ハイパフォーマンス
コンピューティング 92－1
（２００２．１０．２５）

Itaniumプロセッサによる SCoreクラスタ構築に関する検討

石川 裕[†] 高橋大介[††]
朴 泰祐[††] 佐藤三久[††]

4 台の 4 way Itanium (800 MHz) プロセッサから構成されるクラスタ上に SCore クラスタシステムソフトウエアを移植し、Itanium による SCore クラスタの性能を測定する。Pentium-III (933 MHz) プロセッサによるクラスタと比較した結果、姫野ベンチマークでは、単体性能で Itanium プロセッサは Pentium III プロセッサの 3 倍の性能がある。NAS 並列ベンチマークの CG の結果では、16 プロセッサ構成までの比較で、Itanium プロセッサは Pentium III プロセッサの 2.7 倍〜1.3 倍高速である。

A Feasibility Study on an Itanium-based Cluster

Yutaka Ishikawa,[†] Daisuke Takahashi,[††] Taisuke Boku[††]
and Mitsuhisa Sato[††]

The SCore cluster system software has been ported to a cluster of four Itanium-based machines each of which consists of four way 800 MHz Itanium processors. The comparison with a 933 MHz Pentium III-based cluster shows that the Itanium cluster is three times faster than the Pentium III cluster in the Himeno benchmark. As the result of the CG benchmark of the NAS parallel benchmarks up to 16 processors, the Itanium cluster is 2.7 to 1.3 times faster than the Pentium III cluster.

1. はじめに

2001 年、Intel 社の IA64 64 bit アーキテクチャの最初の製品である Itanium プロセッサを搭載したサーバコンピュータが市場に登場した。しかし、IA64 アーキテクチャを使用したクラスタ構築は非常に稀である。これは、コンパイラやオペレーティングシステムの IA64 対応が遅れ、性能や安定面の問題が指摘されたこと、さらに、Itanium プロセッサを搭載したサーバコンピュータは、小規模でも高さが 7U(約 30cm) あり、また、消費電力も高いためである。

2002 年 7 月に Itanium2プロセッサが発表され、秋には本格的に Itanium2 プロセッサを搭載したサーバコンピュータが市場に出回る。Intel 社の発表では、Itanium プロセッサの 1.5〜2 倍の性能向上が見込まれる。また、HP 社からは高さ 2U(約 8.8cm) のサーバ製品が発表された。

ソフトウエア面に関しても、IA64 アーキテクチャ対応の Linux カーネルは安定性および性能面で向上している。また、現在、Intel 社の IA64 アーキテクチャ対応コンパイラは、アカデミックユーザに対して無償で提供されている。

このように、IA64 アーキテクチャを基にしたクラスタ構築が現実的となった。しかし、クラスタ構築に必要なソフトウエアが整備されていないため、実際にどの程度の性能が出るのかが分からない。

そこで、クラスタ用システムソフトウエアである SCore クラスタシステムソフトウエア [10] を IA64 アーキテクチャに移植し、Itanium プロセッサによるクラスタ上で評価を行う。

以下、第 2 章で SCore の特徴を紹介した後、第 3 章で IA64 アーキテクチャと Linux の特徴について紹介する。第 4 章で SCore の移植におけるアーキテクチャ依存部分について紹介した後、第 5 章および第 6 章で評価および検討を行なう。

2. SCore

SCore ソフトウエアアーキテクチャを図 1に示す。

2.1 PMv2 通信ライブラリ

PMv2 通信ライブラリは、クラスタ コンピューティング用低レベル通信ライブラリである [4]。PMv2 API (Application Program Interface) は、クラスタにおける複数種類のネットワークや共有メモリ

† 東京大学大学院情報理工学系研究科
†† 筑波大学電子・情報工学系

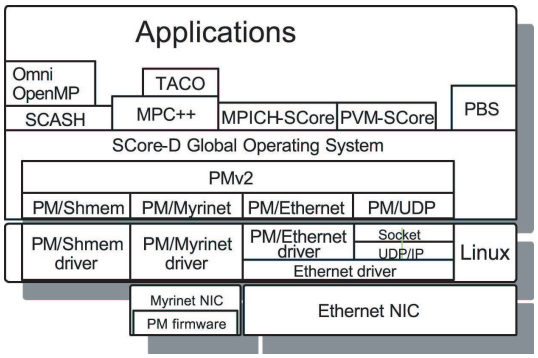


図 1 SCore

に同一の方法でアクセスできるように設計されている。

- PM/Myrinet
  Myricom 社 Myrinet ネットワーク用 [11] の PM 通信レイヤである。PM/Myrinet は、Myrinet NIC 上のプログラム、Linux カーネルドライバ、ユーザレベルライブラリから構成されている。PM/Myrinet では、ユーザレベル通信およびゼロコピー通信と呼ばれる手法を用いて低遅延、高バンド幅ネットワーク通信を実現している。
- PM/Ethernet
  PM/Ethernet は、Ethernet 上における PM 通信プロトコルを実現している。Ethernet デバイスドライバの上に構築されたカーネル内プロトコル処理ルーチンとユーザレベルライブラリから構成される。PM/Ethernet は、TCP/IP と比べて軽量なプロトコル処理で済むように設計されている。
- PM/Shmem
  PM/Shmem はオペレーティングシステムの共有メモリ機構を利用して実現されている。PM/Shmem により同一コンピュータ内で複数のプロセスが PM 通信 API で通信が可能となる。

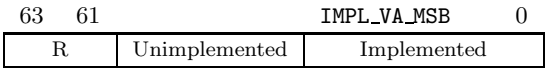
2.2 MPICH-SCore

MPICH-SCore は、MPI 通信ライブラリを実装したフリーソフトウエアである MPICH を PM 通信ライブラリ上で稼動するようにしたソフトウエアである。MPICH-SCore では、単一プロセッサから構成されるクラスタだけでなく、共有メモリ型並列コンピュータから構成されるクラスタ上でも効率良く稼動するように工夫している [7]。

2.3 SCore-D

SCore-D は、複数のユーザが同時にクラスタを利用するときに効率良くコンピュータ資源を管理する機能を提供するグローバルオペレーティングシステムである。SCore-D は、カーネルを変更することなくデーモンプロセス群で実現されている。ギャングスケジューリングと呼ばれるスケジューリング手法を用いて、並列アプリケーションを時分割スケジューリングしている [6]。さらに SCore-D では次のような機能を提供している。

- 実時間ロードモニタ
- デッドロック検出
- チェックポイント・リスタート機能
- 対話型デバッガ起動



R: Region

図 2 仮想アドレス

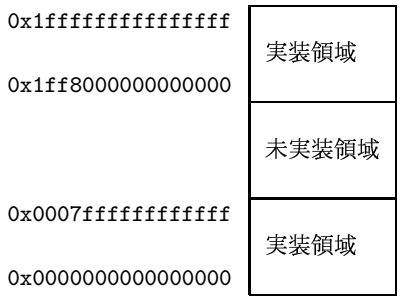


図 3 リージョン内でのアクセス可能なメモリ領域例

2.4 Omni OpenMP on SCASH

SCASH は、カーネルを変更することなく、PM 通信ライブラリを用いてユーザレベルで実現したソフトウエア分散共有メモリシステムである [5]。Omni OpenMP は SCASH ソフトウエア分散共有メモリシステムを使い、OpenMP のプログラムを変更しなくてもクラスタ上で動く [3]。

3. IA64 アーキテクチャと Linux

IA64 は EPIC (Explicitly Parallel Instruction set Computing) パラダイムに基づいたアーキテクチャであり、predication、control speculation、data speculation、register stack を有する [1]。また、オペレーティングシステムによる仮想空間を柔軟に実現できるために、TLB やページ管理機構に特徴を有する [1]。

本章では、SCore の機械依存部分およびカーネル依存部分に関係する IA64 アーキテクチャおよび Linux の特徴についてまとめる。なお、以降、IA64 アーキテクチャ用 Linux を Linux/IA64 と呼ぶことにする。

3.1 メモリ空間

IA64 アーキテクチャでは、64bit アドレス空間を上位 3bit を使って 8 つのリージョンに分けている。各リージョンは、2^{61} バイト、すなわち、2 Exabytes

表 1　ユーザプログラムが実際に使用しているリージョン

region	address	purpose
3	0x6000000000000000	データおよびヒープ領域
	0x60000fffffffc000	スタック
2	0x4000000000000000	テキスト領域
1	0x2000000000000000	共有メモリおよびヒープ領域

の空間を持っているが、実際には、図 2 に示す通り、60 ビット目から IMPL_VA_MSB + 1 ビット目までのビットは使用されない。IMPL_VA_MSB はプロセッサモデルに依存した値であり、Itanium プロセッサの場合、50 である。IMPL_VA_MSB + 1 ビット目が 1 の場合は、60 ビット目まで全てが 1 にセットされなければならない。図 3 に IMPL_VA_MSB が 50 の時の、リージョン内でアクセス可能なメモリ領域を示す。

IA64 用にコンパイルされたユーザプログラムは、表 1 に示すような使われ方をしている。文献 [2] では、リージョン 4 がプロセスのメモリおよびレジスタスタック用に使われていると述べられているが、実際には使われていない。

3.2　ページサイズ

IA64 アーキテクチャでは、リージョン毎にページサイズを 4 KBytes から 256 MBytes まで設定可能である。Linux/IA64 では、カーネル構築時に、ページサイズを 4, 8, 16, 64 KBytes のいずれかに設定することが出来る。

3.3　RSE

RSE(レジスタ・スタック・エンジン) は、Sun Sparc アーキテクチャのレジスタウインドウ同様、汎用レジスタを仮想的に無限のレジスタ群として抽象化する機能を提供する。サブルーチンコール時に、論理レジスタは新しい物理レジスタに rename される。仮想的に無限のレジスタ群を提供するために、プロセッサは必要に応じて物理レジスタの内容をメモリに退避／復帰する。

ユーザレベルスレッドを実現する場合には、退避されている物理レジスタの内容も考慮しなければならない。

4. 実　　装

Itanium 用 Red Hat 7.1 が使用している Linux 2.4.3 カーネルは安定していない。今回は、Linux 2.4.18 標準カーネルに IA64 パッチを当てたカーネルを使用した。SCore のプログラム中、アーキテクチャに依存して変更した大きな箇所は以下の通りである。

4.1　ユーザレベルスレッドライブラリ

スレッド切り替えには、C ライブラリの setjmp および longjmp ルーチンを使用している。スレッド生成時には、setjmp ルーチンを呼ぶことによって、スレッド生成時のレジスタおよびレジスタスタックの退避を行う。そして、setjmp/longjmp が使用する構造体の中で、グローバルポインタ、プログラムカウンタ、RSE 用特殊レジスタ退避部分の値設定を行なっている。

表 2　評価環境

Itanium800 クラスタ

プロセッサ	Itanium 800 MHz x 4 processors
キャッシュ	4 MB(L3) 96KB(L2) 32KB(L1)
メモリバンド幅	2.1 GB/s
ネットワーク	Intel EEPRO100 (100Mbps Ethernet)
	Intel EEPRO1000 (1Gbps Ethernet)
	Myrinet (Lanai9) SAN ケーブル
	66MHz PCI64 に接続
ソフトウエア	SCore 5.1 (Linux 2.4.18)
台数	4 台

SCore III クラスタ

プロセッサ	Pentium 933 MHz x 2 processors
キャッシュ	512KB(L2) 32KB(L1)
チップセット	ServerSet III LE
FSB	133MHz
ネットワーク	Intel EEPRO100 (100Mbps Ethernet)
	Myrinet-2000 (Lanai9) 33MHz PCI64 に接続
ソフトウエア	SCore 4.1.1 (Linux 2.2.16)
台数	512 台

4.2　クロックカウンタ

PM 通信ライブラリはタイミングの計測にプロセッサクロックカウンタを使っている。IA64 アーキテクチャが持つプロセッサクロックカウンタを参照するアセンブラルーチンを記述した。

4.3　ページサイズ

ページサイズは 16KBytes とした。

4.4　PM 通信ライブラリの 64bit アドレス対応

通信領域の仮想メモリアドレスと物理メモリアドレスの対応を管理するために、PM 通信ライブラリでは、pmAddrHandle と呼ぶデータタイプを用意している。pmAddrHandle は 64bit 長で、16bit のプロセス ID 部と 48bit のアドレス部の対で構成されるが、PM/Myrinet ではアドレス部は 32bit しか使用していない。

今回の移植では、このアドレス空間のままとした。表 1 に示すように、Linux/IA64 では、通信領域はリージョン 3 あるいはリージョン 1 のいずれかである。そこで、pmAddrHandle のアドレス部の内の 32bit アドレス空間を 2 分割し、各リージョンで 31bit 空間内の仮想アドレスを指定できるようにした。

5. 評　　価

表 2 に評価環境を示す。Itanium800 クラスタの

表 3　PM/Myrinet によるバス性能

	SCore III 933 MHz with Lanai9	Itanium800 800 MHz with Lanai9
CPU READ	5.6 Mbyte/s	5.8 Mbyte/s
CPU WRITE	8.3 Mbyte/s	98.0 Mbyte/s
DMA READ	260 Mbyte/s	248 Mbyte/s
DMA WRITE	259 Mbyte/s	232 Mbyte/s
DMA READ&WRITE	129 Mbyte/s	121 Mbyte/s

Myrinet では、ネットワークインターフェイスは Myrinet-2000 を使用している。しかし、スイッチは 1.28Gbps Myrinet なので、通信性能は 1.28Gpbs となる。

SCore III は、株式会社日本電気が所有し、PC クラスタコンソーシアム [9] のメンバが利用できるクラスタである。今回は、512 台の内の 16 台までを使用している。SCore III は、2Gbps 通信性能を持つ Myrinet-2000 を使用している。

5.1 バ ス 性 能

表 3 に、PM/Myrinet によるホストメモリと Myrinet NIC メモリ間の転送性能を示す。CPU READ および CPU WRITE は、プロセッサが Myrinet NIC 上のメモリを読み書きした時の性能である。SCore III では 4 バイト単位での読み書き、Itanium800 クラスタでは 8 バイト単位で読み書きした時の性能を示している。

DMA READ および DMA WRITE は、Myrinet NIC が持つ DMA 機構を使って読み書きした時の性能である。SCore III では、33MHz 64bit PCI バス理論ピーク性能である 264 Mbyte/sec とほぼ同じ性能が出ているのがわかる。Itanium800 クラスタでは、66MHz 64bit PCI バス理論ピーク性能である 528 Mbyte/sec の半分程の性能となっている。

5.2 基本通信性能

図 4 に MPICH-SCore の通信バンド幅性能を示す。SCore III の場合には、MPICH-SCore のゼロコピー通信性能も示している。

Itanium800 クラスタでは、Myrinet の物理通信性能である 1.28Gbps にほぼ近い、130MByte/sec の性能が出ている。Gigabit Ethernet の場合は、102MByte/sec の性能が出ている。

Myrinet を使用した時の 4 バイトメッセージの往復遅延時間 (RTT) は、Itanium800 クラスタで、22.4 マイクロ秒、SCore III で、17.1 マイクロ秒である。

5.3 姫野ベンチマーク

圧縮流体解析コードを使用した性能評価プログラムであり、ポアソン方程式をヤコビ反復法で解いた時の主ループにおける処理速度を計測している姫野ベンチマークプログラム [8] を 1 台のプロセッサで実行した時の結果を表 4 に示す。ここでは、Pentium III 以外に 1.5GHz Pentium 4 の結果も示す。

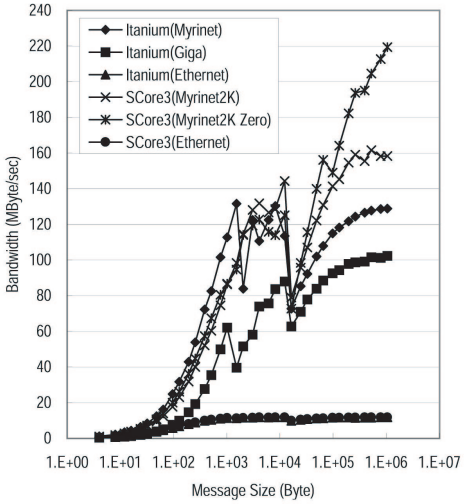


図 4　MPI 通信性能

表 4　姫野ベンチマーク
問題サイズ: Middle (128 x 128 x 256)。表中の単位は MFlops

	SCore III 933MHz	Pentium4 1.5GHz	Itanium800 800MHz
Intel	168.4	407.6	546.9
GNU	167.5	251.1	76.5

表 4 において、Intel とは、Intel 製コンパイラパッケージ ID l_cc_p_6.0.139 を使用し、コンパイルオプションに-O3 を指定してコンパイルした時の性能である。Pentium4 では、コンパイルオプションとして-O3 -tpp7 -xW を指定した。GNU とは、GNU C コンパイラ version 2.96 20000731 を使用し、コンパイルオプションに-O3 を指定してコンパイルした時の性能である。

5.4 NAS 並列ベンチマーク

NAS 並列ベンチマークプログラム集の中から, 整数並び換え問題 (IS), 共役勾配法による最小固有値を求める問題 (CG), フーリエ変換を使った 3 次元偏微分方程式解法（FT）, 3 次元ナビアストークス方程式を SSOR 法により解く疑似アプリケーション (LU)、3 次元ポアソン方程式を簡略化したマルチグリッド法 (MG) の性能差を、それぞれ、図 5、図 6、図 7、図 8、図 9 に示す。問題サイズは Class A である。

Itanium800 クラスタでは、表 5 のような構成でプロセッサを使用した。SCore III はプロセッサが 2 台搭載されたホストから構成されているが、測定には各ホスト 1 台のプロセッサのみを使用した。

表 5　使用プロセッサ構成

プロセッサ数	構成
2	1プロセッサ x 2 台
4	2プロセッサ x 2 台
8	4プロセッサ x 2 台
16	4プロセッサ x 4 台

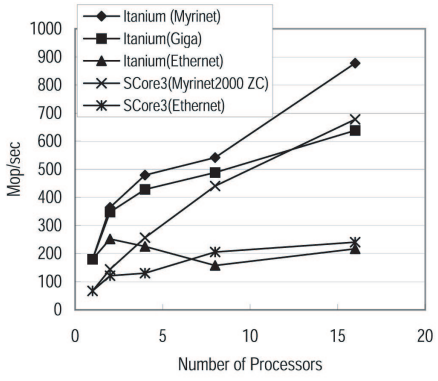


図 5　NAS 並列ベンチマーク CG Class A

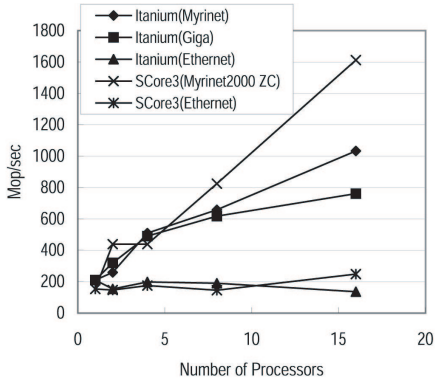


図 6　NAS 並列ベンチマーク FT Class A

6. 検　　討

単体性能について議論する。表 4 より、姫野ベンチマークの結果から単体性能において、800 MHz Itanium プロセッサは 933MHz Pentium III プロセッサに比べ 3 倍高速である。さらに、1.5GHz Pentium 4 プロセッサと比べても 1.3 倍高速である。これはメモリバンド幅の大きさが大きく影響している例であると考えられる。

Itanium の GNU コンパイラを使用した時の結果は、Pentium III よりも性能が出ていない。GNU コンパイラが Itanium 用に最適なコードを生成していないことがわかる。

並列性能について考察する。図 10 は、Itanium800 クラスタと SCore III を NAS 並列ベンチマークの性能で比較している。1 台使用時の性能

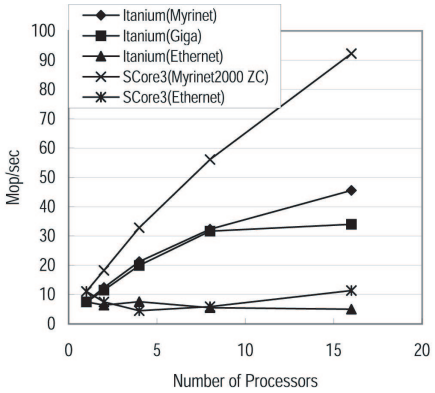


図 7　NAS 並列ベンチマーク IS Class A

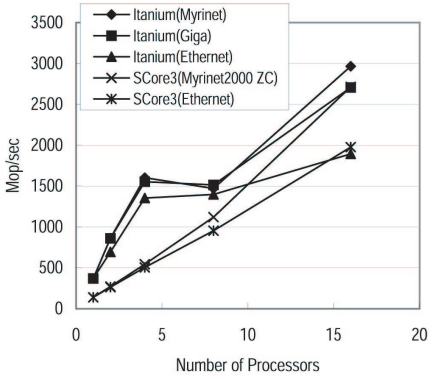


図 8　NAS 並列ベンチマーク LU Class A

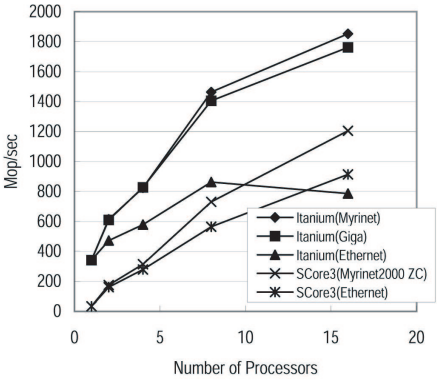


図 9　NAS 並列ベンチマーク MG Class A

では、MG で Itanium が 5 倍高速、CG、LU で 2 倍以上高速であるが、IS では約 1.5 倍遅くなっている。いずれのベンチマークでも、台数が多くなると、SCore III との性能差が縮まっている。

Itanium800 クラスタで台数効果が低いのは、16 プロセッサにおいて、Itanium800 クラスタでは、4

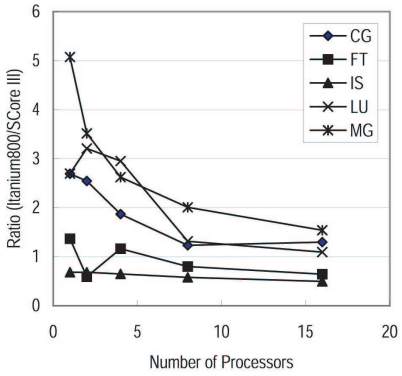


図 10 Itanium800 と SCore III の比較

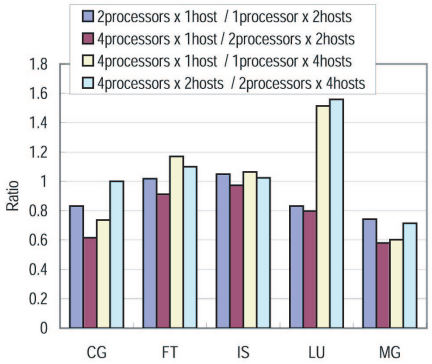


図 11 SMP とクラスタの比較

way の SMP を 4 台使用しているためである。Itanium800 クラスタにおいて、ホスト内のプロセッサを使った場合と、Myrinet ネットワークを使って同数のプロセッサを使った場合の性能比を図 11 に示す。本グラフより、FT, IS において、2 way の SMP と 2 台構成のクラスタでは、SMP の性能がわずかに高いだけで、多くの場合は、クラスタの方が良い結果となっている。4 way の SMP と 4 台構成のクラスタで比較すると、LU において顕著に SMP の性能が高い。しかし、図 8 に示す通り、絶対性能では、Pentium III のクラスタとほぼ同じ性能しか出ていない。ネットワーク性能が低いことにより、クラスタのスケーラビリティが減少しているためと考えられる。

7. おわりに

本稿では、IA64 アーキテクチャ向け SCore クラスタシステムソフトウエアの移植と 4 台構成の 4 way Itanium SMP における性能評価を行なった。

Itanium 2(1GHz) では、メモリバンド幅が最大 8.5GB/s(HP 社 zx6000 の場合)、チップ上に L3、L2、L1 がそれぞれ 3 MB、256KB、32KB 搭載され、表 2 に示した Itanium の仕様と比較すると、Pentium 系プロセッサ以上の性能を有するクラスタがコンパクトに構築できる可能性がある。また、バス性能の向上により、SMP 構成でのクラスタの性能向上が期待できる。

今後、Itanium2 と Myrinet-2000 あるいは Myrinet 2XP[11] を使用した小規模クラスタを構築して、性能評価を行なっていく予定である。

謝　辞

本研究の一部は、科学研究費補助金基盤研究 (A)(1) 課題番号 14208026 「OpenMP による次世代高性能クラスタ計算機向け高度並列プログラミング環境の研究」による。PC クラスタコンソーシアムにおいて専任で SCore のメインテナンスを行なっている SRA 社の亀山氏には、移植の一部を行なって頂いた。また、移植において、OSDL(Open Source Development, URL: www.osdl.jp) の機材を使用させて頂いた。ここに感謝致します。

参　考　文　献

1) “Intel Itanium Architecture Software Developer’s Manual Volumes 1, 2 and 3,” Intel, December 2001.
2) D. Mosberger and S, Eranian, “ia-64 linux kernel,” Hewlett-Packard Books, Prentice Hall PTR, 2002.
3) 佐藤、原田、長谷川、石川、『Cluster-enabled OpenMP: ソフトウエア分散共有メモリシステム SCASH 上の OpenMP コンパイラ』並列処理シンポジウム JSPP’01、情報処理学会、pp. 15 – 22、2001.
4) 住元, 堀, 手塚, 原田, 高橋, 石川, 『高速通信機構 PM2 の設計と評価』情報処理学会論文誌、Vol. 41, No. SIG 5(HPS 1), pp. 80–90, 2000.
5) 原田, 手塚, 堀, 住元, 高橋, 石川, 『ソフトウェア分散共有メモリシステムにおけるページ転送方式の比較』情報処理学会論文誌、Vol. 41, No. 5, pp.1410–1419, 2000.
6) 堀、手塚、石川、『ギャングスケジューリングの高速化技法の提案』情報処理学会論文誌、Vol. 40, No. 5, pp. 2072–2083, 1999.
7) T. Takahashi, F. O’Carroll, H. Tezuka, A. Hori, S. Sumimoto, H. Harada, Y. Ishikawa, and P.H. Beckman, “Implementation and Evaluation of MPI on an SMP Cluster,” IPPS’99 2nd Workshop on Personal Computer Based Networks of Workstations, pp. 1178 – 1192, 1999.
8) 姫野ベンチマーク, http://w3cic.riken.go.jp/HPC/HimenoBMT/
9) PC クラスタコンソーシアム, http://www.pccluster.org
10) SCore クラスタシステムソフトウエア, http://www.pccluster.org
11) Myricom 社, http://www.myri.com