1D-6

データベースマシンGRACEの資源管理方式

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はhashとsortに基づく並列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。GRACEはMIMD方式の並列データベースマシンであるが，この様なデータベースマシンの実行制御に於いては，演算処理に必要なプロセッサ，メモリ等の計算資源の割当て，開放を司る資源管理方式が重要である。即ち，並列データベースマシンに於いては一般に資源管理等の実行制御を行うモジュールが過負荷になる傾向にあり，そのオーバヘッドは無視できない。例えば，DIRECT〔1〕では固定長のページ単位に計算資源を割付ける方式を採用しているが，その評価結果によれば処理時間の大部分は実行制御モジュールに於けるプロセッサ群へのページ割付け処理に費されている。

GRACEは実行制御方式として，実行制御単位をページ等の小さなgranule ではなく，オペランドリレーション全体が構成する巨大なデータストリームとし，演算間の並列制御にデータフロー制御を用いた，データベース処理環境に於いて自然で効率の良い並列実行制御方式を採用している〔2〕。マシン内の資源は関係代数演算単位に割付けられ，演算の実行環境を形成する。個々の演算の処理はオペランドリレーションが得られると発火可能となり，オペランドリレーションを構成するタプル群が割付けられたマシン資源中を一つのデータストリームとなって流れることによって一演算の処理を行う。本方式では実行制御が介入することなく演算が実行される為，資源割付け等の実行制御時間の対演算時間比は飛躍的に改善され，実行制御オーバヘッドが大きく減少することが期待される。

2．計算モデル

GRACEではユーザから発行された問合せは関係代数木に翻訳された後，以下の様な計算モデルに基づいて実行される。データベースはページ分割されており，問合せが参照するページの集合を問合せのread set，また問合せが更新するページの集合及び結果の出力資源（ディスプレイ等）を問合せの write setと呼ぶ。関係代数木中の各ノードは一つの関係代数演算，SORT等に対応し，タスクと呼ばれる。特に，葉に相当するタスクを基底タスク，根に相当するタスクを出力タスク，他のタスクを中間タスクと呼ぶ。基底タスクはデータベースそのものにページを単位としてアクセスし，SELECTION,PROJECTION等を実行する。一方，出力タスクは結果のデータベースへの書込み，結果の表示等を行う。JOIN等は中間タスクとして実行される。タスクの構造を図1に示す。全てのタスクはSource空間，Filter処理，Clustering処理，Sink空間の4つの要素から構成される。タスクはそのSource空間からSink空間へのデータの送出を行うことにより実行され，これによって1つの関係代数演算の処理が行われる。GRACEの計算モデルに於ける基本原理は，全ての演算をこのタスク構成によって実行することである。即ち，オペランドリレーション全体を予め当該演算に関して互いに独立なcluster に分割しておき（Clustering処理），これらcluster に属するタプルが構成する複数のデータストリームの束に対して並列に演算を行う（Filter処理）。全ての演算はデータストリーム中のタプルのふるい落し，結合，sort等の組合せであり，ここではこれらを一括してデータストリームに対するFilter処理とみなす。タスク中ではFilter処理と同時に次タスクに対するClustering処理がデータストリームに沿って施される為，Clustering処理に関するオーバヘッドは実効的に存在しない。

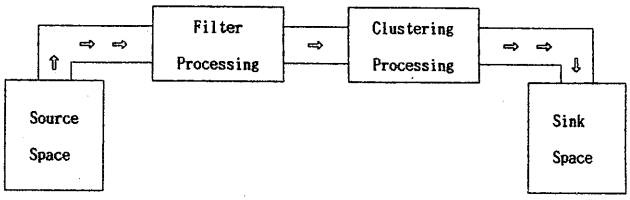

Fig. 1　The General Structure of Task

3．実行制御

GRACEに於ける問合せの実行制御は，所謂データフロー制御を基本としている。問合せはそのread setが確保可能の時実行可能となる。問合せの実行開始と共に，問合せ中のタスクはその木構造が許す限り並列に発火，実行され，出力タスクの終了をもって問合せの実行が終了する。タスクの接続は，タスクの終了後そのSink空間を次タスク（親タスク）のSource空間とすることによりなされ，問合せの木構造は，子タスクがそのSink空間を共有することによって実現される。

タスクは，その実行環境の割付が完了し，そのタスクに対する一意的なId（タスクId）が与えられた時発火可能であると言う。即ち，タスクのSource空間に対して子タスク

からのデータフローが全て流れ込み，処理に必要な実行環境が与えられ，各モジュールに対し処理すべき関係代数演算に応じたコマンド群が与えられ，更にこれらモジュール群がタスクIdによってタスクの構成を認識することによってタスクは駆動される。この意味でタスクはＧＲＡＣＥに於いて最小の制御／実行の単位であり，その発火の後は実行制御を必要とせず，自律的に動作する。

4．資源管理

一般に，タスクの実行環境は　＜Source空間構成ユニットの集合，Filter処理ユニット（Filter Processor）の集合，Clustering処理ユニット（Clustering Processor）の集合，Sink空間構成ユニットの集合，通信資源の集合＞の５つ組で表される。これら５種の資源はＧＲＡＣＥのハードウェアアーキテクチャに於いて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）の３種のモジュールと，これらモジュールを結合する高速リングバスのチャネル（通信資源）によって実装される。ハードウェアアーキテクチャに於いてはタスクの実行環境はタスクの種別に応じて図２の様に具体化される。即ち，一般に複数台のＭＭがSource空間／Sink空間を構成するが，基底タスクに於けるSource空間構成ユニットはread setに属するページとなり，これらページを格納するディスクとFilter Processor，Clustering Processorが一つのＤＭに一体化されている。また，中間タスクで使用されるFilter Processor，Clustering Processorは一つのＰＭに一体化されている。ＰＭは，更にハードウェアソータを内蔵し，JOIN等を高速に実行する。従って，ハードウェアアーキテクチャに於ける各タスクの実行環境は，基底タスクでは　＜read setに対応するページの集合及びこれらページを格納しているＤＭの集合，ＭＭの集合，チャネルの集合＞　によって，中間タスクでは　＜ＭＭの集合（Source），ＰＭの集合，ＭＭの集合（Sink），チャネルの集合＞　によって，また出力タスクでは＜ＭＭの集合，write set に対応するページの集合及びこれらページを格納しているＤＭの集合，チャネルの集合＞　によって構成されることになる。資源管理を司る実行制御モジュールはSource空間を形成するモジュール群からの要請に従い，タスクの種別に従ってこれらモジュールを各々複数台割付けることによってタスクの実行環境を構成する。また，巨大なリレーションのJOIN処理に際しては，Source空間／Sink空間をＭＭと作業用ディスクモジュール（ＷＤＭ）で構成し，空間を仮想化することによって対処する〔２〕。

5．おわりに

現在，資源管理方式の詳細化を行っており，今後，モジュール割付けのアルゴリズム等を中心に研究を進めてゆく計画である。

参考文献

〔１〕Boral,H.et al,『Implementation of The Database Machine DIRECT』，IEEE Trans. on Software Eng., Vol. SE-8, No.6, Nov. (1982)

〔２〕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，(1984)

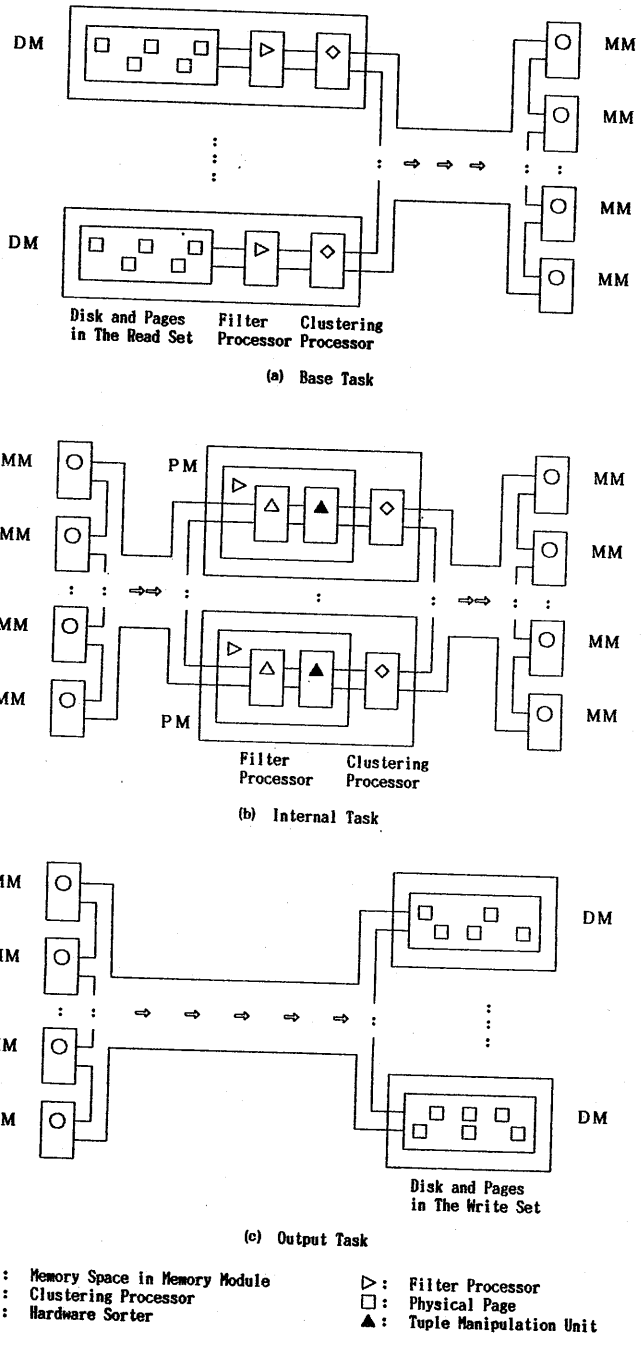

Fig. 2 The Execution Environment of Task in The Physical Architecture of GRACE