ＰＣクラスタにおけるユーザレベルソフトウエア分散共有メモリの実現

緑川 博子、斉木 雅弘*、飯塚 肇　　成蹊大学　工学部(*現在、富士通)
User-Level Software Distributed Shared Memory on PC cluster
Midorikawa H., Saiki M., Iizuka H.　Seikei University

概要
PCクラスタ上にユーザレベルソフトウエアによる分散共有メモリシステムの実装を試みた。Weakメモリコンシステンシモデルを採用し、ページベースの管理を行う。その設計、実装結果について報告する。

【システムの特徴】

・ユーザレベルソフトウエアによる実装

通信ハードウエアに依存しないソケット通信を用い、OSに手を加える必要のないユーザレベルソフトウエアにより実装したため、移植性が高い。

・緩和型メモリコンシステンシーの採用

分散共有メモリの一貫性をとるモデルには、Weakメモリコンシステンシモデルを用いた。

・ページベース管理

一貫性維持の粒度はページ単位で、応用プログラムによる共有メモリデータのアクセス検知には、OSで提供されるページ保護機構を利用した。

・差分(diff)データ転送

共有メモリデータの一貫性を保つために、前回のバリア同期時との差分(diff)を、該当ページコピーを所有するプロセッサにのみに更新データとして送り、通信量の削減を図った。

【システムの構成要素】

・ページマネジャ

共有データ（ページ）管理プロセス。各データアロケート時にユーザが指定可能。

```
#include <stdio.h>
#include < sms_usr.h>  /* システム用ヘッダ */
#define MAX 500
#define START  sms_proc_id * MAX / sms_nproc
#define END  (sms_proc_id +1) * MAX/sms_nproc
void main(int argc,char *argv[ ])
{ int  i,  *array;
   sms_startup (argc, argv);   /*システム開始*/
   /* 共有メモリ確保 */
   array = (int *) sms_alloc (sizeof (int) * MAX, 0);
   for( i=START;  i<END;  i++)   /* ジョブ分割 */
     array[ i ]= i;   /* 共有メモリへの書き込み */
   sms_barrier(0);   /* バリア同期 */
   for(i=0;  i<MAX  &&  sms_proc_id ＝0;  i++)
     printf(" %d  ", array[ i ]);   /* 結果出力 */
   sms_shutdown();      /* システム終了 */
}
```

図1　プログラム例

・バリアマネジャ

バリア同期管理プロセス（ユーザ指定可）

・sigsegvハンドラ

共有メモリアクセス時に呼び出され、ページ要求や、diff生成のための準備を行うハンドラ。

・メッセージハンドラ

ページ要求、diff転送など、プロセッサ間通信時に呼び出される通信ハンドラ。

・分散共有メモリライブラリ（libsms）

図１のプログラム例に使用されている４種のCの関数、２種の定数をAPIとして用意した。

【評価】

本実験は図２の環境で評価を行った。図３に基本部分のオーバヘッド時間を示す。図４にNPB (EP) プログラムの速度向上比を示す。

ＰＣ	CPU:MMXPentium166MHz
	メモリ:64MB
	OS:FreeBSD2.2.2R
ネットワーク	100Mbpsイーサネット

図2　システム環境

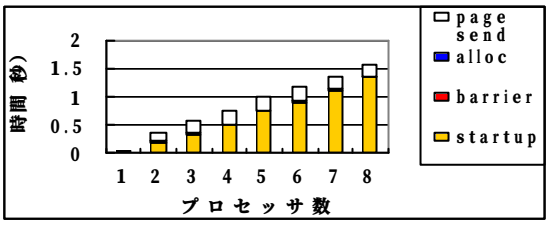

図3　システム開始時のオーバーヘッド

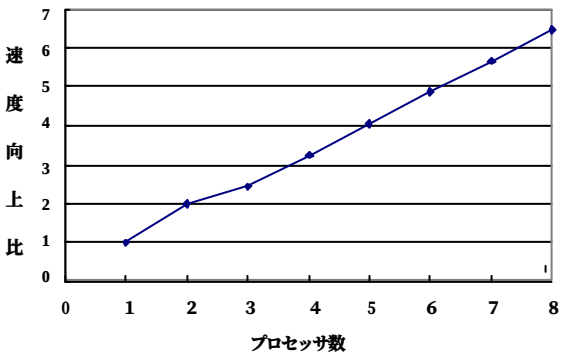

図4　NPB（EP）プログラムの性能向上比