HP N4000 – neuer Compute-Server im RZ

Seit Ende des vergangenen Jahres steht mit dem HP-Server N4000 [1] ein neuer Rechner für den Compute-Service am Rechenzentrum zur Verfügung. Der HP-Server ist ein SMP-Rechner (Symmetric Multiprocessing), der sich in die Entwicklungslinie der HP X-Klasse und V-Klasse als Rechner der N-Klasse [6] einreiht.
Er löst die Convex Exemplar SPP 1200 ab. Wie diese ist der neue Rechner mit 8 PA-RISC Prozessoren ausgerüstet, allerdings mit den neuesten (PA-RISC 8500). Durch den großen Hauptspeicher und die Technik des *Symmetric Multiprocessing* ist er sowohl für skalare als auch niedrig parallele Anwendungen geeignet. Durch die von HP garantierte *High Availability* ist das Rechnersystem sehr ausfallsicher, was besonders bei langen Rechnungen von Vorteil ist.

Hard- und Software

Prozessoren	8 PA-RISC 8500
Hauptspeicher	16 GB
Data Caches	512 KB Onchip Instruction Cache 1 MB Onchip Data Cache
Plattenspeicher	208 GB Diskarray
Taktfrequenz	440 MHz
Peak Performance	8x1,76 Gflops
Netzkomponenten	Fast und Gigabit Ethernet, ATM
externe Geräte	4 mm DAT-Laufwerk, 6x4 GB CD ROM Laufwerk

Tabelle 1: Hardware der N4000

Betriebssystem	HP-UX 11.0
Compiler	Fortran 90/95, C, C++
Parallelisierung	HP MPI (MPI 1.2 Standard)
Bibliotheken	HP MLIB, NAG Libraries
Netzsoftware	NFS, TCP/IP, Netscape
Batch	LSF (Load Sharing Facility)
Debugger	dde, gdb, mpidebug
Performance	gpm (Glance Plus Manager), cxperf
Java	Java Developement Kit
Online Help	Dynatext Beschreibung
Nutzersoftware	Chemie: Gaussian 98

Tabelle 2: Software der N4000

Batch-Service

Wie auf allen Compute-Servern des Rechenzentrums steht auch auf der N4000 ein komfortables Batch-System zur Verfügung. *LSF* (Load Sharing Facility) ermöglicht eine optimale Lastverteilung in einer heterogenen Umgebung (UNIX- und NT-Systeme). Es wird von der Firma *Platform Computing Corp.* in Kanada vertrieben. In Deutschland wird die Software durch die Firma *s+c* (Science and Computing) bereitgestellt. LSF unterstützt auch parallele Anwendungen und erlaubt damit ein gleichzeitiges Rechnen von skalaren und parallelen Jobs auf der N4000.

Die Installation auf der N4000 hat eine lokale Funktion. Die von LSF benötigten Dienste: *LIM* (Load Information Manager), *PIM* (Process Information Manager) und *RES* (Remote Execution Server) werden beim Start des Betriebssystems aktiviert. Der *LIM* startet dann den Master Batch Daemon und den Slave Batch Daemon.

Auf der N4000 sind Queues mit unterschiedlichen Ressourcen und Limits installiert. Jobs, die nicht sofort zur Abarbeitung kommen, weil das Job-Limit für die Queue oder das Job-Limit des Nutzers bereits erreicht sind, werden in eine *Pending-Liste* gestellt und warten darauf, dass die entsprechende Queue wieder frei wird. Ein ausgewogenes Job-Scheduling wird durch die Vergabe von „Shares“ für jeden Nutzer ermöglicht. Berechtigte Nutzer können skalare und parallele Jobs an das Batchsystems abschicken. Jobs können interaktiv, über ein Skript oder mit Hilfe des Graphic User Interfaces (GUI) *xbsub* abgeschickt werden. Für die Überwachung der laufenden Jobs steht mit *xlsbatch* ein weiteres grafisches Werkzeug zur Verfügung.

Mitarbeiter des gleichen Instituts werden zu einer Gruppe zusammengefasst, so dass auch eine ausgewogene Jobverteilung unter den Instituten möglich ist. Weitere Informationen sind unter [2] zu finden.

Speichern von Daten

Den Nutzern der N4000 wird, wie an allen anderen Rechnern auch, das Homeverzeichnis zur Verfügung gestellt. Für das Speichern von temporären Daten werden die Dateisysteme */scratch* für Dateien kleiner als 2 Gigabyte sowie */scratch2* für Dateien größer als 2 Gigabyte bereitgestellt. In den genannten Dateisystemen wird ein Unterverzeichnis mit dem Nutzerkennzeichen als Verzeichnisnamen angelegt, in dem der Nutzer dann seine temporären Daten ablegen kann. Die Dateisysteme mit den temporären Daten werden vom Betriebssystem überwacht, d. h. wenn eine Hochwassermarke (75 %) erreicht ist, werden Daten entsprechend ihrem Alter entfernt. Daten, die länger aufgehoben werden sollen und im Homeverzeichnis keinen Platz finden, können mit Hilfe von *ADSM* (Adstar Distributed Storage Manager) hierarchisch permanent gespeichert werden. Das externe DAT-Laufwerk kann ebenfalls zum Einlesen und Speichern von Daten verwendet werden. Es unterstützt 4-mm-DAT-Bänder mit einer Kapazität von bis zu 24 GB.

Programme

Auf der N4000 können sowohl Programme gerechnet werden, die die schnellen PA-RISC Prozessoren für eine skalare Leistung nutzen, als auch parallele Programme, die auf bis zu 8 Prozessoren laufen. Für die Programmentwicklung werden die notwendigen Compiler, Debugger und Bibliotheken (Tabelle 2) zur Verfügung gestellt. Es gibt verschiedene Möglichkeiten, Programme zu parallelisieren. Durch die Angabe der Optionen *+O3* und *+parallel* beim Compileraufruf wird der Quelltext automatisch parallelisiert. Dieses Vorgehen ist aber nur bei etwa einem Viertel aller Programme sinnvoll, bei allen anderen Programmen ist das explizite Parallelisieren, d. h. Einfügen von Compiler Direktiven in das Programm, Anwendung von Pthreads oder Nutzung des *MPI* (Message Passing Interface) vorzuziehen [3], [5]. Der Anwender entscheidet hier selbst, an welchen Stellen es sinnvoll ist, das Programm zu parallelisieren. Die in Tabelle 2 aufgeführten Debugger sollen bei der Programmentwicklung hilfreich sein. Mit dem „Performance Analyzer“ *cxperf* können Programme, die in *C*, *C++* oder *Fortran 90* geschrieben wurden, analysiert werden. Dieses Werkzeug ist auch für *MPI*-Programme nutzbar.

Für Nutzer der Chemie wird das Programm *Gaussian 98* [4] zur Verfügung gestellt. Es ist möglich, *Gaussian 98* skalar oder parallel zu rechnen. Für einen parallelen Lauf ist beispielsweise die Zeile "%Nproc=4" in die Eingabedatei **.com* einzufügen, wenn die Rechnung auf 4 Prozessoren ausgeführt werden soll. Skalare Jobs werden in die Queue *g98* und parallele in die Queue *mpi* gestellt. Testrechnungen haben aber gezeigt, dass das *Gaussian 98*-Programm nicht sehr gut skaliert. Bei 4 CPUs benötigt eine Berechnung mit der MP2-Methode 52 %, mit der DFT-Methode 37 % der Zeit einer CPU. Das liegt daran, dass Gaussian schon Ende der 80-er Jahre entwickelt wurde und der Code nach und nach weiter optimiert wurde.

Literatur:

1. http://www.hu-berlin.de/rz/compsv/n4000_all.html
2. http://www.hu-berlin.de/rz/compsv/lsf.html
3. http://www.hu-berlin.de/rz/compsv/n4000.htm
4. http://www.hu-berlin.de/rz/softw/g98.html
5. http://www.hp.com/rsn/mpi/mpihome.html
6. http://www.unixsolutions.hp.com/products/servers/nclass/index.html

Daniela-Maria Pusinelli
pusinelli@rz.hu-berlin.de