブルとが含まれる。

[0095] スキーマ名テーブルには、ロック制御処理部 33 により獲得されたデータ、つまり更新処理にかかるロック範囲が含まれるスキーマ名が設定される。同じスキーマ名を持つロック範囲が全て解放された場合、全てのロック範囲が解放されたスキーマ名は、このスキーマ名テーブルから削除される。

テーブル名テーブルには、ロック制御処理部 33 により獲得されたデータ、つまり更新処理にかかるロック範囲が含まれるテーブル名が設定される。同じテーブル名を持つロック範囲が全て解放された場合、全てのロック範囲が解放されたテーブル名は、このテーブル名テーブルから削除される。

[0096] ロックキーテーブルには、ロック制御処理部 33 により獲得されたデータ、つまり更新処理にかかるロック範囲が含まれる行を一意に決定するロックキーが設定される。同じロックキーを持つロック範囲が全て解放された場合、全てのロック範囲が解放されたロックキーは、このロックキーテーブルから削除される。

獲得列パターンテーブルには、スキーマ名、テーブル名、ロックキーの各テーブルによって特定される行において、ロック制御処理部 33 により獲得されている列パターンが設定される。獲得列パターンは、列パターンを持つデータ、つまりロック範囲が解放された場合に削除される。

[0097] 獲得列パターンは、上述した列パターンの如く、例えばロック制御処理部

3 3による獲得対象列を“1”、非獲得対象列を“0”としたビット列であり、1行×m列（mは1行を形成する列数）のビット列である。

例えば、図7に示すように、データベース内の1行に8列存在し、ロック粒度判定処理部32により1列目、3列目、8列目がロック範囲に決定された場合、獲得列パターンは“10100001”となる。

[0098] 獲得スレッドIDテーブルには、獲得列パターンテーブルに格納された、ロック制御処理部33により獲得されている列パターンを持つ獲得スレッドIDが設定される。獲得スレッドIDは、列パターンを持つロック範囲が解放された場合に削除される。

なお、獲得スレッドIDとは、ロック獲得指示の要求元のOS識別子と、スレッドIDとを組み合わせた識別子（獲得識別子）である。

[0099] 待ちスレッドIDテーブルには、獲得列パターンテーブルに格納された列パターンの獲得を待機中（待ち合わせ中）のスレッドのIDが、待ちスレッドIDとして設定される。待ちスレッドIDは、待ち状態が解除された場合に削除される。

なお、待ちスレッドIDとは、ロック獲得指示の要求元のOS識別子と、スレッドIDとを組み合わせた識別子（獲得識別子）である。

[0100] 待ち列パターンテーブルには、待ちスレッドIDテーブルに格納された待ち合わせ中の待ちスレッドIDにおける、ロック範囲、つまり処理対象の列パターンが設定される。待ち列パターンは、待ち状態が解除された場合に削除される。

ロック制御処理部33は、上述したロック管理テーブル43に基づいて、更新処理にかかるロック範囲についてロックが獲得可能か否かを判断する。

[0101] 具体的には、ロック制御処理部33は、ロック範囲、つまり処理対象の列パターンと、獲得列パターンテーブル内の全ての獲得列パターンとの列毎のANDを取り、ANDの結果、全ての列が“0”であればロック範囲のロックが獲得可能であると判断する。一方、ロック制御処理部33は、ANDの結果、“0”以外の列があればロック範囲のロックが獲得不可能であると判

断し、上述した待ち合わせの処理を行なう。

[0102] そして、ロック制御処理部 33 は、ロックが獲得可能であると判断すると、指示元の OS 識別子、ロックキー、獲得列パターン、及びスレッド ID を、ロック獲得情報としてロック管理テーブル 43 に設定する。

一方、ロック制御処理部 33 は、ロックが獲得不可能であると判断すると、待ち列パターン、及び待ちスレッド ID を、更新処理にかかるロック範囲がロック待ち状態であることを示す待ち情報としてロック管理テーブル 43 に設定する。

[0103] また、データ更新部 12 又は 22 より、ロック解放指示を入力されると、ロック制御処理部 33 は、ロック解放指示にかかるロック範囲をロック獲得列パターンから削除するとともに、ロック範囲の解放によりロックの獲得が可能な待ち列パターンがあるかを判定する。そして、ロック獲得可能な待ち列パターンがあった場合、ロック制御処理部 33 は、当該待ち列パターンにかかる指示元 OS 識別子やロック範囲について、新たなロック獲得指示として処理する。

[0104] なお、待ち列パターンによっては、複数の獲得列パターンが解放されないと、更新処理にかかるロック範囲のロックを獲得することができない場合がある。この場合、ロック制御処理部 33 は、この待ちスレッド ID 及び待ち列パターンを、これら複数の獲得列パターンの獲得スレッド ID にそれぞれ対応付けて登録を行なう。そして、ロック制御処理部 33 は、複数の獲得列パターンにおけるロックが解放されたものについて、待ち列パターンにかかるロック範囲の一部についてロックを獲得させる。このように、待ち列パターンにかかるロック範囲について分割してロックを獲得させ、待ち列パターンにかかるロック範囲の全てについてロックが獲得できたときに、ロック制御処理部 33 は、当該待ち列パターンにかかる反映指示を出力する。

[0105] 上述のように、ロック制御処理部 33 は、1 つのロック管理テーブル 43 、つまり 1 つのロック管理機構により、第 1 DBMS 11 及び第 2 DBMS 21 の複数のデータベースのロックを制御することができる。

上述のことから、更新データ制御部 30 と共有メモリ 40 とは、第 1 仮想 OS 10 及び第 2 仮想 OS 20 間のデータベースの同期制御を行なうデータ同期制御装置として機能するといえる。

[0106] 〔 1 - 2 〕 一実施形態の動作

次に、上述の如く構成された仮想マシン 1 によるデータ同期処理を、図 8 に示すシーケンス図を参照しながら説明するとともに、仮想マシン 1 の各部の詳細な処理については、図 9 ～図 15 に示すフローチャートを参照しながら説明する。

図 8 は、本実施形態の一例としての仮想マシン 1 によるデータ同期の処理を説明するためのシーケンス図である。

[0107] 図 9 は、本実施形態の一例としてのデータ更新部 12 又は 22 の処理を説明するためのフローチャートであり、図 10 は、更新データ制御部 30 のデータ変換及びロック獲得の処理を説明するためのフローチャートである。

また、図 11 は、本実施形態の一例としてのデータ変換処理部 31 のデータ変換の処理を説明するためのフローチャートであり、図 12 は、ロック粒度判定処理部 32 のロック範囲決定の処理を説明するためのフローチャートである。

[0108] さらに、図 13 は、本実施形態の一例としてのロック制御処理部 33 によるロック獲得の処理を説明するためのフローチャートであり、図 14 は、更新データ制御部 30 のロック解放の処理を説明するためのフローチャートである。

また、図 15 は、本実施形態の一例としてのロック制御処理部 33 によるロック解放の処理を説明するためのフローチャートであり、図 16 は、非同期反映部 13 又は 23 の処理を説明するためのフローチャートである。

[0109] 以下、第 1 仮想 OS 10 のアプリケーション 14 が第 1 DBMS 11 に対する更新を行なう場合の仮想マシン 1 による処理について説明する。

はじめに、図 8 に示すように、第 1 仮想 OS 10 のアプリケーション 14 による第 1 DBMS 11 への更新処理の要求が発生し、当該要求がデータ更

新部 12 へ出力される（ステップ T 1）。

[0110] 更新要求が入力されると、図 9 に示す如く、データ更新部 12 により、アプリケーション 14 から第 1 DBMS 11 に対して行なわれる更新処理にかかる更新前データ及び更新後データ 41-1 が取得される（ステップ S 1）。なお、更新前データ及び更新後データ 41-1 は、アプリケーション 14 によりアクセス可能なメモリ上に格納されているため、データ更新部 12 は、アプリケーション 14 経由で当該メモリ上から更新前データ及び更新後データ 41-1 を取得する。

[0111] そして、データ更新部 12 により、更新前データ及び更新後データ 41-1 から更新差分データが作成される（ステップ S 2, T 2）。

次いで、データ更新部 12 により、更新後データ 41-1 及び更新差分データと、変換指示及びロック獲得指示とが、更新データ制御部 30 へ出力される（ステップ S 3, T 3）。データ更新部 12 は、上記各データ及び各指示を更新データ制御部 30 へ出力すると、更新データ制御部 30 における処理（図 10 のステップ S 11 ~ S 23）が完了し、更新データ制御部 30 から応答が入力されるまで、待ち合わせを行なう（ステップ S 4）。

[0112] 更新データ制御部 30 では、図 10 に示す如く、データ更新部 12 から変換指示及びロック獲得指示を入力されると（ステップ S 11）、データ変換処理部 31 が指示待ちであるか否かが判断される（ステップ S 12）。

データ変換処理部 31 が指示待ちである場合（ステップ S 12 の Yes ルート）、更新データ制御部 30 により、変換指示とともに、更新後データ 41-1、並びに指示元の OS 識別子及び DB 識別子がデータ変換処理部 31 に入力され、データ変換処理部 31 が起動される（ステップ S 14）。データ変換処理部 31 では、変換指示に対する所定の処理（ステップ T 4 及び後述する図 11 のステップ S 31 ~ S 37）が実行される。

[0113] 一方、データ変換処理部 31 が他の処理を行なっている場合等、指示待ちではない場合（ステップ S 12 の No ルート）、更新データ制御部 30 により、データ変換処理部 31 が指示待ちになるまで待ち合わせが行なわれ（ス

テップS 1 3)、指示待ちになった場合にステップS 1 4に移行する。

また、更新データ制御部30では、データ更新部12から変換指示及びロック獲得指示を入力されると(ステップS 1 1)、ロック粒度判定処理部32が指示待ちであるか否かが判断される(ステップS 1 5)。

[0114] ロック粒度判定処理部32が指示待ちである場合(ステップS 1 5のYesルート)、更新データ制御部30により、更新差分データがロック粒度判定処理部32に入力され、ロック粒度判定処理部32が起動される(ステップS 1 7)。ロック粒度判定処理部32では、ロック粒度(ロック範囲)の決定にかかる所定の処理(ステップT 5及び後述する図12のステップS 4 1～S 4 3)が実行される。

[0115] 一方、ロック粒度判定処理部32が他の処理を行なっている場合等、指示待ちではない場合(ステップS 1 5のNoルート)、更新データ制御部30により、ロック粒度判定処理部32が指示待ちになるまで待ち合わせが行なわれ(ステップS 1 6)、指示待ちになった場合にステップS 1 7に移行する。

なお、上述したステップS 1 2～S 1 4の処理と、ステップS 1 5～S 1 7の処理とは、並行して行なわれても良いし、いずれかの処理が先に実行されても良い。

[0116] 更新データ制御部30では、図10に示すように、ステップS 1 2～S 1 4、及びステップS 1 5～S 1 7の両方の処理が完了するまで待ち合わせが行なわれる(ステップS 1 8)。

ここで、上述したステップS 1 4における、データ変換処理部31の動作について説明する。

[0117] データ変換処理部31では、図11に示す如く、更新データ制御部30により、更新後データ41-1、並びに指示元のOS識別子及びDB識別子が入力データとして入力される(ステップS 3 1)。

そして、データ変換処理部31により、予め設定された変換定義ファイル42が参照され、指示元のOS識別子及びDB識別子に対応する変換方法が

決定される（ステップS 3 2，図 3 参照）。

[0118] 次いで、データ変換処理部 3 1 により、ステップS 3 2 で決定された変換方法に従って、更新後データ（更新データ）4 1 - 1 が第 2 DBMS 2 1 の形式に対応した更新データ（変換後データ）4 1 - 2 に変換される（ステップS 3 3，T 4）。

なお、データ変換処理部 3 1 では、ステップS 3 3 において、データ同期対象の他の DBMS の種別数分、つまりデータ同期対象の他の DBMS の種別毎に、更新データ 4 1 - 1 が変換される。また、データ変換処理部 3 1 では、変換定義ファイル 4 2 中のデータ表現形式が異なる部分を、データ同期対象の他の DBMS の種別に対応した形式に変換される。つまり、更新データ 4 1 - 1 の項目（列）のデータ表現形式と、データ同期対象の他の DBMS における項目（列）のデータ表現形式とが異なる場合等に、データ変換処理部 3 1 により、更新データ 4 1 - 1 の該当項目（列）の変換が行なわれる。

[0119] 更新データ 4 1 - 1 の変換が完了すると、データ変換処理部 3 1 により、エラー判定が行なわれる（ステップS 3 4）。エラー判定では、変換定義ファイル 4 2 の異常や更新データ 4 1 の異常、ハードウェア異常等の原因、或いはその他の原因による変換エラーの有無が確認される。

エラー判定の結果、変換エラーが無い場合（ステップS 3 4 の N o ルート）、データ変換処理部 3 1 により、第 1 DBMS 1 1 及び第 2 DBMS 2 1 における更新データ 4 1 が含まれる行を特定するロックキーが生成される（ステップS 3 5）。

[0120] 一方、エラー判定の結果、変換エラーがあった場合（ステップS 3 4 の Y e s ルート）、データ変換処理部 3 1 により、変換エラーに基づいてエラー情報が設定される（ステップS 3 6）。

そして、データ変換処理部 3 1 により、更新データ制御部 3 0 に対して処理完了が通知され、データ変換処理部 3 1 は、指示待ち状態に移行する（ステップS 3 7）。なお、変換エラーがあり、ステップS 3 6 においてエラー

情報が設定された場合には、更新データ制御部 30 に対して、処理完了通知とともにエラー情報が通知される。

[0121] 次いで、上述した図 10 に示すステップ S 17 における、ロック粒度判定処理部 32 の動作について説明する。

ロック粒度判定処理部 32 では、図 12 に示す如く、更新データ制御部 30 により、更新差分データが入力データとして入力される（ステップ S 41）。

そして、ロック粒度判定処理部 32 により、更新差分データから、変更のあった行のうちのロック対象となる列、つまりロック範囲が決定される（ステップ S 42, T 5）。

[0122] 最後に、データ変換処理部 31 により、ロック範囲が設定され、更新データ制御部 30 に対して、処理完了通知とともに設定されたロック範囲が通知される。そして、データ変換処理部 31 は、指示待ち状態に移行する（ステップ S 43）。

更新データ制御部 30 では、上述したデータ変換処理部 31 及びロック粒度判定処理部 32 の両方の処理が完了すると、すなわち両方の処理完了の通知が入力されると、図 10 に示す如く、ロック制御処理部 33 が指示待ちであるか否かが判断される（ステップ S 19）。

[0123] ロック制御処理部 33 が指示待ちである場合（ステップ S 19 の Yes ルート）、更新データ制御部 30 により、ロック獲得指示とともに、変換後データ 41-1、及びロック範囲がロック制御処理部 33 に入力され、ロック制御処理部 33 が起動される（ステップ S 21）。ロック制御処理部 33 では、ロック獲得指示に対する所定の処理（ステップ T 6, T 7 及び T 11、並びに後述する図 13 のステップ S 51 ~ S 62）が実行される。

[0124] 一方、ロック制御処理部 33 が他の処理を行なっている場合等、指示待ちではない場合（ステップ S 19 の No ルート）、更新データ制御部 30 により、ロック制御処理部 33 が指示待ちになるまで待ち合わせが行なわれ（ステップ S 20）、指示待ちになった場合にステップ S 21 に移行する。

またここで、上述したステップS 2 1における、ロック制御処理部 3 3の動作について説明する。

[0125] ロック獲得指示が入力されたロック制御処理部 3 3では、図 1 3に示す如く、変換後データ 4 1－2 及びロック粒度判定処理部 3 2で決定されたロック範囲が入力データとして入力される（ステップS 5 1）。

そして、ロック制御処理部 3 3により、ロック管理テーブル 4 3が参照され、データ変換処理部 3 1により生成された更新処理にかかるロックキーと同一のロックキーがロック管理テーブル 4 3に含まれるか否かが判定される（ステップS 5 2, S 5 3）。

[0126] 更新処理にかかるロックキーと同一のロックキーがロック管理テーブル 4 3に含まれないと判定された場合（ステップS 5 3のN o ルート）、ロック制御処理部 3 3により、更新処理にかかる変換データ 4 1のロック範囲は、“ロック未”の状態、つまりロックされていない状態であると判断される（ステップS 5 4）。

そして、ロック制御処理部 3 3により、更新処理にかかる更新データ 4 1－2のロックキー、獲得列パターンで示されるロック範囲、指示元のOS識別子及びスレッドIDが、ロック管理テーブル 4 3に設定される（ステップS 5 5, 図6参照）。すなわち、ステップS 5 5においてロック制御処理部 3 3により、更新処理にかかるロック範囲がロックされていることを示す上述した情報が、ロック獲得情報としてロック管理テーブル 4 3に設定される。

[0127] 更新処理にかかるロック範囲がロックされると、ロック制御処理部 3 3により、更新データ 4 1－2について第2DBMS 2 1への反映指示が設定される（ステップS 5 6）。

最後に、ロック制御処理部 3 3により、更新データ制御部 3 0に対して、処理完了通知とともに変換後データ 4 1－2 及び設定された反映指示が通知される。そして、ロック制御処理部 3 3は、指示待ち状態に移行する（ステップS 5 7）。

[0128] 一方、ステップS 5 3において、更新処理にかかるロックキーと同一のロックキーがロック管理テーブル4 3に含まれると判定された場合（ステップS 5 3のY e sルート）、ロック制御処理部3 3により、ロック管理テーブル4 3内の獲得されたロック範囲と、変換後データ4 1－2にかかるロック範囲とに重なりがあるか否かが判定される（ステップS 5 8, S 5 9）。

[0129] すなわち、ステップS 5 8, S 5 9では、ロック制御処理部3 3により、ロック管理テーブル4 3に含まれるロックキーに対応したロック範囲と、更新処理にかかるロック範囲との間に共通した範囲があるか否かが判定される。この判定は、例えば、上述したように、更新処理にかかるロック範囲、つまり処理対象の列パターンと、獲得列パターンテーブル内の全ての獲得列パターンとの列毎のANDを取ることにより行なわれる。

[0130] 獲得されたロック範囲と、変換後データ4 1－2にかかるロック範囲とに重なりがないと判定された場合（ステップS 5 9のN oルート）、ロック制御処理部3 3により、更新処理にかかるロック範囲は、“ロック未”の状態であると判断され、上述したステップS 5 4の処理に移行する。

一方、獲得されたロック範囲と、更新データ4 1にかかるロック範囲とに重なりがあると判定された場合（ステップS 5 9のY e sルート）、ロック制御処理部3 3により、更新処理にかかる変換データ4 1のロック範囲は、“ロック済”の状態、つまり既にロックされている状態であると判断される（ステップS 6 0）。

[0131] そして、ロック制御処理部3 3により、更新処理にかかる第1及び第2のデータベースのロック、つまり更新処理にかかるロック範囲がロック待ち状態であることを示す待ち情報が、ロック管理テーブル4 3に設定される（ステップS 6 1, T 1 1）。

すなわち、ステップS 6 1においてロック制御処理部3 3により、待ち列パターンで示されるロック範囲及び待ちスレッドIDが、ロック管理テーブル4 3内のロック獲得待ちの獲得スレッドIDに対応付けて設定される（図6参照）。

[0132] また、ロック制御処理部 33 により、更新処理にかかるロック範囲のロックが獲得できなかったことを示すエラー情報が設定され（ステップ S 62）、ステップ S 57 に移行する。なお、ロック制御処理部 33 により、ステップ S 62 においてエラー情報が設定された場合、更新データ制御部 30 に対して通知される処理完了通知には、更新データ 41-2 及び反映指示に替えてエラー情報が付加される。

[0133] 更新データ制御部 30 では、上述したロック制御処理部 33 の処理が完了すると、すなわち処理完了の通知が入力されると、図 10 に示す如く、エラーの有無が判定され、復帰情報が作成される（ステップ S 22）。

ここで、更新データ制御部 30 によるエラーの有無の判定は、データ変換処理部 31 やロック制御処理部 33 から入力された処理完了通知に、エラー情報が付加されているか否かによって行なわれる。

[0134] また、復帰情報は、指示元の第 1 仮想 OS 10 又は／及び同期対象の第 2 仮想 OS 20 への通知情報であり、更新データ 41 の各 DBMS への反映指示や、各処理部 31～33 からのデータ変換及びロック獲得の通知、エラー情報が入力された場合に通知するエラー情報などが含まれる。

なお、変換定義ファイル 42 の異常によりデータ変換処理部 31 において変換が行なわれなかった等、更新処理が継続できないエラーが発生した場合には、復帰情報に所定のメッセージ等が設定されても良い。また、ロック制御処理部 33 において、更新処理にかかるロック範囲のロックが獲得できず待ち情報となった場合等、更新処理が一時的に停止するエラーが発生した場合にも、復帰情報に所定のメッセージ等が設定されても良い。

[0135] そして、更新データ制御部 30 により、指示元の第 1 仮想 OS 10 又は／及び同期対象の第 2 仮想 OS 20 に対して、復帰情報が出力され、これらの OS の待ち合わせ状態又は指示待ち状態が解除される（ステップ S 23, T7）。つまり、指示元の第 1 仮想 OS 10 への更新データ 41-1 の反映指示と、同期対象の第 2 仮想 OS 20 への変換後データ 41-2 の反映指示とが、同じ若しくはほぼ同じタイミングで出力される。

[0136] 復帰情報（反映指示）の入力により待ち合わせ状態が解除されたデータ更新部 12 では、図 9 に示す如く、復帰情報にエラー情報が含まれているかが判定される（ステップ S 5）。

エラー判定の結果、エラー情報が含まれていない場合（ステップ S 5 の No ルート）、データ更新部 12 により、自身の OS、つまり第 1 仮想 OS 10 内の第 1 DBMS 11 に対して、更新データ 41-1 を用いて更新が行われる（ステップ S 6, T 8）。

[0137] そして、データ更新部 12 により、更新データ制御部 30 に対して、更新処理にかかるロック範囲についてロック解放指示が出力され（ステップ S 7, T 10）、更新データ制御部 30 においてロック解放指示に対する処理（ステップ T 12 並びに後述する図 14 及び図 15 のステップ S 63 ~ S 74）が実行される。

その後、データ更新部 12 により、アプリケーション 14 に対して更新処理の完了が通知され、アプリケーション 14 の処理が復帰し（ステップ S 8）、データ更新部 12 の処理が終了する。

[0138] 一方、ステップ S 5 におけるエラー判定の結果、エラー情報が含まれている場合（ステップ S 5 の Yes ルート）、データ更新部 12 により、更新処理の実行結果にエラー情報が設定され（ステップ S 9）、実行結果がアプリケーション 14 に出力され、ステップ S 8 の処理に移行する。

また、復帰情報（反映指示）の入力により指示待ち状態が解除された第 2 仮想 OS 20 の非同期反映部 23 では、図 16 に示す如く、更新データ制御部 30 からの更新データ（変換後データ） 41-2 が入力データとして入力される（ステップ S 81）。

[0139] なお、非同期反映部 23 は、例えば、更新データ制御部 30 から、変換後データ 41-2 が格納された共有メモリ 40 のアドレスを入力され、入力されたアドレスにアクセスすることにより、変換後データ 41-2 を取得することができる。

そして、非同期反映部 23 により、自身の OS、つまり第 2 仮想 OS 20

内の第2DBMS 21に対して、更新データ41-2を用いて更新が行なわれる（ステップS 82, T 9）。

- [0140] 第2DBMS 21への更新データ41-2の更新が完了すると、非同期反映部23は指示待ち状態に移行し（ステップS 83）、非同期反映部23の処理が終了する。

次に、図9に示す上述したステップS 7における、ロック解放指示に対する更新データ制御部30の動作について説明する。

更新データ制御部30では、図14に示す如く、データ更新部12からロック解放指示を入力されると（ステップS 63）、ロック制御処理部33が指示待ちであるか否かが判断される（ステップS 64）。

- [0141] ロック制御処理部33が指示待ちである場合（ステップS 64のY e s ルート）、更新データ制御部30により、ロック解放指示とともに、ロック解放指示にかかるロックキー、指示元のOS識別子及びスレッドIDがロック制御処理部33に入力され、ロック制御処理部33が起動される（ステップS 66）。ロック制御処理部33では、ロック解放指示に対する所定の処理（ステップT 12及び後述する図15のステップS 69～S 74）が実行される。

- [0142] 一方、ロック制御処理部33が他の処理を行なっている場合等、指示待ちではない場合（ステップS 64のN o ルート）、更新データ制御部30により、ロック制御処理部33が指示待ちになるまで待ち合わせが行なわれ（ステップS 65）、指示待ちになった場合にステップS 66に移行する。

次に、上述したステップS 66における、ロック制御処理部33の動作について説明する。

- [0143] ロック解放指示が入力されたロック制御処理部33では、図15に示す如く、ロックキー、指示元のOS識別子及びスレッドIDが入力データとして入力される（ステップS 69）。

そして、ロック制御処理部33により、ロック管理テーブル43が参照され、ロック獲得状態にある、上述した入力データ（ロックキー、指示元のO

S識別子及びスレッドID)にかかるロック範囲(獲得列パターン)が削除される。なお、このとき、ロック制御処理部33により、削除したロック範囲の列パターンが記憶される(ステップS70)。

[0144] 次いで、ロック制御処理部33により、ロック獲得待ち状態の指示の中に、削除されたロック範囲によりロックの獲得が可能なものがあるか否かが判定される(ステップS71)。なお、ロック獲得待ち状態の指示は、ロック管理テーブル43内に設定されている待ちスレッドID及び待ち列パターンによって示される(図6参照)。

ロック獲得待ち状態の指示の中に、ロックの獲得が可能なものがあると判定された場合(ステップS72のYesルート)、ロック制御処理部33により、ロック獲得可能な指示について、再度、ロック制御処理部33による上述したロック獲得の処理が行なわれる(ステップS73)。つまり、ロック制御処理部33により、ロック獲得可能な指示について、再度、ロック獲得指示、変換データ41-2、及びロック範囲を入力データとして、上述したステップS51~S62の処理が実行される。

[0145] 次いで、ロック制御処理部33により、更新データ制御部30に対して、処理の完了が通知される。そして、ロック制御処理部33は、指示待ち状態に移行する(ステップS74)。

また、ロック獲得待ち状態の指示の中に、ロックの獲得が可能なものがないと判定された場合(ステップS72のNoルート)にも、ロック制御処理部33は、ステップS74の指示待ち状態に移行する。

[0146] 更新データ制御部30では、上述したロック制御処理部33の処理が完了すると、すなわち処理完了の通知が入力されると、図14に示す如く、エラーの有無が判定され、復帰情報が作成される(ステップS67)。

ここで、更新データ制御部30によるエラーの有無の判定は、ロック制御処理部33から入力された処理完了通知に、エラー情報が付加されているか否かによって行なわれる。

[0147] また、復帰情報は、指示元の第1仮想OS10への通知情報であり、エラ

一情報が入力された場合に通知するエラー情報などが含まれる。

そして、更新データ制御部 30 により、指示元の第 1 仮想 OS 10 に対して復帰情報が出力され、第 1 仮想 OS 10 の処理が復帰し（ステップ S 68）、更新データ制御部 30 の処理が終了する。

[0148] 上述のように、本実施形態の一例としての更新データ制御部 30 によれば、データ変換処理部 31 により、第 1 DBMS 11 に対して行なわれる更新処理にかかる第 1 更新データ 41-1 が取得される。また、データ変換処理部 31 により、予め設定された変換定義ファイル（変換定義情報）42 に基づいて、第 1 更新データ 41-1 が第 2 DBMS 21 の形式に対応した第 2 更新データ 42-2 に変換される。

[0149] これにより、互いに異なる構造を持つ第 1 DBMS 11 及び第 2 DBMS 21 間で更新データ 41 を同期させる際に、アプリケーション 14 は従来のまま、更新データ制御部 30 を備えるだけで、同期させる更新データ 41 を第 1 DBMS 11 及び第 2 DBMS 21 巻で変換することができる。つまり、更新データ制御部 30 により、アプリケーション 14 や第 1 DBMS 11 及び第 2 DBMS 21 は従来のまま、互いに異なる構造を持つ第 1 DBMS 11 及び第 2 DBMS 21 間で更新データ 41 のみを同期させることができる。

[0150] 従って、更新データ制御部 30 により、アプリケーション 14 のプログラムの変更にかかる保守工数増加を抑えることができ、図 17 に示す場合より簡易に、同期させる更新データ 41 を第 1 DBMS 11 及び第 2 DBMS 21 間で変換することができる。

また、本実施形態の一例としての更新データ制御部 30 によれば、ロック粒度判定処理部 32 により、第 1 更新データ 41-1 と、更新処理が行なわれる前の更新前データとに基づいて、ロック範囲が決定される。このとき、第 1 DBMS 11 及び第 2 DBMS 21 におけるテーブルの列を最小単位として、ロック範囲が決定される。また、ロック制御処理部 33 により、ロック粒度判定処理部 32 により決定されたロック範囲に基づいて、第 1 DBM

S 1 1 及び第 2 DBMS 2 1 がロックされる。

[0151] これにより、第 1 DBMS 1 1 及び第 2 DBMS 2 1 間で更新データ 4 1 を同期させる際に、従来の行単位よりも小さい、1 行×1 列のブロック単位で、第 1 DBMS 1 1 及び第 2 DBMS 2 1 のロック範囲を決定することができる。従って、更新対象の行及びロック範囲内の列により特定される領域（1 以上のブロック）によってロック範囲が決定されるため、ロック範囲を最小限に抑えることができる。

[0152] また、例えば、ビューによってあるテーブル内の 1 以上のブロックを参照する場合、他のアプリケーション等により、ビューで参照していないブロックに対する更新が発生しても、更新が発生した 1 以上のブロックのみがロック範囲としてロックされるため、ビュー経由の参照に対する影響を抑えることができる。すなわち、データベースの同一資源に対する同時更新・アクセスの発生を最小限に抑えることができる。

[0153] さらに、何らかの原因により、第 1 DBMS 1 1 及び第 2 DBMS 2 1 のロック範囲がロックされる時間が長引いた場合でも、行単位より細かなブロック単位でロックが行なわれるため、ロックによりデータベース上で排他制御が行なわれる範囲を最小限に抑えることができる。すなわち、1 つのデータベースのロック状態を、他のデータベースのロック状態と一致させて排他制御させることができる。従って、データベース資源の共用度を向上させることができる。

[0154] また、本実施形態の一例としての更新データ制御部 3 0 によれば、ロック制御処理部 3 3 により、ロック粒度判定処理部 3 2 により決定されたロック範囲に基づいて、第 1 DBMS 1 1 及び第 2 DBMS 2 1 がロックされる。そして、ロック制御処理部 3 3 により、第 1 DBMS 1 1 及び第 2 DBMS 2 1 が備えられるそれぞれの OS に対して、それぞれ第 1 更新データ 4 1-1 及び第 2 更新データ 4 1-2 を用いた更新が指示される。このとき、ロック制御処理部 3 3 により、第 2 仮想 OS 2 0 に、第 1 仮想 OS 1 0 及び第 2 仮想 OS 2 0 からアクセス可能な共用メモリ 4 0 を介して、第 2 更新データ

4 1 - 2 が出力される。

[0155] これにより、第 1 DBMS 1 1 及び第 2 DBMS 2 1、つまり複数の異なる種類のデータベースについて、1つのロック管理機構でデータ同期の制御を行なうことができる。また、本実施形態におけるデータ同期は、共有メモリを介して行なうことができるため、図 1 8 及び図 1 9 を用いて説明したネットワーク経由のデータ同期と比べて、データ漏洩のリスクを大幅に低減することができる。

[0156] また、上述の如く、本実施形態におけるデータ同期は、共有メモリを介して行なわれるため、ネットワークが一時的に使用できない場合等、ネットワークに関する障害や遅延の影響を受けずに、複数のデータベース間で確実にデータ同期を行なうことができる。

さらに、第 1 DBMS 1 1 及び第 2 DBMS 2 1、つまり複数の異なる種類のデータベース間において、リアルタイムで更新データ 4 1 の自動同期（更新）を行なうことが可能となる。従って、データ同期が共有メモリを介して行なわれることにより、図 1 8 及び図 1 9 を用いて説明したネットワーク経由のデータ同期と比べて、更新データ 4 1 の反映にかかるデータベース間のタイムラグを大幅に低減することができる。

[0157] [2] その他

以上、本発明の好ましい実施形態について詳述したが、本発明は、かかる特定の実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲内において、種々の変形、変更して実施することができる。

例えば、本実施形態においては、仮想マシン 1 が共有メモリ 4 0 と、共有メモリ 4 0 にアクセス可能な第 1 仮想 OS 1 0 及び第 2 仮想 OS 2 0 を備える構成について述べているが、これに限定されるものではない。例えば 1 以上の情報処理装置（装置）により構成される情報処理システム（システム）が、共用メモリ 4 0 と、共有メモリ 4 0 にアクセス可能な複数の OS を備える構成でもよい。

[0158] また、本実施形態においては、第 1 DBMS 1 1 及び第 2 DBMS 2 1 の

2つのデータベース間のデータ同期について述べているが、これに限定されるものではない。例えば、データ同期対象のDBMSが3つ以上ある場合、更新処理にかかる更新データの変換は、データ変換処理部31により、更新処理が行なわれるDBMS以外のデータ同期対象のデータベース毎に行なわれても良い。また、データ同期対象の更新処理にかかるロック範囲のロックは、ロック制御処理部33により、データ同期対象の全てのDBMSについて行なわれても良い。さらに、DBMSの更新は、ロック制御処理部33により、更新データ又は変換された更新データを用いてデータ同期対象の全てのデータベースがそれぞれ更新されても良い。

[0159] さらに、上述した図9に示すステップS2の処理は、図12に示すステップS41の処理において実行されても良い。つまり、データ更新部12は、図9に示すステップS2において更新差分データを作成せず、ステップS3における更新差分データの出力に替えて更新前データを出力しても良い。このとき、図10に示すステップS17では、更新データ制御部30は、更新差分データに替えて更新前データ及び更新後データ41-1とともに、ロック粒度判定処理部32を起動すれば良い。また、ロック粒度判定処理部32は、図12に示すステップS41において、更新前データ及び更新後データ41-1を入力データとして入力されるとともに、更新前データ及び更新後データ41-1から更新差分データを生成すれば良い。

[0160] なお、データ更新部12及び22、非同期反映部13及び23、更新データ制御部30、データ変換処理部31、ロック粒度判定処理部32、並びにロック制御処理部33としての機能を実現するためのプログラム（データ同期プログラム）は、例えばフレキシブルディスク、CD（CD-ROM、CD-R、CD-RW等）、DVD（DVD-ROM、DVD-RAM、DVD-R、DVD+R、DVD-RW、DVD+RW、HD DVD等）、ブルーレイディスク、磁気ディスク、光ディスク、光磁気ディスク等の、コンピュータ読取可能な記録媒体に記録された形態で提供される。そして、コンピュータはその記録媒体からプログラムを読み取って内部記憶装置または外

部記憶装置に転送し格納して用いる。また、そのプログラムを、例えば磁気ディスク、光ディスク、光磁気ディスク等の記憶装置（記録媒体）に記録しておき、その記憶装置から通信回線を介してコンピュータに提供するようにしても良い。

[0161] データ更新部 12 及び 22、非同期反映部 13 及び 23、更新データ制御部 30、データ変換処理部 31、ロック粒度判定処理部 32、並びにロック制御処理部 33 としての機能を実現する際には、内部記憶装置（本実施形態では情報処理装置、情報処理システムが備えるメモリ又は HDD 等の I/O 装置）に格納されたプログラムがコンピュータのマイクロプロセッサ（本実施形態では情報処理装置、情報処理システムが備える CPU）によって実行される。このとき、記録媒体に記録されたプログラムをコンピュータが読み取って実行するようにしても良い。

[0162] なお、本実施形態において、コンピュータとは、ハードウェアとオペレーティングシステムとを含む概念であり、オペレーティングシステムの制御の下で動作するハードウェアを意味している。また、オペレーティングシステムが不要でアプリケーションプログラム単独でハードウェアを動作させるような場合には、そのハードウェア自体がコンピュータに相当する。ハードウェアは、少なくとも、CPU 等のマイクロプロセッサと、記録媒体に記録されたコンピュータプログラムを読み取るための手段とをそなえており、本実施形態においては、情報処理装置、情報処理システムがコンピュータとしての機能を有しているのである。

符号の説明

[0163] 1, 100 仮想マシン
 10, 110 第 1 仮想 OS
 11, 111 第 1 DBMS, 第 1 のデータベース
 12, 22 データ更新部
 13, 23 非同期反映部
 14, 24, 113, 114 アプリケーション

- 2 0, 1 2 0 第 2 仮想 OS
- 2 1, 1 2 1 第 2 DBMS, 第 2 のデータベース
- 3 0 更新データ制御部
- 3 1 データ変換処理部 (データ変換部)
- 3 2 ロック粒度判定処理部 (ロック範囲決定部)
- 3 3 ロック制御処理部 (ロック制御部)
- 4 0 共用メモリ
- 4 1 - 1 更新データ (第 1 更新データ, 更新後データ, 第 1 DBMS
形式データ)
- 4 1 - 2 更新データ (第 2 更新データ, 変換後データ, 第 2 DBMS
形式データ)
- 4 2 変換定義ファイル
- 4 3 ロック管理テーブル
- 1 1 2, 1 2 2 通信部
- 1 1 5 トリガー機能
- 1 1 6, 1 2 6 データ変換部
- 1 1 7 更新データ
- 1 1 8, 1 2 8 データ同期部

請求の範囲

- [請求項1] 第1のデータベースに対して行なわれる更新処理にかかる第1更新データを取得するステップと、
- 予め設定された変換定義情報に基づいて、前記第1更新データを第2のデータベースの形式に対応した第2更新データに変換するステップと、
- 前記第1更新データと、前記更新処理が行なわれる前の更新前データとに基づいて、ロック範囲を決定するステップと、
- 決定されたロック範囲に基づいて、前記第1及び第2のデータベースをロックするステップと、
- 前記第1及び第2更新データを用いて、前記第1及び第2のデータベースをそれぞれ更新するステップと、をそなえることを特徴とする、データ同期方法。
- [請求項2] 前記第1及び第2のデータベースをロックするステップが、
- 前記ロック範囲に基づいて、前記第1及び第2のデータベースのロック状態が設定されたロック管理情報から、前記第1及び第2のデータベースにおける前記ロック範囲がロック済みであるか否かを判定するステップと、
- 前記ロック範囲がロック済みではないと判定された場合に、前記ロック管理情報において、前記更新処理にかかるロック範囲がロックされていることを示すロック獲得情報を設定するステップと、をそなえることを特徴とする、請求項1記載のデータ同期方法。
- [請求項3] 前記第1及び第2のデータベースをロックするステップが、
- 前記ロック範囲がロック済みであると判定された場合に、前記ロック管理情報において、前記更新処理にかかるロック範囲がロック待ち状態であることを示す待ち情報を設定するステップをそなえることを特徴とする、請求項2記載のデータ同期方法。
- [請求項4] 前記ロック管理情報は、前記第1及び第2のデータベースに共通す

る前記更新処理にかかる行を特定するロックキーを含み、

前記第 1 更新データを前記第 2 更新データに変換するステップが、
前記更新処理にかかるロックキーを生成するステップをそなえ、

前記ロック範囲がロック済みであるか否かを判定するステップが、
生成された前記更新処理にかかるロックキーと同一のロックキーが前
記ロック管理情報に含まれるか否かを判定し、前記更新処理にかかる
ロックキーが前記ロック管理情報に含まれないと判定された場合に、
前記ロック範囲がロック済みではないと判定することを特徴とする、
請求項 3 記載のデータ同期方法。

[請求項5]

前記ロック範囲がロック済みであるか否かを判定するステップが、
前記更新処理にかかるロックキーが前記ロック管理情報に含まれると
判定された場合に、前記ロック管理情報に含まれるロックキーに対応
したロック範囲と、前記更新処理にかかるロック範囲との間に共通し
た範囲があるか否かを判定し、共通した範囲がないと判定された場合
に、前記ロック範囲がロック済みではないと判定する一方、共通した
範囲があると判定された場合に、前記ロック範囲がロック済みである
と判定することを特徴とする、請求項 4 記載のデータ同期方法。

[請求項6]

前記第 1 のデータベースと前記第 2 のデータベースとは、互いに異
なる種類のデータベースであり、

前記第 1 及び第 2 のデータベースをロックするステップが、前記第
1 のデータベースのロックキーと前記第 2 のデータベースのロックキ
ーとの間の対応関係、及び前記第 1 更新データのロック範囲と前記第
2 更新データのロック範囲との間の対応関係、をそれぞれ示す情報に
基づいて、前記ロック管理情報を参照することを特徴とする、請求項
4 又は請求項 5 記載のデータ同期方法。

[請求項7]

前記第 1 及び第 2 のデータベースがそれぞれ更新された後に、前記
第 1 及び第 2 のデータベースのロックを解放するステップと、

ロックの解放後に、待ち情報が設定された更新処理にかかるロック

範囲がロック済みであるか否かを判定するステップと、

前記ロック範囲がロック済みではないと判定された場合に、前記ロック管理情報において、前記更新処理にかかる待ち情報を削除するとともに、前記更新処理にかかるロック獲得情報を設定するステップと、をそなえることを特徴とする、請求項 3～6 のいずれか 1 項記載のデータ同期方法。

[請求項8] 前記ロック範囲を決定するステップが、前記第 1 及び第 2 のデータベースにおけるテーブルの列を最小単位として、前記ロック範囲を決定することを特徴とする、請求項 1～7 のいずれか 1 項記載のデータ同期方法。

[請求項9] 前記ロック範囲を決定するステップが、
前記更新処理による更新前のデータと、前記更新後のデータとに基づいて、更新差分データを生成するステップと、
前記更新差分データに基づいて、前記更新処理によって更新される範囲を判定し、ロック範囲を決定するステップと、をそなえることを特徴とする、請求項 1～8 のいずれか 1 項記載のデータ同期方法。

[請求項10] 前記第 1 更新データを変換するステップが、前記変換定義情報に基づいて、前記更新処理が行なわれる第 1 のデータベース以外のデータ同期対象のデータベース毎に行なわれ、

前記データベースをロックするステップが、データ同期対象の全てのデータベースをロックするとともに、

前記データベースを更新するステップが、前記第 1 更新データ又は変換された更新データを用いて前記データ同期対象の全てのデータベースをそれぞれ更新することを特徴とする、請求項 1～9 のいずれか 1 項記載のデータ同期方法。

[請求項11] 前記第 1 更新データを前記第 2 更新データに変換するステップと、前記ロック範囲を決定するステップとが、並行して行なわれ、前記第 1 及び第 2 のデータベースをロックするステップが、前記第 1 更新デ

ータを前記第2更新データに変換するステップ及び前記ロック範囲を決定するステップの完了後に行なわれることを特徴とする、請求項1～10のいずれか1項記載のデータ同期方法。

[請求項12] 前記第1のデータベースは、第1OS (Operating System) に備えられ、

前記第2のデータベースは、第2OSに備えられ、

前記第1及び第2OSは、1のシステム上で動作するとともに、

前記第1及び第2のデータベースをそれぞれ更新するステップは、前記第2OSに、前記第1及び第2OSからアクセス可能な共用メモリを介して、前記第2更新データを出力することを特徴とする、請求項1～11のいずれか1項記載のデータ同期方法。

[請求項13] 前記第1及び第2OSは、仮想OSであり、

前記システムは、仮想マシンであることを特徴とする、請求項12記載のデータ同期方法。

[請求項14] 第1のデータベースに対して行なわれる更新処理にかかる第1更新データを取得し、

予め設定された変換定義情報に基づいて、前記第1更新データを第2のデータベースの形式に対応した第2更新データに変換し、

前記第1更新データと、前記更新処理が行なわれる前の更新前データとに基づいて、ロック範囲を決定し、

決定されたロック範囲に基づいて、前記第1及び第2のデータベースをロックし、

前記第1及び第2更新データを用いて、前記第1及び第2のデータベースをそれぞれ更新する、

処理をコンピュータに実行させることを特徴とする、データ同期プログラム。

[請求項15] 前記第1及び第2のデータベースをロックする処理は、

前記ロック範囲に基づいて、前記第1及び第2のデータベースのロ

ック状態が設定されたロック管理情報から、前記第 1 及び第 2 のデータベースにおける前記ロック範囲がロック済みであるか否かを判定し、

前記ロック範囲がロック済みではないと判定された場合に、前記ロック管理情報において、前記更新処理にかかるロック範囲がロックされていることを示すロック獲得情報を設定する、
処理をコンピュータに実行させることを特徴とする、請求項 1 4 記載のデータ同期プログラム。

[請求項16]

前記第 1 及び第 2 のデータベースをロックする処理は、

前記ロック範囲がロック済みであると判定された場合に、前記ロック管理情報において、前記更新処理にかかるロック範囲がロック待ち状態であることを示す待ち情報を設定する、
処理をコンピュータに実行させることを特徴とする、請求項 1 5 記載のデータ同期プログラム。

[請求項17]

前記ロック管理情報は、前記第 1 及び第 2 のデータベースに共通する前記更新処理にかかる行を特定するロックキーを含み、

前記第 1 更新データを前記第 2 更新データに変換する処理が、前記更新処理にかかるロックキーを生成するとともに、

前記ロック範囲がロック済みであるか否かを判定する処理が、生成された前記更新処理にかかるロックキーと同一のロックキーが前記ロック管理情報に含まれるか否かを判定し、前記更新処理にかかるロックキーが前記ロック管理情報に含まれないと判定された場合に、前記ロック範囲がロック済みではないと判定することを特徴とする、請求項 1 6 記載のデータ同期プログラム。

[請求項18]

前記第 1 及び第 2 のデータベースがそれぞれ更新された後に、前記第 1 及び第 2 のデータベースのロックを解放し、

ロックの解放後に、待ち情報が設定された更新処理にかかるロック範囲がロック済みであるか否かを判定し、

前記ロック範囲がロック済みではないと判定された場合に、前記ロック管理情報において、前記更新処理にかかる待ち情報を削除するとともに、前記更新処理にかかるロック獲得情報を設定する、処理をコンピュータに実行させることを特徴とする、請求項 16 又は請求項 17 記載のデータ同期プログラム。

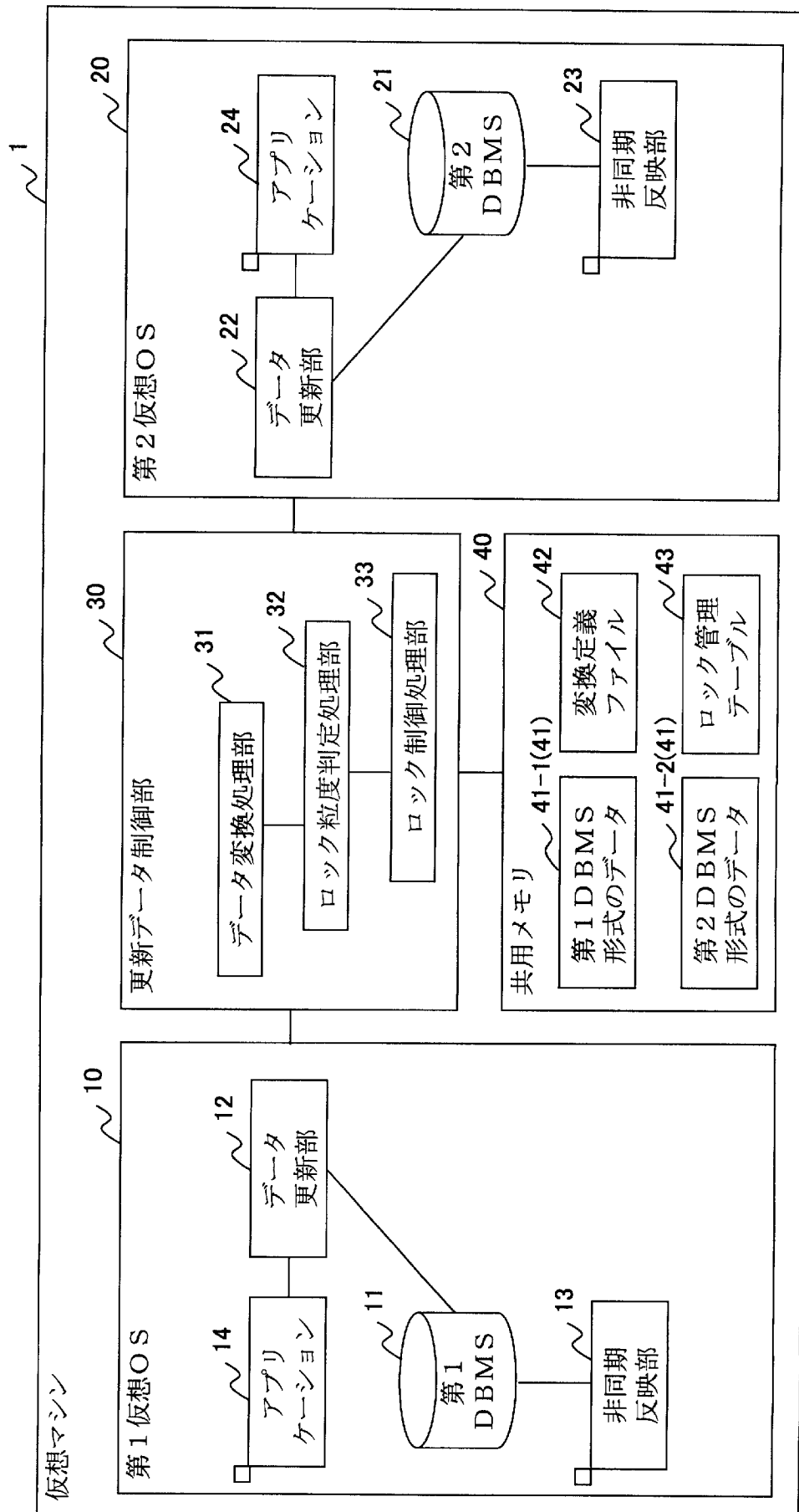
[請求項19] 前記ロック範囲を決定する処理が、前記第 1 及び第 2 のデータベースにおけるテーブルの列を最小単位として、前記ロック範囲を決定することを特徴とする、請求項 14 ～ 18 のいずれか 1 項記載のデータ同期プログラム。

[請求項20] 第 1 のデータベースに対して行なわれる更新処理にかかる第 1 更新データを取得して、予め設定された変換定義情報に基づいて、前記第 1 更新データを第 2 のデータベースの形式に対応した第 2 更新データに変換するデータ変換部と、

前記第 1 更新データと、前記更新処理が行なわれる前の更新前データとに基づいて、ロック範囲を決定するロック範囲決定部と、

ロック範囲決定部により決定されたロック範囲に基づいて、前記第 1 及び第 2 のデータベースをロックするとともに、前記第 1 及び第 2 のデータベースに対して、それぞれ前記第 1 更新データ及び前記データ変換部により変換された前記第 2 更新データを用いた更新を指示するロック制御部とをそなえることを特徴とする、データ同期制御装置。

[図1]



[図3]

42
↘

No.	OS識別子	DB識別子	文字データの のコード系	後方空白の扱い	数値データの 内部表現形式	混在データの列名、 及び混在データの扱い
1	Win001	SQL1	UTF-16	後方空白補正なし	文字列表現	ROW03、文字データ
2	Win002	RW01	Shift-JIS	後方空白補正あり	COBOL表現	ROW03、数値データ
⋮	⋮	⋮	⋮	⋮	⋮	⋮

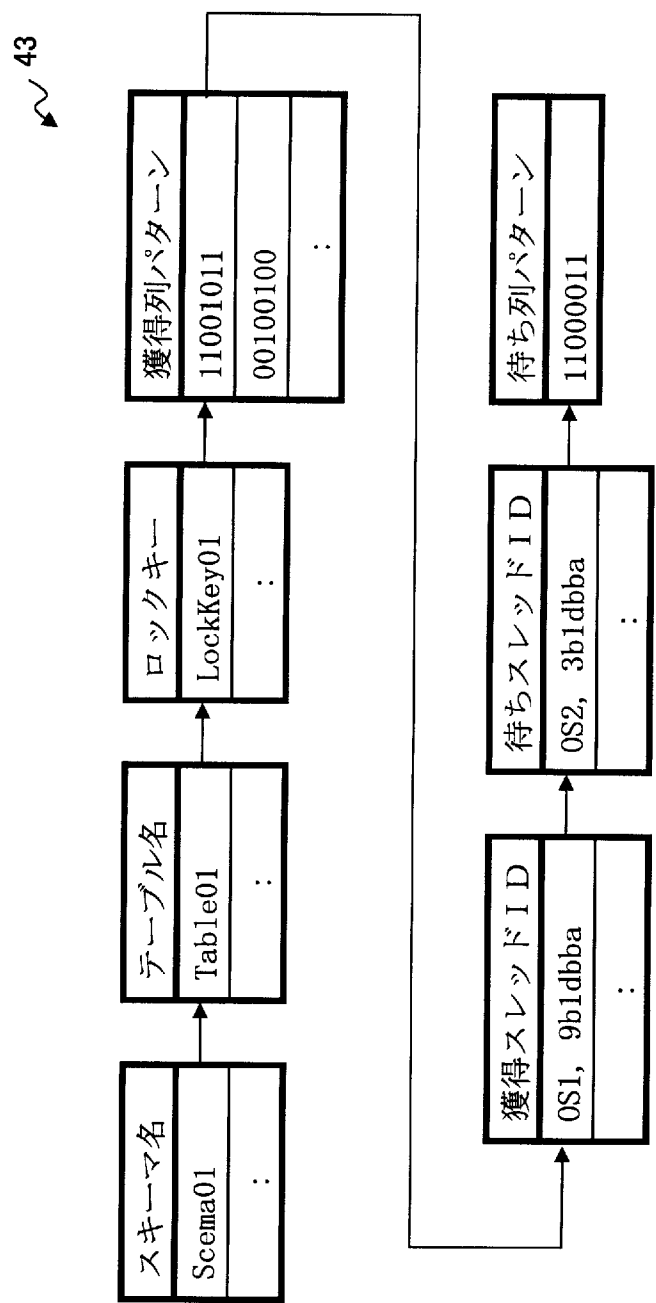
[図4]

内部表現形式		数値データ (123.45)
文字列表現	UTF-16	0x3100320033002e0034003500
	Shift-JIS	0x31323302e3435
COBOL表現		0x31323334c5

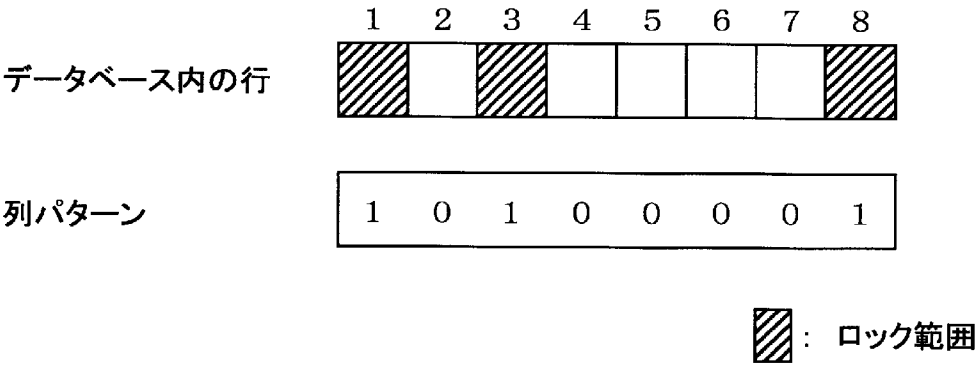
[図5]

データ内容	文字データ	数値データ	混在データ
OS識別子	あい\abc	123.45	12
Win001	0x42304430610062006300	0x3100320033002e0034003500	0x31003200
↓	↓	↓	↓
Win002	0x82a082a2616263	0x31323334c5	0x31c2

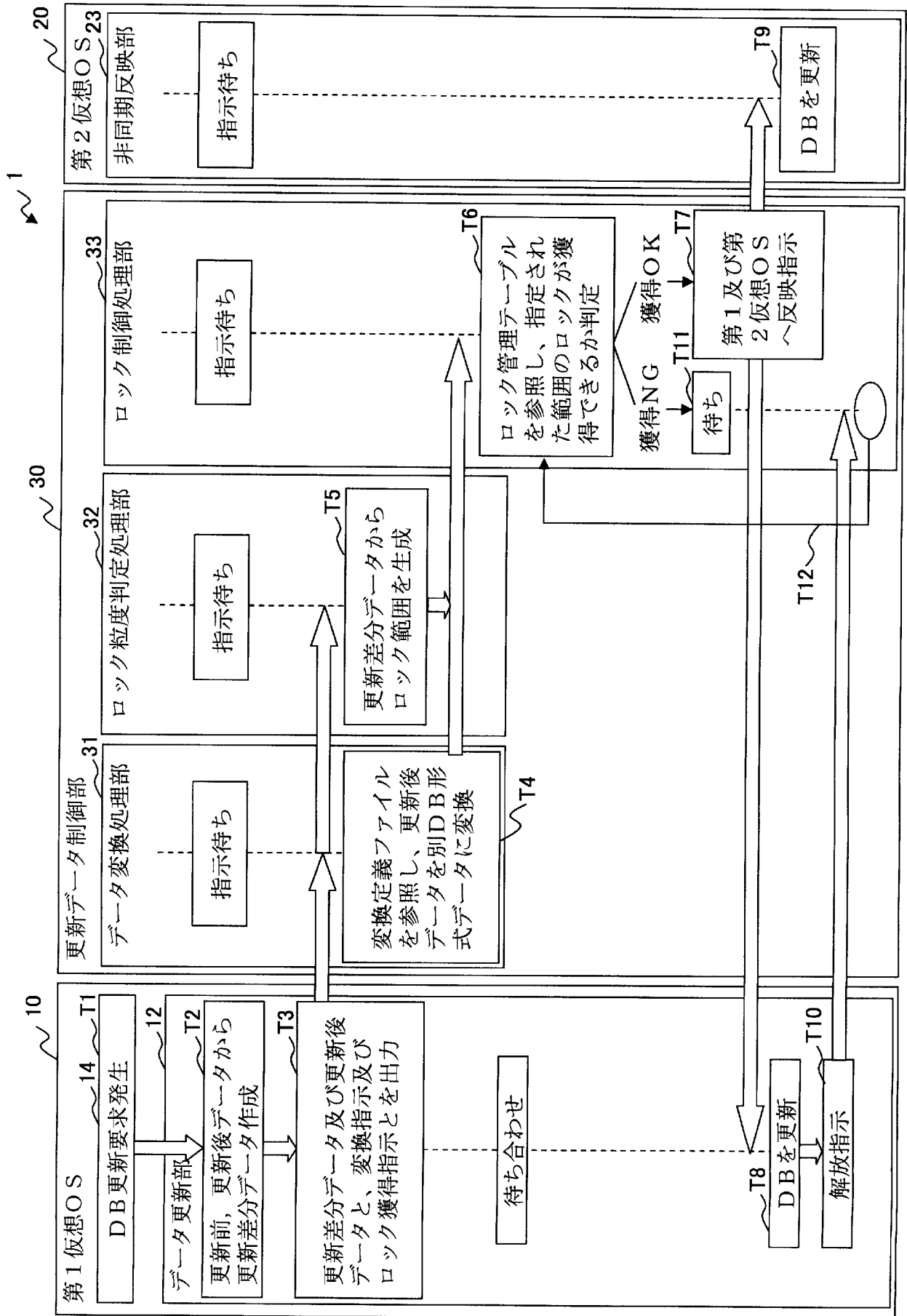
[図6]



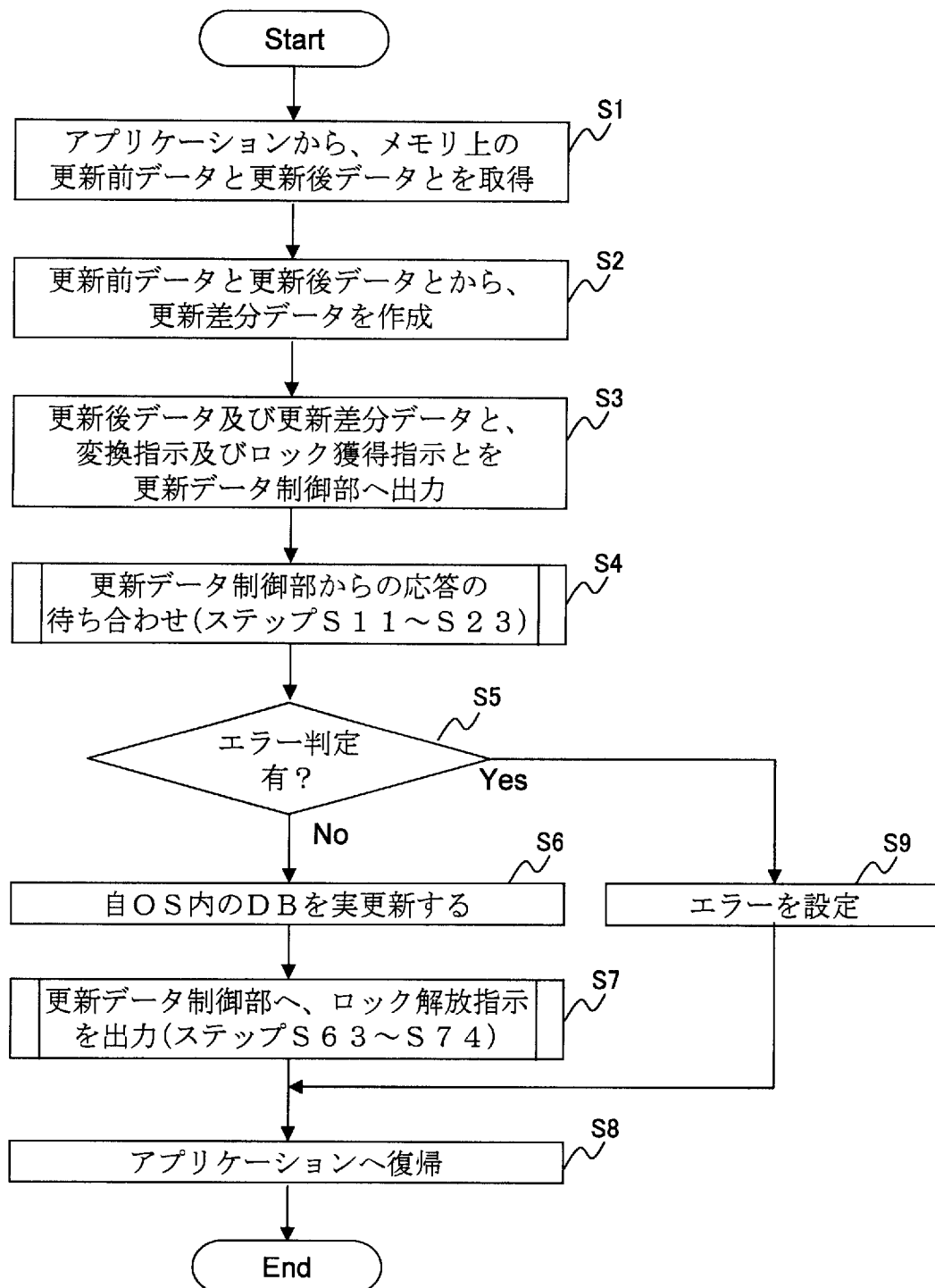
[図7]



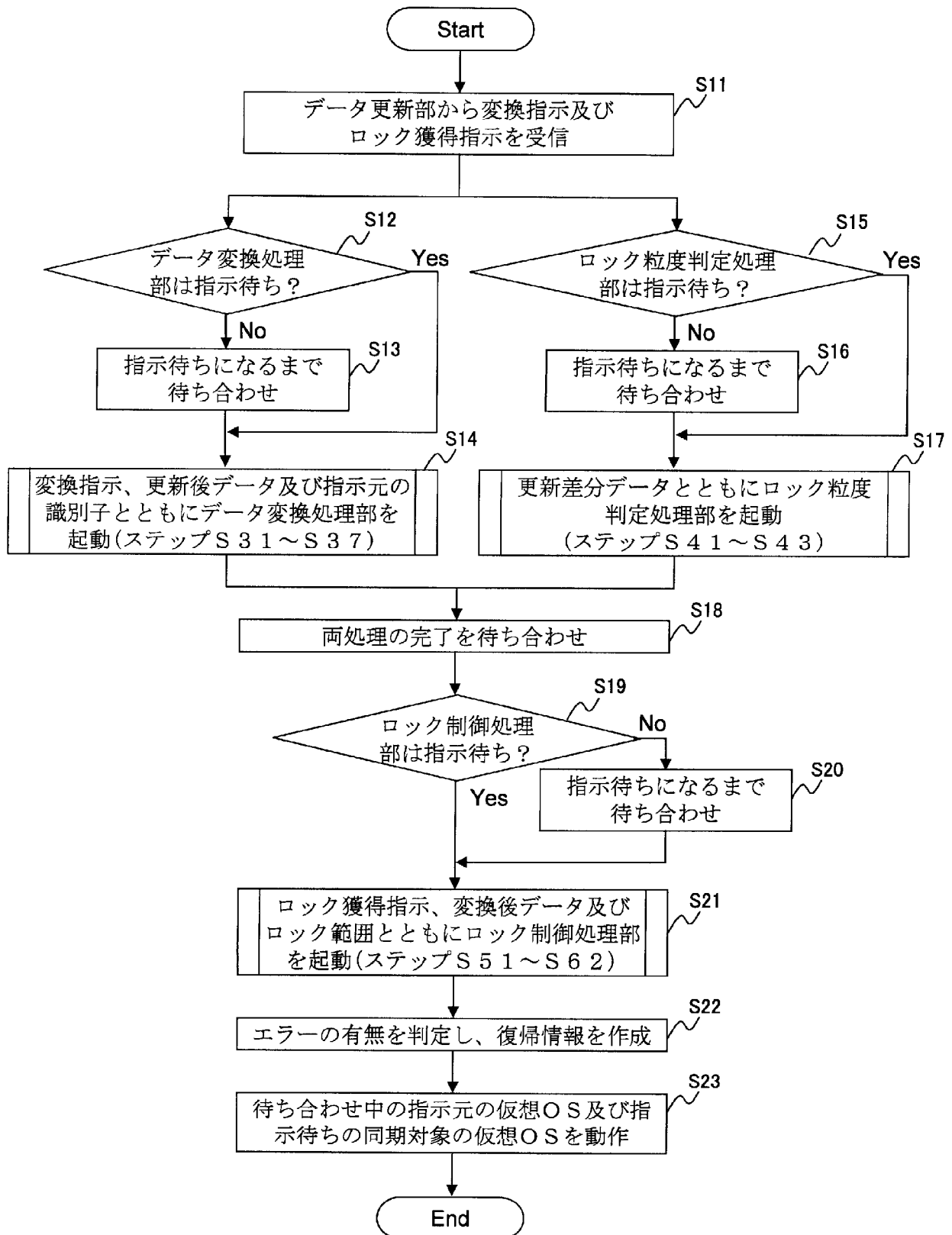
[図8]



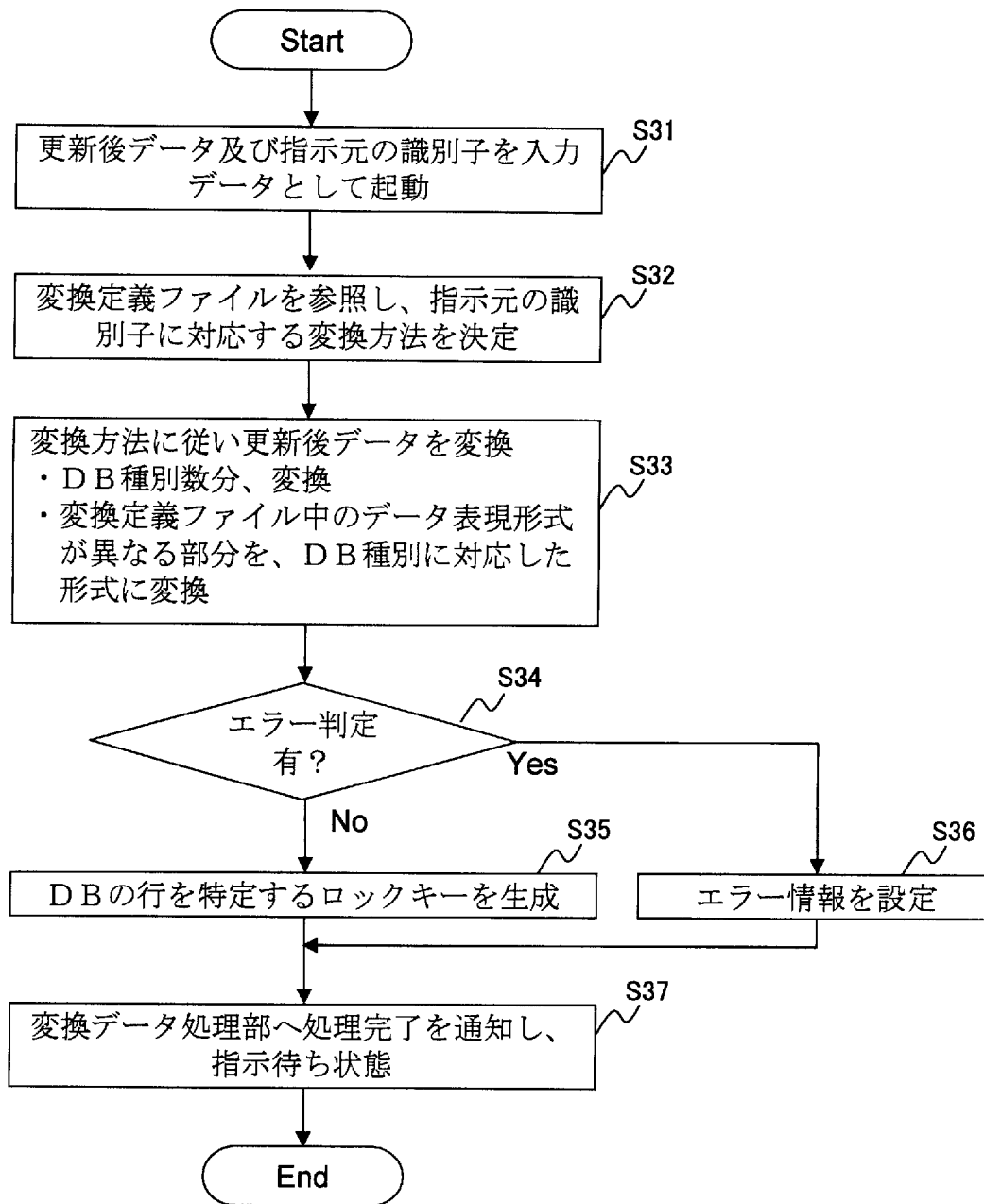
[図9]



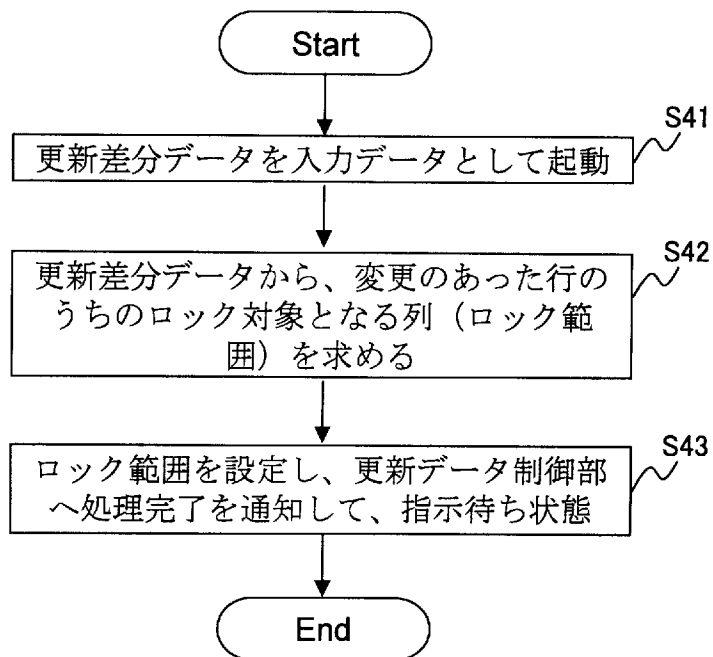
[図10]



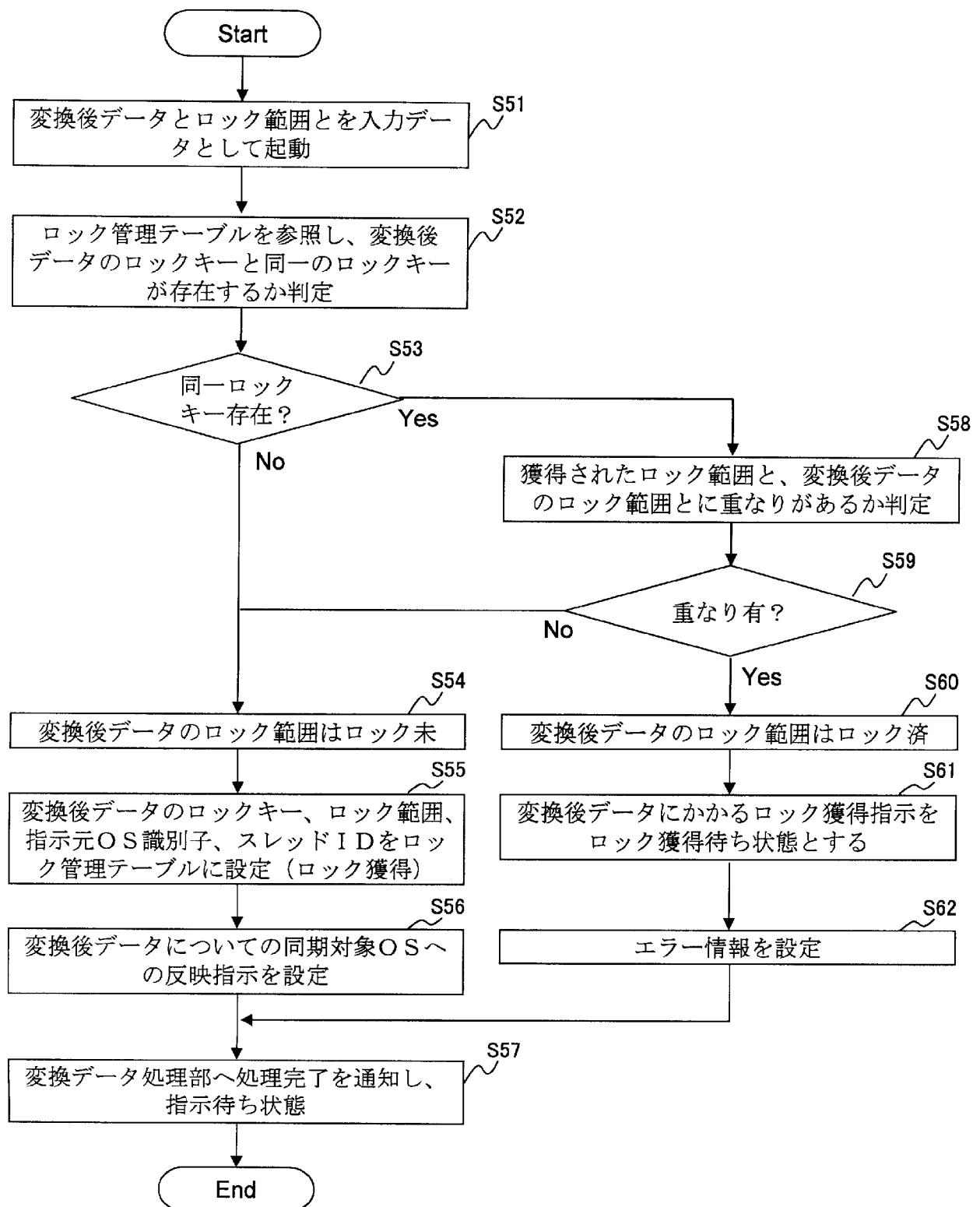
[図11]



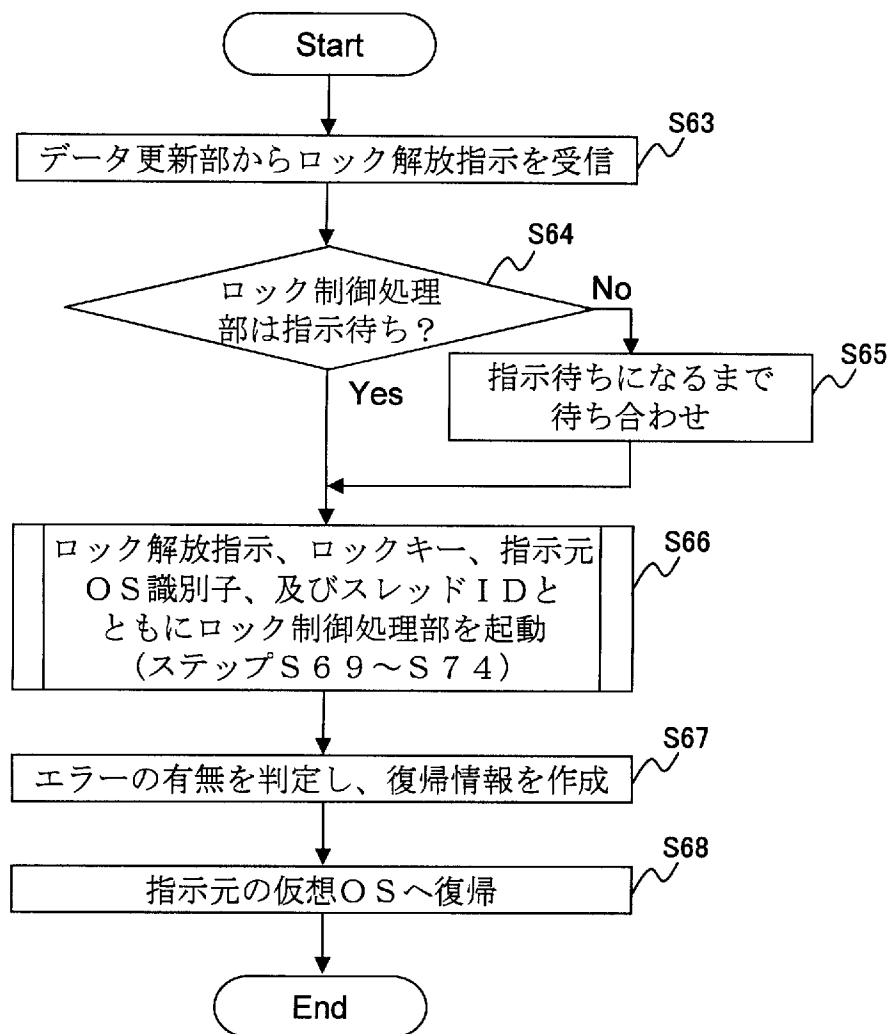
[図12]



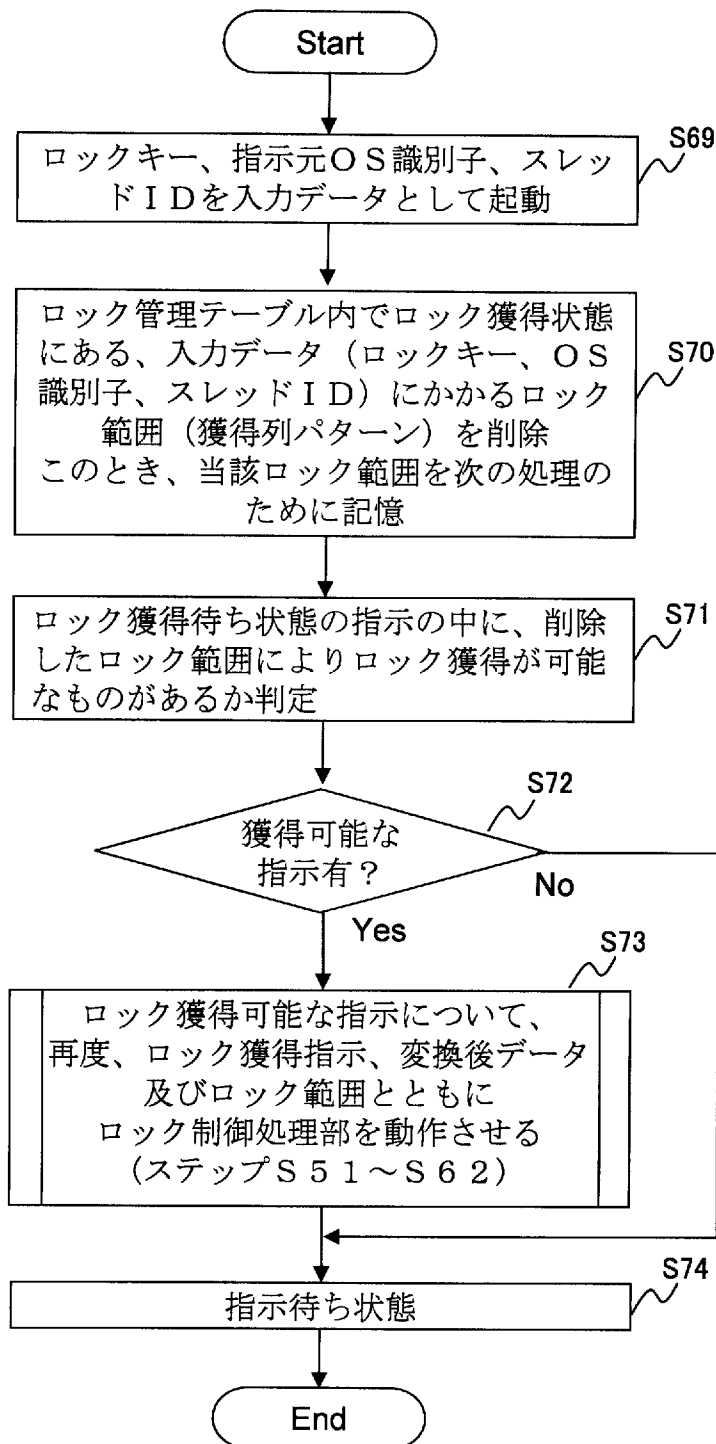
[図13]



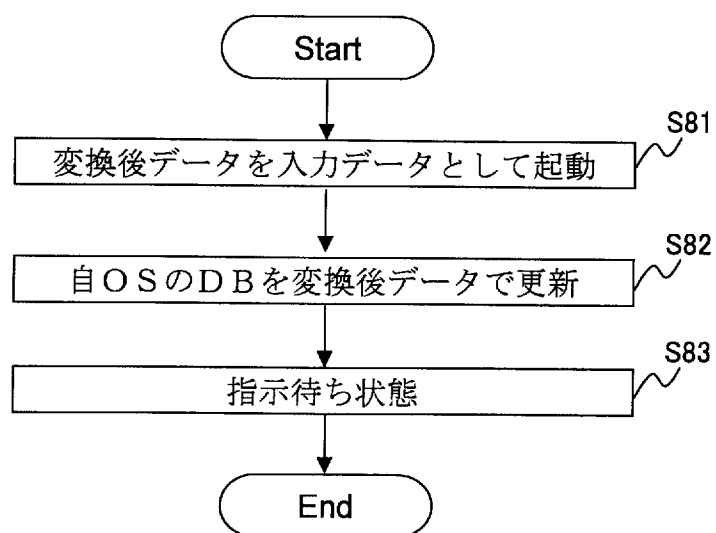
[図14]



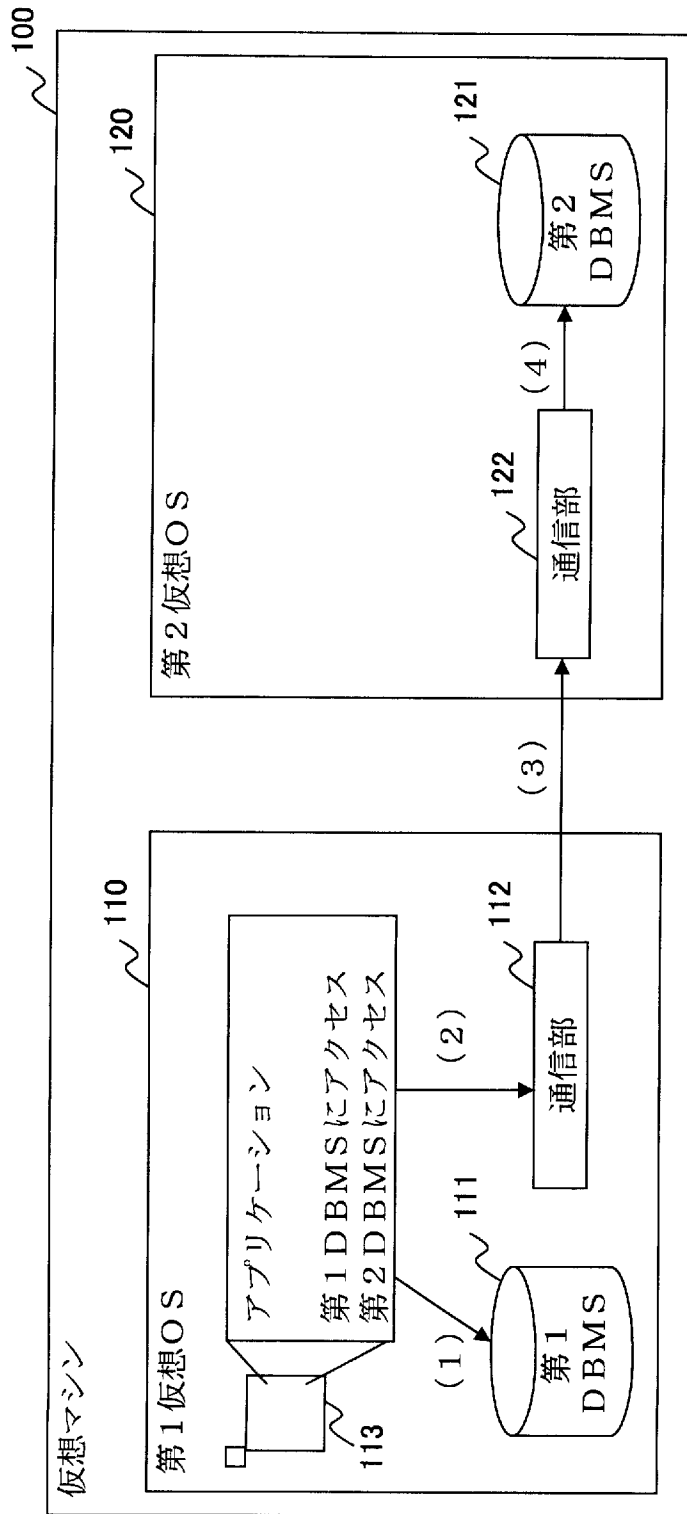
[図15]



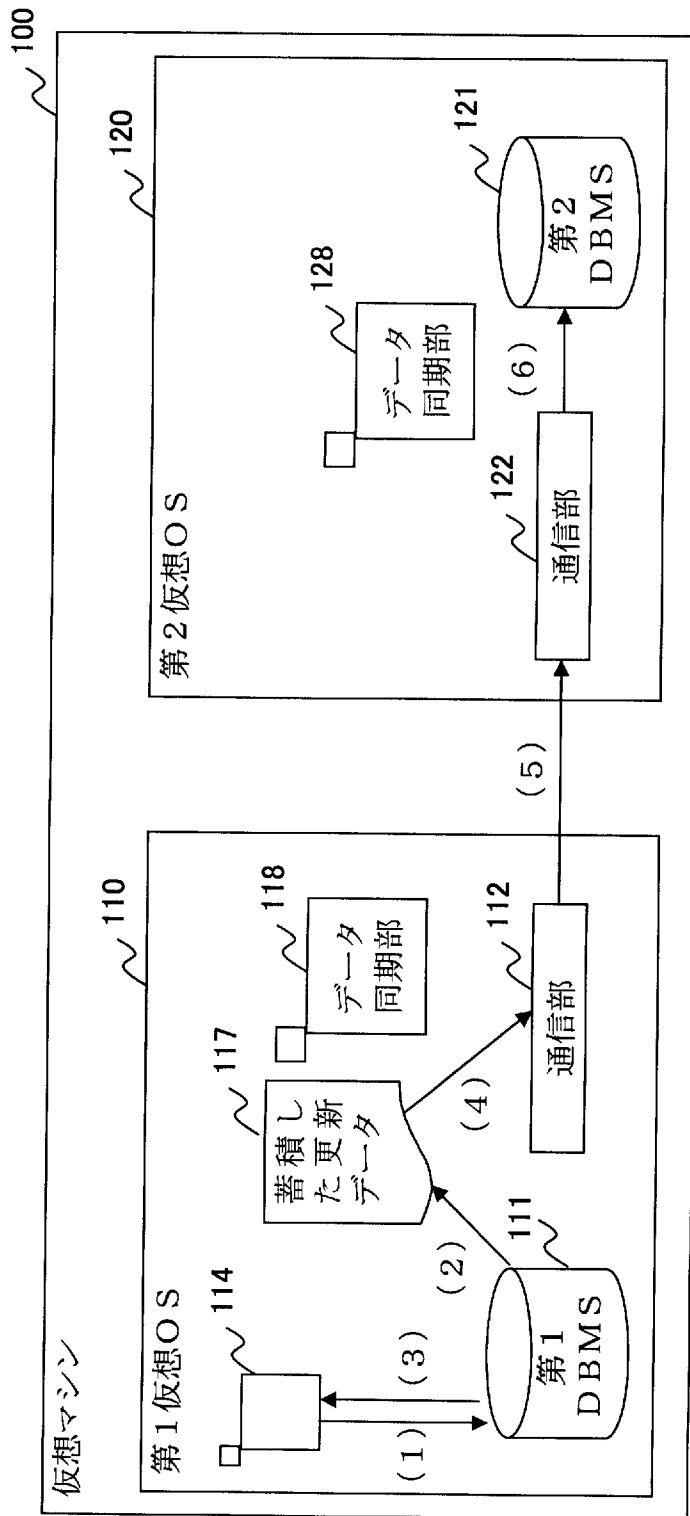
[図16]



[図17]



[図19]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/052758

A. CLASSIFICATION OF SUBJECT MATTER

G06F12/00 (2006.01) 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)

G06F12/00

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 1996-2011

Kokai Jitsuyo Shinan Koho 1971-2011 Toroku Jitsuyo Shinan Koho 1994-2011

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

JSTPlus (JDreamII)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y A	JP 8-179980 A (Hitachi, Ltd.), 12 July 1996 (12.07.1996), claim 1; paragraphs [0006] to [0009] (Family: none)	1, 8, 9, 14, 19, 20 2-7, 10-13, 15-18
Y A	JP 2009-026334 A (Ricoh Co., Ltd.), 05 February 2009 (05.02.2009), paragraphs [0048] to [0055] (Family: none)	1, 8, 9, 14, 19, 20 2-7, 10-13, 15-18
Y A	JP 8-235043 A (NTT Data Communications Systems Corp.), 13 September 1996 (13.09.1996), paragraph [0024] & US 6052695 A & EP 758114 A1 & WO 1996/027157 A1	1, 8, 9, 14, 19, 20 2-7, 10-13, 15-18

☒ 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
01 March, 2011 (01.03.11)Date of mailing of the international search report
15 March, 2011 (15.03.11)Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/052758

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 2003-316617 A (Hitachi, Ltd.), 07 November 2003 (07.11.2003), paragraphs [0012], [0028] (Family: none)	1-20
A	JP 2009-129414 A (Softbank Mobile Corp.), 11 June 2009 (11.06.2009), entire text (Family: none)	1-20

A. 発明の属する分野の分類 (国際特許分類 (I P C))

Int.Cl. G06F12/00 (2006.01) 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. G06F12/00

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

JSTPlus (JDreamII)

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
Y A	JP 8-179980 A (株式会社日立製作所) 1996.07.12, 請求項 1, [0006]-[0009] (ファミリーなし)	1, 8, 9, 14, 19, 20 2-7, 10-13, 15-18
Y A	JP 2009-026334 A (株式会社リコー) 2009.02.05, [0048]-[0055] (ファミリーなし)	1, 8, 9, 14, 19, 20 2-7, 10-13, 15-18

☒ 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

0 1 . 0 3 . 2 0 1 1

国際調査報告の発送日

1 5 . 0 3 . 2 0 1 1

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)

工藤 嘉晃

電話番号 0 3 - 3 5 8 1 - 1 1 0 1 内線 3 5 9 9

5 M

3 9 7 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
Y A	JP 8-235043 A (エヌ・ティ・ティ・データ通信株式会社) 1996. 09. 13, [0024] & US 6052695 A & EP 758114 A1 & WO 1996/027157 A1	1, 8, 9, 14, 19, 20 2-7, 10-13, 15-18
A	JP 2003-316617 A (株式会社日立製作所) 2003. 11. 07, [0012], [0028] (ファミリーなし)	1-20
A	JP 2009-129414 A (ソフトバンクモバイル株式会社) 2009. 06. 11, 全文 (ファミリーなし)	1-20