1B-6

データベースマシンGRACEのプロトタイプシステム

伏見　信也†　喜連川　優††　田中　英彦†　元岡　達†
(†東京大学　工学部　††東京大学　生産技術研究所)

1．はじめに

関係データベースマシン GRACE〔1〕はhashとsortに基づくデータ流指向のアルゴリズムにより，join等の処理負荷の重い関係代数演算を高速に処理することができる。現在，(1)我々の提案するデータ流指向のアルゴリズムの検証，及び(2)その汎用計算機への適用性の検討，等を目的としてGRACE のプロトタイプシステムの実装を進めている。本システムは，汎用ミニコンピュータにデータベース処理専用のプロセッサ（ハードウェアソータ，μプロセッサ）を付加した比較的conventionalなハードウェア構成であるが，データ流指向のデータベース処理を基本としており，そのソフトウェア構成は従来のソフトウェアデータベースシステムのそれとは大きく異なる。また，本プロトタイプシステムは，それ自身小型の高速関係データベースマシンを目指しており，本稿ではそのアーキテクチャ，処理方式等について報告する。

2．論理アーキテクチャ

GRACEはプロセッシングモジュール（PM），メモリモジュール（MM），ディスクモジュール（DM），作業用ディスクモジュール（WDM），及び制御モジュール（CM）を各々複数台，2つのネットワークで結合した並列関係データベースマシンであるが，試作中のプロトタイプシステムは図1に示す様に，データ流指向アルゴリズムの実現に必要とされる最小のモジュール構成を採る。関係代数演算は，演算対象となるデータを格納しているSource空間からFilter処理，Clustering処理をへてSink空間に向かってデータを送出することにより実行される〔1〕。例えば，2つの関係R，Sに selection を行い，その結果をjoinする問合せは本プロトタイプ上で次の様に実行される：DM（Source空間）に格納されているR，S各々に対し，DMは必要なデータページを読出し，指定されたselection 演算を施してタプルの篩落しを行い（Filter処理），更にその結果タプルのjoin属性値に対してhashを行ってclusteringされたデータ流を生成し（Clustering処理），これをMMに転送する。同一hash値を有するタプルの集合が1cluster を構成する。従って cluster間にまたがるjoinの可能性はなく，これらは当該joinに関して互いに独立な処理単位となる。MMは送られてくるデータ流から格納可能なcluster のみをMM内のメモリに格納し，他の溢れたcluster をWDMにdestage する。WDMはcluster を最小単位としてMMからdestage されてくるデータ流を一時的に保持する。このデータ流転送が終了すると同時に，join処理の対象となるデータのMM及びWDMにわたっての格納が完了する（Sink空間）。この際，データ全体は当該演算に対して互いに独立なcluster からなるcluster 空間として再構成されている。続いて，このMM，WDMの対を今度はSource空間とし，join実行の為のデータ流転送を行う。MMは格納されているcluster を順にPMに送り，joinを実行する。PMはハードウェアソータとタプル処理ユニットを用いて，連続的に送られてくるclusterに対し，データ流に遅れることなくjoinを実行する（Filter処理）。MMからcluster がPMに転送されるにつれて，処理すべき新しいcluster がWDMからMMに対し再stage され，MMとWDMは一体となってPMに対し連続的なデータ流を供給する（データ空間の仮想化）。PMでjoinを施されたデータ流は必要ならば次演算に対するCluetering処理を施されて，もう一組のMM及びWDMが構成するSink空間に流れ込む。

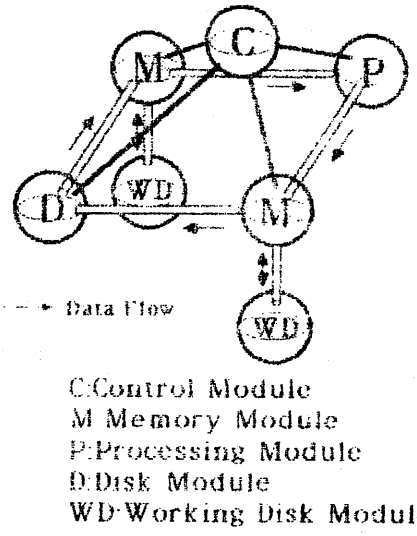

Fig.1 Logical Architecture of Prototype System

3．物理アーキテクチャ

本プロトタイプシステムのハードウェア構成を図2に示す。本システムはミニコンピュータ（三菱MELCOM80モデル500，主記憶8MB，ディスク200MB x3）とデータベース処理専用プロセッサにより構成される。3台のディスクの内，一台はデータベースを格納するDM内のディスク

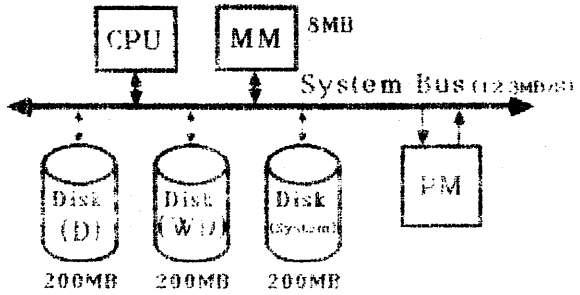

Fig.2 Physical Architecture of Prototype System

として，一台はWDM内のディスクとして，更にもう一台はシステムプログラムの格納等に用いられる。データベース処理プロセッサは内部に改良型ハードウェアソータ〔4〕及びμプロセッサ（MC68000）を持ち，join，aggregation 等を高速に実行する〔3〕。

論理アーキテクチャに於ける各モジュールはミニコンピュータ上のソフトウェアプロセスとこれらディスク，専用ハードウェアの対として実現される。即ち，DMはD－プロセスとDM用ディスクの対によって，2つのWDMは2つのWD－プロセスとこれらが共用するWDM用ディスクによって，更にPMはP－プロセスとデータベース処理専用プロセッサによって各々実現される。

2つのMM及びCMは，ソフトウェアプロセスのみによって各々2つのM－プロセス，C－プロセスとして実現される。これらソフトウェアプロセスは図1に示される経路に従って互いにデータ転送や制御情報の通信を行う。DM，WDM，PMの各ソフトウェアプロセスはミニコンピュータのOSが提供する最下層の非同期入出力プリミティブを用いて，対応するデバイスとのデータ転送を効率良く実行する〔2〕。

4．実装

本プロトタイプシステムは，前述の様に汎用計算機に於けるデータ流指向のデータベース処理の適用性の検討を一つの目的としており，ミニコンピュータ上のソフトウェアの構成，実装にあたっては以下の様な点に留意した。

(1)入出力

外部デバイスと直接入出力を行うプロセス（D－プロセス，WD－プロセス，P－プロセス）は，これらデバイスとの間で大量のデータをデータ流として連続的に転送し合うことが必要である。通常のOS下では，プロセスは入出力を発行すると，その入出力が終了するまでCPUサービスの対象とならない。この為，例えばD－プロセスでは入力されてくるデータに対するselection 処理，hash操作等と，データの入力動作がserialize されてしまい，データ流の連続性が維持できない。本プロトタイプシステムでは，ミニコンピュータ上のOS（DPS-10）が提供する非同期入出力機構とダブルバッファリングを用い，入出力発行後も当該プロセスは入出力と並列に計算を続行できる様に設計されている。

(2)プロセス間通信

プロセス間では図1に示される様な大量のデータ転送が定常的に必要とされる。これら大量データのプロセス間転送を効率良く実行する為，shared memory 方式によるプロセス間通信機構を用いることとした。

(3)仮想空間

従来のOSではプログラムとデータを混在させた空間に対し，固定長ページを単位とし，そのアクセス特性を考慮したリプレイスメントアルゴリズムを用いて主記憶と作業用ディスクにより仮想空間を構成する。これに対し本プロトタイプシステムでは当該演算に対し生成されたclusterを単位とし，データのみに対する仮想空間を主記憶と作業用ディスクを用いて構成する〔1〕。各々の演算に応じた動的clusteringにより，当該演算実行に際しては生成されたcluster を独立に処理すれば良く，データ空間に対するアクセス特性は完全に予測可能となる。従って，WDM→MM間のデータ転送は余分な入出力を伴なうことなく実行され，MM→PM間のデータ転送とデータ流を乱すことなく互いに重畳化出来る。尚，本プロトタイプシステムでは，各プロセスは比較的簡単なプログラムで実現出来，これらプロセスは主記憶に常駐する様設計されている。

5．おわりに

データベースマシン GRACEのプロトタイプシステムの構成について述べた。現在，ハードウェアの実装と共に，同時実行制御，障害回復制御等を含めたソフトウェアの詳細設計，実装を進めている。

〔参考文献〕

〔1〕伏見他，『データベースマシンGRACEのアーキテクチャとその実行制御系』アドバンスト・データベースシンポジウム，昭和59年

〔2〕中山他，『GRACEプロトタイプシステムにおけるソフトウェア構成』，情報処理学会第31回全国大会，1B-7（1985）

〔3〕鈴木他，『GRACEプロトタイプシステムにおけるプロセッシングモジュールの設計』，情報処理学会第31回全国大会，1B-8（1985）

〔4〕楊他，『Length Tuning 機構を有するハードウェア・マージソータの設計』，情報処理学会第30回全国大会，1D-8（1985）