Hardware specifications of cluster compute nodes

- This page provides an overview of the compute cluster's hardware specifications, including node details, partitions, and configuration settings. We strive to keep this information up to date, but changes may occur. For the most accurate and current cluster details, please use the following command: `clusterinfo`
- The LUIS computing cluster is a heterogeneous general purpose system designed for a variety of workloads. All nodes in a sub-cluster (“partition”) are interconnected using Mellanox Infiniband (at least QDR) non-blocking fat tree network. We use SLURM as the job scheduler.
- By policy, compute nodes cannot access the internet. If you need an exception to this rule, contact cluster support with a reason, information about the IP address (must belong to the university network!), port number(s) and protocol(s) needed as well as the duration and a contact person.
- However, the compute nodes have access to cloud storage systems provides by LUIS. For detailed information please refer to the Rclone usage instructions.
- You will notice that the columns “(useable) Memory/Node (MB)” and “Memory Total (GB)” differ slightly, which takes into account the difference of total physical memory per node vs. the memory configured in the batch scheduler SLURM avilable to jobs. The latter number is smaller since the operating system needs memory, too. If you want to autoritatively find out the maximum allocateable memory per node in SLURM, use the command `scontrol show nodes <nodename>`, e.g. “scontrol show nodes amo-n001” for a node in the Amo partition, and look for the “mem=” parameter.
- To avoid unbalanced node allocations, there is another limit, which is the maximum memory that may be requested per cpu core. The scheduler automatically adjusts your ressource request if you request more memory per core than what is configured, resulting in more cores being requested. That mechanism avoids having nodes with no memory but some cores left that would effectively be unuseable. To see how much memory is allocateable per core, use the command `scontrol show partition <partitionname>`, e.g. `scontrol show partition amo` and look for the “MaxMemPerNode=” parameter. For example, if you were to request a job using 4 cores and 40 GB of total memory on an Amo node, the scheduler would change this to request 8 cores, since the configuration limits memory requests to 5120 MB/core and 8*5120 MB = 40.960 MB to keep the allocated cores/memory balanced.
- The line “FCH” in this table aggregates the nodes we run for various institutes of the LUH under the conditions of the service "Forschungsclusterhousing". They contribute significant additional power to the cluster, mostly during the night and over the weekend, but are usually reserved exclusively for institute accounts on week days. Your jobs have a chance of running in the night when they request less than 12 hours of walltime, or during weekends, for jobs that request less than 60 hours.

Cluster	Nodes	CPUs	Cores/Node	Cores Total	(useable) Memory /Node (MB)	Memory Total (GB)	Gflops /Core [1)]	Local Disk /Node (GB)	Partition [2)]
Amo	80	2x Intel Cascade Lake Xeon Gold 6230N (20-core, 2.3GHz, 30MB Cache, 125W)	40	3200	180.000	15360	75	400 (SATA SSD)	amo
Dumbo	18	4x Intel(R) IvyBridge Xeon E5-4650 v2 (10-core, 2.40 GHz, 25MB Cache, 95W)	40	720	500.000	9216	19	17000 (SAS HDD)	dumbo
Haku	20	2x Intel Broadwell Xeon E5-2620 v4 (8-core, 2.10GHz, 20MB Cache, 85W)	16	320	60.000	1280	34	80 (SATA SSD)	haku
Lena	80	2x Intel Haswell Xeon E5-2630 v3 (8-core, 2.40GHz, 20MB Cache, 85W)	16	1280	60.000	5120	38	180 (SATA SSD)	lena
Taurus	24	2x Intel Skylake Xeon Gold 6130 (16-core, 2.10GHz, 22 MB Cache, 125W)	32	768	120.000	3072	67	500 (SAS HDD)	taurus

Cluster	Nodes	CPUs	Cores/Node	Cores Total	(useable) Memory /Node (MB)	Memory Total (GB)	Gflops /Core [1]	Local Disk /Node (GB)	Partition [2]
SMP	9	2x AMD EPYC 9534 (64-core, 2.45GHz, 256MB Cache, 280W)	128	1152	1.024.000	9216	40	800 (NVMe)	smp
	2	2x AMD EPYC 9354 (32-core, 3.25GHz, 256MB Cache, 280W)	64	128	1.020.000	2048	52	3600 (NVMe)	helena
GPU	4	2x Intel Xeon Gold 6230 CPU 2x NVIDIA Tesla V100 16 GB GPU	CPU: 40 GPU: 2×5120	CPU: 160 GPU: 40960	CPU: 125.000	CPU: 512 GPU: 128	—	300 (SATA SSD)	gpu
	3	2x Intel Xeon Gold 6342 CPU 2x NVIDIA A100 80GB GPU	CPU: 48	CPU: 288	CPU: 1.025.000	CPU: 3072	—	3500 (NVMe)	
	4	2x AMD EPYC 9555 CPU 4x NVIDIA H200 141GB GPU	CPU: 128	CPU: 512	CPU: 1.150.000	CPU: 4490	—	5900 (NVMe)	
FCH	various partitions		12-128	~9000	—	[3]	—	—	—

1)

Performance values are theoretical

2)

See section about SLURM usage

3)

This line aggregates all the partitions of institutes participating in the FCH service; there is no partition called FCH. For details run the command `clusterinfo`