(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4285770号
(P4285770)
(45)発行日　平成21年6月24日(2009.6.24)　(24)登録日　平成21年4月3日(2009.4.3)

(51)Int.Cl.　F I
*G06F 17/30 (2006.01)*　G06F 17/30 180D
G06F 17/30 230Z

請求項の数 6　(全 25 頁)

(21)出願番号　特願平9-522575
(86)(22)出願日　平成8年12月16日(1996.12.16)
(65)公表番号　特表2000-502201(P2000-502201A)
(43)公表日　平成12年2月22日(2000.2.22)
(86)国際出願番号　PCT/GB1996/003102
(87)国際公開番号　WO1997/022939
(87)国際公開日　平成9年6月26日(1997.6.26)
審査請求日　平成15年12月5日(2003.12.5)
(31)優先権主張番号　9526096.4
(32)優先日　平成7年12月20日(1995.12.20)
(33)優先権主張国　英国(GB)

(73)特許権者
ブリティッシュ・テレコミュニケーションズ・パブリック・リミテッド・カンパニー
イギリス国、イーシー１エー・７エージェイ、ロンドン、ニューゲート・ストリート　８１
(74)代理人
弁理士　鈴江　武彦
(74)代理人
弁理士　村松　貞男
(74)代理人
弁理士　橋本　良郎
(74)代理人
弁理士　白根　俊郎

最終頁に続く

(54)【発明の名称】リレーショナルデータベースのためのインデックス指定

(57)【特許請求の範囲】
【請求項１】
コンピュータ化されたデータベースシステムにおいて、マシン読取り可能な形式で記憶されているデータベース表と組合わせて使用するためのインデックスのセットを、コンピュータのソフトウエアに基づいてそのハードウエア資源を使用して決定する方法において、前記コンピュータ化されたデータベースシステムは、データ照会に応答して、データ表中に含まれているデータから識別されるデータセットをユーザに提示することができるように構成されており、
対応するインデックスをつけることができる術語を識別するために、コンピュータ化されたデータベースシステムに実行を依頼された複数のデータ照会を解析するステップと、
それぞれのデータ照会に応答してコンピュータ化されたデータベースシステムによって要求される処理量を減少させることのできるインデックスの候補となる複数の候補インデックスを、前記解析ステップで識別されたインデックスをつけることができる術語中から識別するステップと、
識別された候補インデックスを使用して形成されているコンピュータ化されたデータベースシステム中に記憶されている実際のデータの表について、コンピュータ化されたデータベースシステムにより受取られた実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムにより要求される処理量の減少量を示す各候補インデックスに関するコスト節減を、前記識別するステップで識別された複数の候補インデックスのそれぞれに対して評価するステップと、

複数の候補インデックスのそれぞれに対する評価されたコスト節減に基づいてコンピュータ化されたデータベースシステムで使用される好ましいインデックスのセットを決定するステップとを含んでおり、
コスト節減は候補インデックスを使用せずにコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理量と、候補インデックスを使用してコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理量との差を表す処理量であり、
前記コスト節減を評価するステップにおいて、候補インデックスを使用せずにコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理の処理量であるベースコストが計算され、各候補インデックスを使用するコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理の処理量を測定し、この測定された処理量をベースコストから減算することによって評価が行われるインデックスのセットを決定する方法。

【請求項２】
前記コスト節減の評価において、許容される最大数よりも多くのエントリが含まれている実際のデータ表から読取られたエントリから代表的なものとして選ばれたサンプルだけを含む縮小データ表を生成し、この縮小データ表の生成のために実際のデータ表について、コンピュータ化されたデータベースシステムにより受信された実際のデータ照会の代表的なサンプルに対してコンピュータ化されたデータベースシステムによって行われる処理の処理量を測定する請求項１記載の方法。

【請求項３】
前記実際のデータ表についてのデータベース統計が前記縮小データ表においても使用される請求項２記載の方法。

【請求項４】
コスト節減の前記評価には、インデックスを記憶するオーバーヘッドの評価が含まれている請求項１記載の方法。

【請求項５】
コンピュータ化されたデータベースシステムにおいて、マシン読取り可能な形式で記憶されているデータベース表と組合わせて使用するインデックスのセットを決定する装置において、前記コンピュータ化されたデータベースシステムは、データ照会に応答して、データ表中に含まれているデータから導出されるデータセットをユーザに提示することができるように構成され、
対応するインデックスをつけることができる術語を識別するために、コンピュータ化されたデータベースシステムに実行を依頼された複数のデータ照会を解析する解析手段と、
それぞれのデータ照会に応答してコンピュータ化されたデータベースシステムによって要求される処理量を減少させるのに役立つインデックスの候補となる複数の候補インデックスを、前記解析ステップで識別されたインデックスをつけることができる術語から識別する識別手段と、
識別された候補インデックスを使用して形成されている実現されるコンピュータ化されたデータベースシステムに記憶されている実際のデータ表に対してコンピュータ化されたデータベースシステムが受取る実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムにより要求される処理量の減少を示す各候補インデックスに関するコスト節減を、前記識別するステップで識別された複数の候補インデックスのそれぞれに対して評価する評価手段と、
複数の候補インデックスのそれぞれに対する評価されたコスト節減に基づいてコンピュータ化されたデータベースシステムで使用される好ましいインデックスのセットを決定する

決定手段とを備え、
前記コスト節減を評価する評価手段は、前記識別するステップで識別された複数の候補インデックスのそれぞれに対して評価し、コスト節減とは候補インデックスを使用せずにコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理量と、候補インデックスを使用するコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理量との差を表す処理量であり、
前記コスト節減を評価する評価手段は、候補インデックスを使用せずにコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理の処理量であるベースコストを計算し、各候補インデックスを使用するコンピュータ化されたデータベースシステムによって受信された実際のデータ照会の代表的なサンプルに応答してコンピュータ化されたデータベースシステムによって行なわれる処理の処理量を測定し、この測定された処理量をベースコストから減算することによって評価を行うように構成されているインデックスのセットを決定する装置。

【請求項６】

請求項５記載の装置を具備し、データ照会に応答してマシン読取り可能な形式で格納されている前記実際のデータ表の中に含まれているデータから導出されるデータセットをユーザに提示するように構成されているコンピュータ化されたデータベースシステム。

【発明の詳細な説明】

発明の背景

発明の分野

本発明は、リレーショナルデータベースのためのインデックスの指定に関する。本発明はまた、インデックスを指定するためのプロセスを含むリレーショナルデータベースに関する。

従来技術の説明

データの照会（enquiry）に応答してデータベース内に含まれたデータから１セットのデータを得るように実行可能な命令が構成されるデータ処理環境が知られている。データはデータ表から直接にアクセスされてもよく、そこにおいてそれは要求された情報を得るように表内の全てのエントリをサーチするために必要である。その代りに、サーチ過程において、サーチプロセスの速度を実質的に増加させるためにインデックスを利用することもある。

大きく、多用されるデータベース用のインデックス構造の設計は、現在では実行に際して非常に困難であり、エラーが導入され易い。この問題は、人間のデータベース管理者が、日常のベースでデータベースに対してランする一般に何百あるいは何千もの異なる構造化照会言語（ＳＱＬ）ステートメントに索引（インデックス）付けする要求を検知し、それからこれらの要求をデータベース全体を通じて定められた好ましいインデックスのセットに変換することが不可能であるという技術的な要求ならびに制約があるために生じる。しかしながら、貧弱な特定がされたインデックスの設計によって、結果的にＳＱＬステートメントはプロセッサの設備を大幅に消費し、あるべきすがたよりも長時間運行する結果となり、また、結果的に装置がひどくオーバーロードにされてしまう。

長い間、ターゲットワークロードと呼ばれる典型的なＳＱＬワークロードに対して、所定のデータベースの設計に対して定義されたインデックス構造を全体的に指定する方法が必要とされてきた。しかしながら、人間のオペレータの器官での直観的なメンタルプロセスを要求せずに使用できる処理能力を利用することによって技術的な解決方法を実現することが困難である場合、この技術的な問題は解決されない。

発明の概要

本発明の第１の特徴によれば、読取り可能な形態でマシン中に記憶されたデータベースに対するインデックスのセットを指定するために、前記データベースに供給された複数のス

テートメントを解析し、前記データベースの表から得られたインデックスを識別し、前記インデックスが使用可能であるときに達成可能な改良された動作のレベルを評価し、前記データベースに対してインデックスのセット（組、群）を指定するために前記評価されたレベルを処理するステップを含んでいる方法が提供されている。

好ましい実施形態において、改良された動作のレベルは、前記表の性質に関連した情報から得られたデータベース表の縮小されたモデルを生成することによって評価される。一般に、縮小されたモデルは、表毎に領域中に５０００個のデータエントリを含んでいてもよい。モデルデータベースはモデル化されたライブデータベースから得られた代表的なデータエントリで充たされていることが好ましく、前記モデルは、ライブデータベースの現在のインデックスのカーディナリティを考慮することによって充たされる。さらにデータベースモデルはライブデータベースの現在のインデックス内のエントリの分布を考慮することによって充たすことができる。

好ましい実施形態において、データベースの統計は、ライブデータベースからデータベースモデルにコピーされる。ベースレベルのコストは付加的なインデックスがない状態でステートメントを実行するために計算されることが好ましい。コストレベルは、実行時間を評価することによって得られることが好ましい。さらに、コストレベルは、インデックスのメンテナンスオーバーヘッドを査定することによって評価されてもよい。

本発明の第２の特徴によれば、リレーショナルデータベースに対してインデックスのセットを指定するように構成され、前記データベースに供給された複数のステートメントを解析する解析手段と、前記データベースの表から得られたインデックスを識別するための識別手段と、前記インデックスが使用可能である場合に達成可能な改良された動作のレベルを評価する評価手段と、前記データベースに対してインデックスのセットを指定するために前記評価されたレベルを処理する処理手段とを具備しているインデックスのセットの指定手段が提供される。

好ましい実施形態において、可能性のあるインデックスは、前記ステートメントによって定められた述語のセットから識別される。

本発明の第３の特徴によれば、データベースのためのインデックスのセットを指定するように構成され、データ記憶手段と、データ処理手段と、前記データ記憶手段から読取り可能なプログラム命令とを具備しているデータ処理装置が提供され、そこにおいて前記処理手段は、前記命令に応答して、前記データベースに供給された複数のステートメントを解析し、前記データベースの表から得られたインデックスを識別し、前記インデックスが使用可能である場合に達成可能な改良された動作のレベルを評価し、前記データベースに対して好ましいインデックスのセットを指定するために前記評価されたレベルを処理するための手段を提供するように構成されている。

好ましい実施形態において、コストの節減は、古いＳＱＬステートメントのコストのコスト値と新しいＳＱＬステートメントのコストのコスト値を可能なインデックスによって処理することによって計算される。コストの節減は、古いコストから新しいコストを減算することによって計算されてもよい。コストの節減は、新しいそれぞれの可能なインデックスをそのそれぞれの表に関して考慮することによって表に対して計算されることが好ましい。

好ましい実施形態において、可能なインデックスは、好ましいインデックスとして指定される可能性の点から順序付けされる。インデックスの組合わせは、現在可能性のあるインデックスをランダムに組合わせ、好ましいインデックスのセットを指定するために前記評価されたレベルを処理することによって識別される。

本発明の第４の特徴によれば、マシン読取り可能な形態で記憶された複数のデータ表と、ステートメントに応答して前記データ表を処理し、前記データ表の処理を容易にするためにインデックスを生成する処理手段とを具備し、さらに、好ましいインデックスのセットを指定するために前記処理手段によって実行可能な命令を含んでいるリレーショナルデータベースが提供され、そこにおいて前記命令は、データベースに供給された複数のステートメントを解析し、前記データベースの表から得られたインデックスを識別し、前記イン

デックスが使用可能である場合に達成可能な改良された動作のレベルを評価し、前記データベースに対して好ましいインデックスのセットを指定するために前記評価されたレベルを処理するように構成されている。

【図面の簡単な説明】

図１は、データ解析システムを含む遠隔通信環境を示している。
図２は、図１において示され、データ表を記憶するように構成された複数の大容量ディスク記憶装置を有するデータ解析システムの詳細を示している。
図３Ａ、３Ｂ、４Ａおよび４Ｂは、図２に示されたデータ記憶装置上に記憶されたタイプのデータ表の例を示している。
図５および図６は、図３Ａに示された表に含まれたデータから得られたインデックスの例を示している。
図７は、図２に示された環境内において設置されたデータベース構造を示しており、インデックスを指定する処理を含む。
図８は、図７に示された環境内で実行された処理を示しており、インデックスのセットを指定する処理を含む。
図９は、図８において示されたインデックスのセットを指定するプロセスを示しており、ライブデータベースをモデル化する処理と、典型的なＳＱＬステートメントを解析するための処理と、ベースコスト計算のための処理と、可能性のあるインデックスを識別するための処理と、好ましいインデックスを指定するための処理とを含む。
図１０は、図９において示されたライブデータベースをモデル化するプロセスを詳細に示している。
図１１は、図９において示されたＳＱＬステートメントを解析するプロセスを詳細に示している。
図１２は、図１１において示されたプロセスを使用して生成されたデータの表を示している。
図１３は、図９において示されたベースコストを計算するプロセスを示している。
図１４は、図９において示された可能性のあるインデックスを識別し、順序付けるプロセスを詳細に示している。
図１５は、図９において示されたインデックスの好ましいセットを指定するプロセスを詳細に示している。
図１６は、図１５において示されたプロセス中に生成されたデータの一例を詳細に示している。
図１７は、図１５において詳細に示されたプロセスを使用する新しいインデックスの組合わせの生成を示している。
図１８は、ライブデータベース内でインデックスを設定する手順を示している。

実施例

以下、本発明を、先に示された添付図面を参照して単なる例示として説明する。
遠隔通信環境が図１に示されており、そこにおいて、電話の送受話器、ファックスマシンおよびモデム等の複数の通信ユーザ装置101がそれぞれのローカルアナログライン103を介してローカル交換機102に接続されている。ローカル交換機102においてアナログ信号がデジタル化され、それに続いてデジタル領域内でスイッチングおよび再経路設定が行われる。これによって結果的に多数の呼がデジタル時分割多重化チャンネル104を通ってトランク通信ネットワーク105に導かれる。
ローカル交換機102およびトランクネットワーク105は、通常の通信のスイッチングを行い、呼が通常の方法で接続されることを許容する。さらに、一層進んだ（アドバンスト）サービスがアドバンストサービスノード106によって提供され、それによって顧客が記憶および順方向の転送ならびに個人番号の識別等のアドバンストサービスにアクセスできるようにする。顧客は、トランクネットワーク105に接続されたデジタルマルチプレクサ107を介してアドバンストサービスノード106にアクセスする。従って、呼はトランクネットワーク105を介してアドバンストサービスノード106に導かれ、その後、呼出ししている顧客

に情報が送り返され、呼がトランクネットワーク105を介して端末装置101等に再度経路設定される。その代りに、アドバンスト・サービスノード106の機能がトランク遠隔通信ネットワーク105を通して分配されてもよい。
呼が生成されたとき、呼に関する料金の明細は、関連したローカル交換機102において記憶される。続いて、接続された顧客が費用を負担する使用を表すこの呼情報は、通信チャンネル109を介して中央管理装置108に供給される。この方法において、全ての料金情報は、顧客の勘定の発生および分配を管理する中央管理装置108に導かれる。
動作において、端末装置101、ローカル交換機102、トランクネットワーク105およびアドバンストサービスノード106で構成されたシステムは非常に多数の呼を接続し、その結果、動作データの集合体が生成される。第１に、このデータは、発生している呼の性質、目的地の呼の性質および呼のタイプに関する詳細を識別する。呼のタイプに関する情報によって呼は直接のローカルな呼であると識別されることもあり、あるいは呼が長距離あるいは国際間の呼であり、アドバンスト・サービスノード106によって提供されたサービスの使用を含むと識別されることもある。このデータの解析によって、少なくとも２つの顕著な利点が与えられる。第１に、システムの動作特性を表す集められたデータに応答して、システムが動作する方法を変更することができる。従って、特定の領域がその他の領域よりも実質的にアドバンストサービスを使用することが発見された場合、これらのサービスの使用可能性を最適にするためにこれらのサービスの割当てを再度経路指定することが可能である。同様に、ネットワーク使用の評価は、特にオフピーク期間中に使用可能な能力を良好に使用するための努力が行われたときに、マーケティング戦略の設計者が使用できるようにされてもよい。
実際に、ほぼ類似した照会がデータベース上で規則的なインターバルで実行される。動作の分割は特有の関心を有しており、それらの関心が規則的なベースで更新されることを要求する。非常に多数のユーザがデータベースにアクセスでき、一定期間を通じて何百あるいは何千ものＳＱＬ照会がデータベース内に含まれたデータに関して実行され、これらの照会の大部分は新しいデータが含まれた際に更新された結果を生成するために多数回実行される。
図１に示されたシステムは、トランクネットワーク105からデータを受取るように構成されたデータ解析装置110と、アドバンストサービスノード106と、中央管理装置108とを含んでいる。順に、データ解析装置は、データをトランクネットワーク105、アドバンストサービスノード106、および中央管理装置108に戻す。トランクネットワーク105およびアドバンストサービスノード106内で、データ解析装置から戻されて受信されたデータに応答してこれらのシステムの技術的動作に修正が行われる。同様に、中央管理装置108に戻されたデータによって結果的に顧客がインボイスされる（請求書を送られる）方法が変更され、すなわち、一般に顧客がネットワークを使用する方法が修正される。
データは、データ解析装置110において集められ、本来のシステムの設計者によって指定された形態で記憶される。これらの設計者は、後に要求される照会のタイプを予測するように努めるが、重要となる全ての照会を予期することは不可能である。それ故、データはリレーショナルデータベースのタームにおいてデータ解析装置に記憶される傾向があり、それによって特定の照会に応答する後続する操作が容易になる。これらの照会によって結果的にトランクネットワーク105、アドバンストサービスノード106、あるいは中央管理装置108の手順が修正される。さらに、特定の照会に応答して生成されたデータは、参照あるいは次の使用のためにデータ照合装置111において照合される。
データ解析装置110が図２において詳細に示されており、それは実質的にＩＢＭ　ＥＳ９０００等のメインフレームコンピュータ201の周囲に配置され、５０ＭＩＰＳ（million instructions per second）で動作可能な１０個のプロセッサを有して構成されている。ユーザは、複数のネットワークで結合されたユーザ端末202を介してデータベースシステムにアクセスすることができ、データは、データ表の形態でディスク駆動装置203に記憶され、テラバイトの単位の容量のデータを記憶することができる。
トランクネットワーク105等の動作データソースと、アドバンストサービスノード106と、

中央管理装置108が動作データソース204として示されている。さらに、データは外部データソース205によって示されているような別の外部ソースから受取られてもよい。トランクネットワーク105、アドバンスト・サービスノード106、あるいは中央管理装置に戻る動作制御信号流は、動作制御装置206に供給されるデータによって示され、データの照合および印刷は207において示されている。データはディスク記憶装置203上のシステム中に記憶され、出力データは照会に応答して得られ、ユーザがネットワークユーザ端末202を使用することによって始められる。ディスク記憶装置203上に保持されたデータは、図１に示されているようなトランクネットワークあるいは別の装置の動作に応答して継続して集められる。さらに、管理データもまた記憶装置上に保持されてもよく、作動データを管理データに関連させるように照会が始められてもよい。

ディスク記憶装置203上に記憶されるタイプのデータベース表の例が図３Ａ、図３Ｂ、図３Ｃおよび図３Ｄに示されている。通信の各イベントを表す情報が中央管理装置108から受取られる。各イベントは特有のシーケンス番号を与えられ、それによって、図３Ａに示されているようにデータベース中に新しい記録を生成する。記録は、そのイベントの日、開始時間、終了時間、イベントを開始する顧客の電話番号および呼のタイプを識別することによって完了する。従って、1995年12月01日の午前０時１分に、電話番号が404 7241である顧客がタイプＡの呼を行い、午前０時２５分に終了した。この例においては呼のタイプに対して任意の指定が与えられており、ここにおいて、タイプＡの呼はローカルな呼を表しており、タイプＢの呼は長距離の呼を表しており、タイプＣの呼はオペレータによって行われた長距離の呼であり、タイプＧの呼はアドバンスト・サービスの使用を表している。上で識別されたイベントは、イベント番号12345の下に記録されており、次のイベントはイベント番号12346の下に記録される。これは、1995年12月01日の午前０時１分に電話番号が386 4851である顧客によって開始されたものであり、再びこれはタイプＡの呼として識別される。

データ解析システム110内のデータベースはまた、図３Ｂにおいて示されたような管理データを含んでいる。図３Ｂにおいて示された表は、顧客の識別子を顧客の電話番号上にマップする。特定の顧客は異なる電話番号の複数の電話線を有していてもよく、それによって、関連した顧客を識別するために特定の電話番号が必要とされることが理解されるであろう。従って、示された例において、電話番号が404 7241である顧客は、記録303内に顧客識別番号0074895が割当てられている。同様に、記録304は、顧客識別番号057896で識別された顧客に電話番号404 7242が割当てられていることを示している。

図４Ａに示された表は、顧客の識別番号を顧客のアドレスに関連させる。従って、記録305から、識別番号0074895を有する顧客がロンドンのハイホルボーン５２に住んでいることがわかる。

一般的に、特定の都市が常に特定の地理領域内に位置しているとされると、広範囲の地理データを提供する必要はない。しかしながら、地理領域は、商業環境における変化を反映するように特定のオペレータによって調整されてもよい。町および市を特定の地域にマッピングしている別の表が与えられている。従って、記録306によって識別されているように、ロンドンは南東地域にマッピングされ、ラフバラは記録307において東部内陸地域にマッピングされている。地域のオフィスの位置を反映するために地域境界線を調整することができ、それによって、例えば、東部内陸地域は西部内陸地域と結合され、内陸地域とされる。そのような環境の下では、図４Ａに示された表を変更する必要がなく、図４Ｂに示された表を変更することが必要とされるだけであり、図４Ｂに示された表が有している記録は図４Ａに示された表よりも実質的に少ない。

イベント番号を参照して照会が行われた場合、図３Ａに示されたデータ表によってデータ記録が非常に速く読取られることが認識されるであろう。図３Ａに示されたデータ表はイベント番号に関して順序付けられ、そこにおいて、各イベント番号は特定の記録に特有のものである。従って、イベント番号12345が与えられると、データベースは電話番号が404 7247である顧客がタイプＡの呼を行ったことに迅速に応答することができる。同様に、図３Ｂに示された表を参照することによって、この電話番号を顧客の識別子に関連付ける

ことができ、それによって、図４Ａおよび図４Ｂにそれぞれ示された表に関して照会するためにアドレスおよび地域が識別される。
しかしながら、図３Ａの表に示された記録内の別のフィールドに関して照会が行われた場合に問題が生じる。例えば、照会は、特定の電話番号によって開始された全てのイベントについてのリストが生成されることを要求してもよい。その代りに、照会は、特定の呼のタイプの全てのイベントに関して行われてもよい。より複雑な照会はこのタイプのエントリを使用して行われてもよく、例えば、照会は南東地域から開始されたすべての呼の詳細を要求して行われ、１０分以上続いてもよい。
先の例によれば、簡単な照会は、特定の電話番号によって開始された全てのイベントの詳細を要求して行われる。図３Ａに示された表は、電話番号のエントリの下で索引付けされておらず、それ故、プロセッサは表内の全ての記録の電話番号データフィールドをサーチする必要がある。明らかに、これにはイベント番号に関する記録の読取りよりも実質的に多くのプロセッサリソースが要求される。図３Ａに示された表はイベント番号のもとに順序付けられており、それによって、イベント番号が与えられると、特定の記録が非常に迅速に識別される。しかしながら、任意の別のフィールドの下で索引付けされないと、関心のある特定のフィールドを識別するためにサーチを行う必要がある。先に識別された一層複雑な例によれば、表内に含まれた全ての記録を通して異なるフィールドを数回サーチすることによって特定の照会を満足させる必要がある。
サーチが行われる速度を改良するために、迅速なサーチが別のデータフィールドに関しても行われるように特定の表のためのインデックスを生成することが可能である。図５に示されているように、表３Ａ内に含まれたデータは、電話番号に基づいてインデックスを生成するために使用されてきた。各電話番号は１つずつ考慮され、インデックスは特定の電話番号によって開始されたそれぞれの特定のイベントを識別するために生成される。従って、記録301は電話番号404 7241に対して識別され、イベント12345はこの電話番号に対して記録されている。続いて、別のイベント14876,15739,15928および16047がこの電話番号によって開始された。インデックスは、考慮されている電話番号に関する全てのイベントが記録されるまで継続する。その後、インデックスは、この例においては404 7242である次の電話番号に連続し、これに対してイベント13728,14937,15821,および14723等が記録された。
その後、電話番号404 7243がこの電話番号に対して記録されたイベントと共に考慮され、それは図３に示されたデータ表内の全ての電話番号のエントリが考慮され終わるまで続けられる。図５に示されたインデックスにおいて、与えられた記録の数は図３Ａに示された元の表にある記録の数と等しい。しかしながら、図５の表は単なるインデックスであり、従って、特定の電話番号に対して、インデックスは図３Ａに示された主要な表内の特定のイベント番号に戻って指示することになる。従って、インデックスによって電話番号に基づいてサーチを迅速に行うことができ、その後、図３Ａに示された表の記録内に含まれた残りのデータが得られる。
図５に示されたものに類似したインデックスが図６に示されており、そこにおいて、図３Ａに示された表内に含まれたデータは、イベントのタイプに関して索引付けされている。記録301は、図６に示されたインデックス中の記録601として再生される。イベント番号12345として示されたイベントは、その後生じるタイプＡの呼により生成されたものであり、このイベント番号は、イベントタイプＡのためのエントリに対してリストされたものである。その後、イベント番号13856,14024,15752および14831は、タイプＡの呼を生じるイベントであった。
その後、図６に示されているように、タイプＢの呼に対するイベントが、イベント13728を含んで記録されて記録602に配置され、順次イベントがタイプＢのイベントに対して記録される。一度、タイプＢのイベントに対する全てのイベントが記録されると、表は、イベント番号12350によって開始されたタイプＣのイベントに連続する。
一度、図５に示された電話番号に対するインデックスおよび図６に示されたタイプのイベントのインデックスが生成されると、イベント番号、電話番号およびイベントのタイプに

基づいて図３Ａに示された表内の記録に迅速にアクセスするためにこれらのインデックスを使用することができる。照会に応答したとき、これらのインデックスを使用可能にすることが望ましいことは明らかである。しかしながら、インデックスが使用可能であるという利点は、記憶スペースに対する付加的な要求と併せて、インデックスの生成およびさらに重要であるインデックスの更新について含まれたコストと比較されなければならない。従って、多くの実際に具体化した場合において、使用可能なディスク記憶スペースが不十分である場合には可能性のある全てのインデックスを生成することは不可能である。時には、ディスク記憶スペースの増加が可能であるが、実際に具現化した大半の場合においては、全記憶スペースおよびインデックスの生成のために割当てられる記憶スペースの全体量に関して上限がある。

図２に示されたシステムは、非常に多量のデータを記憶するように構成されている。さらに、このデータは、新しいデータのセットを生成するために照会を行っているユーザからかなり強く要求される。これらの照会の幾つかは一回限りの性質を有しているが、ソースデータに対して行われた追加および変更に応答して要求されたデータのセットに対する変更を査定するために、照会は高い割合で比較的規則的なインターバルで反復的に照合される。これらの状況の下で、データベースに対して行われた数百あるいは数千もの異なるＳＱＬステートメントに応答して最適な動作を達成するためにデータベース管理者が索引付けの要求を正確に受取ることは実質的に不可能となる。しかしながら、最適なインデックスが提供されなかった場合、照会によって中央処理装置に過剰な要求が課せられてしまう。さらに、そのような照会は必要以上に長く実行する傾向があり、それによって遅延が生じる。これらの問題が残っているため、マシンには重い過負荷がかかり、単位時間中に実行できる照会の数が制限される。この結果、マシンの使用が少くなり、実際に照会当りのシステムのコストが増加してしまう。

本発明のシステムは、上述の技術的な問題を克服するための試みにおいて索引付け構造を指定するように構成されている。このシステムは、可能性のあるインデックスの好ましいセットを指定するためにデータベースに与えられるＳＱＬステートメントを解析するように構成され、データベースを参照するＳＱＬステートメントの実行を改良するために使用できる。可能性のあるインデックスがデータベースシステム内で実際に使用可能である場合に達成できる改良された動作のレベルに関して推定評価が行われる。これらの評価から、インデックスの好ましいリストが指定される。従って、データベースの管理者における主観的な制限がマシンによって取除かれて、数字による表示が計算され、インデックスを実際に生成するようにリソースを割当てる前に、システム内に特定のインデックスを含むことが望ましい程度が示される。これらの数字による指示は、インデックスが配置されたときに改良された動作の評価から実際に得られる。

図２に示されたデータベースハードウェアは、前記ハードウェアによって実行される関連したプロセスと共に図７に概略的に示されている。データベースシステムは、ＩＢＭ社の商標“ＤＢ２”の下で許可されたデータベース命令に従って構成されてもよい。結果として生じたＤＢ２環境は図７において710として示され、それはデータ記憶装置702およびＳＱＬ実行プロセス703を含んでいる。

図示された実施例において、３つの表がデータ記憶装置内で定められており、それらは第１の表704、第２の表705および第３の表706として示されている。図示されたデータベースシステム内では、６個のインデックス全てが各表に対して生成され（これは構成に依存して可変である）、それらは破線の領域707により示されている。従って、各表は関連したインデックスのセットを有していてもよく、インデックスのセットの中に含まれた実際のインデックスは、本発明の好ましい特徴によって定義されているように、インデックスのセットの指定装置により実行される方法に従って決定される。データベースはまた、データベースの定義を記憶するように構成されたカタログ709と、カタログ統計とを含んでいる。

ＤＢ２内では、データベース内の表、カラムおよびインデックスに関する統計を得るように構成された“ＲＵＮＳＴＡＴＳ”がユーティリティとして設けられている。カタログは

、表の寸法についての詳細な情報を提供し、ライブデータ記憶装置から類似したデータ記憶装置のコピーにデータが転送される前に空の表が生成されることを可能にする。さらに、ＳＱＬ実行プロセスは、“オプティマイザ”として知られるプロセスを含み、それは特定のＳＱＬステートメントの実行を最適化するようにカタログの統計を解析するように構成されている。

表の寸法を定めるのに加えて、カタログの統計はカラムカーディナリティ、二番目に高い値および二番目に低い値、クラスター比およびデータの分布も記録する。

カラムカーディナリティは、特定のカラム中の異なる値の数を定義し、一方、フル・キーカーディナリティは、ある表について定義されたインデックス中の異なる値の全体数を同定する。顧客表は、１００万行を上回るカーディナリティを有していてもよく、それらのそれぞれは異なる顧客であるが、これらの顧客の性別を識別するカラムは２つのカーディナリティを有しているだけである。同様に、地理的な領域を識別するインデックスのカーディナリティは、１００の桁の低いほうであってもよい。

ＨＩＧＨ２ＫＥＹとして示された二番目に高い値は、特定のカラム内の二番目に高い値を表している。同様に、ＬＯＷ２ＫＥＹとして識別された値は、特定のカラムの二番目に低い値を表しており、この情報を使用すると、特にカラムのカーディナリティと組合わせて考慮したときに、カラム内の可能性のある値の範囲を表示することができる。

クラスタ比によって、インデックス中のデータの順序付けがどの程度良好に元の表におけるデータの順序付けに従うのかが示される。クラスタ用のインデックスが表について定められ、その表中のデータが再構成されたとき、表中の全ての行はクラスタとされたシーケンスにされ、クラスタ比は１００パーセントであると考えられる。異なるカラムと、クラスタ用のインデックスに対するカラムの順序とを有する別のインデックスはそれらのデータがクラスタ用のインデックスに関してうまくクラスタとされていないこともあり、値の低いクラスタ比を有する傾向がある。行が表に挿入されたり、表から削除されたりすると、インデックスのクラスタ比は徐々に減少し、より多くの行がクラスタ用のシーケンスから外れるようになる。従って、クラスタ比は、この実施形態においてデータがどの程度うまく特定のデータインデックス内で順序付けられ、また、それぞれの表内でライブデータのサンプルから発生されるかの表示を与える。データの分布の統計によって、最も頻繁に発生する１０個の値がそれらの発生の百分率と共にカラム中で定められる。

入力された情報は実質的に２つの形態のライブデータベースシステムに供給される。第１に、入力ライン715によって示されているように、新しいデータがシステムに供給され、第２に、入力ライン716によって示されているように、ＳＱＬステートメントの照会がデータベースに供給される。ライン716上の入力された照会は、ＳＱＬ実行プロセス703によって実行され、出力ライン717によって示されているように出力を生成する。入力ライン715上で受取られた新しいデータによって結果的にデータ記憶装置702内の表が更新され、この更新プロセスは、ＳＱＬ指令に従って実行される。従って、入力されたデータおよびＳＱＬの両者の変更、削除および照会は、ＳＱＬ実行プロセス703に供給され、両方とも前記プロセスの制御の下で実行される。

一般に、入力ライン716上に供給された問合わせすなわち照会の形態のＳＱＬステートメントは、ＳＱＬプロセッサ703がそれぞれの表からの主データに加えて多数のインデックスにアクセスした場合に一層迅速に実行される。結果的に、このタイプの照会に主として応答するためにデータベースシステムが要求される場合、多数のインデックスをシステム内に含むことが望ましい。しかしながら、入力ライン715上で受取られた新しいデータに応答して、あるいは変更または削除されたデータに応答して表が更新されることが要求されたとき、多数のインデックスが存在する場合にはＳＱＬ実行プロセス703の上に大きい処理オーバーヘッドが置かれる。従って、データベースシステムが主としてデータの保管所として要求され、システムへの照会が最小である場合、ハウスキーピング（準備）オーバーヘッドを減少するようにシステム内のインデックスの数を最小にすることが望ましい。大半の実際のシステムにおいて、両方のタイプの入力が受取られ、それ故、ハウスキーピングオーバーヘッドを減少するためにシステム内にあるインデックスの数が最小にされ

るが、記憶のスペースがあるならば、最適な動作性能を提供するために現在のインデックスの選択が最適にされるべきである。
表のインデックスの明細が理想的な解決方法に近付く程度は、入力ライン716上に供給されたＳＱＬ照会の性質に依存している。システムは多数のインデックスを供給されるが、これらのインデックスは、典型的なＳＱＬ照会内で指定された述語に関連していない場合にはほとんど利益がない。同様に、頻繁でないＳＱＬ照会に優先して規則的に生じるＳＱＬ照会を満足させるようにインデックスが構成された場合、使用可能なインデックスから最大の利益が得られる。しかしながら、特に一般的ではないが、このタイプの照会に対して高い優先度が与えられなければならないと言うような（既存のシステムの正当化に）関係がある動作に対して幾らかのＳＱＬ照会は非常に重要である可能性があることが認識されなければならない。それ故、多くの競合する制限がインデックスのセットの指定装置（スペシファイヤ）708に課せられることがわかり、それ（708）は表のインデックスの最適なセットを提供しようと試みるものであることが認められる。ＳＱＬ実行プロセス706は、ＳＱＬステートメントを満足させるためにデータ記憶装置702内に含まれたデータを獲得および操作するのに最適な方法を評価するように構成されたオプティマイザプロセスを含んでいる。オプティマイザは、データベースに対して実行される各ＳＱＬステートメントを検査し、ステートメントを満足させることができる多数のアクセスパスのそれぞれを評価する。それぞれ可能性のあるアクセスパスは、実行される特定のパスに対して要求される処理の量ならびにディスクアクセスの量を表すコストを割当てられる。その後、オプティマイザは、ＳＱＬステートメント内で要求された機能を実行するための実際の通路としてコストが最低のアクセス通路を選択するように構成される。
ステートメントの最適化は動的ＳＱＬとして知られている実行の際に行われてもよく、そこにおいて、各ＳＱＬステートメントの内容は通常、ある回の実行と次の回の実行とでは変化している。その代りに、最適化はステートメントが実際に実行される前に一度実行されてもよく、その結果はＤＢ２プラン内に記憶される。このタイプの最適化は静的ＳＱＬとして知られており、ＳＱＬステートメントが実行の前に知れたときに使用される。静的ＳＱＬは、最適化処理がＳＱＬステートメントのためにたった１度だけ行われ、その結果が後の反復実行のために記憶された場合には、システムのもつ特徴よりも一層効果的となる。
好ましいインデックスセットの指定装置（スペシファイヤ）708は最適化の準備をし、別の段階、すなわち、使用可能なインデックスのどのインデックスを使用するかについてＳＱＬ実行プロセス703内でオプティマイザがオンラインで決定する前に、どのインデックスが実際に生成されるべきかを指定するように構成されている段階を処理する。しかしながら、最適化の実行に加えて、指定装置708は、インデックスのハウスキーピング、実行頻度およびステートメントの優先度を考慮に入れなければならない。
指定装置708は、システムによって実行される典型的なＳＱＬステートメントのサンプルに応答してインデックスの好ましいセットを指定する。この情報を得るために、ＳＱＬ追跡（トレース）プロセス718は、入力ライン716上に供給された照会を追跡し、それによって、適切な期間の後、ＳＱＬステートメントの代表的なサンプルがライン719を通って指定装置708に供給される。指定装置708は、カタログの統計ならびにデータ記憶装置からの表データのサンプルを読取り、結果的に表データがライン720を通って供給される。インデックスセットの指定プロセスが実行された後、出力ライン721を通って出力信号が供給され、プリンタ722により目で読取ることができる形式で指定データを生成することができるようになる。さらに、ＳＱＬの指示がライン723を通ってＳＱＬ実行プロセッサ703に供給され、結果的にインデックスの好ましいセットがデータ記憶装置内で生成される。さらに、プリンタ722によって生成された情報は、インデックスの好ましいセットが構成されるようにするためにデータ記憶装置内に付加的な記憶装置が必要であることをオペレータに知らせることができる。
最適なインデックスのセットの指定装置708によるプロセスの大要が図８に示されている。最初に、ステップ801においてＳＱＬ追跡装置718が活性化され、それによって、一定期

間にわたって、データベースにアクセスするために使用されたＳＱＬステートメントがＳＱＬ追跡装置によって記録される。最終的に、ＳＱＬステートメントのサンプルが集められ、表のインデックスが再度指定されることが決定される。
この段階において、データベースは事実上ラインから外されることができ、それによって、新しい表のインデックスが指定されるまでさらに照会を行うことはできない。このような環境の下では、インデックスのセットの指定プロセスはデータベースそれ自体に共通のハードウェアプラットフォーム上で実行されることが好ましい。その代りに、別の実施形態において、インデックスのセットの指定プロセスの手順は、データベースのプラットフォームとの間で送受されるデータを使用して別個のプラットフォーム上で独立して実行されてもよい。
インデックスのセットの指定命令は、磁気ディスク、光学ディスク、あるいは光磁気ディスク等の適当なデータ搬送媒体を使用して外部プラットフォームに供給される。その代りに、命令はネットワーク能力を介して付加的なプラットフォームに供給されてもよい。インデックスのセットの指定装置708に対する命令の負荷は、取外し可能なディスク724によって示されている。
ステップ802において、ライブデータベースの各表のスペースのカタログ統計はＲＵＮＳＴＡＴＳを使用して更新され、それによって、情報がインデックスのセットの指定装置708に供給されたときにカタログからの更新されたデータが使用可能となることが確実にされる。
ステップ803において、インデックスのセットの指定プロセスが実行され、インデックスのセットが指定される。ステップ804において、ディスクのスペースがより多く設けられるか否かに関して質問が行われ、その答えが肯定であった場合、インデックスを生成するためのディスクの記憶の割当てが増加する。その代りに、ステップ804において行われた質問が否定であった場合、結果的に制御はステップ806に導かれる。
ステップ806において、指定されたセットの詳細が印刷されるかどうかに関して質問され、答えが肯定である場合には、恐らくは通常の並列のインターフェイス接続の形態で印刷信号がプリンタ接続721を通じてプリンタ722に供給される。その代りに、ステップ806において行われた質問が否定である場合、結果的に制御はステップ808に導かれる。
ステップ808において、ステップ803において行われた指定がライブシステムについて実行されるか否かに関して質問が行われ、答えが肯定である場合、ディスクのスペースに制限があることを条件としてステップ809において実行される。その代りに、制御がステップ810に導かれると、結果的にデータベースはライン上に戻って置かれる。
好ましいインデックスのセットの指定方法803が図９に示されている。ステップ901において、ライブデータベースは図１０に詳細に示される手順に従って指定装置708のプロセス内でモデル化される。その後、ＳＱＬ追跡プロセス718によって追跡されたＳＱＬステートメントは、図１１に詳細に示された手順に従って指定装置708のプロセスによって解析される。
ライブデータベースのモデル化によって、結果的に図７に示された表に類似して表が生成されるが、実質的にオンラインのライブシステム中にある表よりも小さい。指定プロセス内でカタログ統計手順をランすることが可能であり、それによって結果的にモデル化された表内のエントリのサイズを反映するカタログ統計が生成される。しかしながら、指定装置708によるプロセスはライブシステムの効果的な動作に関連しており、それ故、それはライブシステム内に含まれ、ライブカタログ702に記憶されたようなカタログ統計にさらに一層関連している。従って、ステップ903において、前記カタログからのライブ統計はインデックスのセットの指定装置にコピーされ、それはライブインデックスの不履行（デフォルト）のセットと共に一次キーインデックスを構成し、各表に対してインデックスをクラスタとする。
ステップ904において、ベースレベルのコストの評価が図１２に詳細に示されているように計算され、それによって、潜在的に最適なインデックスが追加されたときにコストの改良が推定される。このコストの差によって、インデックスのセットの指定に関係する次の

処理を行うための目的関数が提供される。
インデックスのセットを識別するために実行された計算のほとんどは基本的に表毎に行われ、その後、表は、インデックスの生成のためにディスクのスペースが使用可能である場合に、どの特定のインデックスがライブシステム上で生成されるかについての査定が行われるときに再び組合わせで考慮される。結果的に、ステップ905において表が選択され、ステップ906において候補のインデックスが識別され、ステップ907においてそれらの目的関数に従って資格のあるインデックスが順序付けられ、ステップ908において最適なインデックスのセットが生成される。その後、ステップ909において別の表が使用可能であるかどうかについての質問が行われ、答えが肯定であった場合、制御はステップ905に戻される。ステップ909における質問の答えが否定であった場合、結果的に制御はステップ910に導かれ、そこでライブシステムに対して適用できそうなインデックスの最適なセットが指定される。
インデックスのセットの指定プロセス内でデータベースの縮小されたモデルを生成するために、図１０においてライブデータベースをモデル化する手順901が示されている。ステップ1001において、カタログ統計がカタログ702から読取られ、その後、空の表がモデル中に生成され、それぞれのカタログの定義によって説明されたように、ライブ表の704,705,706の性質をコピーする。各表のサイズは５０００行に制限され、これは実質的にライブデータベース表中にある行数よりも少ない。
インデックスのセットの指定装置708内で表の縮小されたモデルがデータ記憶装置702中のライブ表を正確に反映することを確実にするために、ステップ1002において生成された空の表にそれらのそれぞれのライブ表から読取られたデータエントリの代表的なサンプルで充たされる必要がある。ステップ1003において、ライブ表はそのそれぞれのカタログと共に選択される。
ステップ1004において、一次キーのカード値が最高であるＨＦ1として示された特定のインデックスを識別するために、既に選択された表に関係付けられ、ライブデータベース内で動作可能なインデックスが考慮される。ステップ1005において、ＨＦ1の第１のカラムのためのＨＩＧＨ２ＫＥＹ、ＬＯＷ２ＫＥＹおよびＣＯＬＣＡＲＤが識別され、ステップ1006において前記第１のカラムに対するデータの分布が決定される。
その後、ステップ1007において、ＬＯＷ２ＫＥＹおよびＨＩＧＨ２ＫＥＹによって定められた範囲内でＨＦ1の第１のカラムに対して１セットのランダムな値が発生される。値の有用性は、ステップ1006において判断して決定されたような頻度分布に従って重み付けされ、それによって、ライブ表から得られた５０００までのエントリでモデル表が充たされたときに、そのモデルにおける値の分布は、その処理の際の要求に関してモデル上でプロセスが行われるのに十分となり、それはそれぞれのライブ表上で実行されたときに出された類似した要求を実質的に反映している。ステップ1007において識別されたランダムに選択された値は、ステップ1008においてライブ表のエントリから読取られ、ステップ1008において読取られたエントリは、ステップ1009においてモデル表に順次書込まれる。
ステップ1010において、第１に、データ表においてエントリがクラスタキーに従って再編成されるようにモデルデータが処理される。各表に対して可能性のあるインデックスが生成され、これらのインデックスの性質に関して統計が集められる。得られたインデックスの統計は製造寸法まで拡大され、それによって拡大された値が蓄えられ、元の表データが削除される。
ステップ1011において行われた質問の答えは、モデル化された表の全てが、データ記憶装置702内に保持されたライブ表からランダムに選択され、頻度に従って加重されたエントリで形成されるまで肯定である。
捕捉されたＳＱＬステートメントを解析する手順902が図１１に詳細に示されている。ステップ1101において、ＳＱＬステートメントは、それが初めて発生した特定のステートメントであるかどうか、あるいはそのステートメントが前に見られたものであるかどうかについて判断して決定するために処理される。従って、それぞれの特有のＳＱＬステートメントは特有のラベルを与えられ、同じＳＱＬステートメントが再び識別された場合、発生

の回数が頻度のカラム中に記録される。
ステップ1102において表が選択され、次に、ステップ1101においてラベルをつけられたＳＱＬステートメントがステップ1103において識別される。従って、手順1103乃至1106は、捕捉されたステートメントのそれぞれの特有の発生に対して実行されるだけである。
ステップ1103において選択されたステートメントがステップ1102において選択された表を使用するかどうかについて、ステップ1104において質問が行われる。質問の答えが肯定である場合、ステップ1105においてステートメントのラベルが適切な表のリストに加えられる。その代りに、ステップ1104における質問の答えが否定である場合、ステップ1105はバイパスされ、制御はステップ1106に導かれる。
ステップ1106において、別のステートメントが考慮されるかどうかに関して質問が行われ、その答えが肯定である場合、制御はステップ1103に戻され、次のラベルをつけられたステートメントが選択されることを可能にする。その代りに、答えが否定である場合、別のステートメントは使用可能でなく、ポインタを識別するステートメントはリセットされ、制御はステップ1107に導かれる。ステップ1107において、別の表が存在するかどうかについて質問され、その答えが肯定である場合、制御はステップ1102に戻され、次の表が選択される。
最終的に全ての表が考慮され、結果的に、ステップ1107において行われた質問の答えは否定である。
ステップ1105において生成された表のリストが図１２において詳細に示されている。そのリストは、元の表を識別する第１のカラム1201と、ステップ1101において指定されたようなそれらのラベルに関してＳＱＬステートメントを識別する第２のカラム1202と、ステートメントの頻度、すなわち、追跡されたセットの内で特定のＳＱＬステートメントが生じる回数を識別する第３のカラム1203とで構成されている。
図１２に示されているように、最初にステップ1102において表１が選択され、その結果、ステップ1005における反復された動作に応答して、ＳＱＬステートメントＡ，Ｂ，Ｃ，乃至ＳＱＬ　Ｚが表のリストに加えられる。その後、表２が選択され、結果的にＳＱＬラベルがこの表で識別され、最終的に表３が選択され、結果的にステートメントラベルがその表と関連される。本発明の実施形態において３つの表が存在しているが、大きいリレーショナルデータベース内で使用される際には任意の数の表が存在してもよいことを認識すべきである。
第３のカラム1203において、発生の頻度が記録されており、それは一般には数千回測定されたものである。従って、ステートメントＡに対してｘ回の発生が記録され、ステートメントＢに対してｙ回の発生が記録され、ステートメントＣに対してｚ回の発生が記録される。
ベースレベルコストを評価する手順904が図１３に示されている。表がステップ1301において選択され、ステップ1302においてＳＱＬステートメントが選択される。ステップ1303において、ステップ1301において選択された表に適用されたときに、ステップ1302で選択されたＳＱＬステートメントを実行するためのコストが評価される。コスト計算は、ＳＱＬ実行プロセッサ703内にあるオプティマイザから得られたタイマーオン値を使用して達成される。しかしながら、タイマーオン値はインデックスの使用だけを考慮し、インデックスのメンテナンスは考慮に入れない。結果的に、好ましい実施形態では、米国フロリダ州のInnovation Management Solutions社により“ＱＣＦ”（商標）の名の下で開発された命令が実行され、それによって、ＣＰＵの使用およびＳＱＬステートメントが実行されるための経過時間に関して、インデックスのメンテナンスの評価と組合わせてインデックスのコスト値が与えられる。これらのコスト値は絶対的な測定結果を表してはいないが、付加的なインデックスが存在するときに類似した方法を実行することによって関連したコスト値を得ることができ、それは付加的なディスクスペースに対する要求と比較されたときに、より高価なインデックスに優先して特定のインデックスを選択することができる目的関数を提供する。
ステップ1304において、各ステートメントに対してステップ1303において計算されたコス

ト値が実行頻度の係数で乗算され、ステップ1305において、結果的な積が優先度係数によって乗算される。その後、ステップ1306において、コストが特定の表に対するベースコストの和に加算され、ステップ1307において、別のステートメントがあるかどうかについての質問が行われる。質問の答えが肯定である場合、制御はステップ1302に戻り、結果的に次のＳＱＬステートメントが選択され、コスト計算手順が繰り返される。
最後に、ステップ1307における質問の答えが否定である場合、ステップ1308において別の表があるかどうかに関して質問される。その答えが肯定である場合、別の表の和がステップ1309において生成されて制御がステップ1301に戻され、それによって、次の表が選択できるようにする。最終的に、ステップ1308において行われた質問の答えは否定となり、それによって制御がステップ905に導かれる。
資格のあるインデックスを識別するために候補のインデックスのコストを計算する手順が図１４に詳細に示されている。ステップ1401において、考慮中の表に関連するＳＱＬステートメントから得られた述語のセットから候補のインデックスが識別され、それは図１２において示されたリストに従ってステップ905で選択されることによって定められる。従って、考慮中の表に関連するＳＱＬステートメントを解析することによって、索引付け可能な述語が識別できる。識別された索引付け可能な述語は特定のカラムを参照し、これらのカラムは表およびＳＱＬステートメントによってグループにされて述語のセットを形成する。これらの述語のセットは、関連したＳＱＬステートメントを満足させるときに利益をもたらす可能性のある候補のインデックスを識別する（すなわち、候補のインデックスが構成される）ための開始点を与える。さらに、識別された各インデックスに対してカタログ統計が生成される。
ステップ1402において候補のインデックスが選択され、ステップ1403においてステップ1302で選択されたインデックスがインデックスセット指定装置708内に保持された表のモデルの一部分として生成される。ステップ1403において新しいインデックスが生成された後、モデル内の関連したカタログのエントリがステップ1404において更新され、それによってインデックスが完全なサイズとなる。
候補のインデックスがモデルデータベースに対して生成される。インデックスのユーティリティを回復するＤＢ２は、インデックスを送り込むために表に対してランされ、ＤＢ２　Ｒｕｎｓｔａｔｓのユーティリティは、統計を集めるためにデータベースに対してランされる。統計は各インデックスに対して集められ、それらはデータベース中に記憶され、プロセス全体を通して使用するためにライブ容量まで拡大される。
ステップ1401において識別された可能性のあるインデックスは、特定のカラムのエントリを使用する可能性のある全てのインデックスの組合わせを含む。従って、カラムは異なる順序で配置され、可能性のある順序が含まれる。同様に、各カラム内のエントリは順次上昇および下降してもよく、再びこれら可能性のある全ての組合わせが存在するようになる。これらの組合わせの全てが実際に候補のインデックスとして要求される訳ではなく、それ故、候補のインデックスを識別するために選択プロセスがステップ1402において行われる。カラム位置はカラムのシーケンスと呼ばれ、上昇および下降している可能性は順序付けと呼ばれる。同じカラムの順序付けを共用するインデックスは一緒のグループにされ、それらの順序付けに関連した差だけを示す。グループ内で定められた全てのインデックス、すなわち、カラムのシーケンスが同じである全てのインデックスがモデル内で生成される。これらのインデックスのそれぞれに対するカタログ統計もまた生成され、その後、表を参照する全てのトラップされたＳＱＬがインデックス上に目標を定められる。その後、データベース内の説明機能が、実際に使用されたグループ内の特定のインデックスを識別するために訓練される。次に、これらのインデックスは選択された候補のインデックスとなり、可能性のあるセットからの残りのインデックスは排除される。その後、生成されたインデックスが削除され、そのグループの全てが考慮されてしまうまで次のグループが考慮に入れられ、結果的に、ステップ1402において開始されたループに制御が導かれる前に、最後に生成されたインデックスが再び削除される。
従って、前述されたように、ステップ1402において候補のインデックスが選択され、ステ

ップ1403において候補のインデックスが生成され、新しく生成されたインデックスに応答してステップ1404においてカタログが更新される。
新しいインデックスがモデル内で生成されたので、それぞれの表を参照するＳＱＬステートメントは、図１３に詳細に示されたベースコストレベルを計算する方法にほぼ類似した方法に従ってコストを計算される。ステップ1405において適当な新しいインデックスによりＳＱＬのコストが計算された後、新しいコスト値がステップ1406において記憶され、その後、ステップ1303において生成されたインデックスがステップ1407において削除される。
ステップ1408において、別のインデックスが存在するか否かに関して質問が行われ、その答えが肯定である場合、制御はステップ1402に戻され、結果的に次に可能性のあるインデックスが選択される。最終的に、可能性のある全てのインデックスに対して全てのＳＱＬステートメントのコストが計算され、結果的にステップ1408において行われた質問の答えは否定となる。
ステップ1406において計算され、表中に記憶されたコスト値によって、ステップ1401において識別された可能性のある各インデックスに対して新しいコストが定められる。これらのインデックスのそれぞれに対して、ステップ904において計算されたベースコストから新しいコストを減算することによってコストの節減が計算される。このコストの節減値は目的関数を表し、そこにおいて、コストの節減値が低いインデックスの方がコストの節減値が高いインデックスよりも適していると考えられる。このコストの節減値は、ステップ1410において各インデックスに対して記憶され、ステップ1411において、別のインデックスが存在しているか否かに関しての質問が行われる。答えが肯定である場合、ステップ1409において次のインデックスに対するコストの節減値が計算され、その後、可能性のある全てのインデックスに対してコストの節減値が計算され、結果的にステップ1411において行われた質問の答えは否定となる。
ステップ1410においてコスト節減値を記憶した結果、それぞれに割当てられたコスト節減値によりインデックスのリストが生成される。これは目的関数を表しており、それ故、ステップ907においてこの目的関数に従って適当なインデックスが順序付けられ、それによって、コスト節減値が高いインデックスが適格性のリストの最上部の近くに配置される。順序付けされた可能性のあるインデックスを処理するための図９に示されたプロセス908が図１５において詳細に示されており、図１５の手順に従って実行された動作の一例が図１６および図１７に示されている。
図１６において、１１個の可能なインデックスが示されており、それらは特有の識別番号1625,1616,1604,1673,1612,1646,1635,1691,1622,1683および1617で表されている。これらのインデックスに対してコスト節減を先に識別し、定めた手順は、最適なインデックスの指定されたセットに含むための優先度を表している。従って、図１６に示された適当な資格のあるインデックスは、各インデックスに対して記録された関連したコスト節減に関して指定される優先度に基づいて順序付けされたものである。これらのコスト節減値に絶対的な意味はなく、前述された手順に従って計算されたコスト節減に関連した指示を与える。すなわち、インデックス1625は７３のコスト節減を行うものとして識別され、インデックス1616は７２のコスト節減を行うものとして、インデックス1604は６８のコスト節減を行うものとして、そして最後にインデックス1617は４のコスト節減を行うものとして識別されている。従って、インデックス1625乃至1617をそれらのコスト節減の適格性の点から順序付けするのに要求された情報は、ステップ803において詳細に説明された手順に従って計算される。
図１５のステップ1501において、コスト節減が考慮され、次の処理を容易にするために標準化される。標準化することによって、コストの節減値が予め定められた範囲内で考慮されることが可能になり、この例においては０乃至９９９９の範囲内で選択される。図１６の1681において示されているように全体のコスト節減が計算され、その値はこの例においては５００である。全範囲がこのコスト節減全体1681で除算されて単位範囲値が与えられ、その後、それはコスト節減値と乗算され、それによって分布値が与えられる。分布値の

計算はステップ1502において行われ、結果的に標準化されたコスト節減が計算される。従って、ステップ1502において実行された手順に従って、インデックス1625に対するコスト節減が1660の値に標準化され、インデックス1616は（７２のコスト節減値から）標準化されて1440の標準化された値になる。同様に、標準化された値は考慮中の全てのインデックスに対して計算され、それによって、合計されたとき、標準化された値は分布内の全範囲の値に等しくなり、それはこの例においては９９９９である。ステップ1503において、別の遺伝学的繰返し（genetic iteration）が要求されるか否かについて質問が行われ、第１の反復においてその答えは肯定である。要求される遺伝的繰返しの数に関して事前に選択が行われ、ステップ1503において計数動作が行われる。

質問の答えが肯定である場合、ステップ1504において０乃至９９９９の範囲内でランダムな数が生成される。このランダムな数は、特定のインデックスを選択するためにステップ1505において使用される。従って、ランダムなインデックスの選択はステップ1505において行われ、特定のインデックスによって提供されたコスト節減の点から加重される。従って、０乃至８０の範囲内の数によって結果的にインデックス1617が選択され、８１乃至４００の範囲内の数によって結果的にインデックス1683が選択され、４０１乃至８２０の範囲内の数によってインデックス1622が選択され、最後に、８５４０乃至９９９９の範囲内の数によってインデックス1625が選択される。特定のインデックスに割当てられた分布数の数はその関連したコスト節減に比例し、それによって、多数回の反復を通じて、平均的にコスト節減が低いインデックスよりもコスト節減が高いインデックスが選択される。しかしながら、コスト節減の低いインデックスが依然としてプールに残っており、遺伝手順に従ってこれらのインデックスを選択することができる。

それぞれの反復の際にインデックスの新しい組合わせが生成され、これらのインデックスの組合わせによって特定のコスト節減が与えられる。これによってインデックスの組合わせがそれ自体のインデックスの識別を与えられるようにし、複合のインデックスが図１６に示された表内に含まれるように、目的関数に基づいて順序付けされる。

従って、ステップ1505において特定のインデックスが選択され、ステップ1506において選択されたインデックスがインデックスバッファに書込まれる。インデックスバッファ1791が図１７に示されており、それは考慮中の特定の表に対して許容されたインデックスの最大の数を表す６個のバッファ位置を含んでいる。図１７を参照すると、各表は特定の実施形態内に最大の６個のインデックスを有して示されているが、この形態は特定のローカルな動作条件を満足させるために調節されてもよい。従って、バッファ1791は６個の位置を有しており、インデックス1625等の選択されたインデックスの識別はこれらの任意の位置に配置され、ランダムなベースで選択されることができる。従って、図１７に示されているように、インデックス1625の識別は、インデックスバッファ1791の第２のバッファ位置に位置されている。

ステップ1507において、第２のランダムな数が０乃至９９９９の範囲内で生成され、結果的に第２の親インデックスがステップ1508において選択される。選択されたインデックスの指示は、図１７の1792に示されているように、第２のインデックスバッファに書込まれる。この例において、ステップ1507において生成されたランダムな数によって結果的にインデックス1604が選択され、バッファ1792内の第４の位置にランダムに位置付けられる。

ステップ1510において、特定のバッファ内の位置間の任意のインターフェイスにおいてバッファ切断位置がランダムに選択される。この例において、切断位置は、図１７の矢印1793によって示されているように、第２の位置と第３の位置との間に位置されている。この切断位置によって、第１の親であるバッファ1791と、第２の親と見なされてもよいバッファ1792とが結合できる。この交換の結果がバッファ1794と1795に示されている。バッファ1794において、インデックス1625は、第１の親1791から得られて第２の位置に配置され、インデックスの識別子1604は親1792から得られて第４の位置に配置される。バッファ1795に示されているように、この結合の別の子孫はインデックス識別子を含んでおらず、それ故、それは役立たずの子（void child）として考えられ、それ以上は考慮されない。従って、図１５のステップ1511において、２つの親の“繁殖”が行われ、結果的にインデック

ス1625および1604を含んでいる子1794が生成される。
潜在的に最適なインデックスの使用可能性をさらに関心の高いものにするために、バッファ1794の内容によって定められた子が突然変異するという遺伝子操作の段階が設けられる。別のランダムな数が生成され、結果的に別のインデックスが選択される。ステップ1512におけるこの突然変異のプロセスの結果、バッファ1796は子1794から得られたインデックス指示で負荷され、また、第６の位置にランダムにインデックス1635が加えられる。ステップ1513において、ステップ1511で生成された子のコスト節減およびステップ1512で生成された突然変異体のコスト節減が評価される。
ステップ1514において、新しいインデックス1794および1796が適格性の順に潜在的なインデックスのプールに加えられる。コスト節減は、考慮中の表を参照して各インデックスに対して計算され、それによってインデックスは、結果的なコスト節減に従ってランク付けされた図１６のリストに加えられることができる。全体のコスト節減が再度計算され、その後、新しく標準化されたコスト節減が新しく加えられたインデックスを含む全てのインデックスに対して再計算される。この標準化されたコスト節減の分布から、ステップ1502において各インデックスに対して値が計算され、さらに遺伝の繰返しの必要があるか否かについて再び質問される。
ステップ1503において行われた質問の答えが再び肯定である場合、０乃至９９９９の範囲内でランダムな数が生成され、ステップ1505において新しいインデックスが選択され、その後、ステップ1506においてこのインデックスの指示が第１の親バッファ1791に書込まれる。また、第２のランダムな数がステップ1507において生成され、それによってステップ1508において第２の親が選択され、その後、ステップ1509においてこの親の指示がバッファ1792に書込まれる。
ステップ1510において切断位置が再びランダムに選択され、ステップ1511において親が育てられ、それらの子孫がバッファ1794および1795に書込まれ、その後、有効な子供から突然変異体が発生する。従って、それぞれ反復することによって、インデックス結合プールに４つまでの新しいインデックスが加えられる。
最後に、ステップ1503において行われた質問の答えが否定である場合、結果的に制御は図８のステップ805に導かれる。図１５に示されたプロセスが繰り返されている間、新しいインデックスの計算が実質的にランダムな方法で識別される。しかしながら、新しい各インデックスは、その結果として生じたコスト節減を判断し決定するために試験され、コスト節減値が高いインデックスが図１６に示されたリストの最上部の近くに配置される。さらに、比較的高いコスト節減値を有することによって、選択する番号の分布の範囲もまた大きくなり、それによってこれらのインデックスが結合のために選択される可能性を大きくする。しかしながら、比較的可能性の低いインデックスはプール中に残るため、そのようなインデックスが選択されることも可能である。十分に反復すると、コスト節減値が非常に高いインデックスの組合わせが識別され、これらのインデックスは図１６に示されたリストの最上部の近くに配置される。最終的に、図１５に示されたプロセスが終了したとき、新しく育てられたインデックスを含むインデックスは、指定の際の生成のためのリストにコスト節減の順に次第に下位になるように記載される。
指定されたインデックスを含むようにデータベースを構成するための図９のステップ910において識別された方法の詳細が図１８に示されている。関連したコスト節減を指定する目的関数は、それらの関連した表に関してのみインデックスに対して考慮される。しかしながら、動作中のデータベースシステムにおいて、複数の表が一緒に機能しなければならない。それ故、所定の記憶スペースが使用可能である場合、記憶スペースは、存在している全ての表に関連したインデックスに対して割当てられなければならない。
ステップ1801において、ライブデータベース内に存在している各表によって使用されたディスクスペースの全体量が計算され、ステップ1802において、ステップ1801で計算された値が合計されて全ての表に必要な全体のディスクスペースが与えられる。ステップ1803において、各表に必要なディスクスペースが全ディスクスペースで除算されて表ごとのベースのディスクスペースの割当てのパーセンテージ（百分率）が与えられる。このパーセン

テージの割当ては、ステップ1804に示されているようにインデックスにスペースを割当て、それによって、インデックスの生成のために割当てられた記憶の相対量は、表に対する記憶スペースの相対的な割当てとほぼ等しくなる。従って、表が多量のディスクスペースを占める場合、この表にわたって動作しているインデックスに対して同様に多量のディスクスペースの割当てが行われる。
ステップ1805において表が選択され、その表に対して得られた最も適切なインデックスがステップ1806において選択される。ステップ1807において、ライブシステム上で生成されるようにステップ1806で選択された好ましいインデックスに対して十分なディスクスペースが割当てられるか否かに関して質問が行われる。この答えが肯定である場合、ステップ1806において選択されたインデックスがステップ1808において生成のために指定される。その代りに、十分なディスクスペースが使用できない場合、ステップ1807で行われた質問の答えは結果的に否定となり、ステップ1808はバイパスされ、制御はステップ1809に導かれる。
ステップ1809において、別の表が存在するかどうかに関して質問が行われ、答えが肯定である場合、制御はステップ1805に戻され、結果的に次の表が選択され、この表に適切なインデックスがステップ1806および1807において考慮される。最終的に、選択された表に対する全てのインデックスが考慮されると、ステップ1809において質問された答えは結果的に否定となる。
ステップ808での質問の答えが肯定である場合、図１８に詳細に示された手順が実行される。従って、新しいインデックス構造がライブデータベース内で生成され、その後、データベースは、その新しいインデックスが配置された状態でステップ810においてオンラインで配置される。
図１４を参照すると、ステップ1401において実行されたプロセスは、述語のセットによって結果的に４つ以上のカラムを要求するインデックスが識別された場合に、非常に時間を浪費するようになる。これらの環境の下で、第１の４つの好ましいカラムが選択され、残りのカラムが暫定的に排除される。選択はフィルタ係数のベースで行われ、フィルター係数が低い４つのカラムが選択される。全ての可能な順序付けの可能性を有してこれらの４つのカラムが処理され、それによって前述されたような候補のインデックスが選択される。
全ての適切なインデックスが決定された後、暫定的に排除された多量のインデックスは、第５の好ましいカラム、第６の好ましいカラムおよび第７の好ましいカラム等を加えることによって組み立てられ、それによって新しい第５のカラムのインデックス、新しい第６のカラムのインデックスおよび新しい第７のカラムのインデックス等が生成される。これらのインデックスは、必要な４つのカラムを含んでいる最も費用効果的な候補のインデックスに関して作られる。これらのカラムは、４つのカラムの候補のインデックスに増加順序で加えられるだけであり、カタログ統計はこれらの新しいインデックスに対して計算され、その後、これらの新しいインデックスはコスト計算されて、適格性で順序付けされたリストに加えられ、先にコストを計算されたインデックスの適格性（適切性）の順に配置される。
図１５を参照すると、ステップ1505において開始された遺伝プロセスのために考慮された適切なインデックスの数は重要であり、結果的に処理時間は比較的長くなる。これらの環境の下で、遺伝プロセスが実行される前に“結合プール”のサイズに上限を置くことが好ましい。一般に、結合プールのサイズは、遺伝プロセスを実行する前に最大３０の適切なインデックスに制限されてもよい。
基本的に、遺伝プロセスの目的は、一般的なＳＱＬの照会のセットを構成するときに、処理オーバーヘッドを著しく減少する複合インデックスが見つかるようにすることである。処理時間を減少するために、特に利点があると考えられるインデックスのセットを加えることによって“プールの準備”を行うことが好ましい。
プールの準備の第１の段階は、存在しているライブデータベースシステムを調査し、それによってどのインデックスが実際にライブシステムにおいて使用されるかについて決定す

ることを含んでいる。その後、これらのインデックスのセットは、前述のように適切なインデックスの集合に加えられる。
プールの準備の第２の段階は、最も適格なインデックスがライブシステム内にあると仮定された後に適格性のランク付けを再考慮することである。従って、先に計算されたような最も適切なインデックスがライブシステムに属しているものとして配置されている開始位置からコスト係数が再考慮される。このライブインデックスがシステムに加えられると、残りの適格なインデックスの幾らかのコスト節減は顕著に変化し、それによって適格性のリスト内でインデックスを効果的に再度順序付けする。再び、最も費用効果的な残りの適格なインデックスがシステムに加えられ、存在しているこの新しいインデックスに基づいてコスト節減が再計算される。このプロセスは繰返され、反復的に６個の新しいインデックスのセットまで行われる。これらのインデックスのセットのそれぞれは、適切な回数で結合プールに加えられる。

【図１】

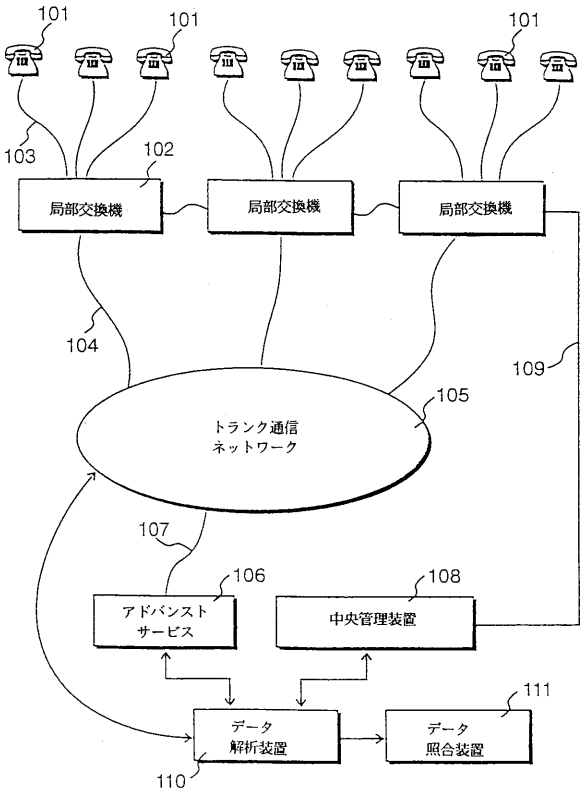


Figure 1

【図２】

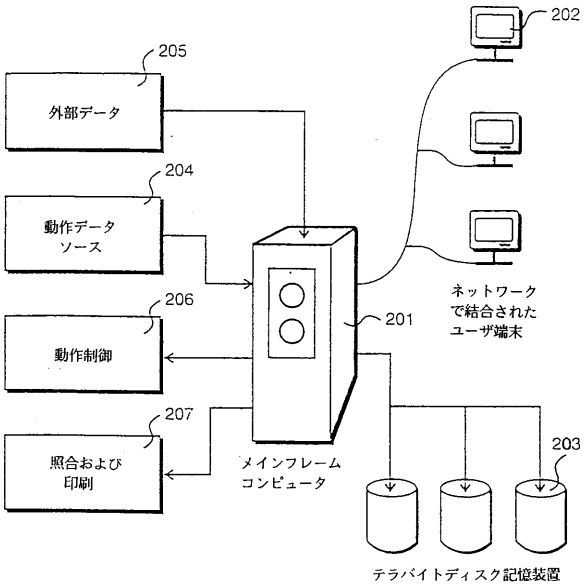


Figure 2

【図３Ａ】

| イベント番号 | 日 | 開始時間 | 終了時間 | 電話番号 | 呼のタイプ |
|---|---|---|---|---|---|
| 12345 | 01-12-95 | 00:01 | 00:25 | 404 7247 | A |
| 12346 | 01-12-95 | 00:01 | 00:10 | 386 4851 | A |
| 12347 | 01-12-95 | 00:01 | 00:53 | 339 1234 | G |
| 12348 | 01-12-95 | 00:01 | 00:14 | 586 1495 | B |
| 12349 | 01-12-95 | 00:02 | 00:15 | 833 1497 | B |
| 12350 | 01-12-95 | 00:02 | 00:36 | 947 8281 | C |
| 12351 | 01-12-95 | 00:02 | 00:05 | 321 4875 | C |
| | | | | | |

301, 302

Figure 3A

【図４Ａ】

| 顧客ＩＤ | ストリート番号 | ストリート | 都市 |
|---|---|---|---|
| 74 893 | 16 | シェパード | イプスウィッチ |
| 74 894 | 47 | ライトフット | リバプール |
| 74 895 | 52 | ハイホルボーン | ロンドン |
| 74 896 | 195 | シアランド | シェフィールド |
| 74 897 | 10 | ダウニング | ロンドン |
| 74 898 | 85 | アベイ | ロンドン |
| 74 899 | 198 | ベーカー | ロンドン |
| | | | |

305

Figure 4A

【図３Ｂ】

| 電話番号 | 顧客ＩＤ |
|---|---|
| 404 7241 | 0074895 |
| 404 7242 | 057896 |
| 404 7243 | 86149 |
| 404 7244 | 83176 |
| 404 7245 | 3238561 |
| 404 7246 | 33987 |
| 404 7247 | 3398762 |

303, 304

Figure 3B

【図４Ｂ】

| 都市 | 地域 |
|---|---|
| リバプール | 北西 |
| ロンドン | 南東 |
| ラフバラ | 内陸東部 |
| 〃 | 〃 |
| 〃 | 〃 |
| | |

306, 307

Figure 4B

【図５】

| 電話番号 | イベント番号 |
|---|---|
| 404 7241 | 12 345 |
| | 14 876 |
| | 15 739 |
| | 15 928 |
| | 16 047 |
| | |
| 404 7242 | 13 728 |
| | 14 937 |
| | 15 821 |
| | 17 423 |
| | |
| 404 7243 | 13 846 |
| | 15 736 |
| | 16 842 |
| | |

501

Figure 5

【図６】

| イベントタイプ | イベント番号 |
|---|---|
| A | 12 345 |
| | 13 856 |
| | 14 024 |
| | 14 572 |
| | 14 831 |
| | |
| B | 13 728 |
| | 14 937 |
| | 15 821 |
| | 17 423 |
| | |
| C | 12 350 |
| | 13 896 |
| | 14 742 |
| | 14 937 |
| | |

601, 602

Figure 6

【図７】

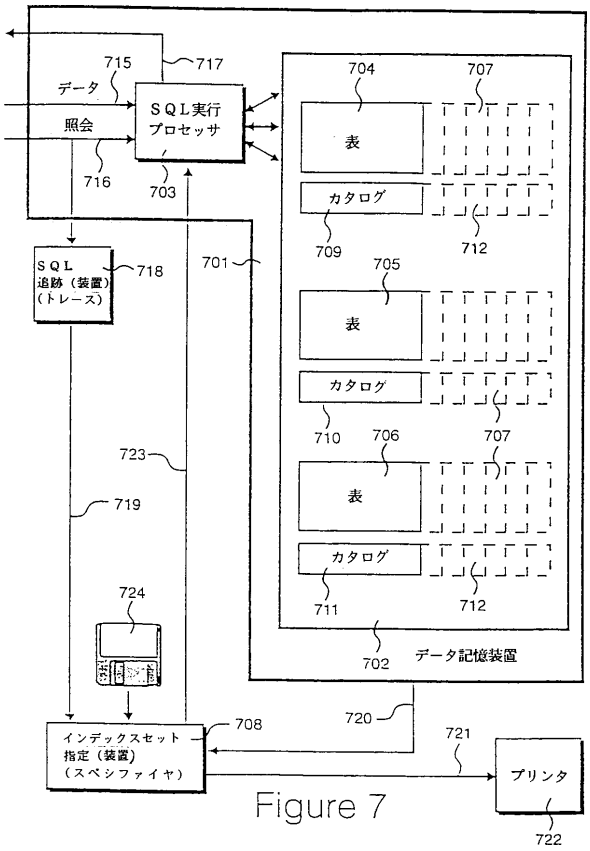


Figure 7

【図８】

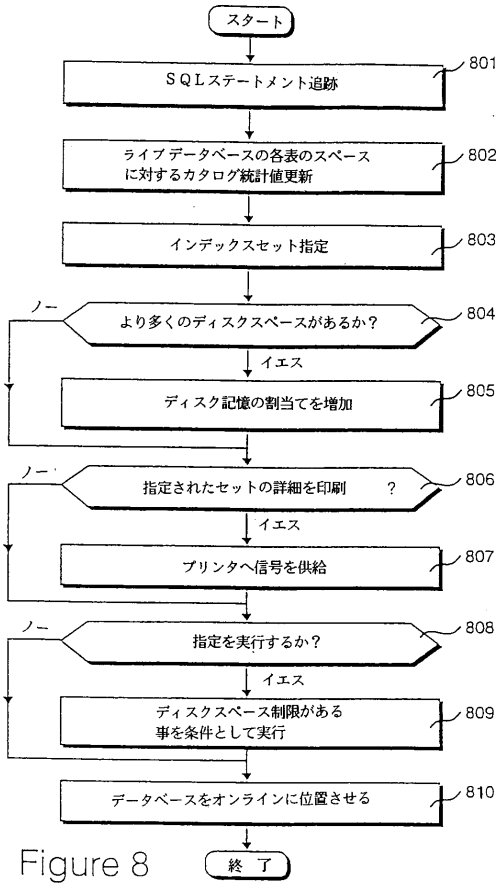


Figure 8

【図９】

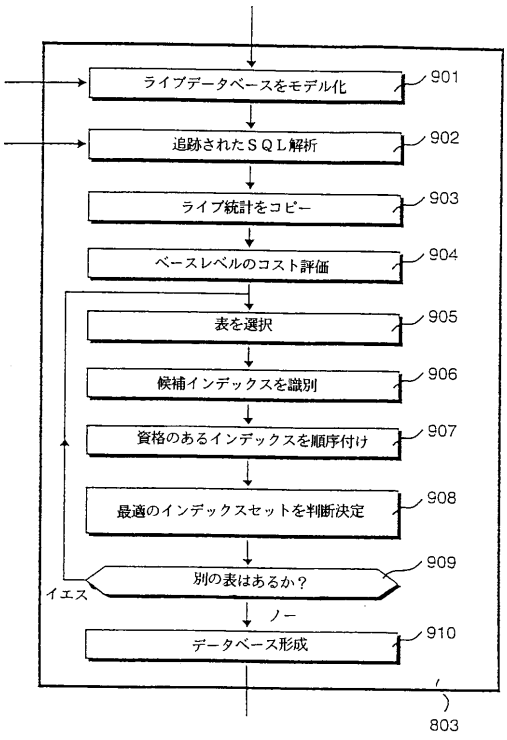


Figure 9

【図１０】

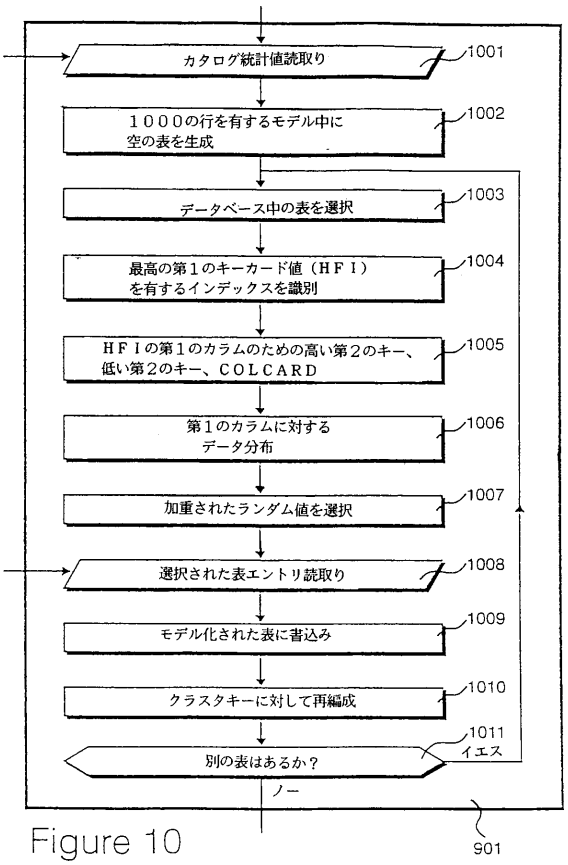


Figure 10

【図１１】

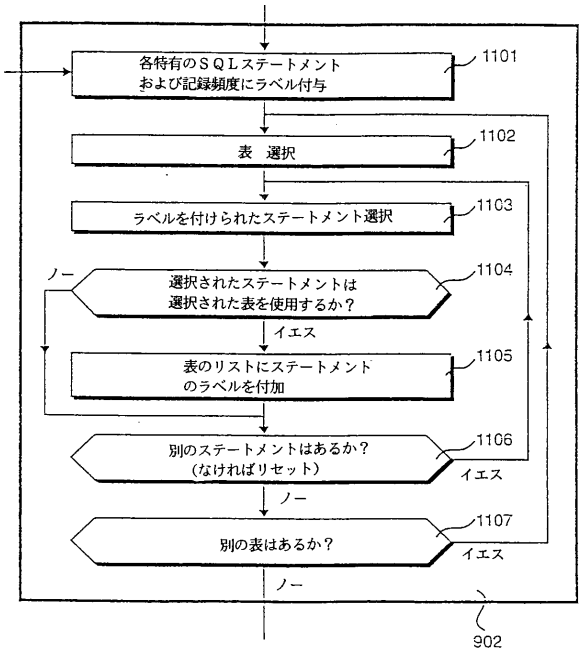


Figure 11

【図１２】

| 1201 表 | 1202 ステートメントラベル | 1203 頻　度 |
|---|---|---|
| 表　1 | SQL A | x |
| | SQL B | y |
| | SQL C | z |
| | ⋮ | |
| | SQL Z | a |
| 表　2 | SQL B | y |
| | SQL D | m |
| | SQL F | p |
| | ⋮ | |
| | SQL Y | b |
| 表　3 | SQL A | x |
| | SQL D | m |
| | SQL F | p |
| | ⋮ | |
| | SQL X | c |

Figure 12

【図１３】

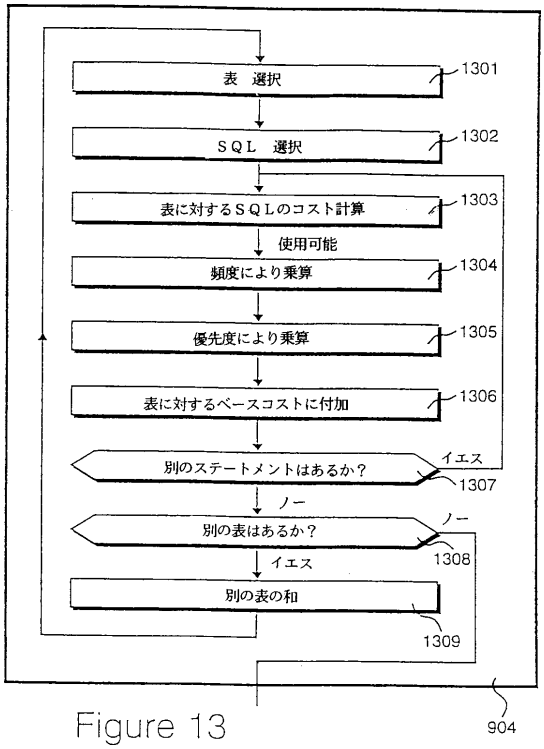


Figure 13

【図１４】

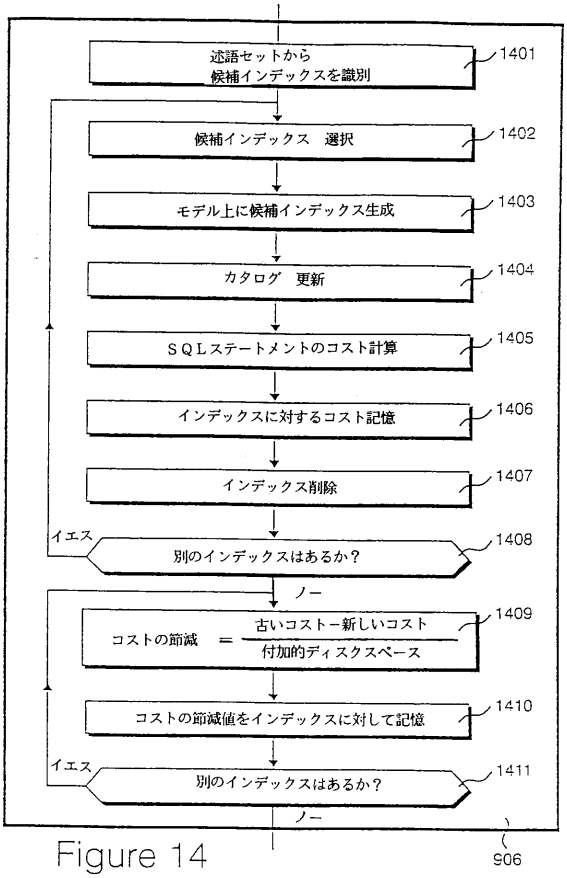


Figure 14

【図１５】

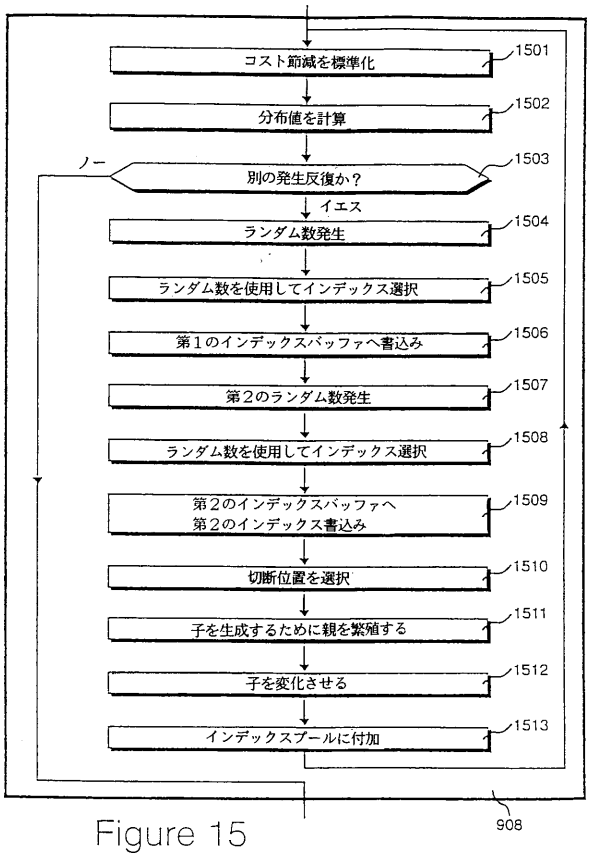


Figure 15

【図１６】

| | インデックス | コスト節減 | 標準化 | 分布 |
|---|---|---|---|---|
| 1 | 1625 | 73 | 1460 | 8540 - 9999 |
| 2 | 1616 | 72 | 1440 | 7101 - 8540 |
| 3 | 1604 | 68 | 1360 | 5741 - 7100 |
| 4 | 1673 | 59 | 1180 | 4561 - 5740 |
| 5 | 1612 | 58 | 1160 | 3401 - 4560 |
| 6 | 1646 | 49 | 980 | 2421 - 3400 |
| 7 | 1635 | 43 | 860 | 1561 - 2420 |
| 8 | 1691 | 37 | 740 | 821 - 1560 |
| 9 | 1622 | 21 | 420 | 401 - 820 |
| 10 | 1683 | 16 | 320 | 81 - 400 |
| 11 | 1617 | 4 | 80 | 0 - 80 |
| | 合計 | 500 | | |

1681

Figure 16

【図１７】

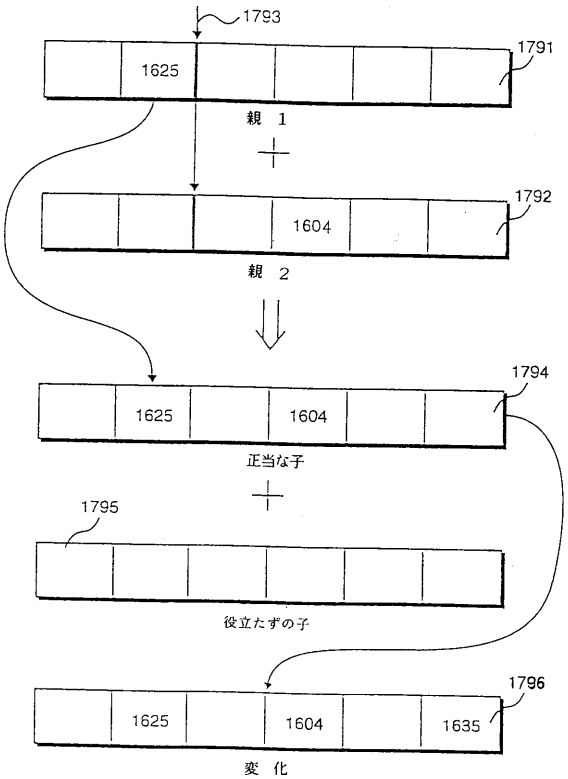


Figure 17

【図１８】

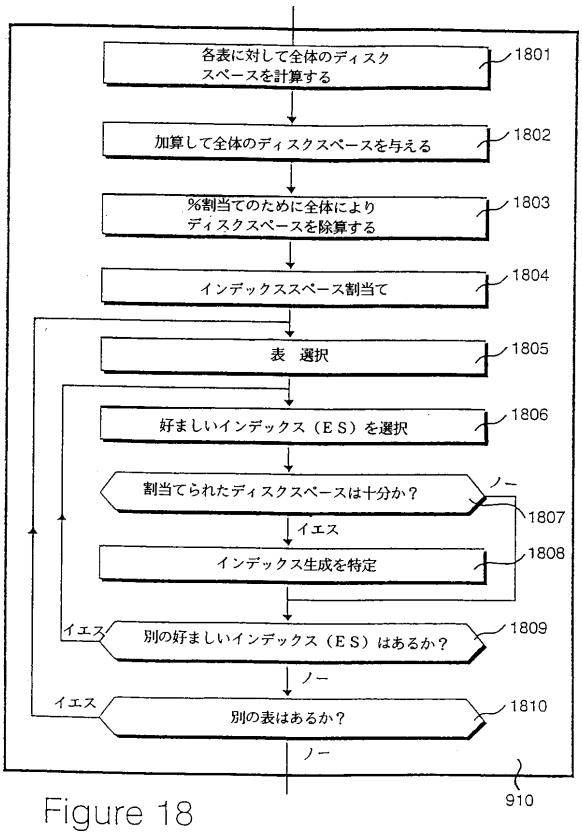


Figure 18

フロントページの続き

(72)発明者　レンジー、ロバート・エス
イギリス国、エーエル５・４ジェイゼット、ハートフォードシャー、ハーペンデン、ウエストフィールド・ロード　１２９

審査官　鈴木　和樹

(56)参考文献　米国特許第０４９５６７７４（ＵＳ，Ａ）
米国特許第０５４０４５１０（ＵＳ，Ａ）
特開平０２－０５４３４７（ＪＰ，Ａ）
特開昭６３－２０１７１７（ＪＰ，Ａ）
C.J.Date、外１名，ＤＢ２入門，株式会社オーム社，１９９５年　9月２０日，第1版，ｐ．６４，６５

(58)調査した分野(Int.Cl.，ＤＢ名)
G06F 17/30
G06F 12/00
NRIサイバーパテント