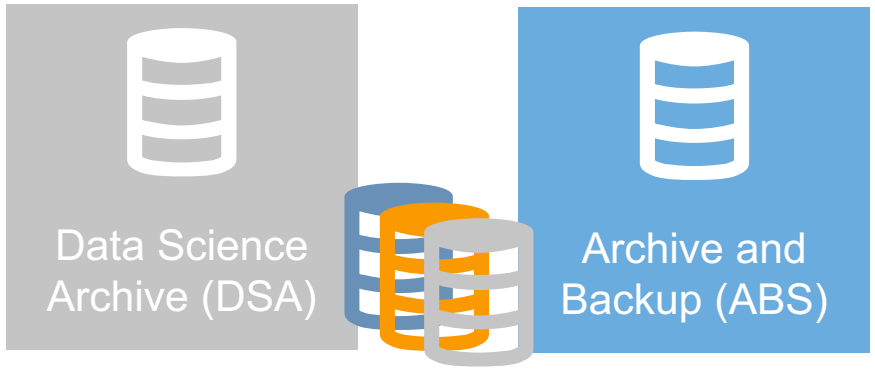
Introduction to Multiuser Cluster Systems at LRZ

Panorama of Systems at LRZ & User perspective

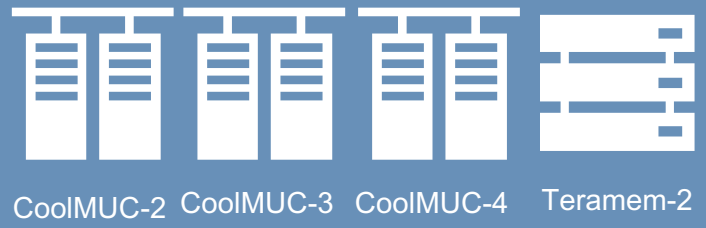
April, 10th 2024

HPC & BDAI Systems for Bavarian Universities



Data Science Storage
(DSS)

LRZ Linux Cluster

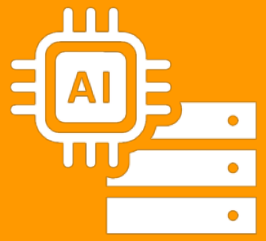


lxlogin<X>.lrz.de

https://doku.lrz.de/linux-cluster-10745672.html

LRZ AI Systems

- "Big Data" CPU nodes
- HPE P100 node
- V100 nodes
- DGX-1 P100, DGX-1 V100
- DGX A100



https://login.ai.lrz.de
ssh login.ai.lrz.de

https://doku.lrz.de/lrz-ai-systems-11484278.html

LRZ Compute Cloud

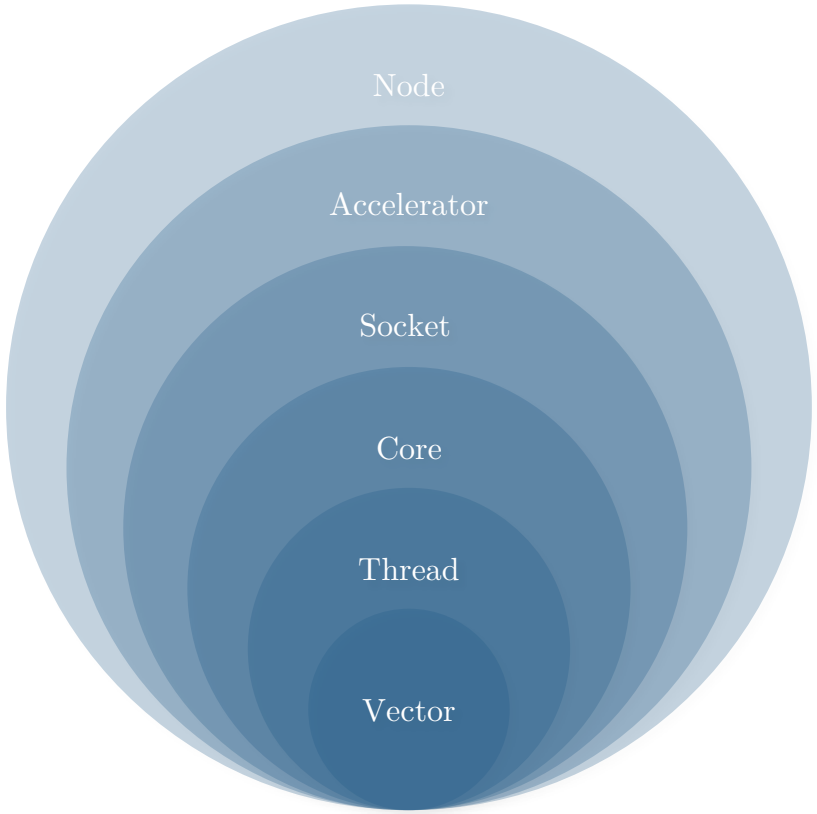
LRZ Compute Cloud
(w/ some GPUs)

https://cc.lrz.de

https://doku.lrz.de/display/PUBLIC/Compute+Cloud

SuperMUC-NG

Levels of Parallelism



- **Node Level** (*e.g.*, SuperMUC-NG has 6480 nodes)
- ***Accelerator Level*** *(e.g., a Nvidia DGX A100 has 8 GPUs)*
- **Socket Level** (*e.g.*, Linux Cluster Teramem has 4 sockets [with 24 cores each])
- **Core Level** (*e.g.*, Linux Cluster CoolMUC-3 nodes have 64 cores [on a single socket])
- ***Thread Level*** *(e.g., Linux Cluster CoolMUC-2 nodes allow 2 threads per core)*
- **Vector Level** (*e.g.*, AVX-512 has 32 512-bit vector registers)

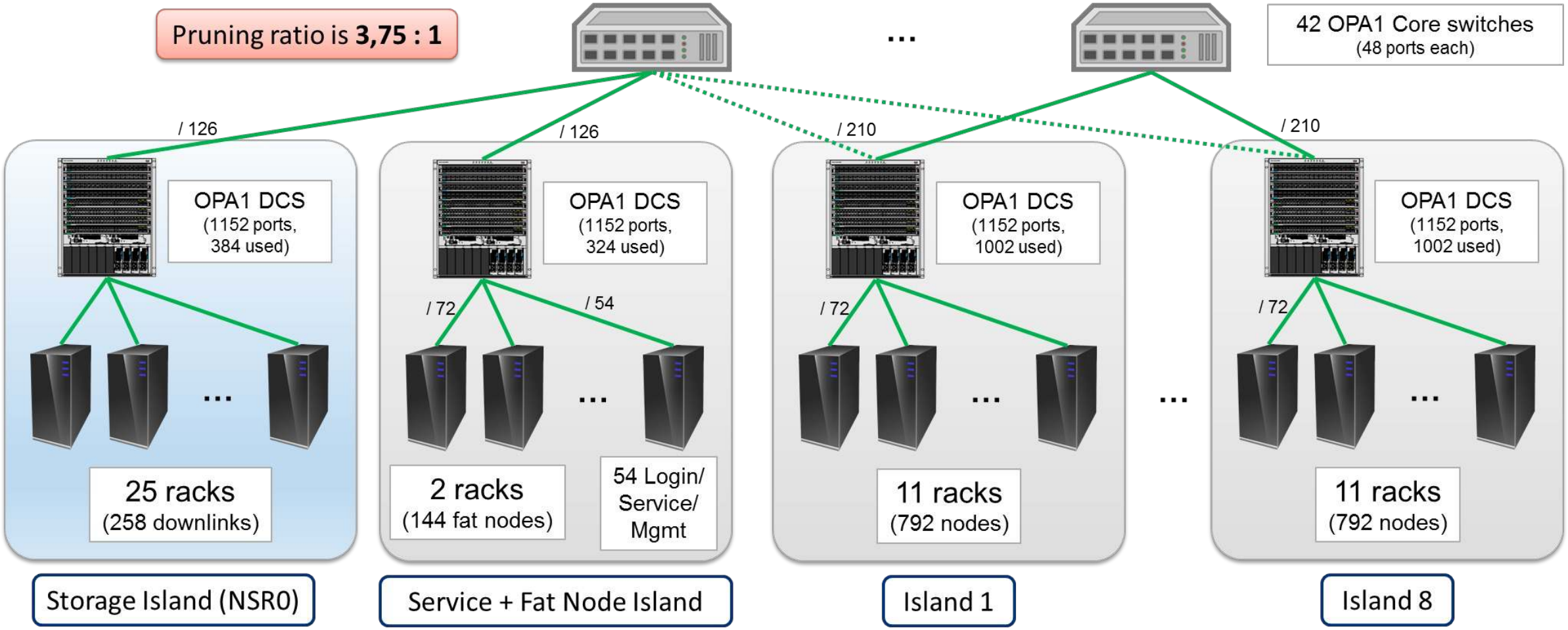
SuperMUC-NG theoretical peak performance:

6480 Nodes x **2** Sockets x **24** Cores x **32** Vectors x **2,7** GHz

= 26 873 856 000 000 000 Flop/s

top500 list

High-Level System Architecture



Hardware Overview

Phase 1

Compute Nodes	Thin Nodes	Fat Nodes	Total (Thin + Fat)
Processor Type	Intel Skylake Xeon Platinum 8174	Intel Skylake Xeon Platinum 8174	Intel Skylake Xeon Platinum 8174
Cores per Node	48	48	48
Memory per Node [GByte]	96	768	N/A
Number of Nodes	6,336	144	6,480
Number of Cores	304,128	6,912	311,040
Peak Performance @ nominal [PFlop/s]	26.3	0.6	26.9
Linpack [PFlop/s]	–	–	19.476
Memory [TByte]	608	111	719
Number of Islands	8	1	9
Nodes per Island	792	144	N/A
Filesystems			
High Performance Parallel Filesystem	50 PiB @ 500 GB/s		
Data Science Storage	20 PiB @ 70 GB/s		
Home Filesystem	256 TiB		
Infrastructure			
Cooling	Direct warm water cooling		
Waste Heat Reuse	For producing cold water with adsorption coolers		
Software			
Operating System	Suse Linux Enterprise Server (SLES)		
Batch Scheduling System	SLURM		
High Performance Parallel Filesystem	IBM Spectrum Scale (GPFS)		
Programming Environment	Intel Parallel Studio XE, GNU compilers		
Message Passing	Intel MPI, (OpenMPI)		

(Phase 2)

Nodes	
Processor	Intel Sapphire Rapids Intel Xeon Platinum 8480+
CPUs per Node	2
Cores per Node	112
Memory per Node	512 GByte DDR5
GPUs	Intel Ponte Vecchio Intel Data Center GPU Max 1550
GPUs per Node	4
Memory per GPU	128 GByte HBM2e
Number of Nodes	240 (incl. 4 login nodes)
Total CPU Cores	26,880
Total Memory	122.88 TByte DDR5
Total GPUs	960
Total GPU Memory	122.88 TByte HBM2e
PEAK (fp64; PFlop/s)	27.96 PFlop/s
Linpack (fp64; PFlop/s)	17.19 PFlop/s
Compute network	
Fabric	NVIDIA/Mellnox HDR Infiniband (200 GBit/s)
Topology	fat tree
Interconnects per Node	2
Number of Islands	1
Filesystems	
HPPFS (same as Phase 1)	50 PB @ 500 GByte/s
DSS (same as Phase 1)	20 PB @ 70 GByte/s
Home Filesystem	256 TByte
DAOS	1 PB @ 750 GByte/s
Infrastructure	
Cooling	Direct warm water cooling
Software	
Operating System	Suse Linux (SLES)
Batch Scheduling System	SLURM
High Performance Parallel Filesystem (HPPFS)	IBM Spectrum Scale (GPFS)
Programming Environment	Intel OneAPI
Message Passing	Intel MPI, (OpenMPI)

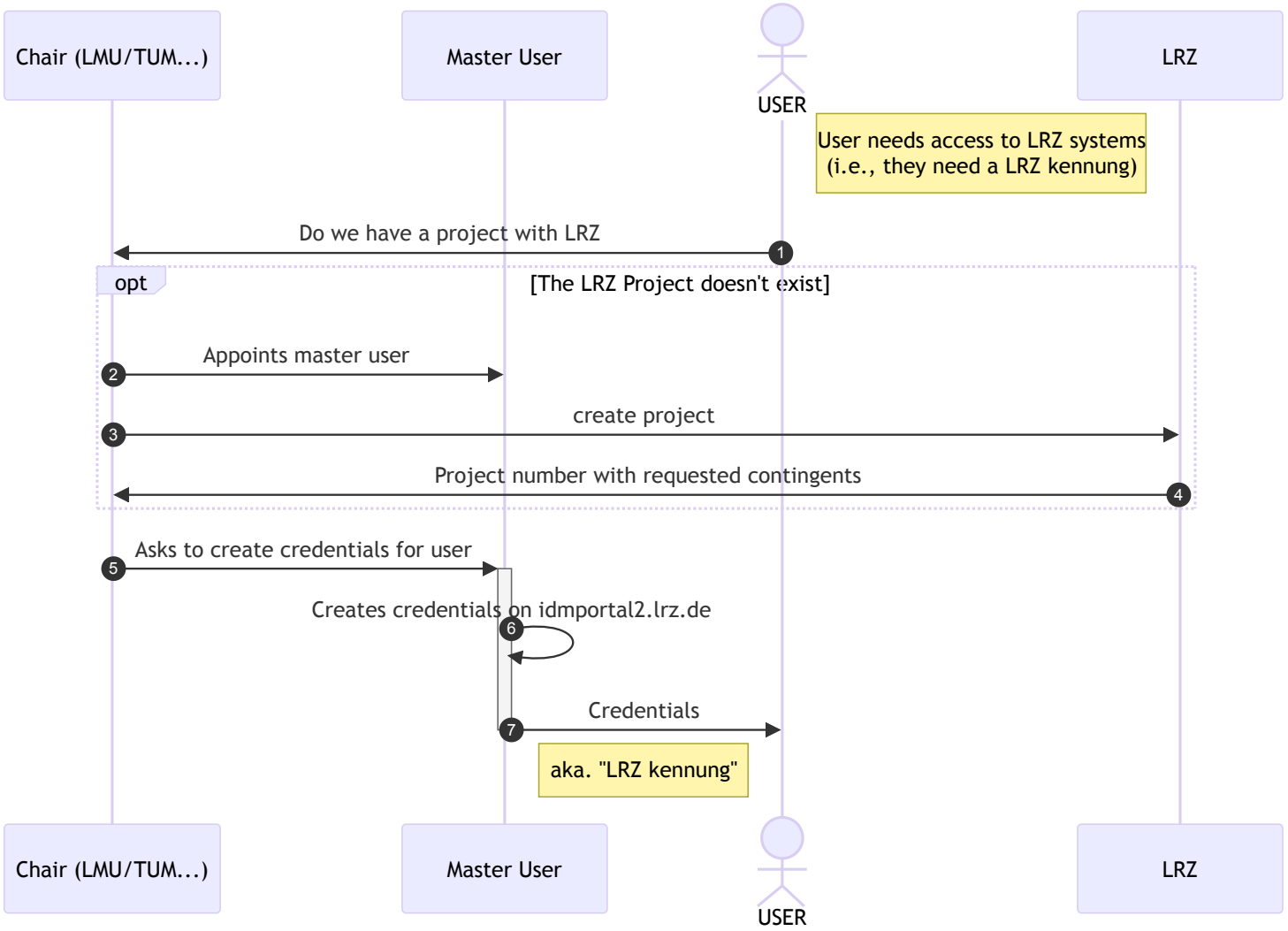
SNG Phase 1: Getting access

There are three (well, four) ways to apply for using SuperMUC-NG:

1. GCS test project: rolling call, fast review (short abstract), $<$ 300.000 core-h
2. GCS regular project: rolling call, technical & scientific review, $<$ 45m core-h
3. GCS large scale project: biannual, technical & scientific review, $>$ 45m core-h
4. (biannual PRACE calls for academic users from any European country)

For further details, see https://doku.lrz.de/x/XAAbAQ

The complete Perspective



Linux Cluster

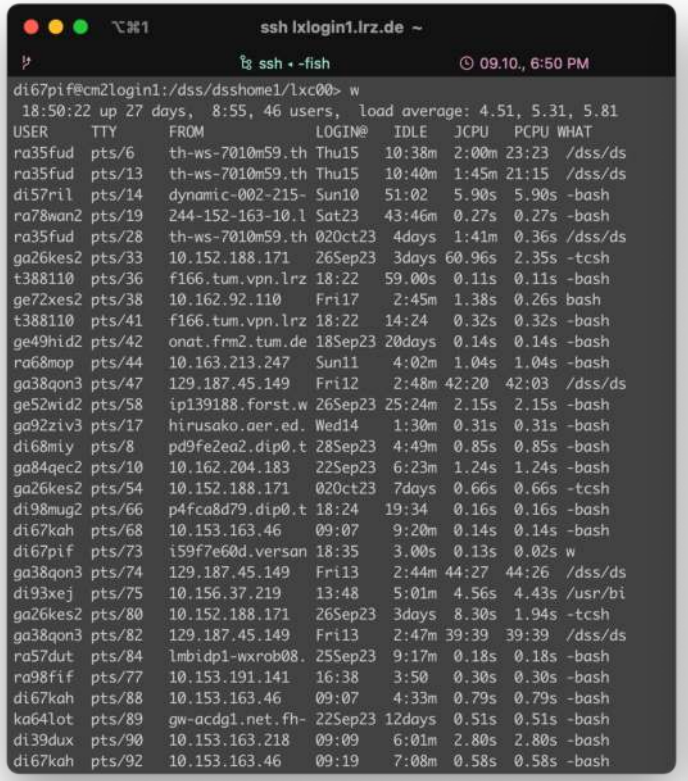
Linux Cluster: Hardware Overview

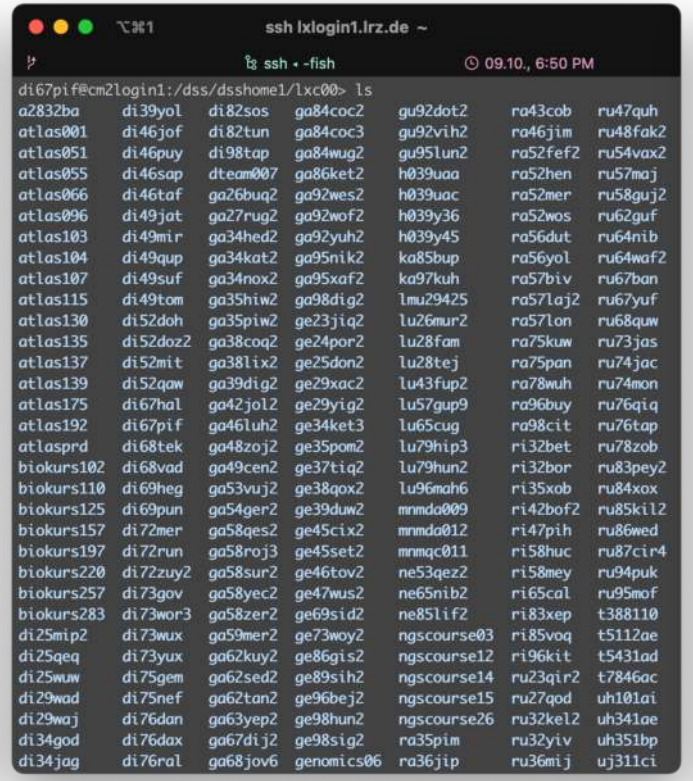
Name	CPU	Cores/Node	RAM/Node (GB)	Nodes (total)	Cores (total)
CoolMUC-2	Intel Xeon E5-2690 v3 ("Haswell")	28	64	812	22736
CoolMUC-3	Intel Xeon Phi ("Knights Landing")	64	96	148	9472
Teramem	Intel Xeon E7-8890 v4 ("Broadwell")	96	6144	1	96
CoolMUC-4	-	-	-	-	-

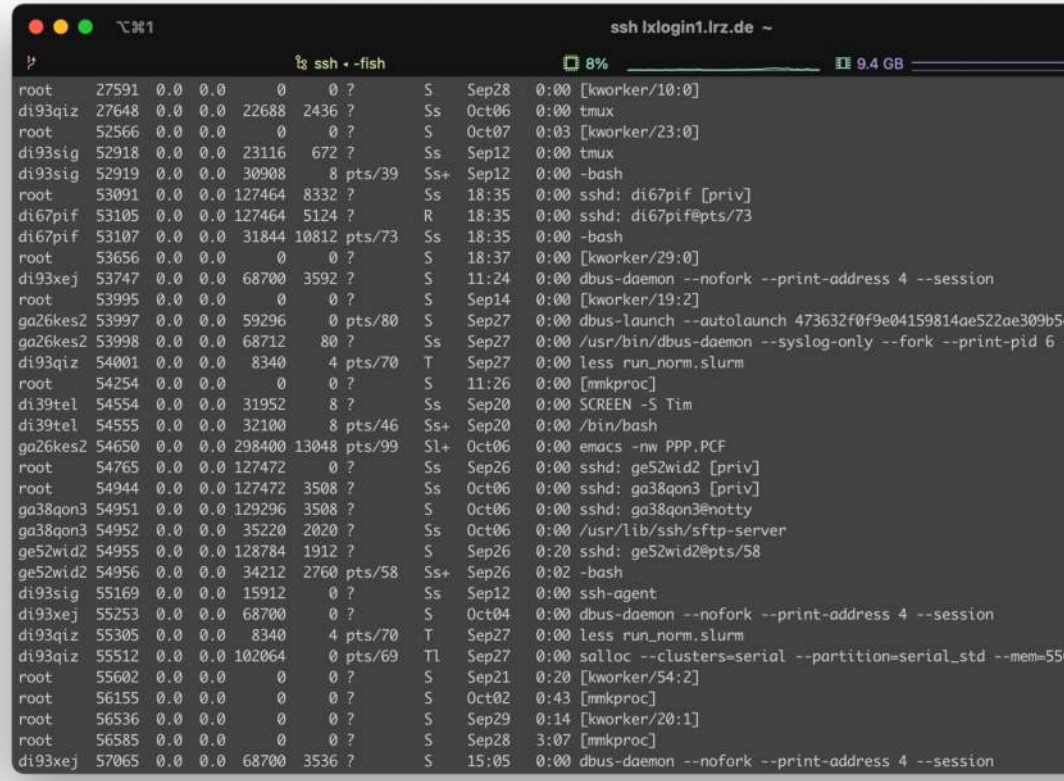
https://doku.lrz.de/linux-cluster-10745672.html

User Perspective: Environment & Workspace

- These are systems shared by many users, i.e. other people will be working on the same (login) node at the same time.
- Be aware of your surroundings and considerate of your fellow colleagues!







User Perspective: Environment & Workspace

You don't have administrative rights on these systems, i.e. no root access.

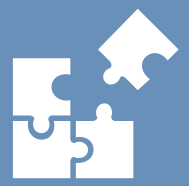
- You will not be able to use the `sudo` command
- You're prohibited from making system-wide modifications
- Disk access is restricted to your home directory (and possibly other storage areas accessible to your account, e.g., your DSS containers)

→ That said, your home (directory) is your castle – there, anything goes!

User Perspective: Environment & Workspace

- If available on the system, modules allow for the dynamic modification of environment variables, e.g., they provide a flexible way to access various applications and libraries available on the system
- List the currently active modules (loaded by default):
  `$ module list`
- Search for available modules:
  `$ module available <module>` or
  `$ module av <module>`
- Get more information about a specific module:
  `$ module show <module>`
- Use `$ module load <module>` to apply the changes of a module to the environment

User Perspective: Package Managers and Binaries



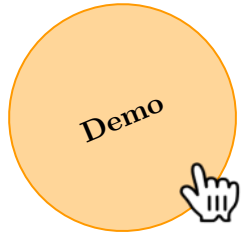
- **Conda** (https://conda.io) is “a package, dependency and environment management for any language – Python, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, FORTRAN, and more”.
- **pip** (https://pip.pypa.io) is “the package installer for Python. You can use it to install packages from the Python Package Index and other indexes”.

  Make sure to install packages to the home directory instead of the system-wide default location:

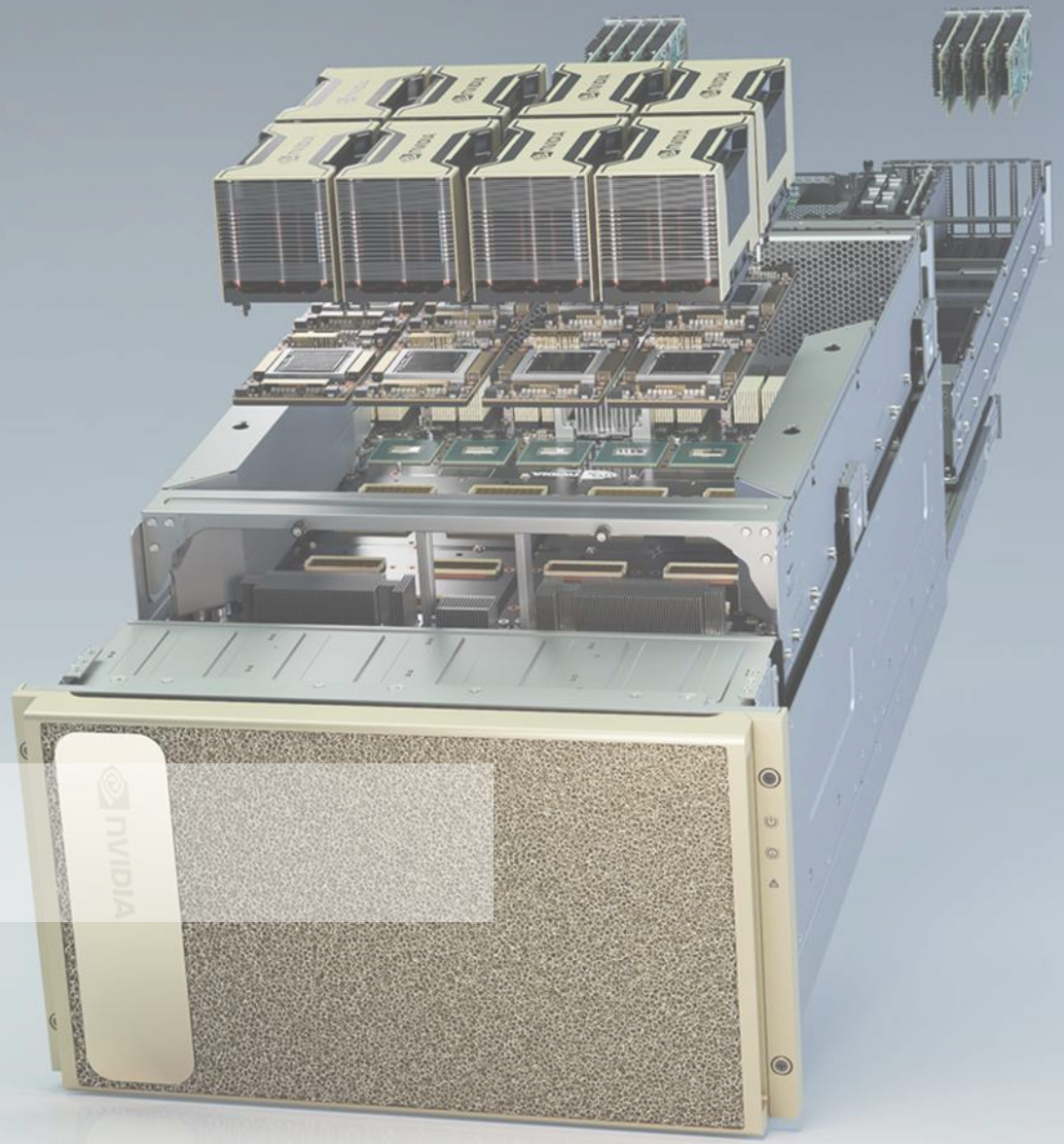
  ```
  ~$ pip install --user <package>
  ```
- **wget** a binary from the internet (be careful!)

  ```
  ~$ wget http://free-software.ru/download/not-malware.bin
  ```
- **Compile** yourself

  ```
  ~$ git clone https://github.com/ggerganov/whisper.cpp
  ~$ cd whisper.cpp
  ~$ make
  ```

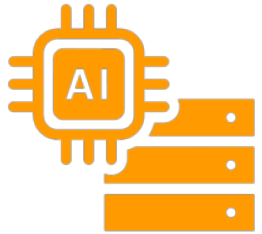


LRZ AI Systems

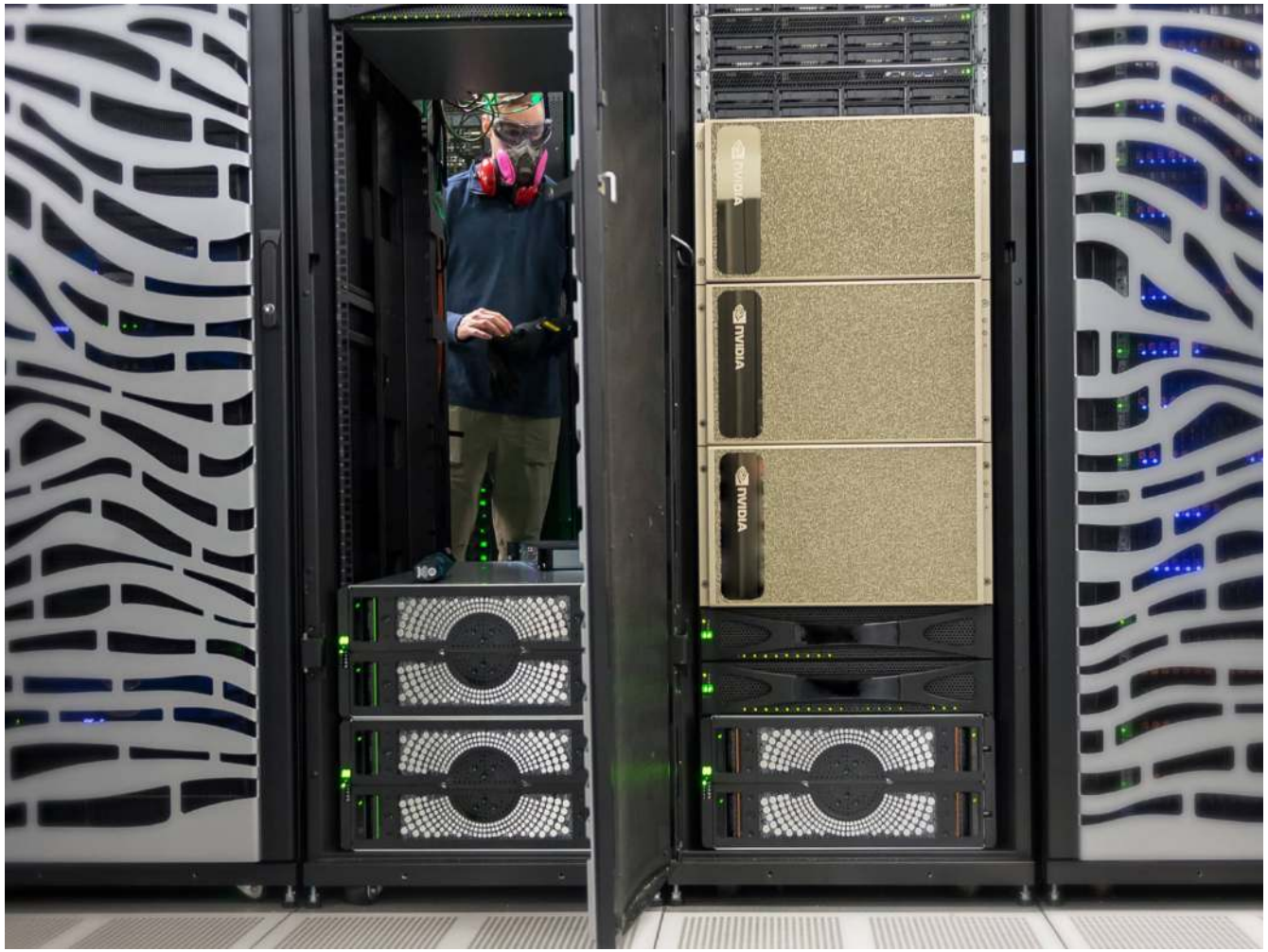


(BD)AI Systems: Hardware Overview

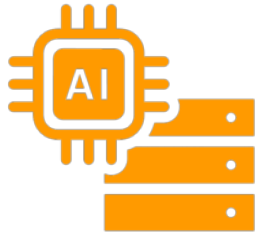
Type	Nodes	CPUs (Node)	Memory (Node)	GPUs (Node)	Memory (GPU)
CPU Nodes	9	up to 20	up to 850GB	-	-
HPE P100 Node	1	64	256 GB	4x P100	16 GB
V100 Nodes	4	40	368 GB	2x V100	16 GB
DGX-1 P100	1	80	512 GB	8x P100	16 GB
DGX-1 V100	1	80	512 GB	8x V100	16 GB
DGX A100/40	1	256	1 TB	8x A100	40 GB
DGX A100/80	4	256	2 TB	8x A100	80 GB

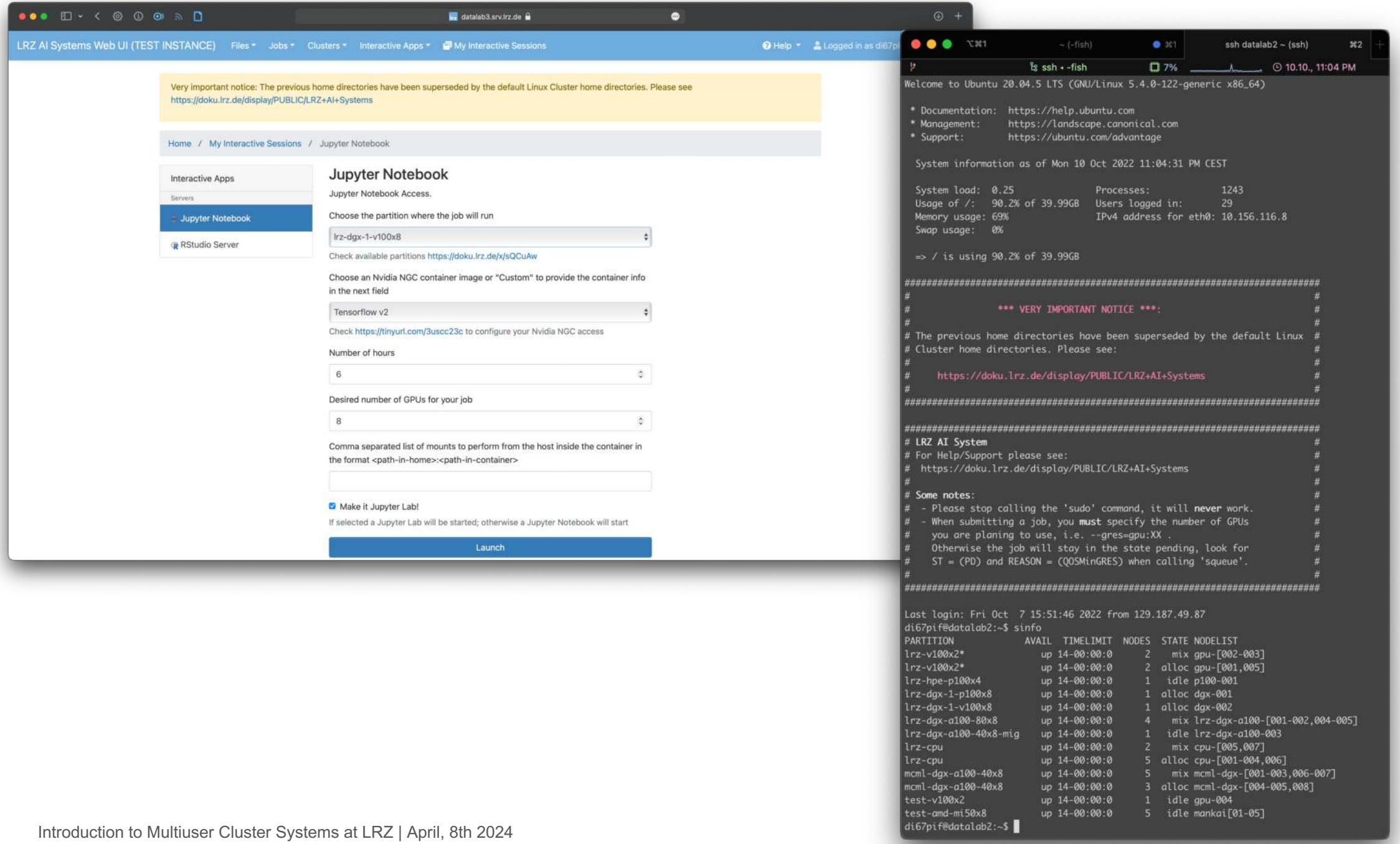


(BD)AI Systems: Hardware Overview



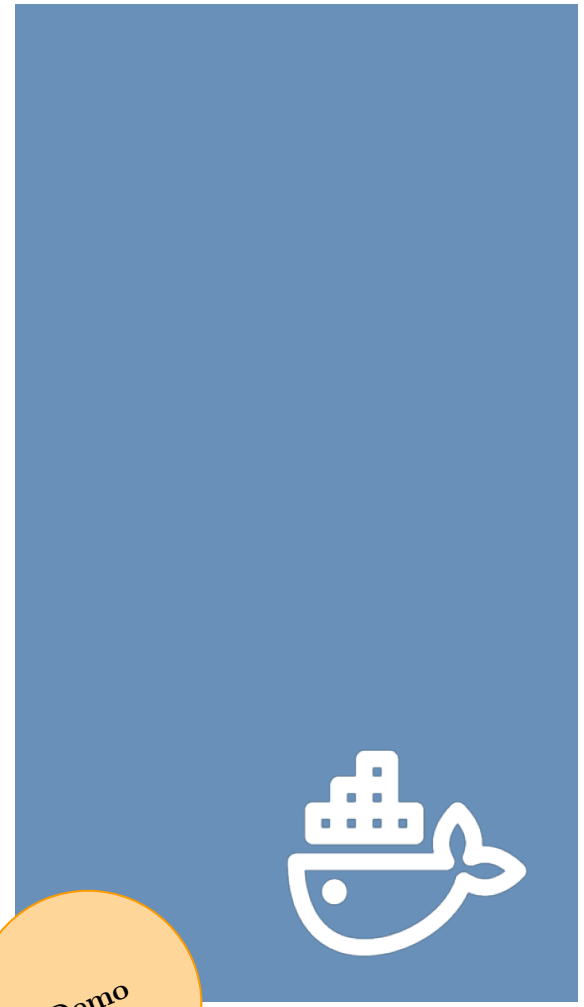
- 3 NVIDIA A100 rack mounted at the Argonne National Lab
- 143kg / node
- 8 GPUs / node
- 400 W
- (not actually made of gold)



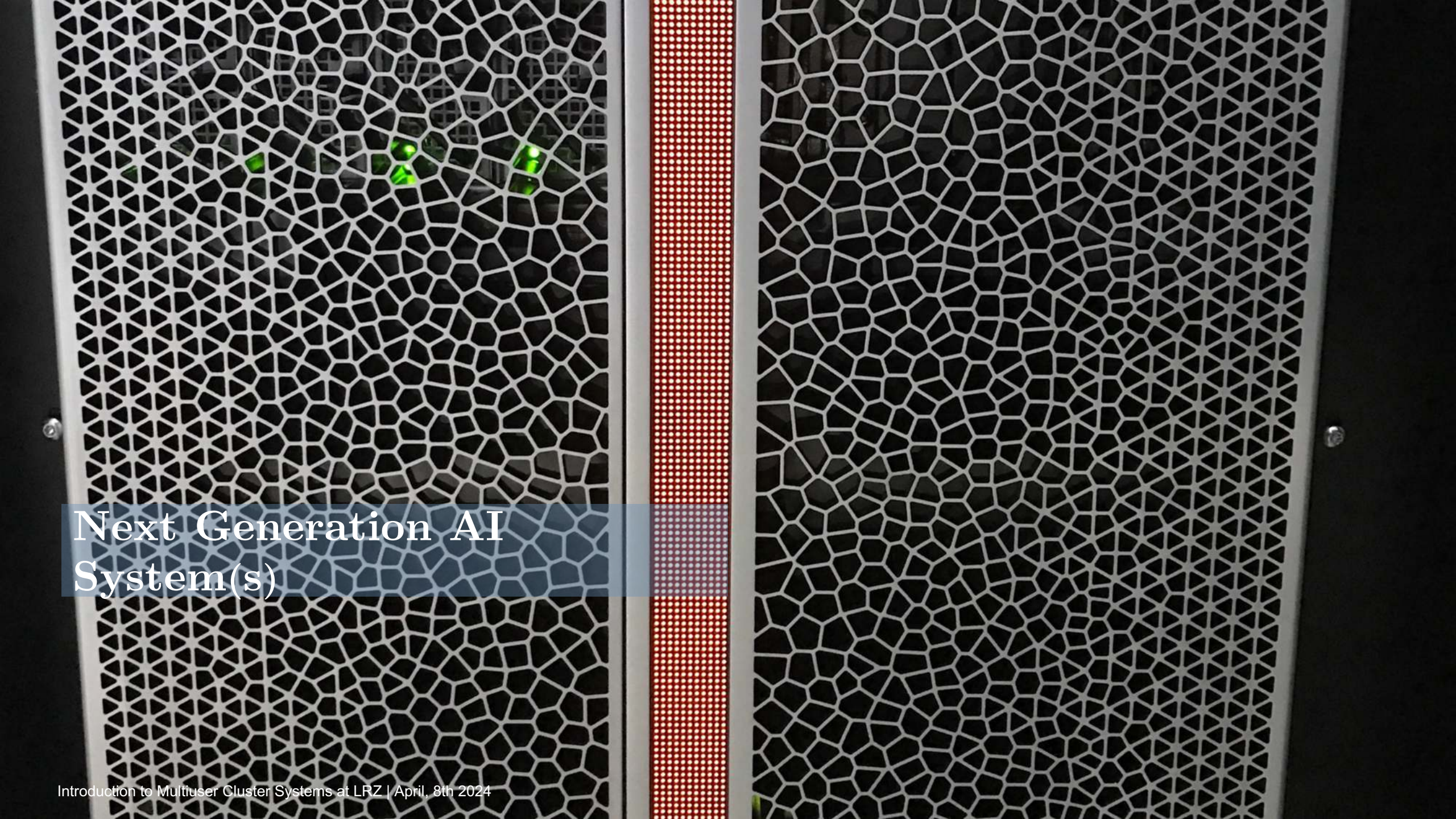

datalab3.srv.lrz.de
LRZ AI Systems Web UI (TEST INSTANCE)
Files
Jobs
Clusters
Interactive Apps
My Interactive Sessions
Help
Very important notice: The previous home directories have been superseded by the default Linux Cluster home directories. Please see
https://doku.lrz.de/display/PUBLIC/LRZ+AI+Systems
Home / My Interactive Sessions / Jupyter Notebook
Interactive Apps
Servers
Jupyter Notebook
RStudio Server
Jupyter Notebook
Jupyter Notebook Access.
Choose the partition where the job will run
lrz-dgx-1-v100x8
Check available partitions https://doku.lrz.de/x/sQCuAw
Choose an Nvidia NGC container image or "Custom" to provide the container info in the next field
Tensorflow v2
Check https://tinyurl.com/3uscc23c to configure your Nvidia NGC access
Number of hours
6
Desired number of GPUs for your job
8
Comma separated list of mounts to perform from the host inside the container in the format <path-in-home>:<path-in-container>
Make it Jupyter Lab!
If selected a Jupyter Lab will be started; otherwise a Jupyter Notebook will start
Launch
ssh datalab2 ~ (ssh)
ssh • -fish
7%
10.10., 11:04 PM
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.4.0-122-generic x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
System information as of Mon 10 Oct 2022 11:04:31 PM CEST
System load: 0.25 Processes: 1243
Usage of /: 90.2% of 39.99GB Users logged in: 29
Memory usage: 69% IPv4 address for eth0: 10.156.116.8
Swap usage: 0%
=> / is using 90.2% of 39.99GB
*** VERY IMPORTANT NOTICE ***:
The previous home directories have been superseded by the default Linux Cluster home directories. Please see:
https://doku.lrz.de/display/PUBLIC/LRZ+AI+Systems
LRZ AI System
For Help/Support please see:
https://doku.lrz.de/display/PUBLIC/LRZ+AI+Systems
Some notes:
- Please stop calling the 'sudo' command, it will never work.
- When submitting a job, you must specify the number of GPUs you are planing to use, i.e. --gres=gpu:XX .
Otherwise the job will stay in the state pending, look for ST = (PD) and REASON = (QOSMinGRES) when calling 'squeue'.
Last login: Fri Oct 7 15:51:46 2022 from 129.187.49.87
di67pif@datalab2:~$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
lrz-v100x2* up 14-00:00:0 2 mix gpu-[002-003]
lrz-v100x2* up 14-00:00:0 2 alloc gpu-[001,005]
lrz-hpe-p100x4 up 14-00:00:0 1 idle p100-001
lrz-dgx-1-p100x8 up 14-00:00:0 1 alloc dgx-001
lrz-dgx-1-v100x8 up 14-00:00:0 1 alloc dgx-002
lrz-dgx-a100-80x8 up 14-00:00:0 4 mix lrz-dgx-a100-[001-002,004-005]
lrz-dgx-a100-40x8-mig up 14-00:00:0 1 idle lrz-dgx-a100-003
lrz-cpu up 14-00:00:0 2 mix cpu-[005,007]
lrz-cpu up 14-00:00:0 5 alloc cpu-[001-004,006]
mcml-dgx-a100-40x8 up 14-00:00:0 5 mix mcml-dgx-[001-003,006-007]
mcml-dgx-a100-40x8 up 14-00:00:0 3 alloc mcml-dgx-[004-005,008]
test-v100x2 up 14-00:00:0 1 idle gpu-004
test-amd-mi50x8 up 14-00:00:0 5 idle mankai[01-05]
di67pif@datalab2:~$

User Perspective: OS-level Virtualization, Containers

- Isolated **user space** instances, called containers, allow programs running inside to only see the container's contents and devices assigned to the container.
- Thus, the environment inside a container can essentially be modified freely, typically **providing (encapsulated) root privileges**
- The most prominent container runtime, Docker, is typically not available on multiuser systems, but you will encounter alternatives
  - Charliecloud (https://hpc.github.io/charliecloud/)
  - Enroot (https://github.com/NVIDIA/enroot)
- Containers imposes no noticeable overhead, i.e. there should be no performance impact and parallelization, GPU access, etc. should – if set up correctly – work as expected
- Containers are UDSS: User Defined Software Stacks: you're basically independent from the environment created by system administrators, but you will only receive limited support for the environment created instead (inside the container).



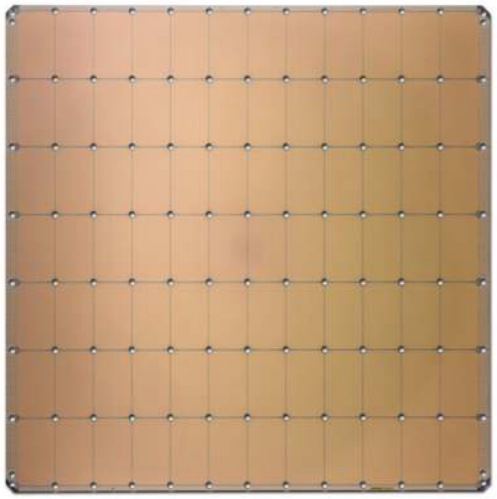
Demo



Next Generation AI System(s)

(BD)AI Systems: Hardware Overview

Cerebras CS-2 Wafer Scale Engine (WSE2) in Numbers



Cerebras WSE-2
2.6 Trillion Transistors
46,225 mm² Silicon

Largest GPU
54.2 Billion Transistors
826 mm² Silicon

A Systems Approach to Deep Learning

"Cluster-scale acceleration on a single chip"

	Cerebras WSE-2	NVIDIA A100	WSE 2 Advantage
Chip Size	46,225 mm^2	826 mm^2	56 X
Cores	850,000	6,912 + 432	123 X
On Chip memory	40 Gigabytes	40 Megabytes	1,000 X
Memory B/W	20 Petabytes/sec	1,555 Gigabytes/sec	12,862 X
Fabric B/W	220 Petabits/sec	4.8 Terabytes/sec	45,833 X

Data Source: https://www.cerebras.net/whitepapers/

LRZ Compute Cloud

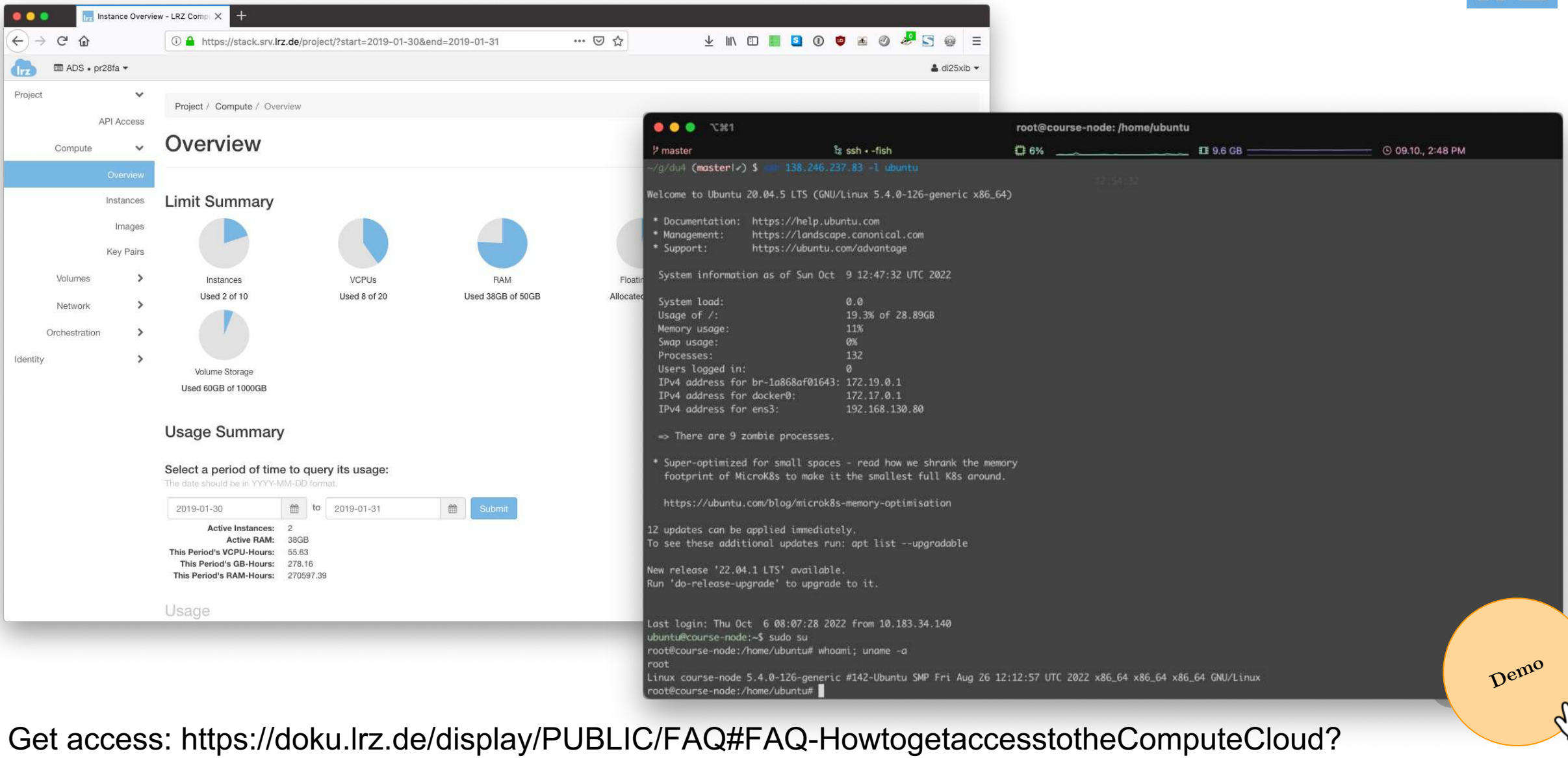
LRZ Compute cloud: Hardware Overview

Compute	200 Nodes 192 GB to 1024 GB RAM Intel® Xeon® ~2.40 GHz
	32 x 2 GPUs Nodes 2x Nvidia Tesla V100 16 GB/node 768GB RAM/node
Storage	15 nodes 2 PB Raw Storage
Networking	100G Intel OmniPath
Software	OpenStack & CEPH

Access to more than 10 vCPUs and/or other restricted resources can be requested by contacting the cloud support team: https://servicedesk.lrz.de/ql/create/105

40000 vCPU capacity with overcommitment
2000 users and 1500 active VMs

Compute Cloud: Hardware Overview



Get access: https://doku.lrz.de/display/PUBLIC/FAQ#FAQ-HowtogetaccesstotheComputeCloud?

LRZ Data Storage

https://doku.lrz.de/display/PUBLIC/Data+Science+Storage

Data Storage: Overview

- The LRZ HPC/HPDA/HPAI Infrastructure is backed by the Data Science Storage (DSS)
  - Long-term storage solution for potentially vast amounts of data
  - Directly connected to the LRZ computing ecosystem
  - Flexible data sharing among LRZ users
  - Web interface for world-wide access and transfer
  - Data sharing with external users (invite per e-mail, access per web interface)
- Additionally, we also provide a new type of Data Archive, based on the DSS Solution stack, called Data Science Archive (DSA) (this basically relates to DSS like AWS Glacier relates to AWS S3).
- Disk space and access is managed (as DSS projects and containers) by data curators. This can be LRZ personnel (e.g., Linux Cluster $HOME directories) or PIs/master users/dedicated data curators (e.g., project storage).

https://doku.lrz.de/display/PUBLIC/File+Systems+and+IO+on+Linux-Cluster

Data Storage: Linux Cluster & AI Systems

- `$HOME` (DSS-backed home directory, managed by LRZ)
  - 100GB per user
  - Access: /dss/dsshome1/lxc##/<user>
  - Automatic tape backup and file system snapshots (see "/dss/dsshome1/.snapshots/" directory)
  - All your important files/anything you invested a lot of work into should be here
  - BUT Not suitable for heavy and/or high-frequency I/O operations, i.e. most machine learning applications. Use the AI Systems DSS instead.

https://doku.lrz.de/display/PUBLIC/2.+Storage+on+the+LRZ+AI+Systems

Data Storage: AI Systems

- AI DSS
  - Up to 5 TB per project **upon request**, shared among project members
  - Access: `$ dssusrinfo all`
  - Configuration (e.g., exports, quota) to be managed by data curator
  - Use this for e.g., high bandwidth, low latency I/O
  - Can not (yet) be accessed from Linux Cluster

https://doku.lrz.de/display/PUBLIC/File+Systems+and+IO+on+Linux-Cluster

Data Storage: Linux Cluster

- DSS project storage
  - Up to 10 TB per project **upon request**, shared among project members
  - Access: `$ dssusrinfo all`
  - Configuration (e.g., exports, backup, quota) to be managed by data curator
  - Use this for e.g., large raw data (and consider backup options)
  - Can be accessed from the AI systems

https://doku.lrz.de/display/PUBLIC/File+Systems+and+IO+on+Linux-Cluster

Data Storage: Linux Cluster

- Legacy `$SCRATCH` (scratch file system, “temporary file system”)
  - 1.4 PB, shared among all users
  - Access: `/gpfs/scratch/<group>/<user>`
- New `$SCRATCH_DSS` (not yet available on CoolMUC-2 compute nodes)
  - 3.1 PB, shared among all users
  - Access: `/dss/lxclscratch/##/<user>`
- No backup (!) and sliding window file deletion, i.e. old files will eventually be deleted (!!) – a data retention time of approx. 30 days may be assumed, but is not guaranteed
- This is the place for e.g., very large, temporary files or intermediate results, directly feeding into additional analyses
- Data integrity is not guaranteed. Do not save any important data exclusively on these file systems! Seriously, don’t do it!

https://doku.lrz.de/display/PUBLIC/DSS+documentation+for+users

Data Storage: Compute Cloud

- The storage backend of the Compute Cloud is used to host the virtual disks belonging to the VMs in the cloud. It is not meant to store large data sets. No backups are created.
- DSS containers can be made available for VMs running in the LRZ Compute Cloud without the need to copy data into the VM.
  - The data curator of the data project, to which the relevant container belongs, needs to export the container to the IP address used by your VM via NFS.
  - You should only export DSS containers to IPs that are statically assigned to and trusted by you. NFS exports follow a "host based trust" semantic, which means the DSS NFS server will trust any IP/system to which a DSS container is exported. There is no additional user authentication between NFS server and client enforced.