スーパーデータベースコンピュータ第二版 (SDC2) における マルチジョインの実装方式

1F－4

中村 稔　田村 孝之　喜連川 優　高木 幹雄
東京大学生産技術研究所

1 概要

我々は現在スーパーデータベースコンピュータ第二版 (SDC2) を開発中である．本システムはスーパーデータベースコンピュータ第一版 (SDC1) に対して各モジュールの処理性能の向上と高機能オメガネットワークによる多モジュールでの相互接続を行ない，多モジュール構成下での評価を行なっている．本論文ではまず，SDC2 のハードウェア並びにソフトウェア構成について簡単に触れたのち,SDC2 上でのマルチジョインの実装方式について述べる．

2 SDC2 の構成

SDC2 の構成を図 1 に示す．

SDC2 は最大でプロセッサ 7 台と磁気ディスク装置 4 台を密に結合したデータ処理モジュール，並びに，複数の処理モジュールを疎に結合する高機能オメガネットワーク [2] からなるハイブリッドアーキテクチャをとる．

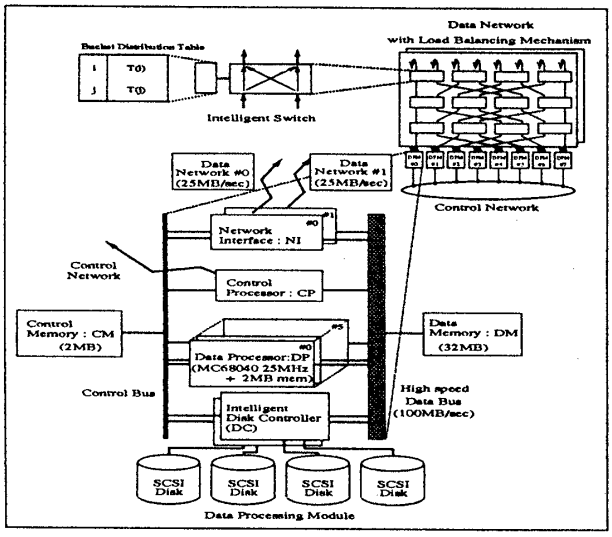

図 1:SDC2 の構成

SDC2 ではデータメモリを固定長のページに分割しモジュール内でのデータ交換はすべてこのページを受け渡すことで行なわれる [1]．ページの受渡しはコントロールメモリ上のバッファ構造体を通じて行なわれる．ページ内には複数のタプルが格納され，ページ長を越えない範囲でタプルの長さに制限はない．

ページをデータ交換の単位とすることでディスク，プロセッサ，ネットワークにまたがるデータの受渡しを単純化し統一的な扱いを可能にしている．

3 SDC2 におけるマルチジョインの実行

SDC2 において以下のようなマルチジョイン演算を実行することを考える．

select * from $R_1, R_2, R_3, \cdots, R_n$
where $R_1.a = R_2.a$
and $R_2.b = R_3.b$
and $R_3.c = R_4.c$
$\vdots$
and $R_{n-1}.m = R_n.m$

このような結合演算を行なう場合図 2 のような Left Deep と Right Deep およびそれ以外の，Bushy Tree とに分類される．

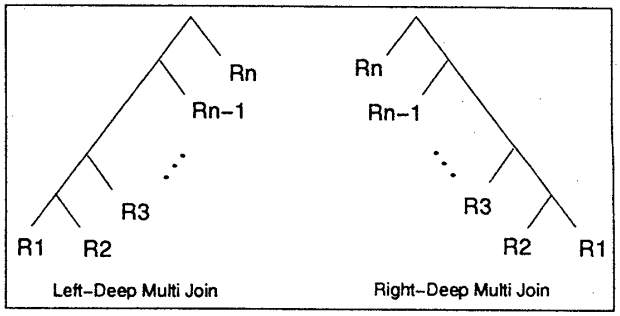

図 2:Multi-join query tree

4 SDC2 における処理の多重化

マルチジョイン処理では単一のジョインと異なり複数の種類のタプルを同時にとり扱わなければならない．このため SDC2 のシステムソフトウェアに多重処理の機能を採り入れた．

4.1 データ構造の拡張

SDC2 ではモジュール上の共有メモリ上に各種のバッファやテーブルなどのデータ構造が格納されている．処理の多重化を実現するために，共有メモリ上のデータ構造の拡張を行なう (図 3)．まず，ハッシュテーブルを多重化し，同時に複数のハッシュテーブルを使用できるようにする．次にバッファ構造体を拡張

Impremention method of Multi-Join for the Super Database Computer version II (SDC2)
M.Nakamura, T.Tamura, M.Kitsuregawa and M.Takagi
Institute of Industrial Science, University of Tokyo

する．データ交換のベースとなるページは施すべき処理の内容や行き先が異なるとその生成 / 消費の速度が異なってくる．そのため，異なった種類のページが一つのバッファ中を流れると，デッドロック発生の原因となる．そこで一つのバッファには一種類のページのみが流れるようにし，バッファの数を増やすことで同時に取り扱えるページの種類を増やす．

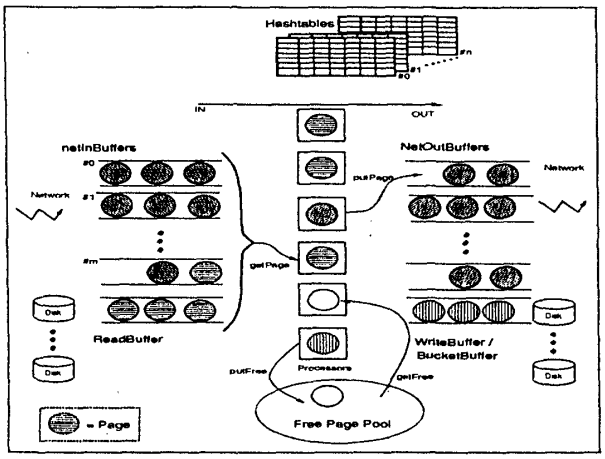

図 3:SDC2 におけるデータの流れ

4.2 システム制御

処理の多重化を行なうと二種類以上のページが処理プロセスに入力されるため処理内容を入力の種類に応じて切替えなければならない．SDC2 では処理の高速化のため，どのコンテキストのどのバッファから受けとったページであるかに応じてそのページに対する処理をデータ駆動的に切替える．処理の切替をタプル単位ではなくページ単位で行なうことで処理の高速化がはかれる．

Scenario

各モジュールの共有メモリ上に scenario と呼ぶ処理の内容を記述したテーブルをおく．scenario は現在の状態と入力の種類に応じて入力に対する処理 (action) と，次の状態を決定する状態遷移表と，処理に必要な付加的情報とからなる．付加的情報としては処理の対象となるアトリビュートの位置や属性，選択処理における述語の指定などがある．

以下に Left-deep マルチジョイン処理に対する scinario の状態遷移表の一部を示す．状態遷移表の action 部はその処理結果によって状態遷移を制御できるようにするため,next 部に action 部の戻り値を加えたものを次の状態として用いている．

```
DPP input       readBuf0, netIn0, netIn1;
DPP Control {
/* stat : input      :  next : output     : action */
    1   : read0      : 2     : keep       : hash;
    1   : netIn0     : 1     : hash0      : build;
    1   : eof        : 4     : none       : sync;
    2   : -          : 1     : netOut0    : netOut;
    3   : -          : 1     : hash0      : build;
    4   : read0      : 5     : keep       : hash;
    4   : netIn0     : 6     : keep       : hash;
    4   : netIn1     : 4     : hash1      : build;
    4   : eof        : 10    : none       : sync;
    5   : -          : 4     : netOut0    : netOut;
    6   : -          : 7     : keep       : probe;
    7   : -          : 8     : keep       : hash;
    8   : -          : 4     : netOut1    : netOut;
    9   : -          : 4     : hash1      : build;
   10   : read0      : 11    : keep       : hash
   10   : netIn0     : 10    : write0     : probe;
   10   : eof        : -1    : none       : done;
   11   : -          : 10    : netOut0    : netOut;
   12   : -          : 10    : write0     : probe;
}
```

Actor

モジュール内の処理プロセッサ上で処理を行なうプロセス群を actor と呼ぶ．actor は scenario によって動く状態遷移機械とみなすことができる．各 actor は監視すべき入力が指定されており，そのいずれかにページが到着すると入力と現在の状態から scenario を検索し処理内容を決定する．

actor は処理の動的拡張をサポートする．これは以下のように実装される．

- scenario の action 部には組み込み関数の ID，関数へのポインタ，機能モジュール名へのポインタのいずれかをセットできる．関数へのポインタは実行時にそのまま直接評価される．
- 組み込み関数の ID は最初に実行時される時に関数へのポインタに変換され action 部の書換えが行なわれる．
- 機能モジュール名が指定された場合は指定された機能モジュールをフロントエンドモジュールからロードし動的にリンクする．action 部はロードされた機能モジュールの処理関数へのポインタに書き換えられる．
  処理速度の低下を防ぐため，機能モジュールはインタープリタ形式をとらずに実行オブジェクトを直接，動的リンクする．
  機能モジュールの動的拡張によって例えば複合的な関数を高速に実行することができる．

5 まとめ

SDC2 におけるマルチジョイン処理の実装方式について述べた．現在，処理性能について評価中である．

参考文献

[1] 中村稔，田村孝之，喜連川優，高木幹雄．スーパーデータベースコンピュータ第二版 (SDC2) におけるデータ流制御の評価．アドバンストデータベースシンポジウム，pages 103–112, 1993.

[2] 田村孝之，中村稔，喜連川優，高木幹雄．スーパーデータベースコンピュータ (SDC2) におけるデータネットワーク系の実装．電子情報通信学会技術報告，*CPSY93-31*, 1993.