3T－2

分散並列マシン Cenju-3 における仮想共有メモリ実現方式について

†稲村 雄,　†浅野 由裕,　‡丸山 勉
†(株) NEC 情報システムズ,　‡ NEC C&C 研究所コンピュータシステム研究部

1　はじめに

Cenju-3 はプロセッシングエレメント (PE) として RISC プロセッサ VR4400 を用い、各 PE を独自設計のネットワークハードウェアで結合した分散メモリ型並列マシンである。その OS である Cenju-3 OS では、柔軟なプログラミング環境を提供するために、『仮想共有メモリ方式』と呼ぶ、他 PE 上のローカルメモリへの透過的なアクセスを可能とするメモリ管理方式をサポートしている。

本稿では、このような仮想共有メモリ方式を Cenju-3 OS 上で実現するために採用した、PE (VR4400) の例外処理に関する特徴を活かした実装手法等についての説明を行なう。

2　VR4400 メモリ管理方式

Cenju-3 では PE として VR4400 を 32 bit モードで使用している。32 bit モードということは、仮想アドレス空間として 4 GByte の領域が使用できることになるが、$MSB = 1$ であるアドレスはシステムプログラム用特権空間として予約されているため、ユーザ空間として実際に使用できる仮想アドレス空間は 2 GByte となる。

ユーザ空間仮想アドレスはプロセッサ内蔵の TLB により物理アドレスへ変換される。この時、TLB 内に該当するエントリが存在しなかった場合には、**TLB Refill 例外** と呼ばれる例外が発生し、OS は独自に管理するページテーブルから必要なエントリを捜し出して、新たに TLB の空きエントリに格納する等の例外処理を行なう。

VR4400 では、TLB Refill 例外発生を効率良く処理するために、他の例外事象とは異なる特別なハンドラが起動される。したがって、例外処理時にはユーザコンテキストのセーブが一部省略できる等、比較的軽微な手間でアドレス変換の失敗から回復できる。

3　仮想共有メモリ方式およびその Cenju-3 OS での実現

仮想共有メモリ方式というのは、Cenju-3 のような分散メモリ型並列計算機上で、他 PE 上のローカルメモリを自 PE 上の仮想アドレス空間にマップすることで、透過的なメモリアクセスを可能とし、仮想的に共有メモリ型並列計算機と同等なプログラミング環境を提供するための手法である。

Cenju-3 OS では、完全にソフトウェアのみによる仮想共有メモリの実装を行なっているのが特徴となっている。以下の各節に、その実装方式について述べる。

3.1　アドレス空間の分割

ユーザ空間仮想アドレスを任意の一点で分割し、その下位アドレス部をローカルエリア、上位アドレス部をリモートエリアとする (図 1)。

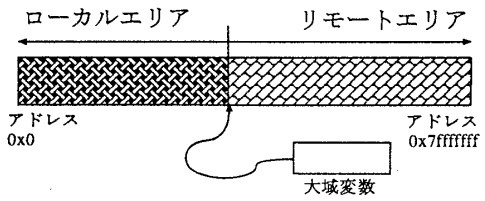

図 1 ユーザ空間の分割

両エリアの分岐点となるアドレスは、図に示した通り OS の持つ大域変数に保持されている。このようにソフトウェアプログラマブルとしたことで、たとえばメモリ容量の異なるシステム構成にも柔軟に対応することが可能となっている。

Virtual Shared Memory Management on Cenju-3
†Yu Inamura, †Yoshihiro Asano, ‡Tsutomu Maruyama
†NEC Informatec Systems and ‡ NEC Corporation

3.2 リモートメモリの仮想アドレスへのマッピング

リモートメモリのマッピング方式として、ナイーブには全ユーザアドレス空間を均等に分割し、分割した各セグメントに他 PE のローカルメモリを静的にマップするという方式が考えられる (図 2 (a))。

しかし、この方式では実現可能なシステム規模が制限されるという問題があり、最大 256 PE 構成が可能な Cenju-3 の場合には明らかに都合が悪い。

そこで、Cenju-3 OS では他 PE のローカルメモリを動的に自アドレス空間へマップ/アンマップするシステムコールを用意することで、この問題に対処することとした。マップは VR4400 のメモリ管理単位であるページ[†1]毎に行なわれる (図 2 (b))。

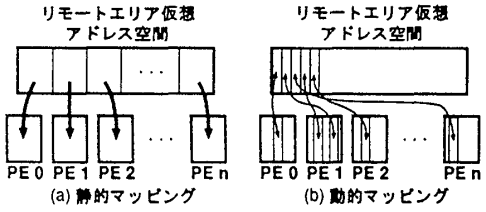

図 2 リモートメモリのマッピング

3.3 リモートページテーブル構成

リモートメモリのマッピング状態を管理するために、Cenju-3 OS は図 3 に示すようなページテーブルを管理している。

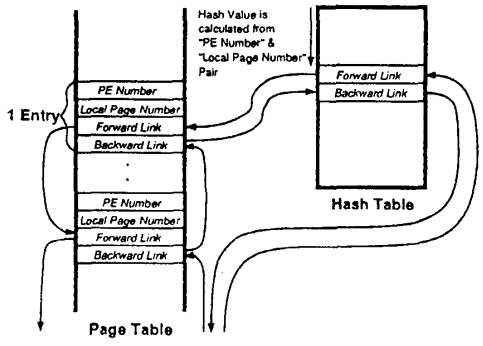

図 3 リモートページテーブル

ページテーブルの各エントリは

- メモリの実体が存在する PE 番号
- PE 内ローカル仮想ページ番号
- 前向きハッシュポインタ
- 後向きハッシュポインタ

の 4 語からなり (図 3)、仮想ページ番号によってインデクスされる。ここで前向き/後向き二本のハッシュリンクを用意しているのは、(1) ある "PE 番号&ページ番号" ペアがマップ済かどうかを高速に判定するためと、(2) アンマップされたエントリを高速にハッシュリンクから除去するためである。

3.4 リモートアクセス処理

OS はリモートエリアに属する仮想アドレスを決して TLB には設定しないため、リモートアドレスへのアクセス時には常に TLB Refill 例外が発生することになる。実際のリモートアクセス処理は TLB Refill 例外ハンドラがメモリアクセス処理をエミュレートすることで行なわれる。すなわち:

- ページテーブルでマップ状態のチェック
- 当該命令の解析
- remote_read/write 命令を用いたリモートメモリへの load/store 処理の実行
- 次命令からの通常実行再開

というような流れで例外処理を行なう。

Cenju-3 OS の場合、ローカル物理メモリ↔仮想アドレスのマッピングは全 PE で一様であるため、リモートアドレスの有効/無効はアクセスを行なう側の PE でローカルに判定可能であり、store 処理は remote_write 命令を当該 PE へ発行するだけでただちに終了できる。load 処理の場合は、例外ハンドラ内で結果を待ち合わせている。

4 おわりに

本稿では、Cenju-3 OS で採用した仮想共有メモリ方式に関する説明を行なった。本方式は完全にソフトウェアのみによる実現であるため、システム構成の変更などに柔軟に対応できるものとなっている。

しかし反面、パフォーマンス的には不利となる面もあるため、今後はリモートメモリに関する情報のキャッシングなどによる性能改善方法を検討したいと考えている。

†1 サイズは可変であり、Cenju-3 OS の場合 8 KByte に設定