



Collaborative Climate Community Data and Processing Grid (C3-Grid)

Richtlinien für die Konfiguration des Workspaces

Arbeitspaket:	AP5 und AP7
Autor:	T.Langhammer (ZIB), V.Achter, V.Winkelmann (ZAIK)
Version:	Jan. 2007
Veröffentlichungsdatum:	Jan. 2007
AP-Koordination:	Konrad-Zuse-Zentrum (ZIB)
Partner:	Universität zu Köln (ZAIK)
Ansprechpartner:	T. Langhammer, V. Winkelmann
Email:	langhammer@zib.de, winkelmann@uni-koeln.de

Inhalt

1 Definition

2 Grundlegende Forderungen

3 Konventionen

- Einrichtung von individuellen C3-User-Accounts sowie einer zentralen C3-Grid-Unix-Gruppe (Konvention zwischen Scheduling und Provider)
- Einrichtung eines speziellen DMS-Accounts (Konvention zwischen DMS und Provider)
- Workspace-Basis und Umgebungsvariable `C3GRID_ROOT` (Konvention zwischen Scheduling und Provider)
- DMS-Verzeichnis `§C3GRID_ROOT/dms_workspace` (Konvention zwischen DMS, Scheduling und Provider)
- Ablage von Ein- und Ausgabefiles in `§C3GRID_ROOT/dms_workspace` (Konvention zwischen DMS und Scheduling)
- Publish-Verzeichnis `§C3GRID_ROOT/dms_workspace/htdocs` (Konvention zwischen DMS und Provider)
- Zugriff per GridFTP (Konvention zwischen DMS und Provider)
- Zwischenprodukte der Tools, temporäre Files (Konvention zwischen Provider und Toolentwickler)
- Überblick

1 Definition

Ein Grid-Workspace ist eine Speicherressource, die jeder Daten- und Compute-Provider im C3-Grid der Community zur Verfügung stellt. Dieser Datenbereich wird als temporäre Ablage für Eingabe- und Ausgabe-Files von Processings verwendet. Ebenso können einzelne Grid-Tools während der Zeit ihrer Ausführung in diesem Bereich Zwischenprodukte ablegen.

2 Grundlegende Forderungen

- Netzsicherheit
 - Nur explizite Zugangswege werden akzeptiert (C3-Grid Frontend per GridFTP & HTTP, Compute-Knoten über Filesystem), bspw. kein unkontrollierter NFS-Mount von außen.
- Informationssicherheit
 - das Datenmanagement DMS muss mit GridFTP übertragen und aufräumen können,
 - C3Grid-User müssen eigene Input-Daten lesen und Output-Daten schreiben können, aber die von anderen nicht lesen dürfen,
 - HTTPD muss C3-Grid-Ergebnisse lesen können, ein C3Grid-User darf nur an seine eigenen Ergebnisse herankommen.

3 Konventionen

Jeder Daten- und Compute-Anbieter richtet einen lokalen Datenbereich ein, der über GridFTP erreichbar ist. Um einen einheitlichen Zugriff durch DMS, WF-Scheduler und Tools zu ermöglichen, sind nachfolgende Konventionen vereinbart. Wünschenswert wäre der Einsatz von Access Control Lists, ist aber bei der heterogenen File-Server-Struktur bislang nicht umsetzbar. Für die Fragen der Sicherheit werden daher nur Unix-Accounts und das einfache Permission-Modell berücksichtigt.

Einrichtung von individuellen C3-User-Accounts sowie einer zentralen C3-Grid-Unix-Gruppe (Konvention zwischen Scheduling und Provider)

Jeder C3-Grid-User erhält einen eigenen Account. Der Account ist insbesondere Mitglied der Unix-Gruppe **c3grid**.

(Erläuterung: das st in jedem Falle auch bei Pool-Accounts, wie sie in Generation 0 verwendet werden, notwendig.)

Einrichtung eines speziellen DMS-Accounts (Konvention zwischen DMS und Provider)

Jeder Compute-Anbieter richtet einen speziellen Account **c3-adm** (o.ä.) für das DMS ein. Der Account ist ebenfalls Mitglied der Unix-Gruppe **c3grid**.

Workspace-Basis und Umgebungsvariable C3GRID_ROOT (Konvention zwischen Scheduling und Provider)

Jeder Compute-Anbieter stellt einen Workspace bereit und setzt beim Ausführen eines Jobs die Umgebungsvariable `C3GRID_ROOT`. Diese beinhaltet den Pfad zum Basis-Verzeichnis des lokalen Datenbereichs. Da C3-Workflows ihre Daten in Unterverzeichnissen ablegen und auch das DMS administrieren soll, darf das Basis-Verzeichnis für die Unix-Gruppe `c3grid` nicht vollständig gesperrt sein.

Vorschlag für die UD/GID und Permissions:

```
c3-adm/c3grid rwxr-x---
```

nicht erforderlich, aber noch restriktiver wäre:

```
c3-adm/c3grid rwx--x---
```

DMS-Verzeichnis `§C3GRID_ROOT/dms_workspace` (Konvention zwischen DMS, Scheduling und Provider)

Dieses Verzeichnis muss angelegt sein und steht dem Datenmanagement (DMS) als File-Pool zur Verfügung.

Erläuterung: Es muss garantiert sein, dass das DMS als einziger Dienst die volle Kontrolle über die Erzeugung und Entfernung von Files und Verzeichnissen hat. Das bedeutet im einzelnen:

- *Der WF-Scheduler weist zwar jeden Grid-Job an, seine Ausgabe im Verzeichnis `§C3GRID_ROOT/dms_workspace` abzulegen, nach Beendigung des Jobs muss der WF-Scheduler die Files aber beim DMS anmelden (Schnittstelle B).*
- *Das Staging von Files erfolgt nur durch Veranlassung des DMS, d.h. das Datenmanagement beauftragt einen Datenprovider, seine Files an einer vorgegebenen Stelle in `§C3GRID_ROOT/dms_workspace` bereit zu stellen (Schnittstelle D).*
- *Alle anderen schreibenden Zugriffe erfolgen ausschließlich durch das DMS.*

`§C3GRID_ROOT/dms_workspace` erhält folgende UID/GID und Permissions:

```
c3-adm/c3grid rwx-wx---
```

Ablage von Ein- und Ausgabefiles in `§C3GRID_ROOT/dms_workspace` (Konvention zwischen DMS und Scheduling)

Pro ausgeführtem Workflow existiert ein Verzeichnis

```
§C3GRID_ROOT/dms_workspace/§WID
```

wobei `§WID` einem beliebigen, möglichst nicht zu erratenden, aber eindeutigen Workflow-Identifikator entspricht. In diesem Unterverzeichnis werden alle für diese Workflowausführung benötigten Files abgelegt. Ein derartiges Unterverzeichnis erhält folgende UID/GID und Permissions:

```
anyc3usr/c3grid rwxrwx---
```

Publish-Verzeichnis `$C3GRID_ROOT/dms_workspace/htdocs` (Konvention zwischen DMS und Provider)

In diesem Verzeichnis werden Dateien für den Export ins Portal abgelegt. Das Verzeichnis muss vom Provider über HTTP zugreifbar gemacht werden. Um für eine ausreichende Sicherheit zu sorgen, müssen HTTPD und das Verzeichnis `$C3GRID_ROOT/dms_workspace/htdocs` aufeinander abgestimmt werden. Daher muss der HTTPD unter der UID/GID

```
apache/c3grid
```

gestartet werden, der Account `apache` muss gleichzeitig Mitglied der Gruppe `apache` sein.

Das Verzeichnis `$C3GRID_ROOT/dms_workspace/htdocs` erhält folgende UID/GID und Permissions:

```
c3-adm/apache rwxr-x---
```

Zugriff per GridFTP (Konvention zwischen DMS und Provider)

Das lokale Verzeichnis `$C3GRID_ROOT` muss per GridFTP zugreifbar sein. Es bietet sich an, den GridFTP-Server so zu konfigurieren, dass das Verzeichnis `$C3GRID_ROOT/dms_workspace` für den GridFTP-Client als Basisverzeichnis erscheint.

Zwischenprodukte der Tools, temporäre Files (Konvention zwischen Provider und Toolentwickler)

Es wird ein Basisverzeichnis benötigt, in dem die Tools (oder gar Workflows) Zwischenprodukte als temporäre Files ablegen können. Die Tools sind verantwortlich für das Entfernen dieser Files nach Beendigung der Ausführung. Es bieten sich zwei Lösungen hierfür an:

- Anlegen eines Scratch-Verzeichnisses unter `$C3GRID_ROOT/tool_scratch` (Vorteil: alles unter einem Dach, Nachteil: unkontrollierte Nutzung kann Workspace lahmlegen, Alternative hierzu ein SymLink auf einen unabhängigen TMP-Bereich.)
- Verwendung eines unabhängigen TMP-Bereichs, Verwendung der Variablen `$C3GRID_SCRATCH` (Vorteil: flexible Lösung, Nachteil: Variable muss dem Tool/Workflow bekannt sein, ist aber durch Tool-Setup einfach möglich.)

Überblick:

```
$C3GRID_ROOT/          drwxr-x---      c3-adm   c3grid
|
+-- dms_workspace/     drwx-wx---      c3-adm   c3grid
|   |
|   +-- wid0815/       drwxrwx---      c3usr1   c3grid
|   +-- wid0822/       drwxrwx---      c3usr2   c3grid
|   +-- wid0843/       drwxrwx---      c3usr1   c3grid
|   |   ...
|   +-- htdocs/        drwxr-x---      c3-adm   apache
|
+-- tool_scratch/      drwxrwxrwt      root     root
```

Alternativ zu tool_scratch:

```
$C3GRID_SCRATCH/      drwxrwxrwt      root     root
```