NHR Data-Intensive Project Cheat Sheet

Version 1.0 September 13, 2024

Introduction

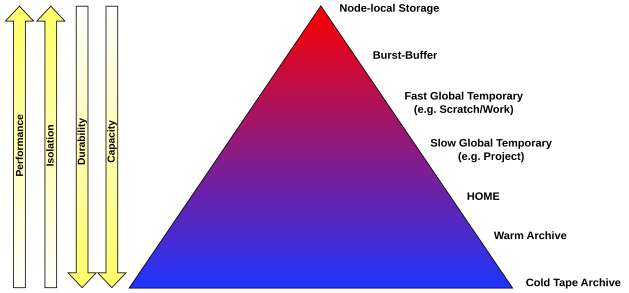
This is a quick onboarding Cheat Sheet for users intending to do data-intensive projects on one of the NHR systems. The goal of this cheat sheet is to provide a quick, yet quite overarching, overview of the most important aspects when doing data-intensive projects. This should help users identify missing aspects in their current experimental plan and provide ideas about possibly missing aspects. For all the points below, you can ask the local support of your NHR center for help.

Storage Tiering

There is a wealth of different storage systems available, which all offer different advantages and disadvantages. Out of this vast amount of available options, HPC centers provide a subset within a tiered storage environment. This means users have to (actively!) choose the right storage system for their use cases. Typical attributes in which these storage tiers differ are:

- Available capacity
- Performance, both streaming and IOPS
- Data durability (including backups, and snapshots)
- Lifetime (during job, weeks and months, quarters and years)

These different attributes always come as a trade-off:



Data Volumes

Different dataset sizes have different requirements:

- Utilize storage tiers according to your dataset size.
- Check the quotas of the storage tiers you are using.
- Look for data mover services/commands provided on your cluster.
- In case there are workspaces: Use them.

Data Parallelism

Data-intensive projects are typically not only I/O bound and require large data volumes but also that data parallelism is employed to parallelize the project on an HPC system. Data parallelism is a special paradigm for parallel computing, where either different tasks, e.g. for a hyperparameter search, are running on the same input data, or the overall data set is split into multiple independent segments, where the same task is then used to process the larger data set in parallel.

To minimize stalling times due to slow I/O requests, it is important to stage on a performant storage tier. Even if single processes run fine, choosing the wrong storage tier can lead to performance problems when fully scaling out since multiple processes can then overload the storage system. This will impact all users, and HPC admins are therefore monitoring the system for suspicious tasks. To determine the best storage tier, the access patterns (see below), have to be considered carefully.

Data Ingest

Transferring data from and to the cluster can be challenging and there are several things that need to be considered. Due to the high processing speed of HPC systems the data should be stored on the cluster and not be requested during runtime, otherwise the data transfer is a mayor bottleneck for the performance. However, the amount of data, the number of files, tools that are used for transfer and the network connection all have a big impact. Quick tips:

- Don't transfer millions of files, combine into 10 - 1000 files.
- Compare tools for file transfer. Performance varies greatly.
- Analyse the local network. Identify bottlenecks.

Data Lifecycle Planning and Management

Reproducibility is a requirement that can be easily overlooked during an HPC project. Documenting the workflow including pre- and postprocessing steps, storing the raw and processed data where it cannot be reproduced, and using a versioning system for the codes can be a good start. Building up on that domain-specific metadata schemata can be defined to provide detailed information for anyone who wants to reuse the data.

Data Access Patterns

File systems on HPC

- support processing few big files better than many small files
- support in-order processing of data better than random I/O access patterns
- are shared among users: bad I/O patterns affect all users!
- if you are doing small file, random IO try to use node local storage
- monitoring..

Consider mounting archives with many small files rather than extracting archives!

Data Acquisition

- Be careful when downloading data with parallel jobs: Rate limits of data providers, which may lead to a (temporary) ban of IP addresses, may affect your jobs and other users.
- Look at your center for publicly available data sets before staging standard data sets yourself

Archiving

Archives are suited for longer-term storage of small amounts of files, possibly of large size, but are slow. Usually, tape storage is used. *Best practice*: create a (packed) archive of your data, e.g. a gzipped tar file (`.tar.gz`) like as follows:

```
$ tar -czf my-archive.tar.gz /path/to/my/data
$ mv my-archive.tar.gz /path/to/archive
```

How to get Help?

Refer to the NHR center documentation or contact support

Center	Website	E-Mail
Berlin	nhr.zib.de	support@zib.de
Göttingen	gwdg.de/hpc	support@gwdg.de
Aachen	RWTH HPC Help	hpc-projects@itc.rwth-aachen.d