

データベースプロセッサ RINDA の設計と実現†

井上 潮** 速水 治夫** 福岡 秀樹**
鈴木 健司** 松永 俊雄**

リレーショナルデータベースプロセッサ RINDA のアーキテクチャと実現方式について述べる。RINDA は、インデックスの利用が困難な非定型的検索処理を高速化するためのデータベースプロセッサであり、汎用計算機にとって負担が重いサーチ、ソート等の処理を専用ハードウェアにより超高速に実行する。RINDA は、汎用計算機と入出力インタフェースにより接続され、汎用計算機上で走行するデータベース管理システムにより制御される。本論文では、RINDA のアーキテクチャと実現方式をハードウェアとソフトウェアの両面から述べる。さらに、ウィスコンシン・ベンチマークを用いた実測評価結果により、従来のソフトウェアのみのシステムと比較してリレーショナルデータベースの非定型検索処理が最高 100 倍以上に高速化できたことを示し、その要因を分析する。

1. はじめに

データベースマシンの開発は、研究段階から実用化段階へと移ってきた。商用化されたデータベースマシンは、汎用計算機との機能分担という点に着目すれば後置型と付加型の 2 つのグループに分類することができる。後置型のデータベースマシンは、汎用計算機とは独立かつ自己完結的にデータベース処理を行い、汎用計算機に対するバックエンドプロセッサとして利用される。例えば、IDM/BL シリーズ¹⁾、DBC/1012²⁾ が後置型の分類に入る。

一方、付加型のデータベースマシンは、データベース処理機能のうちの一部を専用化し、特定の汎用計算機に付加または内蔵したものであり、汎用計算機上のデータベース管理システムによって制御される。付加型のデータベースマシンは、機能の専用化の範囲と手段によって種々の実現形態をとりうる。例えば、CAFS³⁾ は、ディスク中にファイルとして格納されたデータのサーチとふるい落とし機能を専用化したものであり、検索用のハードウェア論理を組み込んだ専用のディスク制御装置として実現されている。IDP⁴⁾ は、主記憶上に読み上げられたデータに対するソートと基本的な関係演算機能を専用化したものであり、汎用計算機に内蔵されたベクトルプロセッサをデータベース処理用に拡張することによって実現されている。一般に、付加型のデータベースマシンは汎用計算機の性能

ネットワークとなっている機能のみを専用ハードウェア化するため、小さな追加コストで大きな性能向上が実現できるという利点がある。

著者らが開発した RINDA (Relational database processor)^{5),6)} は、リレーショナルデータベースに対するインデックスの利用が困難な非定型的検索処理を高速化することを目的とする付加型のデータベースマシンである。具体的には、非定型的検索処理を実行する際に汎用計算機にとって負担が重いサーチ、ソート等の処理を専用ハードウェアにより超高速に実行する。これにより、従来のシステムでは実時間で処理が困難であった大規模データベースに対する非定型検索処理を最高 100 倍以上に高速化することを可能にした。

RINDA は汎用計算機 DIPS シリーズ^{7),8)} と入出力インタフェースにより接続され、DIPS 上のソフトウェアであるデータベース管理システム (DEIMS-5: DEnden Information Management System-5) によって制御される。以下、RINDA の接続された汎用計算機をホスト計算機、RINDA を制御するデータベース管理システムを制御ソフトウェアと呼ぶ。RINDA が実現する機能は非定型的検索処理のみであり、その他の機能、例えばデータベースの生成処理や、定型の検索・更新処理は従来どおり制御ソフトウェアによって実現される。ユーザプログラムとのインタフェースは SQL⁹⁾ に準拠したものであり、ユーザプログラムから RINDA を特に意識する必要はない。また、ホスト計算機およびデータベースが格納されるディスクとディスク制御装置は汎用のものがそのまま使用できるため、既存ユーザの設備を無駄にすることなく RINDA

† Design and Implementation of a Relational Database Processor -RINDA by USHIO INOUE, HARUO HAYAMI, HIDEKI FUKUOKA, KENJI SUZUKI and TOSHIO MATSUNAGA (NTT Communications and Information Processing Laboratories).

** NTT 情報通信処理研究所

を導入することが可能である。

本論文では、RINDA のアーキテクチャと実現方式について、ハードウェアとソフトウェアの両面から述べる。2章では、従来のリレーショナルデータベース管理システムの問題点と RINDA による解決法を述べる。3章では、RINDA のハードウェアとソフトウェアのアーキテクチャ、実現方式を述べる。4章では、RINDA による性能向上効果についての評価と分析結果を示す。

2. 基本概念

2.1 従来システムの問題点

リレーショナルデータベースの主要な処理は、定型の検索・更新処理と非定型の検索処理である。定型の検索・更新処理の代表的な例は、表（リレーション）の中からユニークなキーを用いて単一の行（タプル）を選択する処理であり、インデックスを用いることによって従来のソフトウェアでも十分に実用に耐えうる性能が実現されている。

一方の非定型の検索処理には、あらかじめインデックスの作成されていない列（アトリビュート）に対して条件を設定した検索や、列内の文字列データに対する任意文字列の部分一致検索等があり、これらを従来のソフトウェアのみのシステムで実行すると多大の処理時間がかかっていた¹⁰⁾。その原因の1つは、インデックスを使用できないために表の中のすべての行を逐次的に読み出して検索条件の判定を行うサーチ処理に多大の CPU 時間と I/O 時間がかかることである。原因の2つめは、非定型の検索処理では単一の表のグループ化、または複数の表を対象とする副問合せや結合演算を伴うことが多く、そのために表を特定の列の順序で並べ換えるソート処理にも多大の CPU 時間がかかることである。

2.2 RINDA での解決法

サーチ処理の基本的な問題は、すべての行をディスクから汎用計算機の主記憶に転送した上で検索条件の判定を行うため、多大の CPU 処理とデータ転送が必要になることである。一方のソート処理の問題は、対象となる行の数を N とすると、通常のソフトウェア・アルゴリズムではオーダー $N * (\log N)$ の時間がかかることと、高速処理のためには大量の主記憶を使用することが不可欠であることである。

以上述べた問題の解決を図るため、著者らはサーチ処理とソート処理を専用ハードウェアで実行するデー

タベースプロセッサ RINDA を開発した。サーチ処理に関しては、ディスクのデータ転送速度で、ディスク中に格納された表のオン・ザ・フライ処理、すなわちディスクからのデータ読み出しと並行した検索条件の判定処理を行い、条件に合致した行の中の必要な列のみをホスト計算機に転送する。ソート処理に関しては、I/O チャンネルのデータ転送速度でホスト計算機から表を受け取り、ソートした結果をホスト計算機に返送する。以上により、RINDA を接続したホスト計算機の CPU 時間と入出力時間を大幅に削減することが可能となった。

3. アーキテクチャと実現方式

3.1 設計方針

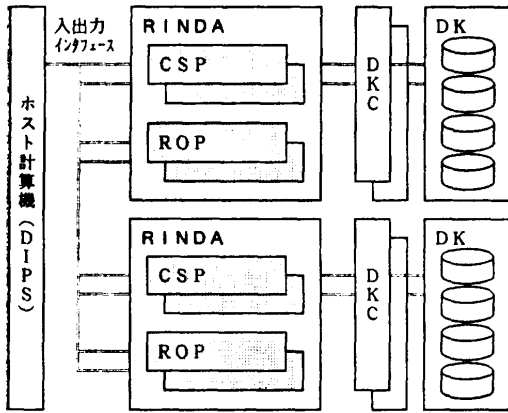
RINDA の設計に際しては、ユーザにあまり負担をかけずに稼働中のシステムにも比較的容易に導入できるように考慮した。具体的には、以下の方針に基づいて設計を行った。

- RINDA とホスト計算機との接続は、入出力インタフェースを使用する。また、データベースを格納するディスクとその制御装置は汎用のものを使用する。その理由は、ホスト計算機シリーズ内での共通化、すなわち大型から小型まですべての機種と接続できることと、性能に対する拡張性、すなわち稼働中のシステムに対して後から追加導入できることによる。
- RINDA は既存のデータベース管理システムと同一のデータベースを処理の対象とする。すなわち、1つのデータベースに対して RINDA を使用したアクセスと使用しないアクセスを1人のユーザが混在して実行したり、複数のユーザが同時に実行することを可能とする。
- ユーザプログラムとのインタフェースは、標準データベース言語 SQL に準拠したものとする。このとき、ユーザプログラムからは RINDA の存在を特に意識する必要はなく、システムの構築および既存システムからの移行を容易にする。

3.2 ハードウェア

(1) 構成

ホスト計算機に RINDA を接続したシステム構成例を図1に示す。2.2節で述べたように、RINDA はサーチ処理とソート処理を高速化するための専用ハードウェアとして、内容検索プロセッサ (CSP: Content Search Processor) とオプションの関係演算プロセッ



CSP: 内容検索プロセッサ, DKC: ディスク制御装置
ROP: 関係演算プロセッサ, DK: ディスク装置

図 1 RINDA システムの構成例

Fig. 1 Typical RINDA system organization.

サ (ROP: Relational Operation Accelerating Processor) から構成した。CSP は、ディスクに格納された表をサーチし、検索条件に合致した行と列を選択し、その結果をホスト計算機に転送する。ROP は、ホスト計算機から転送されてくる表をソートし、その結果をホスト計算機に送り返す。各 CSP, ROP は独立の入出力インタフェースでホスト計算機に接続される。なお、CSP と ROP の 2 つのハードウェア構成としたことにより、システムごとに異なるサーチ処理とソート処理の負荷バランスに対応することを可能にした。

(2) 動作概要

RINDA を使用した検索処理の実行例を図 2 に示す。まず、CSP が選択した行と列をホスト計算機の主記憶に転送する (①)。次に、これらは ROP によってソートと重複値のカウンが行われ、再び主記憶に返送される (②)。最後に、ホスト計算機上の制御ソフトウェアが検索結果を 1 行ずつユーザプログラムに引き渡す (③)。なお、オプションの ROP がシステムに実装されていない場合は、ROP 相当の機能は制御ソフトウェアによって代行される。

(3) CSP の機能

CSP は、ディスクに格納された単一の表に対して表 1 に示す機能を実現する。CSP の特徴は、SQL で規定されている各種の述語判定機能をほぼそのまま専用ハードウェア化した点にあり、これにより制御ソフトウェアが行わなければならない処理量を削減している。例えば、NULL 値に対する条件の判定も SQL の仕様に準拠しているため、制御ソフトウェアで特別な

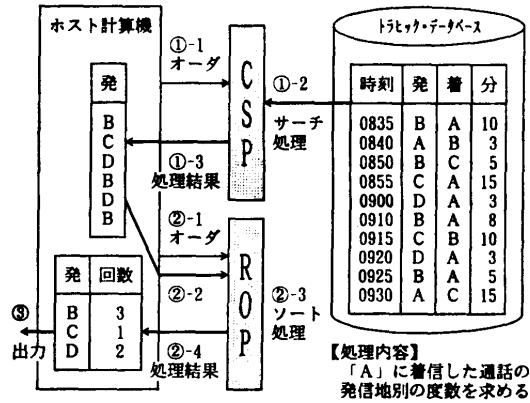


図 2 RINDA を用いた検索処理の実行例

Fig. 2 Typical query execution by RINDA.

表 1 CSP の機能

Table 1 Primary CSP functions.

機能	説明
述語判定	比較述語 <列> <比較演算子> <定数> の条件判定.
	IN 述語 <列> [NOT] IN <定数リスト> の条件判定.
	LIKE 述語 <列> [NOT] LIKE <パターン> の条件判定.
	NULL 述語 <列> IS [NOT] NULL の条件判定.
検索条件判定	述語の AND/OR による任意の論理式判定.
出力列の抽出	上記検索条件を満たす行からの任意の列の出力.
集合関数	上記検索条件を満たす行数のカウン (COUNT (*)).

処理を行う必要はない。CSP を用いた処理を実行する際の概略手順は、以下ようになる。

① 制御ソフトウェアが CSP に対するオーダを作成する。オーダ中には、処理の対象とするディスクの機番、ディスク内アドレス等の物理制御情報と、表 1 の機能を記述した論理制御情報が含まれる。

② 作成したオーダを入出力コマンドによってホスト計算機から CSP へ転送する。

③ CSP は、受け取ったオーダをもとにディスクからデータを連続して読み出し、同時に検索条件の判定と出力列の抽出を実行する。この動作を中断なく行うため、ディスクから読み出すための入力バッファとホスト計算機へ転送するための出力バッファは複数個を交代で使用する。

④ 出力バッファ中に所定の量のデータがたまった段階で、サーチ処理の実行とは非同期に処理結果をホスト計算機へ転送する。

なお、サーチ処理自体は新規に開発した専用 LSI によって並列処理することにより、ほとんどの場合にオン・ザ・フライ処理すなわちディスクからのデータ転送時間内での処理を実現した。また、処理結果のデータ形式はそのまま ROP の入力とすることにより、CSP と ROP の処理の間で制御ソフトウェアが行単位でデータを加工する必要をなくした。

(4) ROP の機能

ROP は、ホスト計算機から転送されてくる 1 個または 2 個の表に対して表 2 に示す機能を実現する。このうち、ふるい落とし機能は副問合せおよび結合演算の実行時のホスト計算機の負荷を削減するため、ハッシュ化ビットアレイ¹¹⁾により一方または両方の表から対応関係のない行を除去するものである。ROP の特徴は、ソート処理のための演算回路とメモリをホスト計算機とは独立に持っている点にあり、このためホスト計算機とは独立かつ非同期に処理を実行できる。ROP を用いた処理を実行する際の概略手順は、以下のようなになる。

- ① 制御ソフトウェアが ROP に対するオーダを作成する。オーダ中には、表 2 の機能を記述した論理制御情報のみが含まれる。
- ② 作成したオーダを入出力コマンドによってホスト計算機から ROP へ転送する。
- ③ ROP は、オーダを受け取ると引き続きホスト計算機から転送されてくる表を入力し、ふるい落とし、ソート、重複値の計数/除去の順にパイプライン処理する。ホスト計算機との間のデータ転送を中断させないため、入出力バッファは複数個を交代で使用する。
- ④ 表の入力が終了した後、結果データをホスト計算機へ転送する。

なお、ソート処理は ROP 内部の大容量メモリとマ

表 2 ROP の機能
Table 2 Primary ROP functions.

機能	説明
ソート	指定された列の値(ソートキーと呼ぶ)による昇順または降順への行の並べ換え。ソートキーは任意の順序の複数列により構成可能。
ふるい落とし	ソートキーの値による 2 つの表からの結合可能性のない行の除去。一方または両方の表からの除去が可能。
重複計数/除去	ソートキーの値が重複する行数のカウント、および 2 番目以後の行の除去。

ルチウェイ・マージソート¹²⁾を行う新規開発の専用 LSI を用いることにより、チャンネルのデータ転送時間内でのパイプライン処理を実現した。マルチウェイ・マージソートの特性としてソート対象となる行の数が少ない場合、具体的にはマージ処理が 2 段階ですむ場合には、マージ処理時間がホスト計算機とのデータ転送時間と完全にオーバーラップされ、全体の処理時間はオーダ N (N は行の数) となる。また、マージ処理が 3 段階以上になる場合でも、ROP 内部では行の中のソートキーのみを扱うためにマージ処理時間がデータ転送時間と比べて十分小さく、処理時間はオーダ N に近い。

3.3 ソフトウェア

(1) 構成

ホスト計算機上の制御ソフトウェアの構成を図 3 に示す。制御ソフトウェアの実現に際しては、既存のデータベース管理システムに RINDA 対応の機能を追加する構成とした。新規に追加した主な部分は、言語処理部の RINDA 用最適化機能と、RINDA 制御部の全体である。このような構成とした理由は以下の 2 点である。

- ① 同一のデータベースに対して RINDA を使用したアクセスと使用しないアクセスを同時に行うことを可能とする。
- ② データベース管理システムを RINDA 対応することによって、RINDA を使用しない既存ユーザに対する影響を性能面に関しても与えないようにする。

(2) 機能概要

ユーザプログラムから発行された SQL 文は、言語

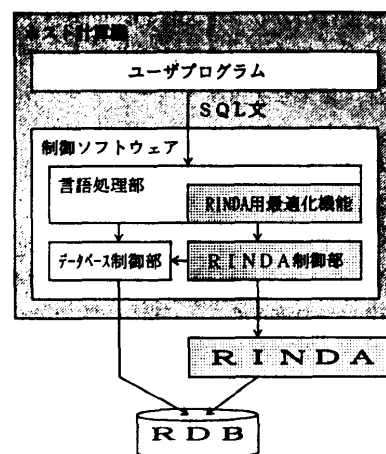


図 3 制御ソフトウェアの構成
Fig. 3 RINDA control software organization

処理部により構文解析, 意味解析, 最適化が行われた後に実行される. RINDA 制御部は, RINDA を用いた検索処理の実行機能を実現する. 主な機能は, RINDA との入出力制御, バッファ管理, 選択・結合等の基本的な関係演算処理である. データベース制御部は, RINDA を用いないでソフトウェアだけによる検索・更新処理の実行機能を実現する. さらに, トランザクション管理, 排他制御, 資源管理等のデータベース処理として一元的な管理が必要な制御機能も実現する. なお, SQL 文の実行を RINDA 制御部で行うかデータベース制御部で行うかは, ユーザプログラムのコンパイル時に SQL 文単位でユーザが指定する.

(3) 高速化のための技法

RINDA を使用した検索処理の実行を高速化するため, 制御ソフトウェアでは以下の工夫を行った.

① 1つの表を複数 (M 個) のディスク・ボリュームに分割して格納しておき, これを検索処理の実行時に M 個の CSP が並列に読み出すことにより, ディスクからのデータ読み出し時間を約 M 分の 1 に短縮する.

② 2つの表に対する結合演算は, 一方の表に対する CSP からの出力データ量に応じて結合演算アルゴリズムを動的に選択することにより, 以後の処理時間を短縮する.

③ 表全体をサーチ処理の対象とする非定型の検索処理とインデックスを用いた単一の行を対象とする定型の更新処理の同時走行を可能にするため, RINDA 側の処理をダーティデータ・リードすなわちデータへの

ロックをかけずに問合せを実行する機能をオプションとして提供する.

④ CSP/ROP が出力する処理結果は, 基本的には主記憶上に格納するが, 領域が不足した場合は作業用ディスクに書き出すことにより, CSP/ROP の処理を中断させることなく大量データの処理を可能とする.

4. 性能評価

4.1 評価条件

RINDA による非定型の検索処理の高速化効果を拡張ウィスコニン・ベンチマーク^{13),14)}により評価した. 評価に用いた表の基本的な大きさは1万行であり, 各行はシステムの制御情報を除いて208バイトの長さである. 評価に用いたシステムは, ホスト計算機が小型汎用機の DIPS-V30E⁷⁾, CSP が2台, ROP が1台と, 容量が1.3GB, データ転送速度が3MB/S のディスクおよび制御装置各2台により構成した.

処理時間を実測した問合せを表3に示す. このうち, Q1~Q6 はベンチマーク標準の問合せであり, Q1~Q3 によって CSP 単独による性能向上効果, Q4~Q6 によって CSP と ROP の両者による性能向上効果の評価した. さらに, RINDA のハードウェア部分に着目した処理性能を評価するため, 検索結果データの転送をせずに行数だけをカウントする Q7~Q9 を評価した. なお, RINDA の開発目的から問合せの評価はすべてインデックスを使用せずにを行った.

4.2 評価結果

RINDA を使用しない場合と使用した場合の性能を比較すると図4のようになる. 処理時間比は, SQL 文

表3 評価に用いた問合せ
Table 3 Executed queries in the test.

問 合 せ	対 応 す る SQL 文
Q1 選択 (1%)	SELECT * FROM tenk WHERE unique2>=5000 AND unique2<5100
Q2 選択 (10%)	SELECT * FROM tenk WHERE unique2>=5000 AND unique2<6000
Q3 関数 (スカラ MIN)	SELECT MIN(unique2) FROM tenk
Q4 関数 (グループ化 MIN)	SELECT MIN(unique2) FROM tenk GROUP BY hundred
Q5 結合 (2表)	SELECT * FROM tenkA A, tenkB B WHERE A.unique2=B.unique2 AND A.unique2<1000
Q6 結合 (3表)	SELECT * FROM onek C, tenkA A, tenkB B WHERE C.unique2=A.unique2 AND A.unique2=B.unique2 AND A.unique2<1000 AND B.unique2<1000
Q7 計数 (スカラ)	SELECT COUNT(*) FROM tenk WHERE unique2>=5000 AND unique2<5100
Q8 計数 (グループ化)	SELECT COUNT(*) FROM tenk GROUP BY hundred
Q9 文字列検索	SELECT COUNT(*) FROM tenk WHERE stringu2 LIKE '%ABC%'

の実行開始から検索結果をすべてユーザプログラムに返却し終わるまでの経過時間の相対比である。この図から、以下のことがわかる。

- ① ホスト計算機の CPU 時間が大幅に短縮される。
- ② ホスト計算機の I/O 時間も大幅に短縮される。
- ③ RINDA による非定型検索の性能向上効果は、最高 100 倍以上に達する。

さらに、公表されている市販のデータベースマシンの性能^{14),15)}と RINDA の性能比較を表 4 に示す。この表からも、RINDA は非定型の検索処理が高速であることがわかる。

4.3 考 察

(1) 性能向上要因

RINDA により上記の性能向上が実現できた要因をまとめると以下ようになる。

(a) ホスト計算機の CPU 時間短縮

非定型の検索処理を従来のソフトウェアで実行した場合、サーチ処理およびソート処理を表の中のすべての行に対して行うため、膨大な CPU 時間がかかっていた。RINDA の利用によりこれらの処理の大半をオフロードでき、CPU 時間を大幅に短縮した。なお、制御ソフトウェアとユーザプログラム間で処理結果を行単位で返却する処理はオフロードされていないため、処理結果の行数が多い場合 (Q2, Q5, Q6) は性能向上効果が相対的に小さくなる。

(b) ホスト計算機の入出力時間短縮

1つの表を複数のディスクに分割して格納し、検索実行時に複数の CSP を並列に動作させることにより、ディスクからの入力時間を短縮した。また、1シリンダ内の全データを途中でディスクヘッドの機械的な動作なしに連続的に読み出すことにより、ディスクからの入力そのものの効率も向上した。

(2) 効果の有効範囲

以上の評価結果をもとに、別の観点から分析する。

(a) 表の大きさについて

RINDA による検索処理時間は、表の大きさ (データ量) に比例する。汎用計算機による検索処理時間も、インデックスを使用できない場合は表の大きさに

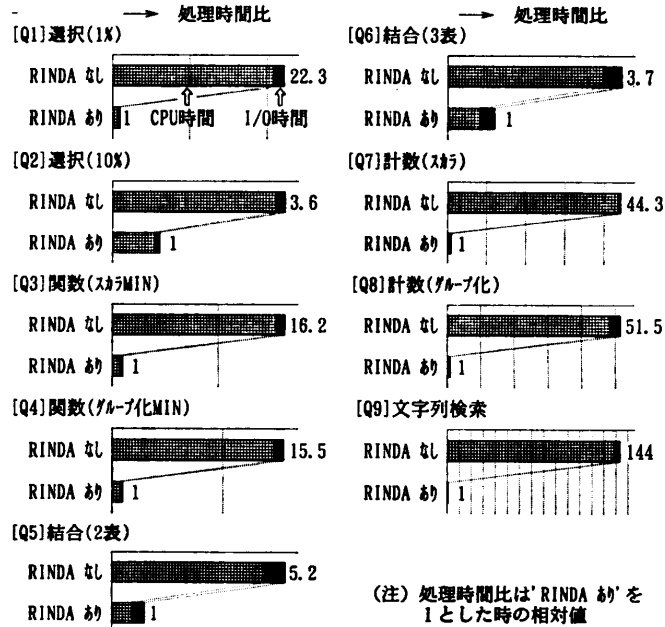


図 4 RINDA による性能向上効果
Fig. 4 Performance improvement by RINDA.

表 4 市販マシンとの処理時間比較
Table 4 Query execution time (in seconds).

(処理時間の単位: 秒)

問 合 せ		マ シ ン (年)		
		RINDA (1989)	IDM/500 da (1985)	DBC/1012 (1987)
Q1	選択 (1%)	1.61	19.9	6.86
Q2	選択 (10%)	12.0	23.4	16.0
Q3	関数 (スカラ MIN)	3.28	19.0	4.21
Q4	関数 (グループ化 MIN)	5.67	36.0	8.66
Q5	結合 (2表)	27.5	84.0	35.6
Q6	結合 (3表)	31.5	36.0	27.8

(注) RINDA: DIPS-V30E, 2CSP, 1ROP の構成。
IDM/500: データベース・アクセラレータ付きの構成。
DBC/1012: 24 プロセッサ, 40 ディスクの構成。

比例する。したがって、RINDA による性能向上効果 (相対比) は表の大きさにはほとんど依存しない。

(b) 問合せの種類について

RINDA は SQL のすべての機能を分担するわけではない。例えば、COUNT 関数は RINDA で処理するが、SUM 関数はホスト計算機で処理する。また、処理結果をユーザプログラムに返却する処理も RINDA は関与しない。このため、問合せの種類および処理結果の大きさによって性能向上効果はかなり変化する。

(c) コストパフォーマンス

RINDA の導入によりシステムのハードウェアコストは数割のオーダーで増加するが、RINDA による性能向上効果は数倍から数十倍のオーダーになる。したがって、非定型の検索処理の比重が高いシステムについては1桁近いコストパフォーマンス向上が得られる。

5. おわりに

本論文では、データベースプロセッサ RINDA のアーキテクチャと実現方式について述べた。RINDA の目的は、汎用計算機上のデータベース管理システムにとって重い負担となる非定型の検索処理を高速化することにある。このため、サーチ処理とソート処理を専用ハードウェアにより高速に実行する機構を実現した。RINDA による性能向上効果はきわめて大きく、従来のソフトウェアと比較して検索処理を最高100倍以上に高速化できた。

RINDA の適用領域は、リレーショナルデータベースでの非定型の検索処理が必要となる業務分野である。例えば情報検索の分野では、文章データに対してキーワードを抽出したりインデックスを作成したりすることなく、そのままデータベースに格納するだけで RINDA を用いた任意の語句をキーとする検索が可能になる。また統計処理の分野では、大量かつ頻繁に追加されるトラヒック情報を RINDA によって実時間で分類・集計することが可能になる。

RINDA は現在いくつかのユーザシステムに導入が進められており、今後はこれらの適用結果をもとにさらに評価していく予定である。

謝辞 RINDA の開発を推進していただいた NTT 研究開発技術本部の戸田巖本部長、ならびに同情報通信処理研究所の石野福彌所長、橋本昭洋元知能処理研究部長、拜原正人専用システム研究部長に感謝します。

参 考 文 献

- 1) Britton Lee, Inc.: *The Intelligent Database Machine—Product Description* (1984).
- 2) Teradata Corp.: *DBC/1012 Data Base Computer System—Introduction* (1986).
- 3) Babb, E.: Implementing a Relational Database by Means of Specialized Hardware, *ACM Trans. Database Syst.*, Vol. 4, No. 1, pp. 1-29 (1979).
- 4) Kojima, K., Torii, S. and Yoshizumi, S.: IDP—A Main Storage Based Vector Database Proc-

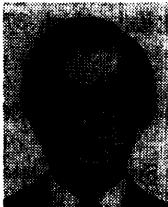
- essor, *Database Machines and Knowledge Base Machines* (Kitsuregawa, M. and Tanaka, H. eds.), pp. 47-60, Kluwer Academic (1988).
- 5) 速水, 井上, 福岡, 鈴木: リレーショナルデータベースプロセッサ RINDA のアーキテクチャ, 情報処理学会計算機アーキテクチャ研究会資料, 88-ARC-73-12 (1988).
- 6) Inoue, U., Hayami, H., Fukuoka, H. and Suzuki, K.: RINDA—A Relational Database Processor for Non-Indexed Queries, *Proc. Int. Symp. Database Systems for Advanced Applications*, pp. 382-386 (1989).
- 7) 矢沢, 平野, 山口, 岡田: DIPS-V 30 E のハードウェア構成, NTT 研究実用化報告, Vol. 37, No. 9, pp. 523-532 (1988).
- 8) 小柳津, 塩川, 木ノ内, 安保: DIPS-11/5 E シリーズの実用化, NTT 研究実用化報告, Vol. 36, No. 1, pp. 49-56 (1986).
- 9) ISO: Information Processing Systems—Database Language SQL, *International Standard*, ISO 9075 (1987).
- 10) 井上, 北村, 速水, 中村: 情報提供サービスに適用可能な超大規模リレーショナル・データベースマシン, 情報処理学会データベースシステム研究会資料, 85-DB-47-5 (1985).
- 11) McGregor, D. R., Thomson, R. G. and Dawson, W. N.: High Performance Hardware for Database Systems, *Systems for Large Data Bases* (Lockemann, P. C. and Neuhold, E. J. eds.), pp. 103-116, North-Holland (1976).
- 12) 佐藤, 武田: 大容量データベース処理に適したソート手法, 信学技報, Vol. 88, No. 56, DE 88-1 (1988).
- 13) Bitton, D., DeWitt, D. J. and Turbyfill, C.: Benchmarking Database Systems—A Systematic Approach, CSTR # 526, Univ. of Wisconsin-Madison (1983).
- 14) DeWitt, D. J. et al.: A Single User Evaluation of the GAMMA Database Machine, *Database Machines and Knowledge Base Machines*, pp. 370-386, Kluwer Academic (1988).
- 15) Simon, E.: *Update to December 1983 'DeWitt' Benchmark*, Britton Lee, Inc. (1985).

(平成元年 5 月 9 日受付)

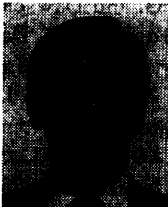
(平成元年 12 月 12 日採録)

**井上 潮 (正会員)**

昭和 28 年生。昭和 50 年名古屋大学工学部電気学科卒業。同年、日本電信電話公社入社。現在、NTT 情報通信処理研究所基本アーキテクチャ研究部主幹研究員。オンライン情報検索システム、データベース管理システム、データベースマシンの研究実用化に従事。電子情報通信学会、IEEE-CS 各会員。

**速水 治夫 (正会員)**

昭和 22 年生。昭和 45 年名古屋大学工学部応用物理学専攻卒業。昭和 47 年同大学院工学研究科応用物理学専攻修士課程修了。同年、日本電信電話公社入社。現在、NTT 情報通信処理研究所情報処理研究部主幹研究員。主に、DIPS ハードウェアシステム、データベースマシンの研究実用化に従事。電子情報通信学会会員。

**福岡 秀樹 (正会員)**

昭和 25 年生。昭和 47 年大阪府立大学工学部電子工学科卒業。同年、日本電信電話公社入社。現在、NTT 情報通信処理研究所情報処理研究部主幹研究員。主に DIPS 本体系装置の研究実用化を行い、現在データベースマシンの研究実用化に従事。電子情報通信学会会員。

**鈴木 健司 (正会員)**

昭和 21 年生。昭和 44 年日本大学理工学部電気工学科卒業。同年、日本電信電話公社入社。現在、NTT 情報通信処理研究所情報システム研究部主幹研究員。主にデータベース管理システム、データベースマシンの研究実用化に従事。電子情報通信学会会員。

**松永 俊雄 (正会員)**

昭和 18 年生。昭和 40 年東北大学工学部通信工学科卒業。同年、日本電信電話公社電気通信研究所入所。現在、NTT 情報通信処理研究所研究企画部長。コンピュータの入出力制御方式、大型ハードウェアの研究実用化に従事。最近、コンピュータシステム構成方式に興味を持っている。電子情報通信学会会員。