Getting started with DKRZ‘s new HPC System „Levante”

See also: docs.dkrz.de/doc/levante

Hendryk Bockelmann
Deutsches Klimarechenzentrum (DKRZ)

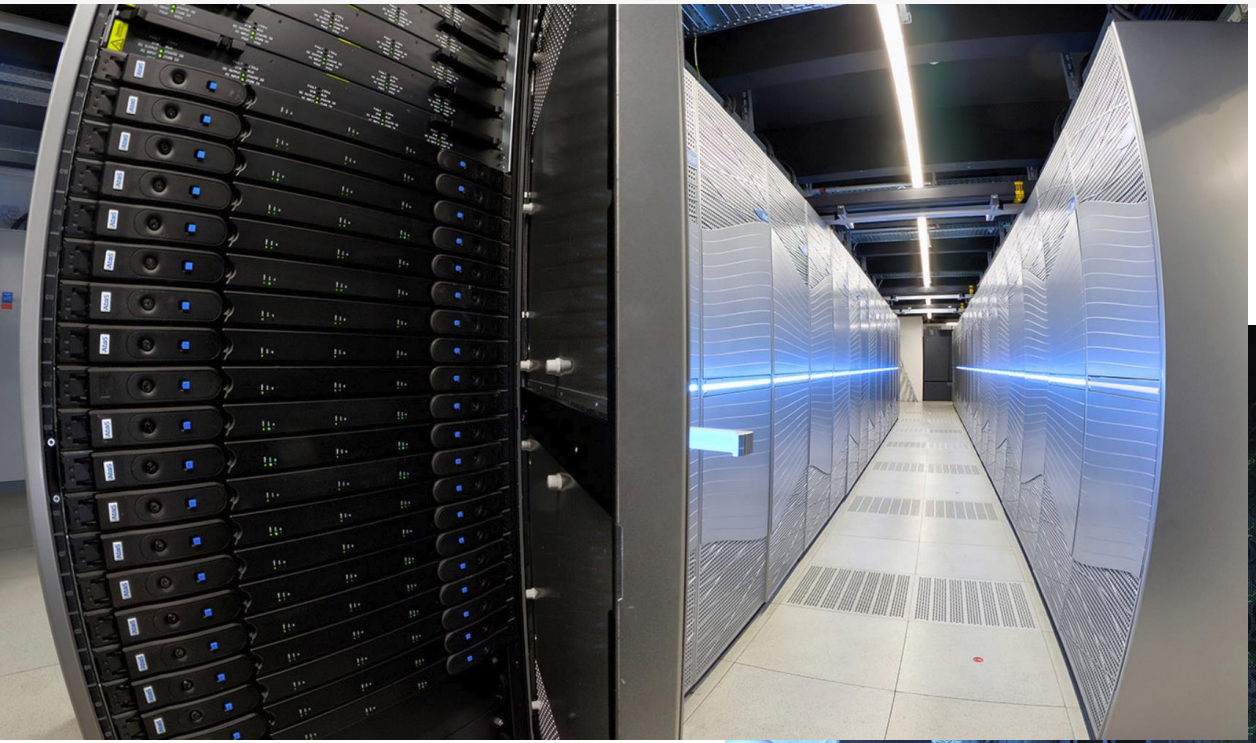
Logistics

This is intended as a high level initial overview

Please post questions here:
pad.gwdg.de/8zKXPOFeRueWmVEGC0hjyw

We will organize detailed sessions on specific topics if needed (i.e. many requests in certain areas)

General System Overview



General System Overview - Hardware

Current status:

- 2679 CPU nodes (AMD EPYC Milan)
- 130 PByte */work* (and */scratch*) on disks
- 100 TByte */home* on SSDs
- 8 GPU/vis-nodes not yet fully installed

Additional 56 GPU nodes (each with 4x Nvidia A100), ~150 CPU nodes, and */fastdata* file system will be available later

Main Differences to Mistral

- Not more nodes, but more cores + more complexity
  - 128 cores on AMD EPYC vs. 36 cores on Intel BDW
  - No local disk on node -> also */tmp* now in RAM
- Less partitions, uniform nodes
- Different file systems, not only namespaces -> */home* should be noticeably faster than on Mistral
- More GPUs … later in 2022

Performance Improvements to be Expected

- 0-30% when using same number of tasks
  - Intel BDW: 2.1 GHz, 77 GiB/s memory bandwidth
  - AMD EPYC Milan: 2.45 GHz, 190 GiB/s memory bandwidth
- Main boost is given by higher scaling
  - Either 4x faster when using same number of nodes
  - Or 4x less nodes required to get same runtime
- MPI environment must be tuned individually

  docs.dkrz.de/doc/levante/running-jobs/mpi-runtime-settings.html

First Steps – Transition from Mistral

```
ssh -X <useraccount>@levante.dkrz.de
```

- User and project accounts as before
- */work* data (i.e. project data) already copied for you
  - Next slide
- */home* not copied – use chance to get rid of old stuff
  - Especially it won‘t make much sense to copy conda envs
  - Settings in shell startup files (*.bashrc, .profile*, …) need to be adapted anyway

Mistral will be online at least until end of May 2022

First Steps – Data Transfer

All project data from Mistral – until a certain date – is available on Levante

```
/work/<project>/from_Mistral/<project>
```

Last sync with Mistral is indicated via file

```
/work/<project>/from_Mistral/last-sync-YYYY-MM-DD
```

- New data on Mistral must be synchronized on your own - please contact us in case of large quantities
- Copied data will be deleted later (after announcement)

Caution: *mv* between **different** projects (or between */scratch* and */work*) causes overhead

First Steps – SW-Tree and Modules

As on Mistral we use *modules* to set the environment

```
$ module avail openmpi
-------------------- /sw/spack-levante/spack/modules --------------------
openmpi/4.1.2-gcc-11.2.0  openmpi/4.1.2-intel-2021.5.0
```

Naming convention

```
<sw>/<version>-<mpi part>-<compiler part>
```

- MPI part (e.g. *openmpi-4.1.2*) might be missing
- Currently just *gcc-11.2.0* and *intel-2021.5.0* compilers used for SW-tree

First Steps – How to Compile (very short!)

- Intel compiler recommended; AMD CPUs must explicitly be specified for the Intel compiler

```
intel-oneapi-compilers/2022.0.1-gcc-11.2.0
-march=core-avx2
```

- OpenMPI-4 recommended, IntelMPI as alternative but without support from Atos

```
openmpi/4.1.2-intel-2021.5.0
intel-oneapi-mpi/2021.5.0-intel-2021.5.0
```

First Steps – How to Compile (very short!)

- Collection of standard libs is continuously extended

```
hdf5, netcdf-c, netcdf-fortran, …
```

- Use module information to get paths for compilation and linking

```
$ module show netcdf-fortran/4.5.3-openmpi-4.1.2-intel-2021.5.0
…
prepend-path    PATH /sw/spack-levante/netcdf-fortran-4.5.3-k6xq5g/bin
…
$ mpifort –I/sw/spack-levante/netcdf-fortran-4.5.3-k6xq5g/include …
```

First Steps – SW-Tree and Modules

Levante uses *Spack* (spack.io) as package manager

- Caution: Paths are no longer self explaining, e.g.

```
/sw/spack-levante/netcdf-fortran-4.5.3-k6xq5g
```

- You can use Spack to create your own SW-tree based on common upstream packages like compiler or MPI

docs.dkrz.de/doc/levante/code-development/building-with-spack.html

- There is a long list of wishes for common SW!
  - We are working hard on the implementation
  - Some requests cannot be served – it is an HPC-system, not a desktop machine

First Steps – a bit more on Spack

```
# list all installed packages
spack find

# use spack load if no module file available, e.g.
spack load mesa
spack find --loaded

# inspect available netcdf-fortran packages and their dependencies
spack find -dp netcdf-fortran

# built with Intel compiler
spack find -dp netcdf-fortran%intel

# use hash
spack find -dp /k6xq5g
```

Usage Model – Partitions

Partition	Size	Usage
compute	2655 nodes, max 512 per job, max 8h runtime 256, 512, and 1024 GB RAM	Simulation runs
shared	10 nodes, max 1 per job, max 7d runtime	Small jobs, e.g. pre/postprocessing
interactive	10 nodes (dyn. growing), max 1 node in sum over all jobs, max 12h runtime; all 512 GB	Jupyterhub, salloc
gpu	Currently just 4 nodes	GPGPU, ML, Vis

Example job scripts can be found here:
docs.dkrz.de/doc/levante/running-jobs/example-batch-scripts.html

Usage Model – Data Processing

Caution:

- Login nodes should not be used for heavy workloads (limits enforced)
- Use dedicated resources instead of fighting with others on shared nodes
  - NO successor to mistralpp.dkrz.de nodes given

docs.dkrz.de/doc/levante/data-processing-on-levante.html

```
$ salloc –p interactive –A <account> -t 240 --mem=10G --x11
…
# now you are directly on a dedicated (not shared) resource
$ echo $SLURM_JOB_NODELIST
l10160
```

```
$ ssh –X l10160
```

Usage Model – HSM

- HSM = new **tape archive** system since Nov 2021: HPSS => **StrongLink**
- Command line access: ***slk***, ***slk_helpers*** replace ***pftp***
- File retrieval: only via shared, compute and interactive partitions (not via login nodes)
- Set *--mem=6GB* in all archival and retrieval jobs
- Documentation and FAQ docs.dkrz.de/doc/datastorage/hsm/index.html

Current Limitations

- GPU nodes still to be installed (SW and HW!)
- SW missing
  - Open ticket at support@dkrz.de
  - Try mounting storage locally: sshfs
- IB unreliable
  - Endpoint timeouts – job will crash, node will be drained
  - Varying runtime – buggy links, will be solved by firmware update soon
- Bus error while I/O instructions
  - DDN is already preparing a patch

Further Topics

1. Details on AMD EPYC Milan & compiler
2. HowTo job scripts and runtime optimization; task-binding, MPI-settings, …
3. GPU usage; ML, GPGPU, Vis

   redmine.dkrz.de/projects/gpu-forum/boards
4. Storage and Archive (HSM)

Q‘n‘A

- Use pad right now: pad.gwdg.de/8zKXPOFeRueWmVEGC0hjyw
- Contact support@dkrz.de

  Please always provide sufficient information - JobId, runscript, log/output, error message