

【公報種別】特許法第 17 条の 2 の規定による補正の掲載

【部門区分】第 6 部門第 3 区分

【発行日】平成20年11月13日(2008.11.13)

【公表番号】特表2004-532480(P2004-532480A)

【公表日】平成16年10月21日(2004.10.21)

【年通号数】公開・登録公報2004-041

【出願番号】特願2002-592023(P2002-592023)

【国際特許分類】

G 0 6 F 12/00 (2006.01)

【F I】

G 0 6 F 12/00 5 3 1 J

【誤訳訂正書】

【提出日】平成20年9月26日(2008.9.26)

【誤訳訂正 1】

【訂正対象書類名】特許請求の範囲

【訂正対象項目名】全文

【訂正方法】変更

【訂正の内容】

【特許請求の範囲】

【請求項 1】

同期式変更データキャプチャのための方法であって、  
複数のトランザクションのうちの各トランザクションに対して、各トランザクションを独自に識別するトランザクション識別子を生成するステップと、  
トランザクション内の各動作に対して、動作に対する変更データ、前記トランザクションに対する前記トランザクション識別子、および前記トランザクション内の各動作の相対的な順序付けを識別する識別子を第 1 のデータベースオブジェクトに記録するステップと  
を含み、前記複数のトランザクションのうちの少なくとも一つは、複数の動作を含み、前記方法は、

トランザクションのコミット中、前記トランザクション識別子およびシステム変更番号を第 2 のデータベースオブジェクトに記録するステップとを含み、前記システム変更番号は、前回コミットされたトランザクションよりも後のトランザクションのコミットの時期を示し、前記方法は、

前記第 1 のデータベースオブジェクトならびに前記第 2 のデータベースオブジェクト上において結合動作を実行することによって、前記トランザクション識別子に基づいて、前記第 1 のデータベースオブジェクト内の前記変更データを前記第 2 のデータベースオブジェクト内のシステム変更番号に関連づけるステップとを含み、前記結合動作は、前記第 1 のデータベースオブジェクト内の各変更データに対する動作が行なわれる時期に基づいて、前記第 1 のデータベースオブジェクト内の前記変更データを順番に並べる、方法。

【請求項 2】

前記トランザクションの前記コミット中に、  
同時実行ロックを得るステップと、  
前記同時実行ロックを得た後、システム変更番号を生成し、前記トランザクション識別子および前記システム変更番号を第 2 のデータベースオブジェクトに記録する前記ステップを実行し、さらには前記コミットを終了させるステップと、

前記トランザクション識別子および前記システム変更番号を前記第 2 のデータベースオブジェクトに記録する前記ステップの後、前記同時実行ロックを解除するステップとを含む、請求項 1 に記載の方法。

【請求項 3】

前記第1のデータベースオブジェクトは変更テーブルを含み、前記第2のデータベースオブジェクトはトランザクションテーブルを含む、請求項1に記載の方法。

【請求項4】

同期式変更データキャプチャのための命令を有するコンピュータ読出可能な媒体であって、前記命令は、実行時に、1つ以上のプロセッサに請求項1に記載の方法のステップを実行させるように構成される、コンピュータ読出可能な媒体。

【請求項5】

同期的にキャプチャされる変更データを処理するための方法であって、

複数のトランザクションのうちの各トランザクション内で実行される複数の動作のうちの各動作に対する変更データ、前記複数のトランザクションのうちの各トランザクションならびに各トランザクション内で実行される複数の動作を独自に識別するトランザクション識別子、および前記トランザクション内の各動作の相対的な順序付けを識別する識別子を含む第1のデータベースオブジェクトにアクセスするステップと、

前記トランザクション識別子およびシステム変更番号を含む第2のデータベースオブジェクトにアクセスするステップと、

前記第1のデータベースオブジェクトおよび前記第2のデータベースオブジェクト上で結合動作を実行することによって、前記トランザクション識別子に基づいて、前記第1のデータベースオブジェクト内の変更データを前記第2のデータベースオブジェクト内の前記システム変更番号に関連付けるステップとを含み、

前記システム変更番号は、前記トランザクション識別子に対応するトランザクションのコミット中に記憶され、かつ前回コミットされたトランザクションよりも後のトランザクションであって前記トランザクション識別子に対応するトランザクションのコミットの時期を示し、前記結合動作は、前記第1のデータベースオブジェクト内の各変更データに対する動作が行なわれる時期に基づいて、前記第1のデータベースオブジェクト内の前記変更データを順番に並べる、方法。

【請求項6】

同期式変更データキャプチャのための命令を有するコンピュータ読出可能な媒体であって、前記命令は、実行時に、1つ以上のプロセッサに請求項5に記載の方法のステップを実行させるように構成される、コンピュータ読出可能な媒体。

【請求項7】

同期式変更データキャプチャのための方法であって、

トランザクションを独自に識別するトランザクション識別子を生成するステップと、

トランザクション内の各動作に対して、動作に対する変更データ、トランザクションに対するトランザクション識別子、および前記トランザクション内の各動作の相対的な順序付けを識別する識別子を第1のデータベースオブジェクトに記録するステップとを含み、前記複数のトランザクションのうちの少なくとも一つは、複数の動作を含み、前記方法は、

トランザクションのコミット中に、以下のステップを実行するステップとを含み、前記以下のステップは、

同時実行ロックを得るステップと、

前記同時実行ロックを得た後、前回コミットされたトランザクションよりも後のコミットの時期を示すシステム変更番号を生成し、前記トランザクション識別子および前記システム変更番号を第2のデータベースオブジェクトに記録するステップと、

前記トランザクション識別子および前記システム変更番号を前記第2のデータベースオブジェクトに記録する前記ステップの後、前記同時実行ロックを解除するステップと、

前記第1のデータベースオブジェクトならびに前記第2のデータベースオブジェクト上において結合動作を実行することによって、前記トランザクション識別子に基づいて、前記第1のデータベースオブジェクト内の前記変更データを前記第2のデータベースオブジェクト内のシステム変更番号に関連づけるステップとを含み、前記結合動作は、前記第1のデータベースオブジェクト内の各変更データに対する動作が行なわれる時期に基づいて

、前記第1のデータベースオブジェクト内の前記変更データを順番に並べる、方法。

【請求項8】

同期式変更データキャプチャのための命令を有するコンピュータ読出可能な媒体であって、前記命令は、実行時に、1つ以上のプロセッサに請求項7に記載の方法のステップを実行させるように構成される、コンピュータ読出可能な媒体。

【誤訳訂正2】

【訂正対象書類名】明細書

【訂正対象項目名】0005

【訂正方法】変更

【訂正の内容】

【0005】

リレーショナルデータベース管理システム内の各トランザクションに対して、リレーショナルデータベース管理システム内のテーブルに対してなされる各修正に関連した独自のシステム変更番号（SCN）が存在する。通例、各修正は、SQL言語内のINSERT、UPDATE、またはDELETEステートメント等の個別のデータベース操作言語（DML）ステートメントによって特徴付けられる。システム変更番号は、単調に増加する通し番号であって、各新しいDMLステートメントには、予め割当てられたいずれかのシステム変更番号よりも大きな新しいシステム変更番号が割当てられる。トランザクションは、複数のDMLステートメントを含み得、各々が、対応するシステム変更番号を有している。さらに、トランザクションがコミットすると、全体としてのトランザクションに当てはまる、ここで「コミットシステム変更番号」（CSCN）と呼ばれる、最終的なシステム変更番号が存在する。リレーショナルデータベース管理システムでは、データベースの内容に対する変更は、実際にはコミット時に適用され、したがって、CSCNは、そのトランザクション内で起こったすべてのDML動作を表す。さらに、変更テーブルが、各変更のためのCSCNを記録するのであれば、リレーショナルデータベース管理システム上で変更が起こった正確な順序を決定することが可能であろう。その場合、アプリケーションは、変更テーブルを用いて、ある特定のトランザクション内で起こった変更を分離し、さらには、それがリレーショナルデータベース管理システム内でコミットした順序で各変更を見ることができる。

【誤訳訂正3】

【訂正対象書類名】明細書

【訂正対象項目名】0006

【訂正方法】変更

【訂正の内容】

【0006】

従来のシステムは、CREATE TRIGGERステートメントを用いることによって、または、同等の機能を備えた内部メカニズムを用いることによって、同期式変更データキャプチャのためにトリガを用いた。トリガとは、ある特定のイベントが起こるときに自動的に実行されるべき一連のアクションを特定するオブジェクトであり、業界規格によると、トリガを活性化させる（または「ファイヤさせる(fired)」）イベントが、DMLステートメントである。同期式変更データキャプチャでは、トリガは、データベーステーブルの行が更新、挿入、または削除されるときにファイヤするように設計されてきた。これらの修正の各々は、トリガによって記録されるそれ自体のシステム変更番号（SCN）に関連する。トランザクションのための真のコミットシステム変更番号（CSCN）は、このときは記録され得ない。なぜならば、コミットは、まだ起こっていないためである。したがって、既存のトリガベースのメカニズムは、CSCNをキャプチャせず、これは、それらの有用性を大きく制限する重大な技術的欠点である。各記録のためのCSCNがない状態では、ある特定のトランザクション内で起こった変更を識別することができず、また、変更がリレーショナルデータベース管理システムに対してコミットされた正確な順序で、順に各トランザクションの効果を考察することもできない。

## 【誤訳訂正 4】

【訂正対象書類名】明細書

【訂正対象項目名】0007

【訂正方法】変更

【訂正の内容】

【0007】

たとえば、OLTPシステム上に複数のユーザが存在し、OLTPシステムは、ユーザが実行する各ステートメントに対してシステム変更番号を割当てると仮定する。さらに、ユーザXが、SCN1においてトランザクションXを開始させ、SCN20においてトランザクションXをコミットすると仮定する。しかし、トランザクションYが、別のユーザによってSCN16において開始され、18のCSCNでコミットすると仮定する。この例は、システム変更番号の割当てが、すべてのトランザクション中で共有されていることを示す。したがって、所与のトランザクションXは、他のいずれのトランザクションにも割当てられていない、排他的で、破られていない範囲のシステム変更番号を有することを保証されておらず、実際には、逆である。トランザクションXが最初に開始したが、実際には、トランザクションYが最初にコミットした。リレーショナルデータベース管理システムのデータの完全性に関する限り、具体的には、変更テーブルに関して、トランザクションXの第1のステートメントがトランザクションYの第1のステートメントよりも早いシステム変更番号を有していたという事実にもかかわらず、トランザクションYがトランザクションXの前にコミットしたため、トランザクションYは、トランザクションXに先立って起こったと考えられる。

## 【誤訳訂正 5】

【訂正対象書類名】明細書

【訂正対象項目名】0010

【訂正方法】変更

【訂正の内容】

【0010】

これらおよび他の必要性は、変更データ内の各ステートメントに対して、各トランザクションを独自に識別するトランザクション識別子を記録することによって、この発明により対処される。トランザクションがコミットされると、コミットに対するシステム変更番号およびトランザクション識別子が、トランザクションテーブルに記録される。変更データ内の各ステートメントのためのコミットシステム変更番号を識別するために、変更データ内のトランザクション識別子を用いて、トランザクションテーブルからコミットシステム変更番号をフェッチする。

## 【誤訳訂正 6】

【訂正対象書類名】明細書

【訂正対象項目名】0011

【訂正方法】変更

【訂正の内容】

【0011】

したがって、この発明の1つの局面は、トランザクションを独自に識別するトランザクション識別子が生成される、同期式変更データキャプチャのための方法を含む。トランザクション内の各動作に対して、動作に対する変更データおよびトランザクション識別子が、第1のデータベースオブジェクト（たとえば、変更テーブル）に記録される。トランザクションのコミット中、トランザクション識別子およびシステム変更番号が、第2のデータベーステーブル（たとえば、トランザクションテーブル）に記録される。トランザクション内の各動作の相対的な順序を識別する識別子も、第1のデータベースオブジェクトに記録され得る。

## 【誤訳訂正 7】

【訂正対象書類名】明細書

【訂正対象項目名】 0 0 1 2

【訂正方法】 変更

【訂正の内容】

【 0 0 1 2 】

一実現例では、コミット中に記録されるシステム変更番号は、他のプロセスが新しいシステム変更番号を生成することを妨げる同時実行ロックが得られた後に、生成される。トランザクション識別子およびシステム変更番号を第2のデータベースオブジェクトに記録した後、同時実行ロックが解除される。

【誤訳訂正8】

【訂正対象書類名】 明細書

【訂正対象項目名】 0 0 2 4

【訂正方法】 変更

【訂正の内容】

【 0 0 2 4 】

この発明の1つの局面に従うと、OLTPデータベース113から抽出される変更データは、データベース管理システム、たとえば、分析データベース123の制御下の「変更テーブル」および「トランザクションテーブル」とここで呼ばれる1つ以上のデータベースオブジェクト内で維持される。図2を参照して、例として、変更データキャプチャの影響を受ける、OLTPデータベース113上の各ソーステーブルまたはデータベースオブジェクトは、分析データベース123内の対応の変更テーブル211、213、221、223に関連付けられる。トランザクションの一貫性のために、同じ変更セット内の変更テーブル（たとえば、変更セット210の変更テーブル211および213）に追加されるすべての新しい変更データが同時に追加されること、たとえば、これらの変更テーブルに対する修正が、同じトランザクション内で実行され、コミットされることが、パブリッシャ125によって確実にされるように、変更テーブル211、213、221、223は、1つ以上のセット、「変更セット」210および220にグループ分けされる。図2に示される例では、2つの変更セット、変更セット210および変更セット220が存在する。変更セット210は、変更テーブル211および変更テーブル213を含み、これらは、OLTPデータベース113上のそれぞれのテーブル（図示せず）に対応する。同様に、変更セット220は、変更テーブル221および変更テーブル223を含み、これらも、OLTPデータベース113上のそれぞれのテーブル（図示せず）に対応する。変更テーブル211、213、221、223、および変更セット210および220の構造を定義する情報は、システムメタデータ230内で維持される。

【誤訳訂正9】

【訂正対象書類名】 明細書

【訂正対象項目名】 0 0 2 7

【訂正方法】 変更

【訂正の内容】

【 0 0 2 7 】

一実施例では、たとえば、トランザクション識別子233は、以下の特性を有する。（a）各トランザクション識別子は、すべての先行するトランザクション識別子よりも大きく、（b）同じトランザクションに属するすべてのDML動作（挿入、更新、削除等）は、トランザクション識別子に関して同じ値を有し、（c）トランザクションを締めくくるために用いられるSQL「コミット」ステートメントにも、同じトランザクション識別子値が割当てられ、（d）トランザクションの一部ではないSQLステートメントは、トランザクション識別子に関して同じ値を有さない。

【誤訳訂正10】

【訂正対象書類名】 明細書

【訂正対象項目名】 0 0 2 8

【訂正方法】 変更

## 【訂正の内容】

## 【0028】

TIME235列は、変更データを生じさせたトランザクションのコミット時間を含む。この列は、特定の期間に起こる変更データをサブスクライバアプリケーション121が選択するか、または見る助けとなる。

## 【誤訳訂正11】

## 【訂正対象書類名】明細書

## 【訂正対象項目名】0030

## 【訂正方法】変更

## 【訂正の内容】

## 【0030】

ROW239列は、変更テーブル内の変更に対する独自の行シーケンス番号を含む。特に、同じトランザクションの一部である各SQLステートメントまたは動作は、同じトランザクション内のいずれの先行する動作よりも大きな行シーケンス番号の値を有する。この列は、同じトランザクション内でコミットされた動作をサブスクライバアプリケーション121が順序よく並べる助けをする。なぜならば、リレーショナルデータベース管理システムは、いずれの順序でも動作を記憶できるためである。また、同じ動作の、古い更新された値の記録（「UO」を備えたOP237列）と新しい更新された値の記録（「UN」を備えたOP237列）との両方は、同じ行シーケンス番号を有する。なぜならば、これらの2つの記録は、OLTPデータベース113上の同じ変更動作（更新動作）に関するためである。

## 【誤訳訂正12】

## 【訂正対象書類名】明細書

## 【訂正対象項目名】0032

## 【訂正方法】変更

## 【訂正の内容】

## 【0032】

加えて、別個のトランザクションテーブル240が提供されて、コミットされたトランザクションのトランザクション識別子241およびトランザクションコミットに関連したシステム変更番号243が記録される。セッション識別子245も提供されて、現在のデータベースセッションが識別され、関係データベース管理システムの現在のすべてのユーザが区別され得る。トランザクションテーブル240は、すべてのユーザに適用可能な広域テーブルとなるように構成され得る。

## 【誤訳訂正13】

## 【訂正対象書類名】明細書

## 【訂正対象項目名】0035

## 【訂正方法】変更

## 【訂正の内容】

## 【0035】

トランザクション内のすべての動作が実行された後、システムは、トランザクションをコミットする準備ができており、実行は、ステップ305からステップ311まで分岐する。ステップ305、307、309は、図3ではループとして示されているが、トランザクション内のすべての動作が、展開され、または単に連続した順序で並べられた、簡素な順序付けを含む他の制御構造が用いられ得ることが理解されるべきである。

## 【誤訳訂正14】

## 【訂正対象書類名】明細書

## 【訂正対象項目名】0036

## 【訂正方法】変更

## 【訂正の内容】

## 【0036】

ステップ 3 1 1、3 1 3、3 1 5、3 1 7、3 1 9 は、相互排除が実施されることが望ましい、トランザクションのコミットステートメントの処理の一部を含む重要なセクションを構成する。キャプチャされるようにセットアップされていないトランザクションのための時間を節約するために、トランザクション特性がチェックされて、変更が第一にキャプチャされるべきかが判断され得る。変更がキャプチャされなくてもよい場合、ステップ 3 1 1、3 1 3、3 1 5、3 1 9 は、省略される。

【誤訳訂正 1 5】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 3 7

【訂正方法】変更

【訂正の内容】

【0 0 3 7】

ステップ 3 1 1 では、好ましくは、コミット処理においてできるだけ遅く、たとえば、最後の DML 動作後、同時実行ロックが得られて、相互排除セクションが保護され、他のユーザプロセスがステップ 3 1 1、3 1 3、3 1 5、3 1 7、3 1 9 を同時に処理することが防止される。さまざまな形の並行処理が実現されてもよく、この発明は、セマフォの使用等のいずれか 1 つの特定の実現例に限定されているわけではない。

【誤訳訂正 1 6】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 3 8

【訂正方法】変更

【訂正の内容】

【0 0 3 8】

ステップ 3 1 3 では、現在のステートメントのためのシステム変更番号は、近似のコミットシステム変更番号 A C S C N として用いられるように割当てられる。近似のコミットシステム変更番号は、後にコミット中に、それを利用するのには遅すぎる時点で生成されようとしている真のコミット S C N よりも小さい。近似のコミットシステム変更番号 ( A C S C N ) は、同時実行ロック下で得られるため、同時実行ロックが保持されている限り、他のいずれかのトランザクション内の変更データキャプチャ動作が、システム変更番号を割当ててことは許可されない。したがって、近似のコミットシステム変更番号は、真のコミット S C N に対して信頼性のあるプロキシを構成する。

【誤訳訂正 1 7】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 3 9

【訂正方法】変更

【訂正の内容】

【0 0 3 9】

ステップ 3 1 5 では、以下の列、( a ) トランザクション識別子 2 4 1、( b ) 近似のコミットシステム変更番号 2 4 3、および ( c ) 現在のセッションのセッション識別子 2 4 5 を含むトランザクションテーブルに行が挿入される。あらゆる変更テーブルのあらゆる行は、そのトランザクションのためのトランザクション識別子 2 3 3 も記録し、したがって、トランザクション識別子は、2 つの場所、トランザクション識別子 2 3 3 として変更テーブル自体の中に、およびトランザクション識別子 2 4 1 としてトランザクションテーブル内に、記録される。

【誤訳訂正 1 8】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 4 0

【訂正方法】変更

【訂正の内容】

【0 0 4 0】

コミット動作の残りが、ステップ 3 1 7 で処理され、真のコミットシステム変更番号が生成され、ステートメントがコミットする。トランザクションテーブルへの追加的な行の挿入も、このときにコミットされる。最後に、ステップ 3 1 9 で、同時実行ロックが解除され、相互排除セクションを終了し、他のプロセスがシステム変更番号を生成することを許可する。

【誤訳訂正 1 9】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 4 1

【訂正方法】変更

【訂正の内容】

【0 0 4 1】

したがって、トランザクションテーブル 2 4 0 は、同期的にキャプチャされた（トランザクション識別子 2 4 1 によって識別された）各トランザクションのための行を維持し、トランザクションテーブルは、（近似のコミットシステム変更番号 2 4 3 によって決定されるような）全トランザクションがコミットされた真の順序の記録を維持する。性能に関しては、トランザクションテーブル 2 4 0 内でコミットシステム変更番号をキャプチャするときの、最悪の事例でのオーバーヘッドは、そうしない場合とは対照的に、オーバーヘッドの 1 % の約 3 / 4（0 . 0 0 7 5）であり、典型的な事例では、オーバーヘッドは、1 % の 1 / 4（0 . 0 0 2 5）よりも少ないことが確認された。したがって、すべての実際的な目的において、この方法を適用するときのオーバーヘッドは、無視できるほどである。しかしながら、これを行なうことの利点は、大きく、先行技術では見つけられない真のトランザクション順序の能力を提供する。

【誤訳訂正 2 0】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 4 2

【訂正方法】変更

【訂正の内容】

【0 0 4 2】

上述のこの発明の実施例では、キャプチャされた各およびあらゆるトランザクションのための変更データとコミット S C N との両方が、それぞれ、変更テーブル 2 1 1、2 1 3、2 2 1、2 2 3、およびトランザクションテーブル 2 4 0 に記録された。動作が元来起こった順序で変更テーブル 2 2 3 の内容を得るために、変更テーブル 2 2 3 とトランザクション 2 4 0 との間でのデータベース結合動作が用いられ得る。たとえば、この発明の一実現例は、以下の S Q L 結合動作を用い得、変更テーブル 2 2 3 は、C 1、C 2、および C 3 と称されるソーステーブル列 2 3 1 を有する。

```
SELECT TT.CSCN CSCN, CT1.RSID, CT1.C1 C1, CT1.C2 C2, CT1.C3 C3
FROM TT, CT1
WHERE TT.XID=CT1.XID
ORDER BY CSCN
```

このステートメントは、所与のトランザクション内のすべての S Q L ステートメントがトランザクション識別子に関して同じ値を有し、この値が、コミットシステム変更番号 2 4 3 と同じ行内でトランザクションテーブル 2 4 0 に記憶されたという事実に依存している。したがって、2 つのテーブルにおける結合は、ある特定のトランザクションに属するすべての S Q L ステートメントをそれらの関連のコミットシステム変更番号 2 4 3 と一致させる。さらに、S Q L O R D E R B Y 句は、それらのコミットシステム変更番号に従って、増大するソート順序で、つまり、トランザクションがコミットした元々の順序で、変更行を戻す。

【誤訳訂正 2 1】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 4 3



## 【訂正方法】変更

## 【訂正の内容】

## 【0043】

さらに、エンドユーザまたはサブスクライバアプリケーション121は、（コミットシステム変更番号を含まない）変更テーブルと（コミットシステム変更番号と他をわずかに含む）トランザクションテーブルとの間に結合があるという事実を認識しなくてもよい。特徴は、変更データの各ユーザに代わって生成されるサブスクライバビュー251を提供することによって、達成され得る。上の例では、CV1と称される対応のサブスクライバビュー251は、以下の構成で作成され得る。

```
CREATE VIEW CV1 (CSCN, RSID, A, B, C) AS
SELECT TT.CSCN CSCN, CT1.RSID, CT1.C1 C1, CT1.C2 C2, CT1.C3 C3
FROM TT,CT1
WHERE TT.XID=CT1.XID
```

このサブスクライバビュー251を用いると、サブスクライバアプリケーションは、以下の簡潔なSQLクエリを実行するだけでよい。

```
SELECT * FROM CV1 ORDER BY CSCN
```

この選択クエリは、トランザクションがリレーショナルデータベース管理システムに対してコミットされた正確かつ正しい順序で、トランザクションのためのすべての変更行を戻す。さらに、変更のみがコミット順序に従って構成されるのではなく、トランザクション内のさまざまな動作が以下のクエリ内でのように下位順序付け(sub-ordered)され得るように、

追加的な順序レベルも可能である。

```
SELECT * FROM CV1 ORDER BY CSCN, RSID
```

さらに、同期変更テーブルおよびトランザクションテーブルの両方の内容は、それらが無制限に増大しないように、ページされ得る。なぜならば、結合を用いることによって、ページされる必要のあり得る変更テーブルのすべての行が識別され、この後、トランザクションテーブル自体からの対応の行がページされ得るためである。

## 【誤訳訂正22】

## 【訂正対象書類名】図面

## 【訂正対象項目名】図3

## 【訂正方法】変更

## 【訂正の内容】

【図 3】

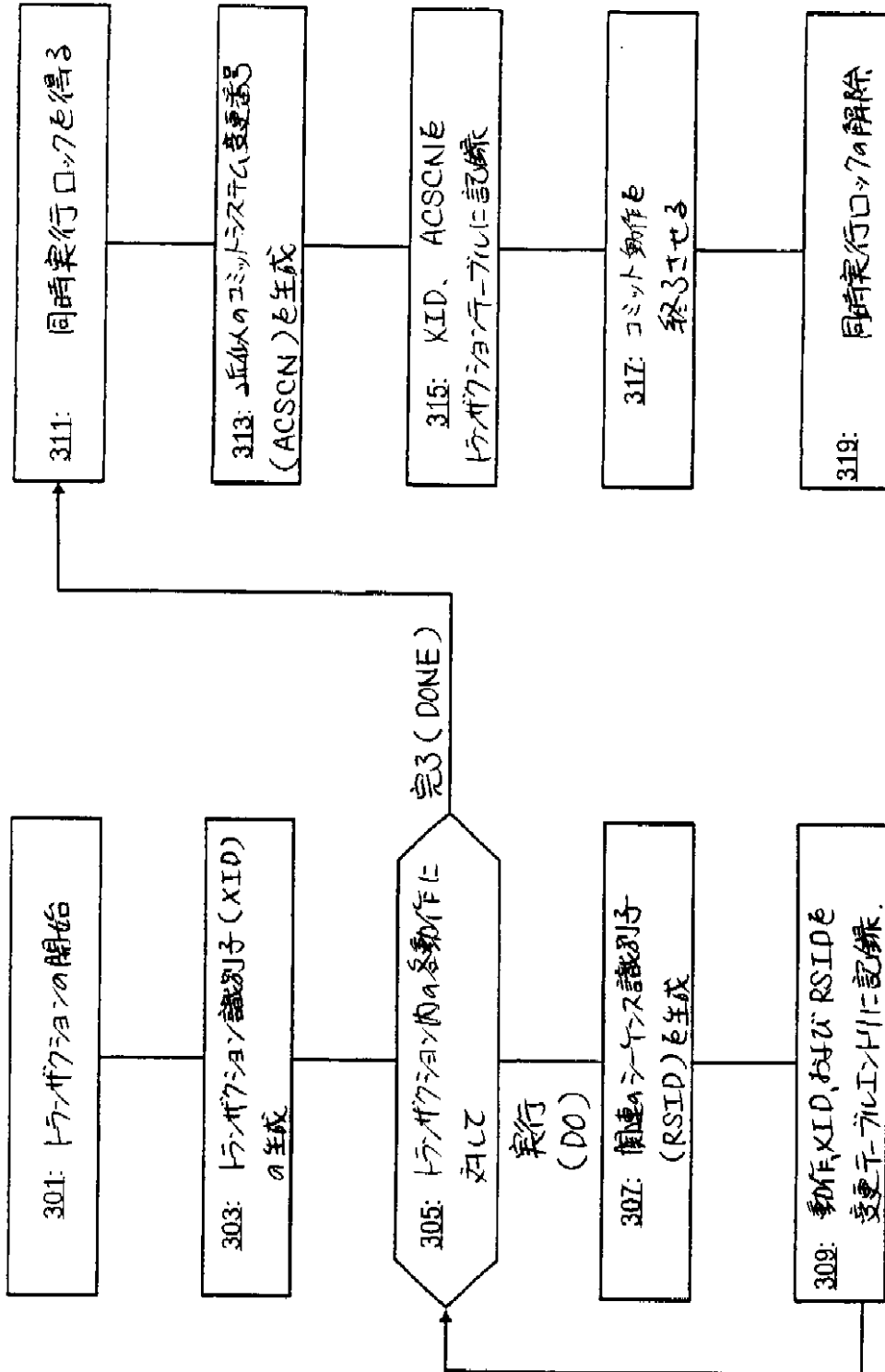


FIG. 3