TECHNISCHE UNIVERSITÄT DRESDEN

DRESDEN concept

Ulf Markwardt, Danny Rotscher, Martin Schroschk

Workspaces - Ein Werkzeug für das Data-Life-Cycle-Management

Konzept, Implementierung und Erfahrungen aus dem mehrjährigen Einsatz im Betrieb

2024-09-24 / ZKI Herbst / Duisburg

ZIH
Center for Information Services & High Performance Computing

Agenda

- Umgebung
- Aufgabenstellung
- Konzept und Implementierung
- Erfahrungen und Learnings
- Diskussion

Disclaimer

- Werkzeug wurde von Holger Berger u.A. entwickelt
  - Keine Blumen für ZIH
- Nur Erfahrungsbericht, keine “Anlageberatung”
  - Die unsachgemäße Konfiguration kann zu Datenverlust führen
  - Für Datenverlust kann keine Haftung übernommen werden

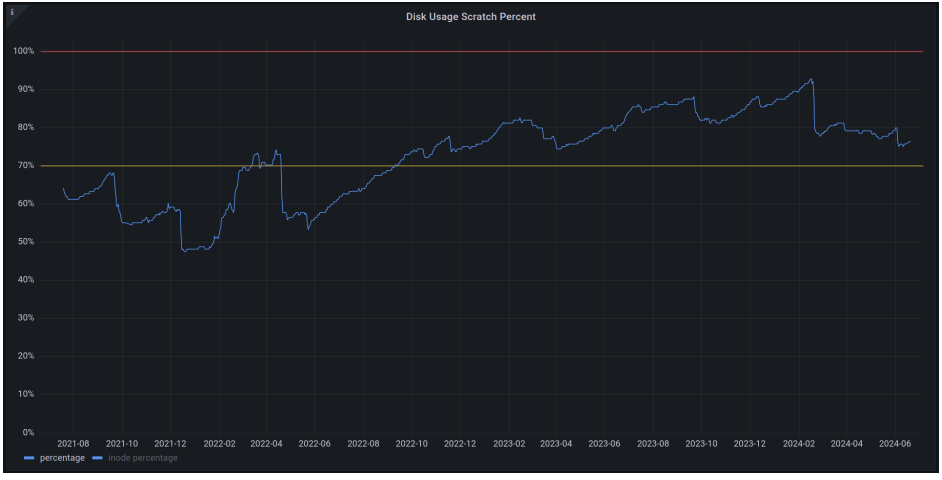
Umgebung

DLR System CARA (2019)

- 200 aktive Nutzende p.M. (~ 30 DLR-Institute)
- 2842 Compute-Knoten (AMD Naples und Rome)
- Lustre *Bulk* (HDDs)
  - 14,5 PB / Füllstand: 83%
  - `/storage/` und `/scratch`
- Lustre *Fast* (SSDs)
  - 0,5 PB / Füllstand: 76%
  - `/home` und `/scratch_fast/`

Foto: DLR (CC BY-NC-ND 3.0)

Füllstand über die Zeit



Speichersysteme vs. Nutzende

- Vielzahl von Speichersystemen
  - Mit verschiedenen Eigenschaften, z.B. Kapazität, IOPS-Rate, Streaming-Bandwidth, Verfügbarkeit, Permanenz, Preis, etc.
- Am Ende des Tages: Speichersysteme laufen immer voll
  - Nicht grundlegend zu klein konzipiert
  - Vielmehr, weil Nutzende nicht aufräumen (müssen)
- Data-Life-Cycle-Management (vgl. Forschungsdatenmanagement)
  - Wo werden Daten erzeugt und gespeichert?
  - Welche Daten sind aufhebenswert? Wie lange müssen sie aufgehoben werden?

Aufgabenstellung

1. Speichersysteme müssen bei der Beschaffung ausreichend bemessen werden.
2. Speichersysteme dürfen im Betrieb nicht unkontrolliert voll laufen.

Mögliche Lösungen für 2.

1. Feste Quota pro Nutzer:in (Fairshare i.S.v. Kapazität / #Nutzende)
   - ggf. mit Submitsperre, d.h. Nutzende müssen aufräumen bevor weitergerechnet werden kann
2. Abschätzung des Bedarfs bei Rechenzeitantrag; daraus wird Quota abgeleitet (ggf. mit Submitsperre)
3. Datamanagment-Plan als Teil des Rechenzeitantrags

Mögliche Lösungen für 2.

1. Feste Quota pro Nutzer:in (Fairshare i.S.v. Kapazität / #Nutzende)
   - ggf. mit Submitsperre, d.h. Nutzende müssen aufräumen bevor weitergerechnet werden kann
2. Abschätzung des Bedarfs bei Rechenzeitantrag; daraus wird Quota abgeleitet (ggf. mit Submitsperre)
3. Datamanagment-Plan als Teil des Rechenzeitantrags
   - Hier fehlen Werkzeuge

Mögliche Lösungen für 2.

1. Feste Quota pro Nutzer:in (Fairshare i.S.v. Kapazität / #Nutzende)
   - ggf. mit Submitsperre, d.h. Nutzende müssen aufräumen bevor weitergerechnet werden kann
2. Abschätzung des Bedarfs bei Rechenzeitantrag; daraus wird Quota abgeleitet (ggf. mit Submitsperre)
3. Datamanagment-Plan als Teil des Rechenzeitantrags
   - Hier fehlen Werkzeuge

Praxis

- Nutzende arbeiten an oder über Quotagrenze
- Erhöhung der Quota für VIPs
- Bedarfe ändern sich über die Laufzeit
- Submittsperren werden nicht umgesetzt (Systemauslastung ist wichtiger)

Workspaces / Konzept

Arbeits-Filesysteme

- Keine Default-Verzeichnisse (z.B. `/scratch/<login>`)
- Arbeits-Filesysteme können nur via Workspaces genutzt werden

Workspace

- Verzeichnis mit einer Lebensdauer, dass auf Nutzendenwunsch erstellt wird

Workspaces / Konzept

Arbeits-Filesysteme

- Keine Default-Verzeichnisse (z.B. `/scratch/<login>`)
- Arbeits-Filesysteme können nur via Workspaces genutzt werden

Workspace

- Verzeichnis mit einer Lebensdauer, dass auf Nutzendenwunsch erstellt wird

Aufräummechanismus *Expirer*

- *Expirer* überwacht die Lebendsdauer von Workspaces
- Abgelaufene Workspaces werden weggeräumt und nach einer Gnadenperiode gelöscht
  - In der Gnadenperiode können Nutzende Workspaces wieder herstellen

HPC-Workspace

Implementierung

- github.com/holgerberger/hpc-workspace

Historie

- Ursprung in 2004 am HLRS
- Lösung von Problemstellungen Rund um das verteilte Filesystem der NEC SX-6 und SX-8
  - Kontrolle über die Lebensdauer von Daten; sehr alte Daten vermeiden
  - Admin-gesteuertes Load-Balancing über mehrere Filesysteme
  - Migration von Filesystem zu Filesystem ohne Nutzerinteraktion

Nutzendensicht / Kommandos und Konfiguration

5+x Befehle

- `ws_list`: Auflisten der aktiven Workspaces und verfügbaren Filesysteme
- `ws_allocate`: Anlegen eines Workspaces
- `ws_extend`: Änderung der Laufzeit eines Workspaces
- `ws_release`: Freigeben eines Workspaces
- `ws_restore`: Wiederherstellen eines Workspaces

Konfiguration

Filesystem	Time Limit	Extensions	Grace Period	Default Time
scratch	60 days	2 times	30 days	1 day
scratch_fast	30 days	2 times	30 days	1 day

Workspaces / Allocate

Command Syntax

```
$ ws_allocate [options] <workspace_name> <duration>
```

Valid characters for workspace names are only alphanumeric characters, -, ., and _.

Workspaces / Allocate

Command Syntax

```
$ ws_allocate [options] <workspace_name> <duration>
```

Valid characters for workspace names are only alphanumeric characters, `-`, `.`, and `_`.

Example

Allocate a workspace named *FancyExp* in */scratch* that will expire in 55 days:

```
$ ws_allocate -F scratch -n FancyExp -d 55
Info: creating workspace.
/scratch/ws/0/marie-FancyExp
remaining extensions  : 2
remaining time in days: 55
```

Workspaces / List

Oh, I don’t remember my workspaces.

```
$ ws_list
id: FancyExp
     workspace directory  : /scratch/ws/0/marie-FancyExp
     remaining time       : 54 days 23 hours
     creation time        : Mon Nov 30 10:04:57 2020
     expiration date      : Sun Jan 24 10:04:57 2021
     filesystem name      : scratch
     available extensions : 2
id: BestExp
    workspace directory  : /scratch/ws/0/marie-BestExp
    remaining time       : 14 days 23 hours
    ...
```

Workspaces / Extend

```
$ ws_list -t
id: FancyExp
      workspace directory  : /scratch/ws/0/marie-FancyExp
      remaining time       : 54 days 23 hours
      available extensions : 2
```

```
$ ws_extend -F scratch FancyExp 4
```

Workspaces / Extend

```
$ ws_list -t
id: FancyExp
      workspace directory  : /scratch/ws/0/marie-FancyExp
      remaining time       : 54 days 23 hours
      available extensions : 2
```

```
$ ws_extend -F scratch FancyExp 4
```

[Q] What is the new remaining time?

Workspaces / Extend

```
$ ws_list -t
id: FancyExp
      workspace directory  : /scratch/ws/0/marie-FancyExp
      remaining time       : 54 days 23 hours
      available extensions : 2
```

```
$ ws_extend -F scratch FancyExp 4
```

[Q] What is the new remaining time?

[A] It will be 3 days 23 hours and 59 minutes!

Remark: The given duration will not add to the remaining time. It will set the new remaining time from the moment the command `ws_extend` is executed.

Workspaces / Automatic Release

- Workspace is released automatically at the end of its life time
- Workspace is moved to hidden directory
  - No data is deleted and data still occupies disk space (quota!)
    - Reminder: Data life cycle
- Grace period starts (30 days)
  - Workspace can be restored using the command `ws_restore`
- After grace period
  - Workspace and its data are irretrievable deleted

Workspaces / Manual Release

- Workspaces can be released manually via

```
$ ws_release -n <workspace_name> -F <filesystem>
```

- Grace period for manually released workspaces is **only 1 day**
  - Workspace can be restored

Good Practice

- Use workspaces allocated from within batch jobs
- Delete data first, than release workspace

Workspaces / Restore

Oh wait, I need my workspace back!

- Released workspace can be restored within grace period into other, active(!) workspace
- ID is different to `ws_list`!

```
$ ws_restore -l
marie-FancyExp-1573826665
        unavailable since Thu Nov 14 11:07:03 2019
```

Workspaces / Restore

Oh wait, I need my workspace back!

- Released workspace can be restored within grace period into other, active(!) workspace
- ID is different to `ws_list`!

```
$ ws_restore -l
marie-FancyExp-1573826665
        unavailable since Thu Nov 14 11:07:03 2019
```

```
$ ws_restore -F scratch marie-FancyExp-1573826665 FancyExp2
to verify that you are human, please type 'chiechi': chiechi
you are human
Info: restore successful, database entry removed.
```

Workspaces / Various

- `ws_allocate` provides the functionality to send an email N days before expiration date, e.g.
  - `$ ws_allocate -n FancyExp -d 17 -r N -m marie@dlr.de`
- Cooperative workspaces via `ws_allocate -g [-G <groupname>] foo 12`
- `ws_send_ical` sends an calender entry via email
- Multiple options to customize output of `ws_list`
- Manage links to all your workspaces using `ws_register`

Workspaces / Use Cases

- For campaign: Allocate and release once outside of batch scripts
- *Per-job storage:* Allocate and release within batch script

```
#!/bin/bash
#SBATCH [...]
module load <MODULES>

wsname=fancy_exp_$SLURM_JOB_ID
wsdir=$(ws_allocate -F scratch $wsname 1)
# check allocation
[ -z "$wsdir" ] && echo "Error: Cannot allocate workspace $wsname" && exit 1

cd $wsdir
# Do work
# Move results
# Delete data

ws_release -F scratch $wsname   # Reduces grace period to 1 day!
```

Workspaces / Admin- und Betreibersicht

- Eine YAML-File pro Workspace

```
$ cat /scratch/ws-db/marie-FancyExp
workspace: /scratch/ws/0/marie-FancyExp
expiration: 1729265074
extensions: 10
acctcode: hpcsupport
reminder: 7
mailaddress: marie@dlr.de
```

Workspaces / Admin- und Betreibersicht II

- Eine Konfigurations-File: `/etc/ws.conf`

```
clustername: CARA                    # name to identify the system
smtphost: smtprelay.dlr.de
mail_from: root@cara.dlr.de
dbgid: 4                             # goup id, owner of workspace top-level directories
dbuid: 11                            # user id, owner of workspace top-level directories
admins: [root]                       # list of admins for ws_list
[...]                                # default life time settings
default: scratch
workspaces:
```

Workspaces / Admin- und Betreibersicht II

- Eine Konfigurations-File: `/etc/ws.conf`

```
clustername: CARA                   # name to identify the system
smtphost: smtprelay.dlr.de
mail_from: root@cara.dlr.de
dbgid: 4                            # goup id, owner of workspace top-level directories
dbuid: 11                           # user id, owner of workspace top-level directories
admins: [root]                      # list of admins for ws_list
[...]                               # default life time settings
default: scratch
workspaces:
  scratch:                          # name of workspace location
    database: /scratch/ws-db        # DB directory
    deleted: .removed               # subdirectory for expired workspaces
    [...]                           # life time settings
    spaces: [/scratch/ws/]
  scratch_fast:
    database: /scratch_fast/ws-db
    deleted: .removed
    [...]
```

Workspaces / Admin- und Betreibersicht III

- Zeit-Konfiguration in `/etc/ws.conf`

```
[...]
duration: 60                    # max. duration in days
maxextensions: 2                # number of extensions
workspaces:
  scratch:
    [...]
    duration: 60
    keeptime: 30                # time in days to keep workspace after expiration
    maxextensions: 2
  scratch_fast:
    [...]
    duration: 30
    keeptime: 30
    maxextensions: 2
```

Zusätzliche Features für Betrieb

Load-Balancing

```
workspaces:
  lustre:
    [...]
    spaces: [/lustre/ws/0, /lustre/ws/1]   # list of directories
    spaceselection: random                 # "random" (default), "uid" (uid%#spaces),
                                           # "gid" (gid%#spaces)
```

Migration

```
workspaces:
  lustre:
    [...]
    allocatable: no                 # do not allow new allocations in this workspace
    extendable: no                  # do not allow extensions in this workspace
    restorable: no                  # do not allow restores from this workspace
```

Gelerntes und Erfahrenes

- Absolut zu vermeiden: Datenverlust
  - Handlungen des Nutzers im Umgang mit Workspace-Tools
  - Automatisches Aufräumen durch Expirer
- Hohe Akzeptanz bei Nutzer:innen
  - Intuitives Verständnis; steile Lernkurve
  - Skriptbar!
  - Stabiles Interface
- Betrieb
  - Alternativlos(TM) auf TU- und DLR-Systemen
  - Gelebte Praxis seit 2020
- `hpc-workspace`: Wartung, Features, Testing
  - Wünschenswert ist eine Community

Backup Slides

Workspace / Reminder

Workspace tools can send email[1] reminder about expiration date.

Allocation Time

- `ws_allocate` provides the functionality to send an email N days before expiration date

```
$ ws_allocate -n FancyExp -d 17 -r N -m marie@dlr.de
```

[1] @dlr only!

Workspace / Reminder

Workspace tools can send email[1] reminder about expiration date.

Allocation Time

- `ws_allocate` provides the functionality to send an email N days before expiration date

```
$ ws_allocate -n FancyExp -d 17 -r N -m marie@dlr.de
```

Reallocation

- Spend one extension to set email reminder

```
$ ws_allocate -n FancyExp -d 17
$ ws_allocate -n FancyExp -d 17 -r N -m marie@dlr.de -x
```

[1] @dlr only!

Workspace / Reminder

Later

- `ws_send_ical` sends an calender entry via email

```
$ ws_send_ical -F scratch_fast -n FancyExp -m marie@dlr.de
```

Workspaces / List Options

- Sort workspaces by remaining time

```
$ ws_list -R
```

- Terse output

```
$ ws_list -t
id: FancyExp
      workspace directory  : /scratch/ws/0/marie-FancyExp
      remaining time       : 54 days 23 hours
      available extensions : 2
```

- Verbose output

```
$ ws_list -v
[...]
      reminder             : Sun Jan 17 10:04:57 2021
      mailaddress          : marie@.dlr.de
```

Workspace / Managing Links

- Create, update and remove links to all your personal workspaces within a specified directory

```
$ ws_register <DIR>
```

- **Tip:** To automatically invoke the update of the links, add the command into your personal shell configuration (e.g., `.bashrc`).

Workspace / Sharing

- Sharing of directories via ACLs (Access Control Lists)
- ACLs allow the owner to manage access rights for colleagues

Workspaces

- A workspace is owned by exactly one user (creator) and a change of ownership is not supported
- A “shared” workspace should be created by a user in a leadership role
- Important: Only the creator of the workspace can extend or release it

Backup 2 / `~/ws_user.conf`

- Store default values for reminder and email
- Defaults in file can be overruled with command line options

Thank you for your attention.