

Ausblick: CODINE

CODINE (COmputing in DIstributed Network Environment) beruht auf einer Client/Server-Struktur. Als Server dient ein Masterrechner (Informations-Server), der über den aktuellen Zustand und die Rechenlast jedes einzelnen CPU-Servers informiert. Die CPU-Server arbeiten unabhängig vom Master-Server, aber konsultieren ihn nach getaner Arbeit. Es ist auch möglich, den Masterprozeß von einer Maschine auf eine andere zu migrieren. Die Vorteile von Client/Server-Modellen liegen im Preis und in der Erweiterbarkeit. CODINE berücksichtigt die Bedingungen eines heterogenen Netzwerkes und versucht, die unterschiedliche Hardware optimal einzusetzen.

Was leistet CODINE?

Folgende Eigenschaften werden von CODINE unterstützt:

- Batch-, interaktive und parallele Jobs
- nutzergesteuertes und transparentes (Programme ohne exec, fork und sockets) Checkpointing
- ausgewogene Lastverteilung
- Ausführung von parallelen Jobs unter PVM, EX-PRESS, p4 und Linda
- NQS Interface
- Accounting und Statistik
- DCE Technologie

Die CONDINE-Queues werden architekturbezogen und nach Gruppen sortiert. Die Queues können hierarchisch organisiert werden. Requests können zu einer speziellen Architektur, zu einer Gruppe oder zu einer speziellen Queue geschickt werden. CODINE

wählt in den ersten beiden Fällen die geeignetste Queue im Netzwerk aus.

Als neue Kommandos kommen **qmod** und **qidle** hinzu, die eine Queue auf einer Plattform suspendieren, um diese dem Besitzer wieder für seine interaktive Arbeit zur Verfügung zu stellen. Die Requests der Queue werden in diesem Falle migriert. Das Kommando **qsh** eröffnet auf einem Host mit "leichter" Last eine interaktive Shell. **qconf** stellt das Nutzerinterface zum Arbeiten mit den Queues dar.

Eine mögliche Konfiguration von CODINE am Rechenzentrum der Humboldt-Universität könnte so aussehen, wie in Abbildung 2 dargestellt. Für parallele Jobs wird ein geeigneter Sub-Cluster von CODINE ausgewählt, nachdem der Nutzer die geforderte Anzahl von Queues über einen allgemeinen Request mitgeteilt hat. Hier ist der Begriff Queue mit Knoten gleichzusetzen.

CODINE bietet 3 verschiedene Nutzerinterfaces. Für die tägliche Routinearbeit ist das Einbetten der Kommandos in Job-Scripts sinnvoll. Das Kommandozeilen-Interface fügt sich den POSIX Anforderungen. Ein OSF/Motif Interface bietet die Möglichkeit, interaktiv auf alle Kommandos zuzugreifen. Ein graphisches Display, das Auskunft über die Nutzung der Cluster gibt, ist verfügbar.

Die HPCN-Kommission (High Performance Computing and Networking) der EG wird in den kommenden 2 Jahren die Weiterentwicklung von CODINE fördern, wobei diese in Richtung Metacomputing gehen soll. Die Perspektive für CODINE ist gegeben!

Daniela-Maria Subklew

Neue Möglichkeiten mit UniTree+ V 1.7.5

Dieser Beitrag setzt die Kenntnis des im Heft Nr. 6 der RZ-Mitteilungen erschienenen Artikels "UniTree an der Humboldt-Universität" voraus.

UniTree+ ist die konsequente Weiterentwicklung von UniTree, das am Rechenzentrum bisher in der Version 1.5.2 seine Anwender gefunden hat. In UniTree+ sind viele Forderungen, die aus der UniTree User Group (UTUG) und von anderen UniTree Nutzern kamen, eingeflossen. Dies betrifft die Nutzerakzeptanz, die Performance und auch die Möglichkeiten der Systemverwaltung. UniTree+ liegt zur Zeit in der Version 1.7.5 vor. Im folgenden werden die Neuerungen und ihre Auswirkung auf die Arbeit mit UniTree+ am RZ dargestellt.

Manuals zu UniTree+ sind unter `/usr/unitree/adm/man` zu finden. Also: `setenv MANPATH $MANPATH:/usr/unitree/adm/man`

- Neue Organisation des Diskcache:

Der Diskcache besteht jetzt aus regulären Filesystemen des ConvexOS, die allerdings von UniTree+ verwaltet werden. Gemounted sind sie auf `/cache/0[1-3]/file`. Der gemeinsame "mount-point" für dieses Filesystem ist auf der C3820 ES `/unitree/data`. Mit dem normalen `df`-Kommando kann man sich die Struktur der Filesysteme anzeigen lassen. Durch die Umorganisation und eine Verbesserung der Algorithmen des Nameservers ist die Performance des UniTree+-NFS etwa so wie die des native UNIX. Aus diesem Grund wird der NFS-Zugang zum Diskcache auf der C3820 ES für alle freigegeben. Es wird allerdings beobachtet, ob es durch die NFS-Nutzung anderweitige Beeinträchtigungen des Betriebs gibt.

Der Nutzer h9999xyz der Gruppe p9999 hat also durch

```
cd /unitree/data/home/p9999/h9999xyz
```

Zugang zu seinen Home-Verzeichnis im Diskcache.

Ob ein File sich im Diskcache befindet oder nur auf einem VHS-Band, wird beim ls-Kommando durch das sticky bit angezeigt.

```
-rw-rw-r-t 1 h9999xyz p9999 4711
                          Nov 13 20:37 foo1
```

zeigt an, daß das File foo1 sich nur auf einem VHS-Band befindet und bei Ansprechen erst eingelagert werden muß.

```
-rw-rw-r-- 1 h9999xyz p9999 4711
                          Nov 13 20:37 foo2
```

zeigt an, daß das File foo2 sich sowohl im Diskcache als auch auf einem VHS-Band befindet.

Wenn mit NFS gearbeitet wird, sollten vorher die Files im Diskcache sein - siehe unten FTP.

- Familykonzept:

Files eines Nutzers bzw. einer Gruppe von Nutzern können sogenannten Familien zugeordnet werden. Files einer Familie werden auf gemeinsamen Datenträgern gespeichert. Dies ist von Vorteil bei Anwendungen, die sehr viele Files gleichzeitig benötigen, da sie beim Einlagern (Caching) nur auf einem VHS-Band gesucht werden müssen. Um Nutzer einer Familie zu werden, muß man in ihr eingetragen werden. Einzelne Nutzer können in mehreren Familien eingetragen sein. Bänder, die einer Familie zugeordnet sind, können nur von den in ihr eingetragenen Nutzern Files enthalten. Da wir mit dem RSS-48 zwar ein Speichermedium für knapp ein 1 TByte besitzen, diese Datenmenge aber auf relativ wenigen Datenträgern verwaltet wird, kann das Familykonzept vorerst nicht zur allgemeinen Nutzung, sondern nur für ausgewählte technologische Aufgaben eingesetzt werden.

- Führen von Duplikaten ausgewählter Files:

Bisher wurden die Files bei der Migration sequentiell auf das aktuelle Migrationsband geschrieben. Mit UniTree+ hat man die Möglichkeit, Duplikate des migrierten Files auf weiteren getrennten Bändern zu fordern (bei uns max. ein Duplikat). Dies bietet eine erhöhte Datensicherheit für wichtige Files, da die Files auf den Migrationsbändern nur einmal vorhanden sind. Das Anlegen von Duplikaten kann nur über den FTP-Zugang erreicht werden (vgl. unten: Erweiterungen des ftp-Kommandos)

- rcp-Zugang:

Neben der Möglichkeit über FTP und NFS Zugang zu UniTree+ zu erhalten, kann jetzt auch rcp benutzt werden. Das UniTree+-spezifische Format des rcp-Kommandos sieht folgendermaßen aus:

```
1.) rcp file1 /utrcp:[[family_id]:[dupl_num]:]file1
                                     # Nach UniTree
```

```
2.) rcp /utrcp:file2 file2           # Aus UniTree
```

```
z.B.: cd; rcp testfile /utrcp:fake/olga
                                     # testfile nach fake/olga unter UniTree
```

Dabei ist /utrcp der UniTree+-Prefix; er muß, so wie oben gezeigt, angegeben werden.

family_id und dupl_num sind die Identifikation einer Familie (bei uns z. Z. für den allgemeinen Gebrauch nicht relevant) und die Anzahl der gewünschten Duplikate.

Die genaue Beschreibung des rcp-Kommandos ist den Manuals zu entnehmen. (**man rcp** für rcp allgemeine Syntax; **man 1 rcp** für UniTree spezifische Syntax)

- Erweiterung des ftp-Kommandos:

Im ftp-Kommando wurden weitere Features eingefügt bzw. vorhandene erweitert.

Durch die Einführung des Familykonzepts wurde die Ausgabe des dir-Kommandos um die Spalte family erweitert. Bei uns wird dort für den allg. Gebrauch nur "common" stehen (Weitere Änderungen: "DK" steht an Stelle von "disk" und "AR" am Stelle von "archive").

Die Erweiterungen werden durch die Benutzung der ftp-Kommandos **quote** und **quote site** ermöglicht.

- **quote site GTRSH** - Anzeige der Minuten, innerhalb der aus UniTree+ gelöschte Files im Verzeichnis ".trash" aufgehoben werden

- **quote site NMDUP [n]** - Anzeigen und Setzen der Dateiduplikationen die in UniTree+ gespeichert werden sollen. Ohne Angabe von "n" wird der gesetzte Wert angezeigt. Der Maximalwert ist bei uns 2. Nachdem dieses Kommando in einer FTP-Sitzung angewendet wurde, wird für jedes File, das mit **put** oder **mput** kopiert wird, die entsprechende Anzahl von Duplikaten auf den VHS-Bändern angelegt. **NMDUP** bewirkt nur eine temporäre Setzung. Nach Verlassen der FTP-Sitzung wird die Setzung wieder aufgehoben und es gilt wieder der Standardwert (1).

- **quote site SETFAM family_name** - Angabe der Familie, zu der die Files gehören sollen. Für die Gültigkeit der Setzung gilt das gleiche wie unter **NMDUP**.

- **quote site STRSH time** - Setzen der Zeit in Minuten, in der gelöschte Files im Verzeichnis

".trash" aufgehoben werden. Standard ist bei uns 30 Minuten. Dieser Wert gilt auch über die FTP-Sitzung hinaus. Bei der ersten Benutzung sollte dieser Wert gesetzt werden.

- **quote chgrp** group_name filename - Ändert den Gruppennamen eines Files. Dazu muß man der Eigentümer des Files und Mitglied der neuen Gruppe oder Superuser sein.
- **quote ln** file1 file2 - Anlegen eines symbolischen Links innerhalb von UniTree+.
- **quote mstage** file[s]_mit_wild_cards - Dieses Kommando verbindet die Funktionalität des

native ftp-Subkommandos **mget** mit der UniTree-Erweiterung **stage** ohne den Operanden waittime (vgl. RZ-Mitteilungen Nr.6).

- **quote umask** maske - Setzen der Maske der Zugriffsrechte für neue Files in UniTree+.

FTP mit seinen Ergänzungen kann den Manuals **man ftp** für ftp allgemein und **man 1 ftp** für UniTree-Spezifika entnommen werden.

Weitere Neuerungen in UniTree+ betreffen die Reorganisation des VHS-Archives, die Möglichkeit, Files, die über einen längeren Zeitraum nicht angefaßt wurden, auf offline-Medien auszulagern, sowie eine günstigere Verwaltung des gesamten System.

Christoph Weickmann
weickmann@rz.hu-berlin.de