行政院國家科學委員會專題研究計劃成果報告

微核心作業系統多處理機上系統軟體之研製-整合型
子計劃三:叢集式微核心作業系統之研究(II)
Research on the microkernel based cluster operating system (II)

計劃編號: NSC87-2213-E009-026

執行期間: 86 年 8月 1日至 87年 7月 31日

主持人: 張瑞川 教授　國立交通大學資訊科學系

系

一、中文摘要

在叢集式計算環境中，內部連接(inter-connection)的效能直接影響到了叢集計算的效能，因此高速的傳輸速率是非常重要的。在這篇報告，我們研究了 OSF/1 AD3.0 叢集上內部連接的性質，移植 Myrinet 到 OSF/1 AD3.0 上，並修改 MCP(Myrinet Control Program)，使其在 MCP layer 就能夠保證可信賴，讓上層的系統核心不需要顧慮可信賴的問題。實驗的結果不僅讓 Myrinet 達到可信賴，同時最高傳輸速率可達 563Mbit/s。透過使用更好的內部連接裝置，也使得叢集式運算的效能以及擴充性都有改善。

關鍵詞:微核心、叢集式系統、對稱式多處理機系統叢集、分散式共享記憶體、Myrinet 界面技術、OSF 1/AD 作業系統。

Abstract

There is strong and growing interest in building servers based on clustering technology. Many high performance and cost efficient uni-processor or SMP are used as the building block and connected to each other through high speed link. High-speed transmission is very important for cluster-based and distributed computing architectures. In this report, we examine cluster inter-connection properties of OSF/1 AD3.0. We port Myrinet to AD 3 and build a efficient cluster. The MCP (Myrinet Control Program) is modified to ensure system reliability. By using a better inter-connection device, we improved our cluster performance and scalability. Experimental results show that the modified Myrinet is reliable and the that the modified Myrinet is reliable and the throughput is 563Mbits/s maximum.

Keyword: microkernel, cluster system, SMP cluster, distributed shared memory, SCI, Myrinet, OSF 1/AD

二、緣由與目的

微核心技術是目前作業系統研究的主流,而SMP及SMP cluster則爲當前硬體架構發展的重要趨勢,有鑑於此,我們準備採用微核心作業系統在SMP及SMP cluster的架構下進行下列各項研究：

1． OSF/1 AD 在Altos 9000 SMP上的移植。

2． 建構SMP cluster。

3. Cluster OS Global shared-memory 的設計與製作。

因爲Dolphin公司在SCI技術的提供上發生了延誤,在慎重的考量下,我們決定採用美國Myricom公司所生產的Myrinet技術來發展SMP cluster。

二、結果與討論

本計劃今年完成的工作如下: 利用SCSI及Myrinet技術建構SMP cluster。我們自行設計了一個可靠的MCP(Myrinet control program)以及驅動程式，提供了一個high bandwidth, low latency且reliable的連結。我們也修了OSF1/AD的DIPC、KKT等部份，將Myrinet移植到OSF/1 AD上，使其能有效的應用我們所發展的快速訊息傳遞技術。我們實驗的設備及環境如下：

1. OSF/1 AD3.0
2. 4-ports Myrinet 交換器(M2F-SW4)
3. Myrinet 網路介面卡 (M2F-PCI32-10556 with 256KB memory on-board) *4
4. Intel Pentium-133*2、Intel Pentium-133

SMP、Pentium II 233

實驗結果

Benchmark api_latency.c

api_latency.c 是 Myricom 所提供的 benchmark，它是一個 ping-pong test， 用來測量 MCP 的round trip time，我們測量的方式是用不同的封包大小，每個封包傳送 10000 次，在pentium133 SMP 與pentium II 233 上傳送，求得其平均round trip time，用原來的 MCP (不可信賴的 MCP)所測得如【表 1】、用我們的 MCP，無論封包大小，全部使用 DMA 所測得如【表 2】、用我們的 MCP，以 32Bytes 爲DMA 及記憶體複製爲分界所測得如【表 3】:

【表 1】原始 MCP 執行 api_latency 所得 Round Trip Time

封包 Size	Round Trip Time (*us*)	Bandwidth (Mbps)
8192	529	236.29494
4096	343	182.21574
2048	248	126.008064
1024	202	77.351486
512	179	43.645276
256	167	23.390717
128	162	12.056327
64	158	6.180776
32	157	3.110072
16	156	1.565004
8	156	0.782502
4	156	0.391251

【表 2】修改過 MCP 執行 api_latency 所得 Round Trip Time(全部使用 DMA)

封包 Size	Round Trip Time (*us*)	Bandwidth (Mbps)
8192	548	228.102189
4096	358	174.581056
2048	264	118.371218
1024	220	71.022757
512	195	40.270617
256	186	21.001367
128	181	10.790746
64	175	5.580357
32	174	2.806214
16	173	1.411217
8	173	0.705609
4	173	0.352805

由【表 2】我們可以看出它與【表 1】的不同在於round trip time 平均增加了 20*us*，這 20*us* 的由來是我們爲了達到可信賴而在封包表頭上多加了幾個欄位，同時在 MCP 中做了可信賴的控制(buffer 流量控制、錯誤訊息傳遞及重送) 所多得的額外負擔。

【表 3】修改過 MCP 執行 api_latency 所得 Round Trip Time(以32Bytes 做爲DMA 及記憶體複製分界)

封包 Size	Round Trip Time (*us*)	Bandwidth (Mbps)
8192	549	227.686713
4096	358	174.581056
2048	265	117.924524
1024	221	70.701356
512	196	39.859712
256	186	21.001356
128	182	10.731456
64	176	5.548651
32	166	2.941455
16	162	1.507041
8	160	0.762939
4	159	0.383869

由【表 3】與【表 2】做比較，當封包大小小於等於 32Bytes 時，由於使用記憶體複製的方式，故明顯的增快了 8～14*us*。

Bandwidth

我們測量的方式是讓發送端盡力的送出封包，而接收端亦盡力的接收封包，傳送 20000 個封包所得的平均值。實驗結果，我們的 throughput 最高可達 563Mbps，由於 Myricom 所附的 MCP 並不可信賴，如果要讓原來的 Myrinet 可以達成可信賴，則必須在 API 之上再加上一個層次(layer)來達成，爲了可以與之比較，我們也用原來的 MCP 及 API，以相同的演算法做了一個可信賴的 Myrinet 程式介面與我們修改過的 MCP 做比較，其結果如【圖 1】所示：

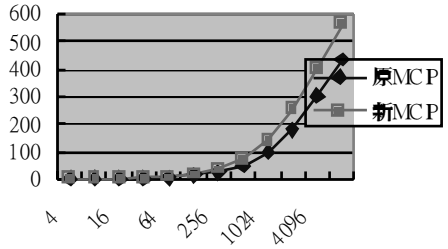

圖 1. Bandwidth 曲線圖

由【圖 1】我們可以看到平均加速比大約是 30%～45%，而很明顯的當封包大小小於等於 32Bytes 時，由於我們使用記憶體複製的方式，減少了 DMA 的延遲時間，故平均加速比在 56%～62%，因此可知當封包大小小於等於 32Bytes 時，使用記憶體複製的方式會得到較好的效率。由【圖 1】很清楚的看出與我們預期的結果一樣，把可信

轉的部份放在 MCP 中會比放在 API 之上快大約 40%。

Debug 核心測試

我們使用 DEBUG 核心裡面的測試 KKT 層的部分來測試效能。底下是測試的結果：

【表 4】KKT 層傳輸延遲

Message Size	SCSI (us)	Myrinet (us)	Myrinet2 (us)
280	1290	2030	1600
1024	1380	2060	1640
4096	1680	2190	1800
8192	2100	2490	2010
16K	2920	2960	2430
32K	4560	4110	3400
64K	7870	6170	5680
128K	14440	10290	9810
256K	27600	18490	18070
512K	53910	34920	34580
1M	106500	67840	67600
2M	211660	133950	133430

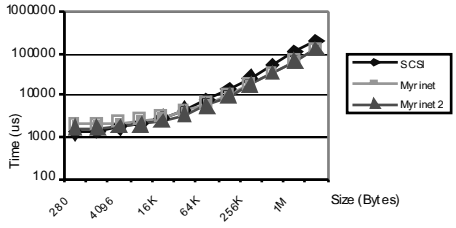

圖 3. KKT 層傳輸延遲

rfork 測試

接下來我們測試應用程式的結果，這個測試是使用 rfork。測試的方式就是在節點 0 執行一個程式，這程式做的事就是使用 rfork()在節點 1 上面建立一個行程，然後等節點 1 的行程結束後，再繼續 rfork 下一個行程。而節點 1 上面的行程開始執行後就直接跳出，不作任何事情。所以，測試的結果大致上就是使用 rfork 在另一台機器建立行程以及結束後刪除動作的時間。我們的測試是跑 1,000 次 rfork 後再算出每次的平均，測試結果如下：

【表 5】每次 rfork 時間

Interconnect Type	Time per rfork (sec)
SCSI	0.08585
Myrinet	0.08682
Myrinet 2	0.08584

由這個結果，我們可以看出改進後的 Myrinet 跟 SCSI 速度十分接近。

討論

根據以上的實驗結果，我們可以發現，在送比較小的訊息時，使用 SCSI 比較有利。而傳送長度大的訊息時，則使用 Myrinet 比較有利。

另外，使用 Myrinet 還有一些量測上比較不容易量出來的好處。第一點就是使用 Myrinet，我們可以很方便的擴充參與叢集運算的節點數。第二點是，當節點數較多時，Myrinet 不限制同時只能有一個訊息傳送，只要傳送目標不同，同時可以有多個訊息一起傳送。這樣等於是提升了傳送時的平行度，整個叢集的擴充性也會比較好。

AD 的問題討論

OSF/1 AD 是一個還在發展中的系統。所以難免會有一些問題以及不夠穩定的地方。在發展及測試我們的系統的時候，我們也遇到了幾個系統上的問題，在此也一起提出。

第一個問題是在測試 KKT 傳送資料時，dmt_do_kkt_test()中測試時會有 race condition 產生。修改方式就是在 KKT_REQUEST()之前先用 splkkt()設定中斷阻擋等級。而 assert_wait()後再用 splx()將中斷阻擋等級降回去。

第二個問題是關於中斷優先等級的問題。這問題主要原因是 KKT 設計時，爲了減少傳輸延遲，很多事情處理都是在中斷處理的時候進行。而當換用了比較快的網路裝置後，中斷的次數會非常多而且頻繁。當在傳送大量資料的時候，有可能會使得其他的裝置 (如硬碟)的中斷被阻擋過久無法正常的運作，而造成系統 crash 的現象。我們嘗試過將 KKT 裝置使用的中斷等級降得更低，這樣確實會大幅減少這問題發生的頻率，但是還是不能完全避免。

三、計劃成果自評

在叢集式計算環境中，內部連接(inter-connection)的效能直接影響到了叢集計算的效能，因此高速的傳輸速率是非常重要的。Myrinet 是一個具有 gigabit-per-second 的高速網路，它可以使用在各種不同的平台。

在這篇報告，我們研究了 OSF/1 AD3.0 叢集上內部連接的性質，移植 Myrinet 到 OSF/1 AD3.0 上，並

修改MCP(Myrinet Control Program)，使其在MCP layer就能夠保證可信賴，讓上層的系統核心不需要顧慮可信賴的問題。實驗的結果不僅讓Myrinet達到可信賴，同時最高傳輸速率可達563Mbit/s。透過使用更好的內部連接裝置，也使得叢集式運算的效能以及擴充性都有改善。

四、參考文獻

[1] Bill Bryant. Design of AD3, a Distributed UNIX Operating System. Open Software Foundation Research Institute, Version 1.0, May 1996.

[2] Nanette J. Boden, Danny Cohen,` Robert E. Felderman, Alan E. Kulawik, Charles L. Seitz, Jakov N. Seizovic, and Wen-King Su. Myrinet: A Gigabit-per-Second Local-Area Network. IEEE-Micro, Vol.15, No.1, February 1995, pp2936.

[3] Anthony Skjellum, Gregory Henley, Nathan Doss, Thomas McMahon. A Guide to Writing Myrinet Control Programs for LANai3.x. Fedruary 28, 1996.

[4] Pakin, Scott. Lauria, Mario. Chien, Andrew. High performance messaging on workstations: Illinois fast messages (FM) for myrinet Proceedings of the ACM/IEEE Supercomputing Conference. v 2 1995. IEEE, Los Alamitos, CA, USA,95CB35990. p 1528-1557.

[5] Gregory Henley, Nathan Doss, Thomas McMahon, Anthony Skjellum. BDM: A Multiprotocol Myrinet Control Program and Host Application Programmer Interface. Technical Report # MSSU-EIRS-ERC-97-3 in progress, Mississippi State University, 1997.

[6] Gregory Henley, Nathan Doss, Anthony Skjellum. BDT: A Thread Library for the Myricom LANai4.x Communications Processor. Technical Report # MSSU-EIRS-ERC-97-2, Mississippi State University, 1997.

[7] Scott Pakin, Vijay Karamcheti, Andrew A. Chien. Fast Message (FM): Efficient, Portable Communication for Workstation Clusters and Massively-Parallel Processors. IEEE Concurrency, 1997

[8] R. Martin, A. Vahdat, D. Culler, T. Anderson. Effects of Communication Latency, Overhead, and Bandwidth in a Cluster Architecture. International Symposium on Computer Architecture , Denver, CO. June 1997

[9] Gilderman I. Barak A., Profiling the Communication Layers Performance of the Myrinet Gigabit LAN, Institute of Computer Science, The Hebrew University, August 1997.

[10] D. Cohen, G. Finn, R. Felderman and A. DeSchon. ATOMIC: A High Speed Local Communication Architecture. JHSN Contents of Volume 3 (1994).

[11] Hsiao-keng Jerry Chu, SunSoft Inc. Zero-Copy TCP in Solaris. Proceeding of the USENIX 1996 Annual Technical Conference, San Diego, California, Tanuary 1996.

[12] Steven H. Rodrigues, Thomas E. Anderson, David E. Culler. High-Performance Local Area Communication With Fast Sockets. USENIX '97.

[13] Andrew Chien, Scott Pakin, Mario Lauria, Matt Buchanan, Kay Hane, Louis Giannini. High Performance Virtual Machines(HPVM): Clusters with Supercomputing APIs and Performance. Eighth SIAM Conference on Parallel Processing for Scientific Computing (PP97); March, 1997

[14] Myricom. Myrinet A Brief, Technical Overview..

[15] Myricom. Myrinet Link Specification.

[16] Myricom. LANai 4. DRAFT December 1,1997.

[17] Myricom. Myrinet User's Guide.

[18] Myricom. Myrinet Performance Measurements.

[19] Myricom. LANai3.1/4.0 Instruction Set.

[20] Cezary Dubnicki, Angelos Bilas,Kai Li and James Philbin. Design and Implementation of Virtual Memory-Mapped Communication on Myrinet. Proceedings of the 11th International Parallel Processing Symposium, April 1997.

[21] Cezary Dubnicki, Angelos Bilas, Yuqun Chen, Stefanos N. Damianakis and Kai Li. Shrimp Project Update: Myrinet Communication. IEEE Micro, Jan/Feb 1998, pp.50-52.

[22] Brent N. Chun, Alan M. Mainwaring, and David E. Culler. Virtual Network Transport Protocols for Myrinet. IEEE Micro, Jan/Feb 1998, pp.53-63.

[23] Steve Sears, et al., "Kernel to Kernel Transport Interface for the Mach Kernel", OSF RI Operating Systems Collected Papers Vol. 3, April 1994.

[24] 蔡源斌. "叢集電腦系統架構的發展趨勢", 電通所訊.