Erfahrungen mit dem Convex/HP-Metacluster

Peter Junglas

14. September 1993

Inhaltsverzeichnis

1 Hardware-Konfiguration

Stand August 1993:

- Convex C3840 (480 MFLOPS Peak)
  1GB Hauptspeicher
  33 GB Platten
  4x DAT
- 6x HP9000/730 (6x 23.7 MFLOPS Linpack 100x100)
  2x 64 MB, 4x 32 MB
  1x 2.8 GB, 5x 0.4 GB
  1 CD-ROM
- Verbinder: Ethernet

ab Oktober:

- 6x HP9000/735 (6x 40.8 MFLOPS Linpack 100x100)
- Verbinder: Shared Memory Interface von Convex

2 Convex Cluster System Administration Tools V1.0

2.1 Cluster-Konfiguration

- physikalischer Cluster: Convex(en) und HPs in einem gemeinsamen Netz
- logischer Cluster: Zusammenschluß mehrerer HPs oder HPs und Convexen zur gemeinsamen Verwaltung oder Adressierung mit Cluster-Kommandos
  - kann mehrere physikalische Cluster umfassen
  - ein Host kann zu mehreren logischen Clustern gehören
  - vom Administrator vorgegeben, weitere beliebig über CLUSTER-Environment-Variable
- Cluster hat einen Master: die Convex (falls vorhanden)

- routet zu den HP's
- bootet die HP's
- ist NFS-Server für die User-Homebereiche
- ist YP-Server für passwd und groups

2.2 Beispielkonfiguration der TUHH

- phys. Cluster: C3 und alle HP's
- logische Cluster:

all:	convex, anton, bert, conny, det, edi, fritz
mmall:	anton, bert, conny, det, edi, fritz
mmuser:	conny, det, edi, fritz
mmrz:	anton, bert

- Convex nur eingeschränkt Master
  - Booten: eigene HP-Konsole
  - NFS: kleine Homebereiche auf den HP's
  - YP: getrennte Benutzer auf Convex und HP's

2.3 Cluster-Kommandos

- generell:

  wirken auf Cluster, der definiert wird durch:
  Kommandozeile (Option -C)
  CLUSTER-Environment-Variable
  cluster database (EVERYTHING)

- Cluster-Verwaltung:

  cladmin: Einrichten von phys. oder logischen Clustern

  clconnect: Verbindung zum Cluster über serielle Leitung

  clupd: Dämon, um Load-Infos zu sammeln (verbraucht 2-3% einer C38-CPU! Tip: $DELAY von 15 auf 60)

- User-Kommandos:

  clinfo: Informationen über die definierten Cluster

clanysh: rsh auf Host mit niedrigster Load (modulo fuzz-Faktor)

clcp: rcp im Cluster, vielseitig durch %-Expansion (%h, %c, time)

clps: ps mit Hostnamen, für user, PID, TTY, Command, Regexp

clkill: Prozeß-Auswahl wie ps, interaktive Abfrage möglich

clsh: rsh auf Cluster, Kommandos nacheinander

cluptime: wie ruptime für den Cluster

- weitere Kommandos (zwecks Einheitlichkeit der Umgebung):

  tcsh, RCS, perl, less, ..

2.4 Beispiele für Anwendungen

- PVM- oder EXPRESS-Jobs aller Benutzer auf den User-Maschinen anzeigen

  ```
  clps -C mmuser -a -r 'pvm|ex'
  ```

- alle eigenen pvm-Dämonen mit Gewalt abschießen (mit Abfrage)

  ```
  clkill -9 -i -c pvmd3
  clsh /bin/rm '/tmp/pvm*.6909'
  ```

- versch. Versionen von /etc/syslog.conf pro Maschine updaten

  ```
  clcp %h:/etc/syslog.conf syslog.conf.%h
  emacs syslog.conf.* &
  clcp syslog.conf.%h %h:/etc/syslog.conf
  ```

3 ConvexMLIB V1.0

- zwei Bibliotheken:
  - veclib: Implementation der CONVEX-Veclib auf HP's (ohne sparse matrix routines)
  - lapack: Nachfolger von LINPACK und EISPACK für Vektorrechner
- Performance-Vergleiche: handgecoded, NAG, MLIB
  - lin. Gleichungssystem mit LU-Zerlegung:

handgecoded:	nach "Numerical Recipes"
NAG:	F01BTF, F04AYF
MLIB:	DGEFA, DGESL

  - Matrix-Multiplikation:

handgecoded:	trivial
NAG:	F06YAF (entspr. DGEMM)
MLIB:	DGEMM

- Wunsch: parallele Version (auf allen HPs oder sogar auch Convex !)

4 ConvexPVM V1.0

- Implementation von PVM2.4.2, keine Ergänzungen wie Xab oder Hence
- Vergleich der Kommunikations-Geschwindigkeit
  - PVM-Demoprogramme timing und timing_slave
    (leicht verändert)
  - Fazit:

    ConvexPVM – wie PVM2.4 – etwas schneller als PVM3.1

    Zeiten für SMI ?
- Convex-Erweiterungen
  - Debugging-Möglichkeit
  - zusätzliche Funktionen in libpvm.a
  - Unterstützung von Convex-native-FP-Format
  - PVM–Online–Tutorial mit X–Oberfläche

4.1 Debugging mit ConvexPVM

- Vorbereitungen:
  - Für Debugging übersetzen
  - .pvmdbinit-File mit Pfaden für hostfile, pvmd,
    Debugger anlegen
  - db=on -Option im Hostfile einfügen
  - pvmdb EXECFILE
- Arbeitsweise:
  - unterstützt Debugger (cxdb, adb, gdb, xdb) im Line-Mode

  - pro PVM-Prozeß eine "Session"
  - pvmdb startet Master-pvmd und geht in Mastersession
  - normale (Line-Mode-)Debug-Kommandos
  - zusätzlich Kommandos:
    * zwischen Sessions umschalten
    * Sessions listen
    * pvmd's restarten
- ist fehlerhaft (konnte es nicht starten) !

4.2 Debugging-Support in PVM3.1

- Bei pvm_spawn Parameter PvmTaskDebug einfügen
- Bei Ausführung wird ein Skript (`debugger`) ausgeführt, das ein xterm-Fenster mit Debugger startet
- Vorteil: flexibler
- Nachteil: etwas unübersichtlich

4.3 Weitere Convex-Erweiterungen

- zusätzliche Funktionen in libpvm.a:
  - rcvn, rcvnmulti, probenmulti
    (rcv,.. aber für spezielle (name,inum))
    vrcvn, vrcvnmulti, vprobenmulti
    (ebenso, aber für vrcv, ..)
    vprobe, vprobemulti
    (probe,.. für TCP)
  - alles überflüssig mit PVM3.1, denn:
    pvm_recv hat Tid und Tag als Parameter
    kein eigener Message-Space für TCP, sondern pvm_advise
- Unterstützung von Convex-native-FP-Format:
  - zwei Architektur-Klassen: C2MP und C2MPCXFP
  - zwei Versionen für Programme und Bibliotheken
  - bei PVM3.1 gibt es CNVX und CNVXN (native FP)

- PVM Online-Tutorial mit X-Oberfläche:
  X11-Browser (wie etwa bei CXdb) mit
  Übersicht über PVM
  "Hands-on"-Beispiel

5 ConvexNQS+ V1.0

Erweiterungen von ConvexNQS+ gegenüber ConvexNQS:

- Demandqueues
- initiator
  anderer NQS-Scheduler (Wartezeit und Share)
  kein Manual, nur Bemerkung in den Release Notes
- qmapmgr -m (für snapshot)
- auf HPs
  qsub, qdel, qstat, qps, qlimit
  qmgr, qsnapshot, qmapmgr, qsa

Load-Balancing bei ConvexNQS+ :

- Load-Average-Queues:
  - load-factor =
    (load-average + weight*queue-length)/cpu-speed
  - Pipedämon berechnet load-factor für jede Maschine
  - Job wird sofort an die Maschine mit niedrigstem load-factor weitergereicht
- Demand-Queues:
  - haben nur Jobs RUNNING, nicht QUEUED
  - Pipequeue hält und verteilt Jobs zyklisch auf freie Demand-Queues
  - Nachteil: Load (interaktiv, andere Queues) geht nicht ein
  - Vorteil: reagiert auf augenblicklichen Zustand

5.1 Einschränkungen auf HPs:

- kein qwatch
- nur folgende Limits:
  - nice value
  - per-process permanent file size
- keine Activity-IDs beim Batch-Accounting
- Checkpoint/Restart nur CX-Queues (aber von HPs aus)

6 DQS - ein Batchsystem für Workstation-Cluster

- frei verfügbar von ftp.scri.fsu.edu
  Makefiles für HP-UX und ConvexOS vorhanden
- nur zwei Dämonen: 1x qmaster, Nx dqs_execd
  - qmaster: Spooling, Scheduling und Dispatching der Jobs
  - dqs_execd: Ausführen der Jobs (auch von mehreren Queues pro Maschine)
- Queue-Gruppen
- zwei Lastverteilungsverfahren (Auswahl zur Compilezeit!):
  - demand-queues ähnlich zu ConvexNQS+
  - demand-queues mit Load-Balancing:
    * dqs_execd's schicken regelmäßig Load
    * qmaster wählt freie Queue nach Load und Maschinen-Gewichtsfaktor

6.1 DQS - weitere Eigenschaften

- Hilfsprogramme:
  - qsub, qdel, qstat, qsh, (m)qmon
  - qconf, qmod, qusage, qacct
  - qidle

- Stoppen und Restarten von Queues

  qmod: stoppt und restartet incl. laufendem Job oder nach Beendigung eines laufenden Jobs

  qidle: Queue wird automatisch angehalten bei Aktivitäten eines X-Servers

  untergeordnete Queues: werden automatisch angehalten, wenn die übergeordnete läuft

- Unterstützung von Parallel-Jobs:
  - "hardünd ßoftAnforderungen
  - explizit mit Skripten und JOBNAME.hosts-File
  - implizit mit Option -pvm3

- Probleme:
  - keine Zusammenarbeit mit NQS (geplant)
  - kein Checkpoint/Restart

7 Fazit

- System Administration Tools V1.0
  - \+ hilfreiche kleine Werkzeuge
  - \+ einheitlichere Umgebung auf Convex und HPs
  - − clupd verbraucht zuviel CPU

- ConvexMLIB V1.0
  - ++ sehr schnell
  - \+ Portabilität zwischen Convex und HP-Cluster
  - − keine Parallelisierung

- ConvexPVM V1.0
  - \+ Online-Tutorial
  - − veraltete Version
  - − keine wesentlichen Ergänzungen

- ConvexNQS+ V1.0
  - ungenügende Load-Balancing-Verfahren
  - wichtige Möglichkeiten für Cluster-Batchsystem fehlen (vgl. mit DQS)
  - fehlerhaftes Installations-Programm