



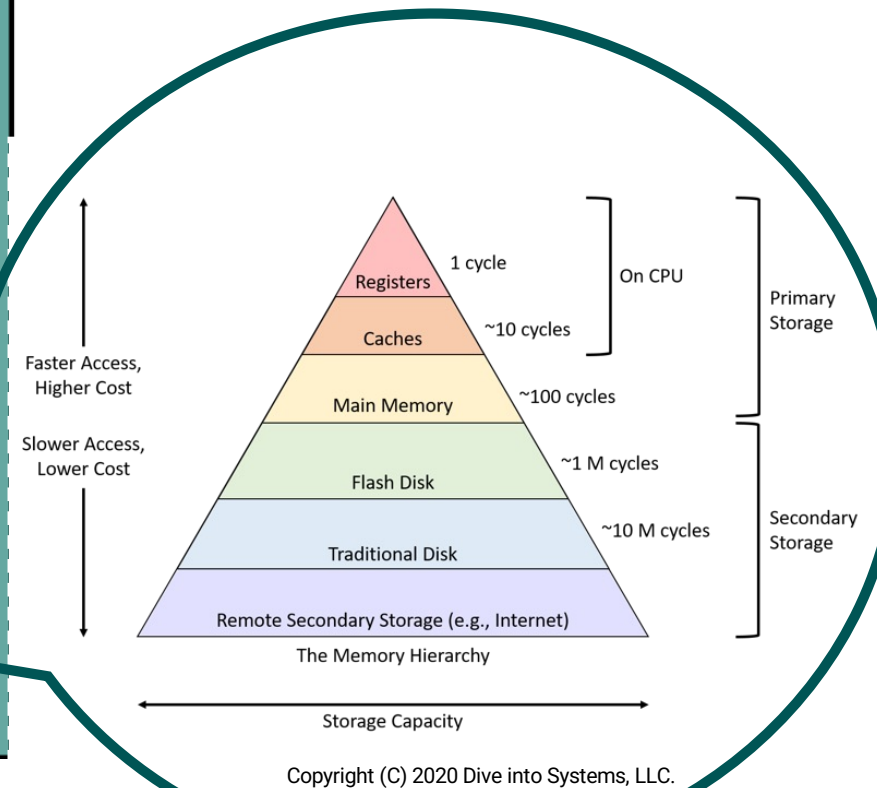
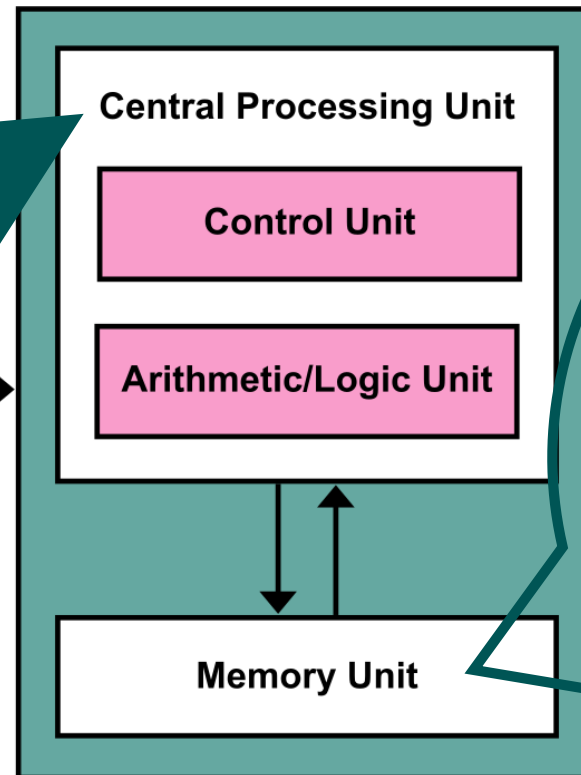
**VORBESPRECHUNG
SEMINAR ON EFFICIENT PROGRAMMING OF HPC
SYSTEMS - FRAMEWORKS AND ALGORITHMS (IN2107)**

Prof. Dr. Erwin Laure



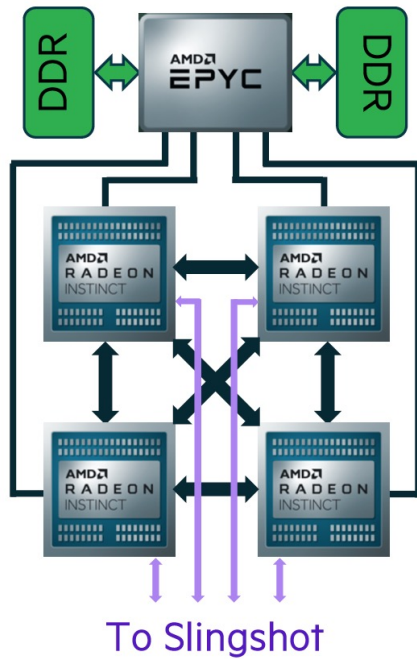
FROM SIMPLE VON NEUMANN ARCHITECTURES TO MODERN HPC SYSTEMS

- **Multi-Core**
 - E.g. 128-core AMD
- **Lots of Optimizations**
 - Pre-fetch
 - Branch prediction
 - FMA
 - Vector
 - Etc.
- **Other features**
 - Encryption
 - Viz
 - Etc.



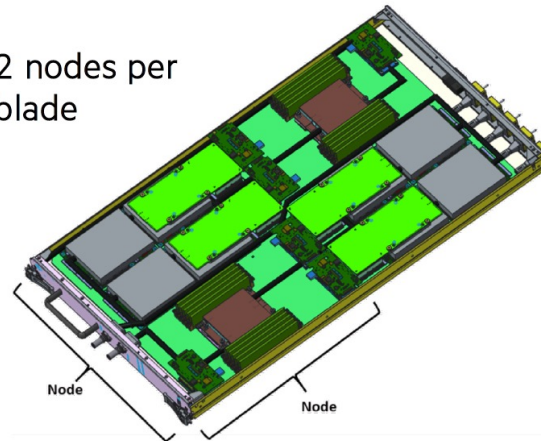


AND THEN WE ALSO ADD ACCELERATORS (GPUS)

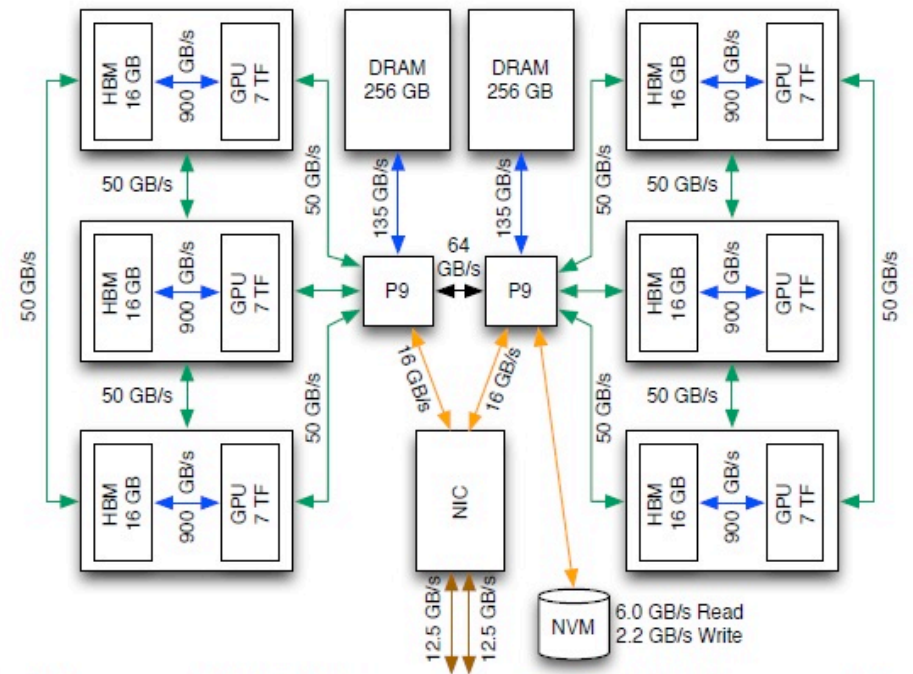


AMD GPU
(ORNL)

2 nodes per blade



COPYRIGHT 2020 HPE

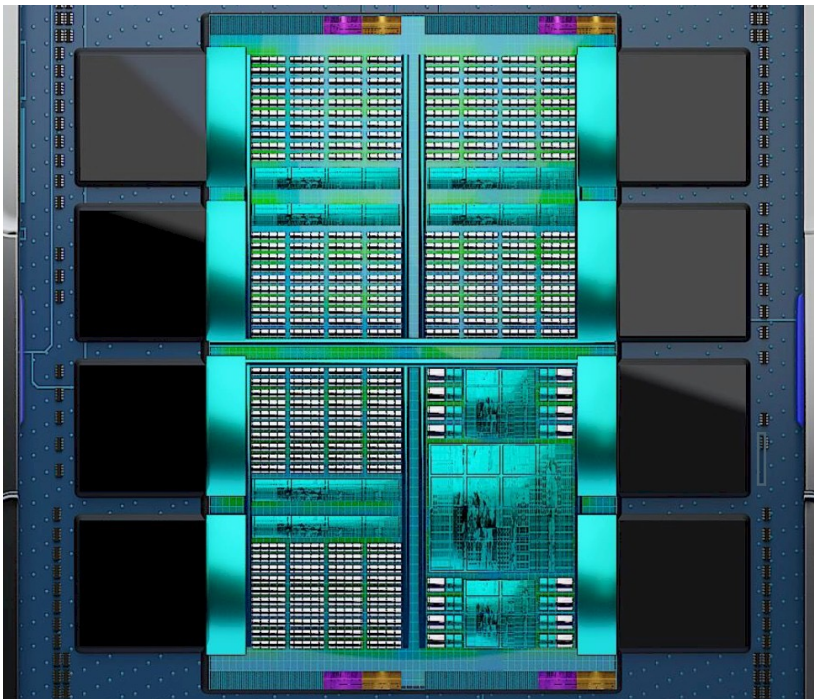


TF	42 TF (6x7 TF)	↔	HBM/DRAM Bus (aggregate B/W)
HBM	96 GB (6x16 GB)	↕	NVLink
DRAM	512 GB (2x16x16 GB)	↔	X-Bus (SMP)
NET	25 GB/s (2x12.5 GB/s)	↔	PCIe Gen4
MMsg/s	83	↔	EDR IB

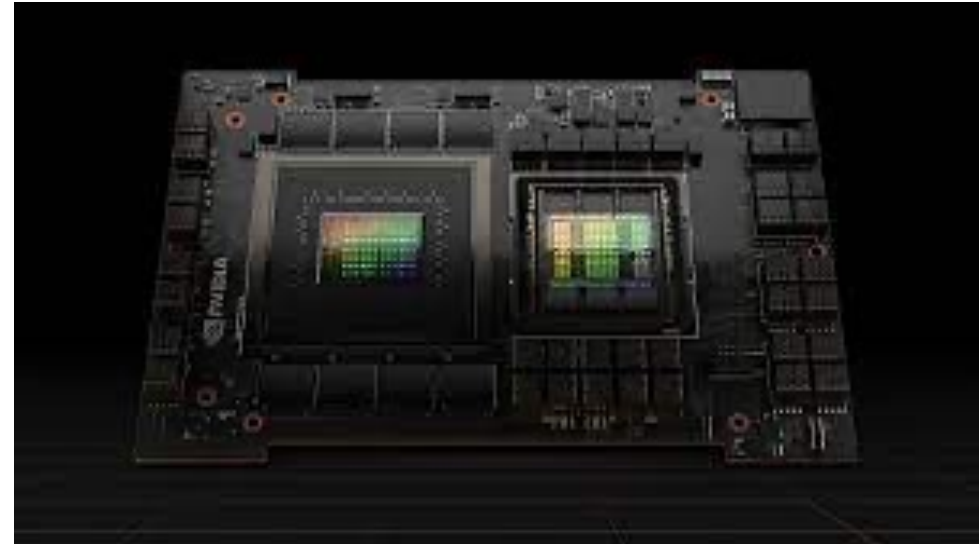
HBM & DRAM speeds are aggregate (Read+Write).
All other speeds (X-Bus, NVLink, PCIe, IB) are bi-directional.



NEW GPU-CONCEPTS MIGHT BE EASIER TO USE (CACHE COHERENT DESIGN)



AMD MI300A



Nvidia Grace-Hopper



AND USE MANY, REALLY MANY OF THESE NODES

- **El Capitan Supercomputer @ LLNL:**

- 11.136 nodes
- 1,74 EF performance
- 30 MW power consumption
- in total over 11 M cores (mostly GPU – MI300A)





(SOME) CHALLENGES IN PROGRAMMING THESE SYSTEMS

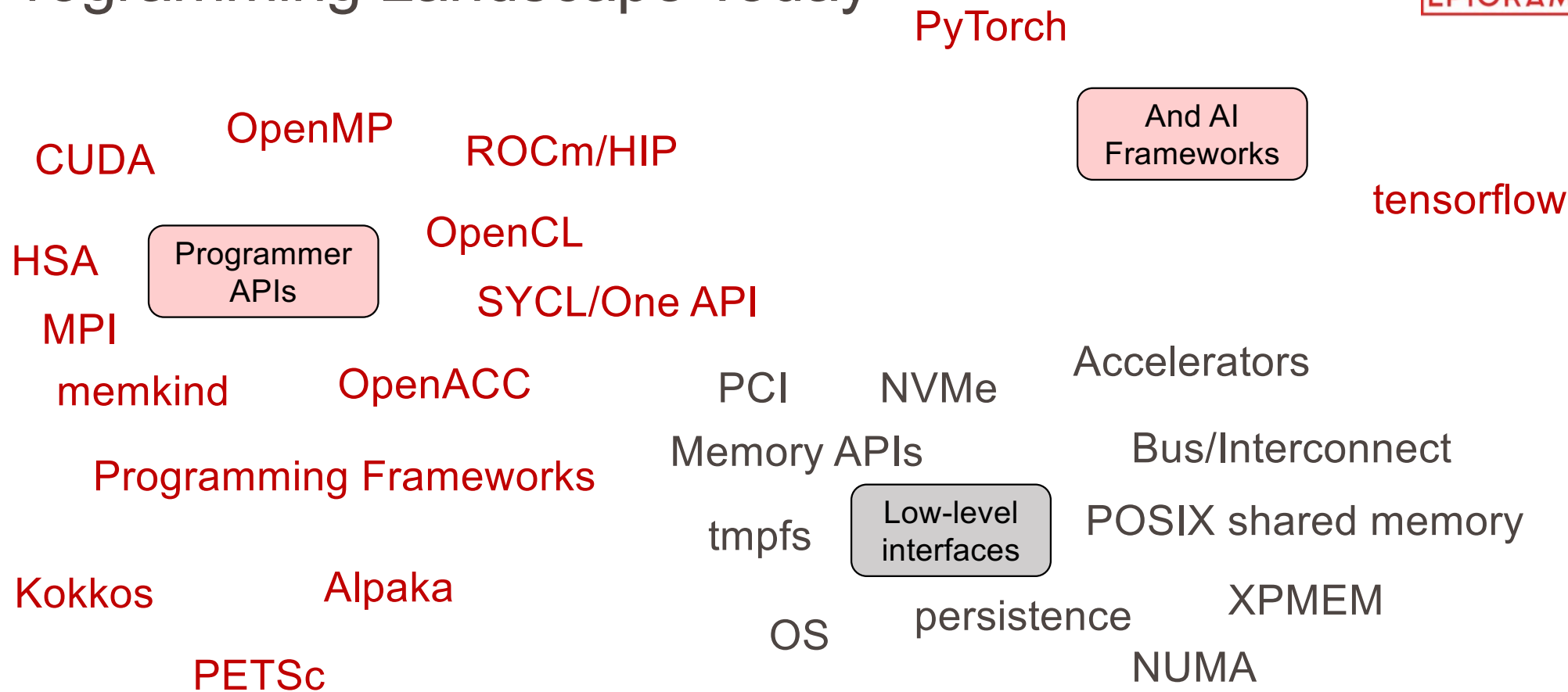
- **Level of parallelism**
 - $O(10^9)$ FPUs
- **Hardware heterogeneity**
 - CPUs, GPUs, other
 - HBM, NVMe, object store
- **Programming/Performance Portability**
- **Novel numerical/methodological approaches**



THE GOOD OLD TIMES

- **Programms written in Fortran (or C/C++)**
- **MPI (Message Passing Interface) for moving data across distributed memory**
- **OpenMP for expressing parallelism on shared memory**

Programming Landscape Today





GOALS OF THE SEMINAR

- **Investigate techniques, frameworks, algorithms to efficiently program such systems**
 - Focus on heterogeneous architectures (GPUs, shared/distributed memory)
- **Examples:**
 - High-level frameworks (Kokkos, Alpaka, Cabana, PETSc, etc.)
 - Numerical libraries (SLATE, Ginkgo, heFFTe, etc.)
 - Mixed-precision and use of non-IEEE data formats
 - Data structures and layouts (AoS, SoA, AoSoA)
 - Adaptive Mesh Refinement (AMReX, p4est, etc.)
 - Adaptive (task) Parallelism (HPX, StarPU, Charm++, etc.)
 - Frameworks for AI (pytorch, tensorflow, etc.)



SEMINAR ORGANIZATION

- **Kick-off meeting 6. May (16-17:30)**
 - Final definition and selection of topics
- **Seminar paper (6-8 pages)**
 - Literature study (scientific papers! Min 3-4)
 - Main concepts (pros & cons) plus (where possible) experiences from real applications
 - Peer reviewed by seminar participants
 - first week of July
- **Presentation (~15 mins)**
 - Workshop July 16th (whole day)
- **Tutors will help in case of questions/problems**
 - Provide help at all stages
 - Review paper/presentation drafts
 - Mandatory to discuss concepts with them
- **Grading**
 - 40% paper, 40% presentation, 20% review
 - All needs to be positive
- **Prerequisites**
 - Understanding of parallel programming (e.g. IN2147)