



bw | HPC - S5



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386



universität
uulm



universität freiburg



University of Stuttgart
Germany

ESSLINGEN
UNIVERSITY

File Systems Introduction

Begatim Bytyqi (KIT/SCC)

Motivation & Goals

Common user I/O issues

- Limited awareness of efficient I/O(input/output) practice
- Using an unsuitable file system for the task
- Harmful I/O access patterns
 - Affecting other users unintentionally
 - Placing excessive load on the file system

In this course you'll learn

- Choosing the correct storage location based on your data characteristics.
- How to check your quota limits?
 - Quotas restrict usable capacity and usable inodes
- How to manage your workspaces?
 - Workspaces are special storage pools on bwHPC systems



Material: Slides & Scripts

- <https://indico.kit.edu/event/5462/>
- BwUniCluster 3.0: /opt/bwhpc/common/workshops/2026-04-13/
- HoreKa: /software/all/workshop/2026-04-13/

How to read the following slides

Abbreviation/Colour code	Full meaning
<code>\$ command -option value</code>	<code>\$</code> = prompt of the interactive shell The full prompt may look like: <code>user@machine:path\$</code> The command has been entered in the interactive shell session
<code><integer></code> <code><string></code>	<code><></code> = Placeholder for integer, string etc
<code>foo, bar</code>	Metasyntactic variables

■ Additional information

- Basics about file systems, file management, access permissions

- Chapter 3 at https://training.bwhpc.de/goto.php?target=lm_310

- Tips about data transfer

- https://wiki.bwhpc.de/e/Data_Transfer

- Information related to this talk


- <https://wiki.bwhpc.de> -> select cluster -> select File Systems

- Dive deeper: Parallel I/O and Distributed File Systems

- https://events.gwdg.de/event/898/sessions/800/attachments/947/1868/IO-Lecture_3.pdf

HPC STORAGE MANAGEMENT GUIDE: DATA & I/O OPTIMIZATION

\$HOME (Home Directory)




- SOFTWARE CONFIGURATION FILES
- FINAL RESULTS

✗ Omit HEAVY I/O
✗ Omit TEMPORARY DATA

Permanent storage for configuration and setup

WORKSPACES (Working Directories)




- INTERMEDIATE RESULTS
- HUGE INPUT/OUTPUT DATA SETS
- SHARED SCRATCH DATA (between nodes)

✗ Omit SMALL FILES
✗ Omit TINY BLOCK SIZES
✗ Omit MANY METADATA OPS

- If not possible, use Workspaces on **FLASH STORAGE** (for KIT & HoreKa users)

Data with defined lifetimes for large files


\$TMPDIR (Local Node Filesystem)



- DATA AVAILABLE ONLY DURING JOB RUNTIME (Local Node)
- DATA TRANSFERS WITHIN BATCH JOB
- ALL SORTS OF I/O ALLOWED

Fast local storage for a single node job

BEEOND (Private Job Filesystem)




- DATA ONLY DURING JOB RUNTIME (Batch Job Nodes)
- TRANSFERS WITHIN JOB
- ALL SORTS OF I/O ALLOWED

Only on bwUniCluster 3.0 & HoreKa

Fast private filesystem
Spanning job nodes

EXTERNAL STORAGE



- ARCHIVE SCIENTIFIC DATA
- MOVE LARGE DATASETS

Examples: RDA, LSDF (e.g., at KIT)

USE HUGE FILES OR COMPRESSED ARCHIVES

Long term archiving
Outside HPC

OPTIMIZED DATA PLACEMENT SUMMARY

- Use **\$HOME** for **PERMANENT DATA**
- Use **WORKSPACES** for **HUGE FILES** and **SEQUENTIAL I/O**
- Use **\$TMPDIR** or **BeeOND** with **MANY SMALL FILES (> 10,000)** or **RANDOM I/O**

\$HOME = Home directory

- \$HOME is visible on all nodes of a cluster
- Properties of \$HOME on different clusters

Cluster	Quota capacity limit	Quota file limit	Backup
JUSTUS 2	400 GB per user	2 mill. per user	Yes
Helix	200 GB per user	unlimited	Snapshots
NEMO 2	100 GB per user	unlimited	No
BinAC 2	40 GB per user	unlimited	Yes
BwUniCluster 3.0	500 GB per user also limit per organization	5 mill. per user	Yes
HoreKa	10 TB per project	10 mill. per project	Yes

\$HOME on bwUniCluster 3.0

HowTo goto:

```
$ cd $HOME
```

```
$ cd ~
```

User's quota usage and limits:

```
$ lfs quota -uh $(whoami) $HOME
```

```
Disk quotas for usr ka_ab1234 (uid 9999):
```

Filesystem	used	quota	limit	grace	files	quota	limit	grace
/home/kit/ka_scc/ka_ab1234	22.59G	500G	550G	-	81408	5000000	5500000	-

current usage

soft limit

hard limit

\$HOME / \$PROJECT on HoreKa

- \$Home and \$PROJECT are identical if your account is member of one project
 - Otherwise to change to another project which will also modify \$PROJECT

```
$ newgrp <another_project_group>
```

- Project group quota usage and limits:

- Show usage and limits of your project group on the \$HOME file system:

```
$ /usr/lpp/mmfs/bin/mmquota -j $PROJECT_GROUP --block-size G -C hkn.scc.kit.edu hkfs-home
```

		Block Limits			File Limits		
Filesystem	type	GB	quota	limit	files	quota	limit
home	FILESET	3467	10240	11264	4747501	10485760	11534336

*Directory tree with
advanced management
capabilities*

current usage

soft limit

hard limit

Exercise 1: Show quotas

- Login to bwUnicluster 3.0 or HoreKa and show list of commands for exercises:

- BwUniCluster

```
$ cat /opt/bwhpc/common/workshops/2026-04-13/pfs_commands.txt
```

- HoreKa

```
$ cat /software/all/workshop/2026-04-13/pfs_commands.txt
```

- Use Cut & Paste to execute the first commands which show your quotas

Workspaces = Working directories with lifetime

- **Workspace:** lifetime on allocated folder
 - Available on all clusters, visible on all nodes of a cluster
 - HowTo:
 - <https://wiki.bwhpc.de/e/Workspace>

<code>\$ ws_allocate foo 10</code>	Allocate workspace <i>foo</i> for 10 days
<code>\$ ws_list</code>	List your workspaces
<code>\$ ws_find foo</code>	Get absolute path of workspace <i>foo</i>
<code>\$ ws_extend foo 5</code>	Extend lifetime of your workspace <i>foo</i> by 5 days from now. Number of extensions depends on cluster.
<code>\$ ws_release foo</code>	Manually erase your workspace <i>foo</i>

Properties of Workspaces on different clusters

Cluster	Capacity limit	File limit	Max lifetime	Max extensions
JUSTUS 2	20 TB per user	5 mill. per user	90 days	unlimited
Helix	10 TB per user	unlimited	30 days	10 times
NEMO 2	5 TB per user	1 mill. per user	100 days	99 times
BinAC	unlimited	unlimited	30 days	3 times
BwUniCluster 3.0	40 TB per user	20 mill. per user	60 days	3 times
HoreKa	250 TB per user	50 mill. per user	60 days	3 times

Exercise 2: Create workspace

- Allocate two workspaces

```
$ ws_allocate ws01 30
Info: creating workspace.
/pfs/work9/workspace/scratch/myuser-ws01
remaining extensions : 3
remaining time in days: 30
```

```
$ ws_allocate ws02 10
Info: creating workspace.
/pfs/work9/workspace/scratch/myuser-ws02
remaining extensions : 3
remaining time in days: 10
```

Exercise 3: List workspace

Find workspace path and switch to it

```
$ ws_list
id: ws01
  workspace directory : /pfs/work9/workspace/scratch/myuser-ws02
  remaining time      : 29 days 23 hours
  creation time       : Wed Oct  6 18:55:17 2021
  expiration date     : Fri Nov  5 17:55:17 2021
  filesystem name     : pfs7wor9
  available extensions : 3

id: ws02
  workspace directory : /pfs/work9/workspace/scratch/myuser-ws02
  remaining time      : 09 days 23 hours
  creation time       : Wed Oct  6 18:55:17 2021
  expiration date     : Fri Nov 15 17:55:17 2021
  filesystem name     : pfs7wor9
  available extensions : 3
```

Exercise 4: Find workspace path

- Find workspace path and switch to it

```
$ ws_find ws01
/pfs/work9/workspace/scratch/myuser-ws01

$ cd $(ws_find ws01)

$ pwd
/pfs/work9/workspace/scratch/myuser-ws01
```

Exercise 5: Release workspaces

■ Release workspaces

```
$ ws_release ws01
```

```
$ ws_list
```