Data Management@HLRS

Thomas Bönisch

Outline

- HLRS
- What our users do
- What we currently provide
- What our users (really) want
- What we (the HPC community) plan to provide
- What we probably should and potentially can provide

Outline

- HLRS
- What our users do
- What we currently provide
- What our users (really) want
- What we (the HPC community) plan to provide
- What we probably should and potentially can provide

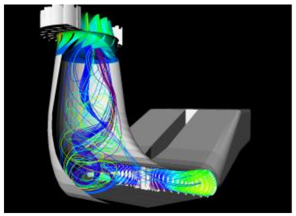
The High Performance Computing Center Stuttgart (HLRS)

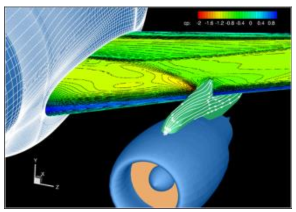
- Central Unit of Universität Stuttgart
  - Supercomputing since 1968
- 1st German National Supercomputing Center
  - Founded 1996
  - service for German researchers
- Gauss Center for Supercomputing
  - Founded 2007, Partners: Jülich and Munich
- Open for European users since 2004
- Partner for German industry

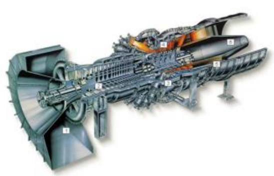
Outline

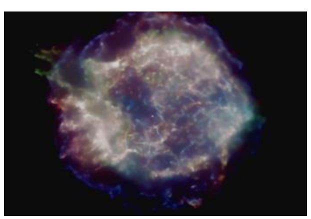
- HLRS
- What our users do
- What we currently provide
- What our users (really) want
- What we (the HPC community) plan to provide
- What we probably should and potentially can provide

Main Areas of Users’ Research

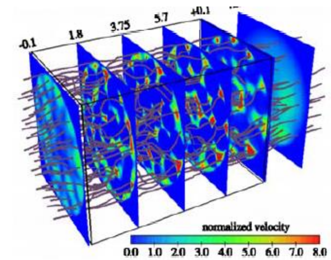


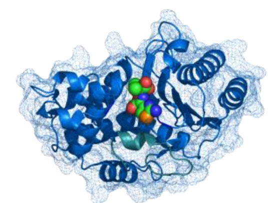


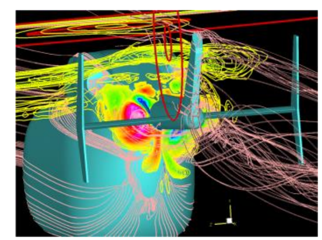




- Aeroacoustics
- Aerodynamics
- Astrophysics
- Bioinformatics
- Combustion
- Fluid-Structure Interaction
- Helicopter Aerodynamics
- Meterology
- Medical Imaging
- Nanotechnology
- Solid State Physics
- Turbo Machinery
- Turbulence Phenomena





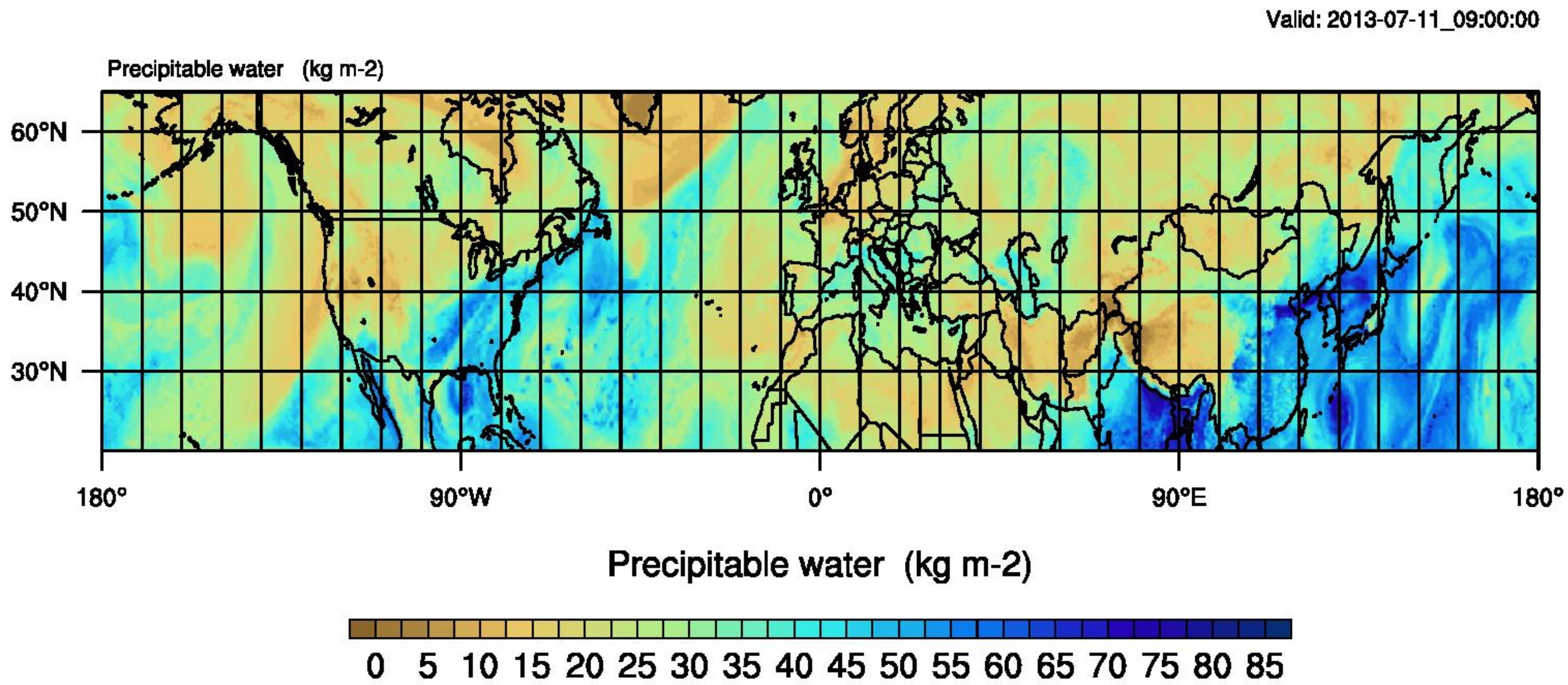


Convection permitting Channel Simulation

- Institut für Physik und Meteorologie, Universität Hohenheim
  - Wulfmeyer, Warrach-Sagi, Schwitalla

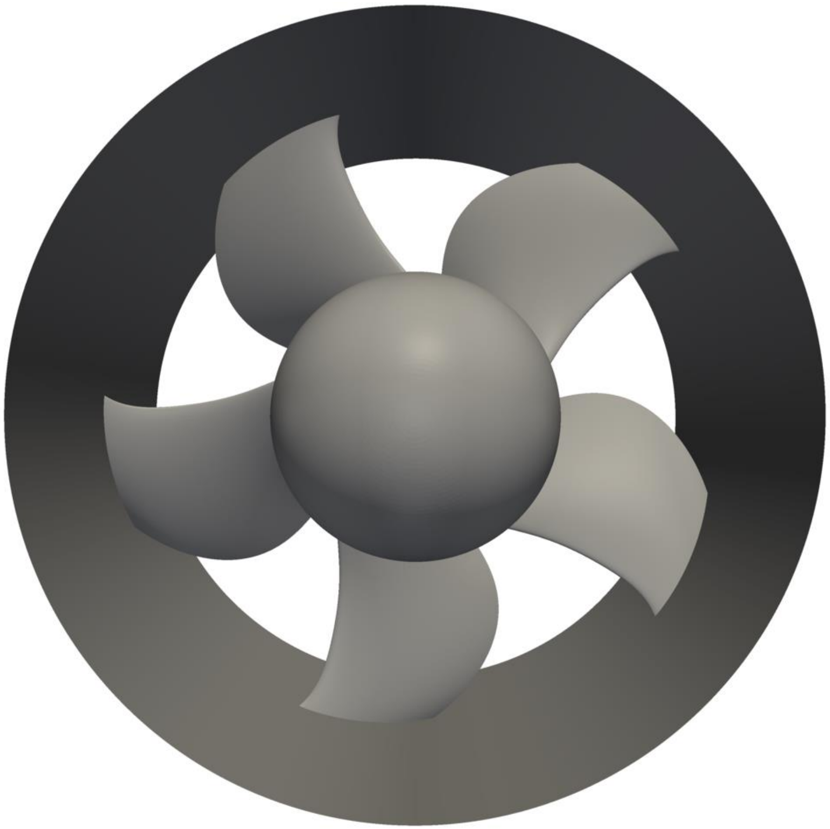
- Vertically integrated water vapor nicely shows the fine scale structure of the atmosphere.
- Visible is the Monsoon circulation over India, Typhoon Soulik close to Taiwan and a tropical depression in the Gulf of Mexico.
- The sharp gradient of moist air masses over the North Atlantic is also visible. Low pressure systems influencing Europe are developing along this line.

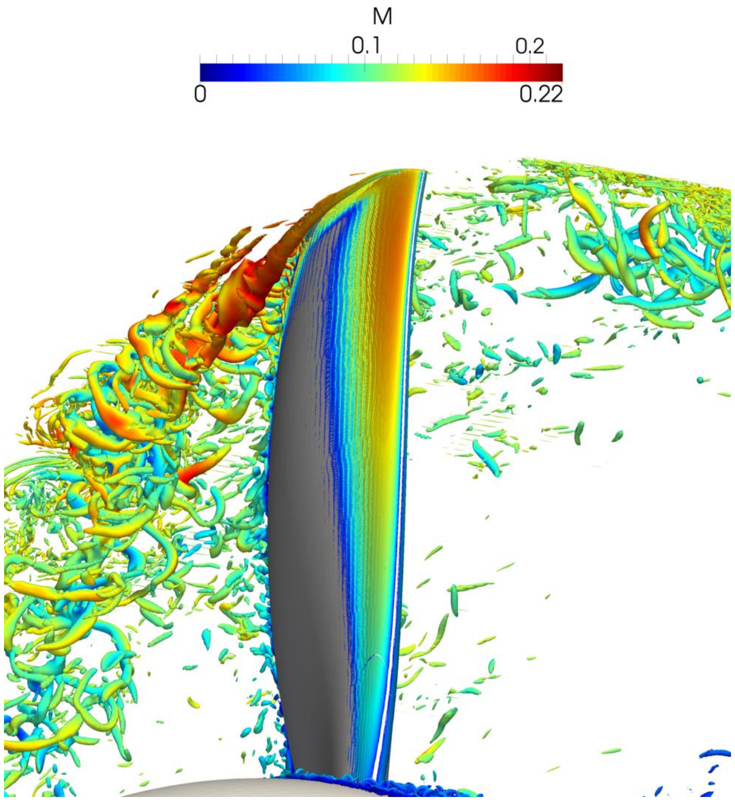
Convection permitting Channel Simulation



- WRF model, 3.3 km resolution
- 3500 nodes=84000 cores; 330 TB data; 84 system hours

Prediction of the turbulent flow field around an axial fan

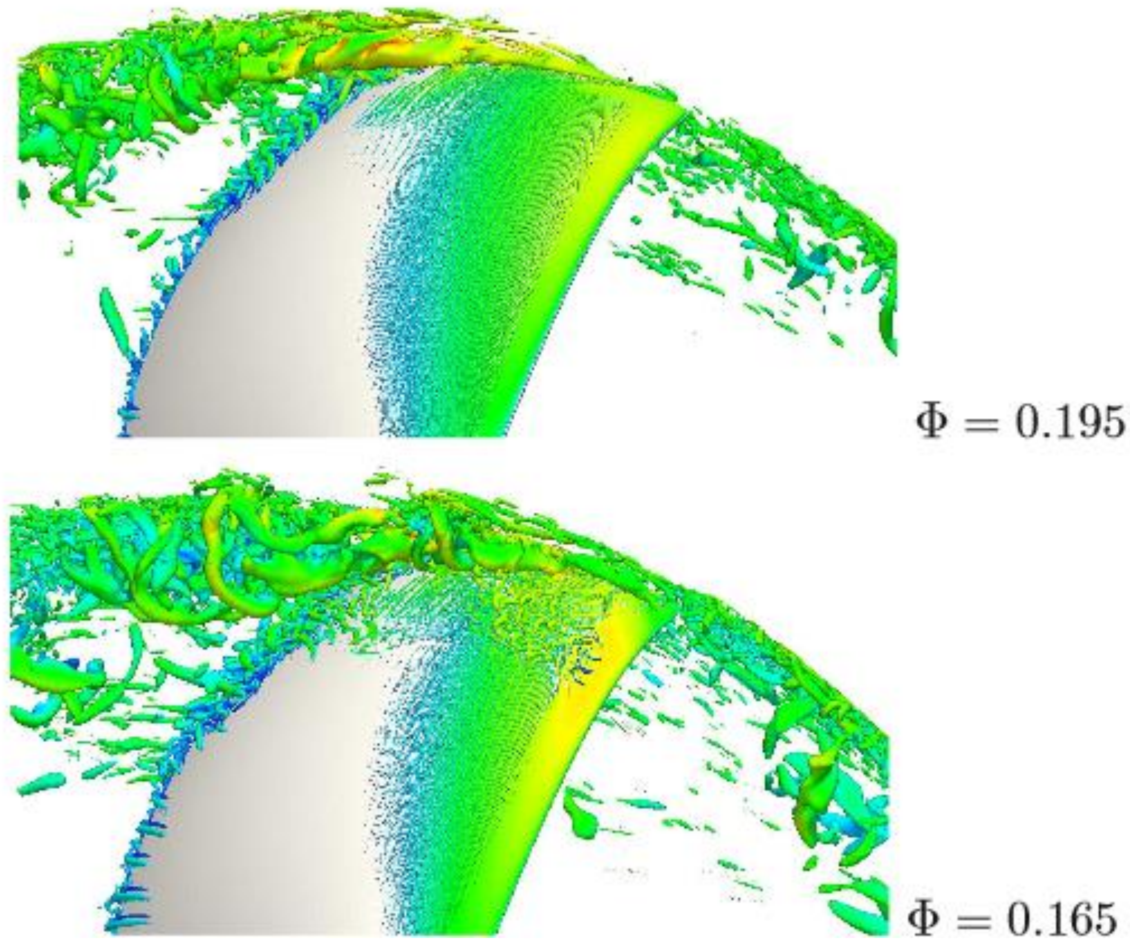




Example: Courtesy of M.Meinke, AIA, RWTH Aachen

Example: Flow around axial fan

- 1 billion cell mesh
- 100 TB of result data
- Statistical analysis
- New methods to detect structures within turbulence
- 1PB data sets foreseeable



Example: Courtesy of M.Meinke, AIA, RWTH Aachen

Outline

- HLRS
- What our users do
- What we currently provide
- What our users (really) want
- What we (the HPC community) plan to provide
- What we probably should and potentially can provide

Hazel Hen

- Cray XC 40
- Performance

  Peak:
  7.42 PetaFlops
  Linpack:
  5.64 PetaFlops
  HPCG:
  138 TeraFlops
  HPCG/Linpack
  2.4%

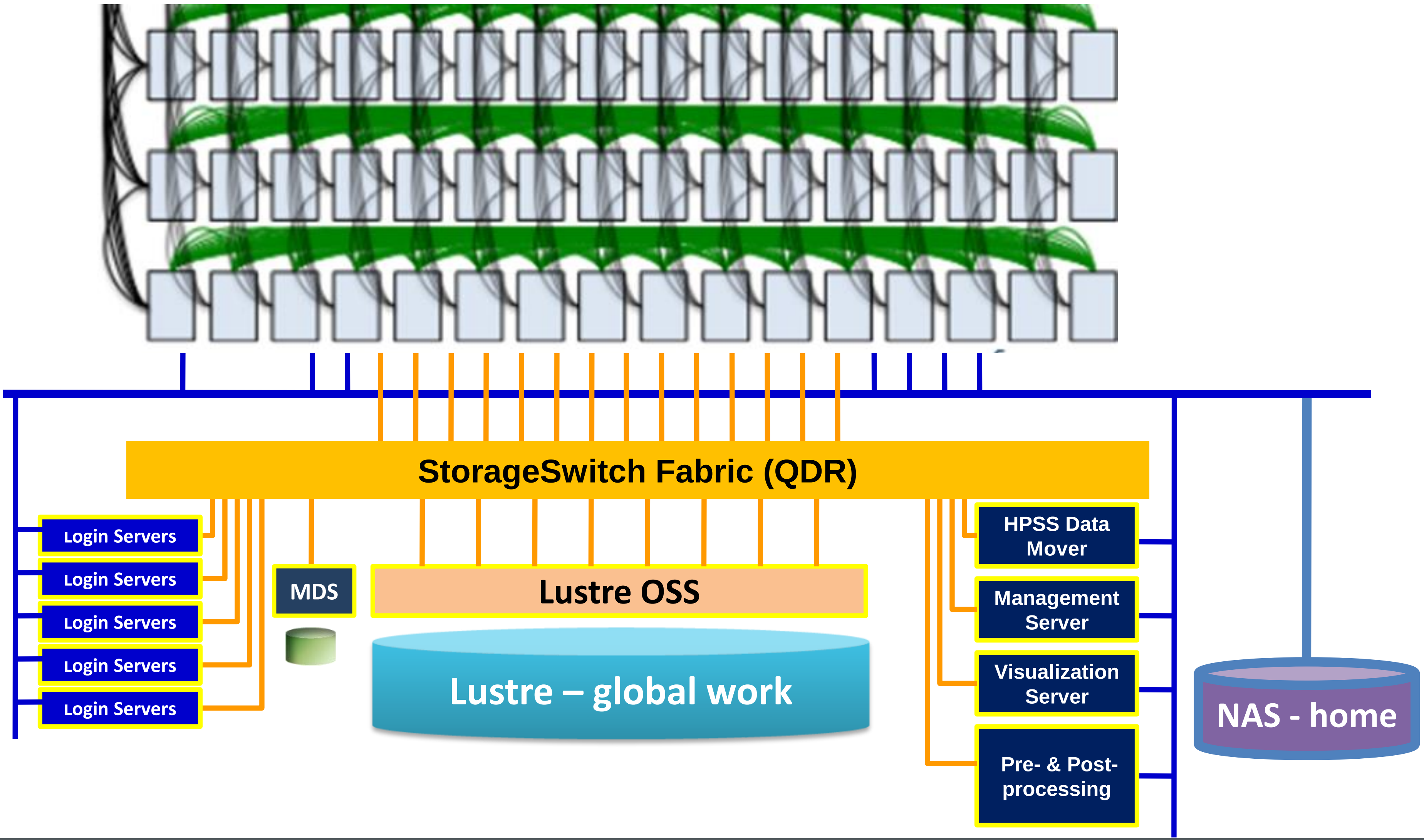
HPCG (Nov.2015): Nr. 1 in Europe

HLRS Phase II – Hazel Hen

- Cray XC 40
- Predesessor System homogenously integrated
- Configuration:
  - Peak Performace ~7.42 Petaflops
  - 7712 nodes
  - Each Node has 2 sockets
    - Intel Xeon E5-2680v3 (Haswell@ 2.5GHz 12 Cores each) leading to 185,088 cores
  - 128 GB main memory per node (5.3 GB/core) → 965 TB in total
  - Aries network
  - 12PB storage capacity @ ~ 350GB/s IO bandwidth
  - External Access Nodes, Pre- & Postprocessing Nodes, Remote Visualization Nodes
  - ~3MW maximal power consumption

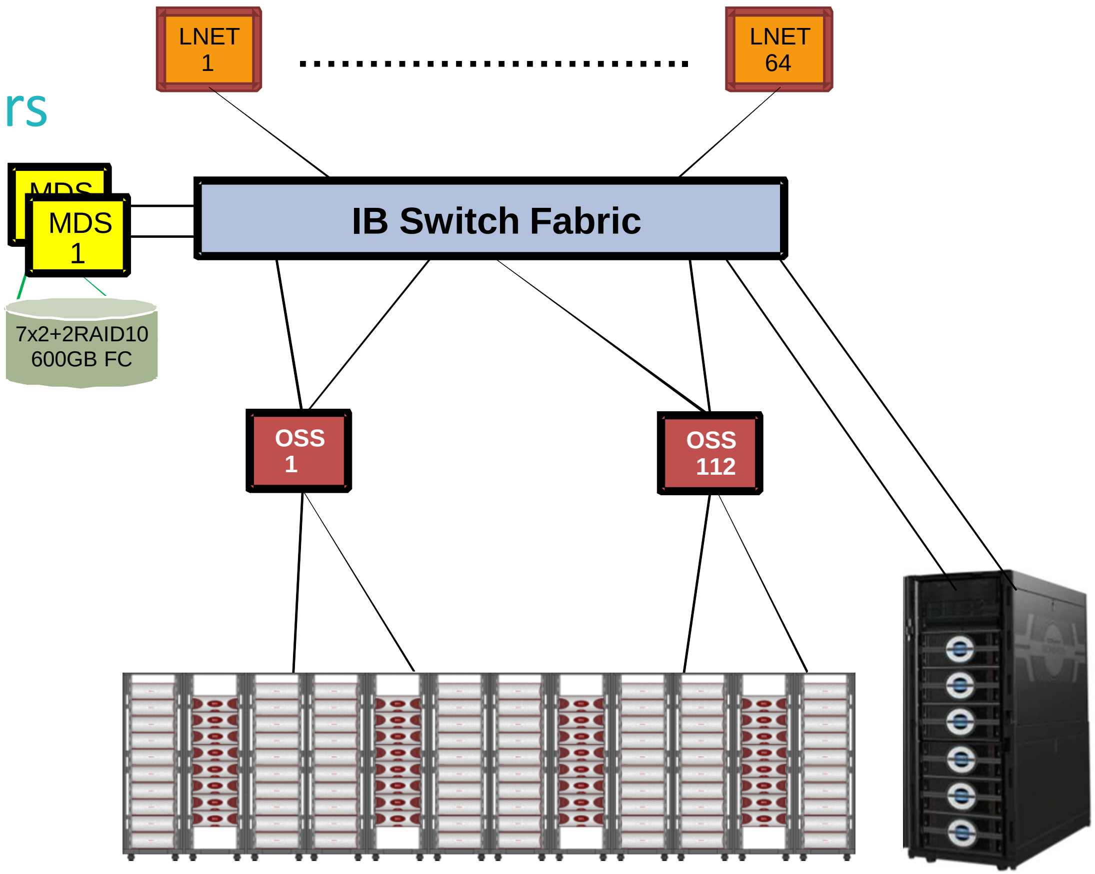
Conceptual Architecture

HLRS



I/O architecture

- Hardware
  - 7+7 MDS/MGS Servers
  - 112 OSS Servers
  - 22 Dual RAID controllers
  - Lustre „appliance“
    - MDS + 13 SSUs
  - 8480 Hard disks



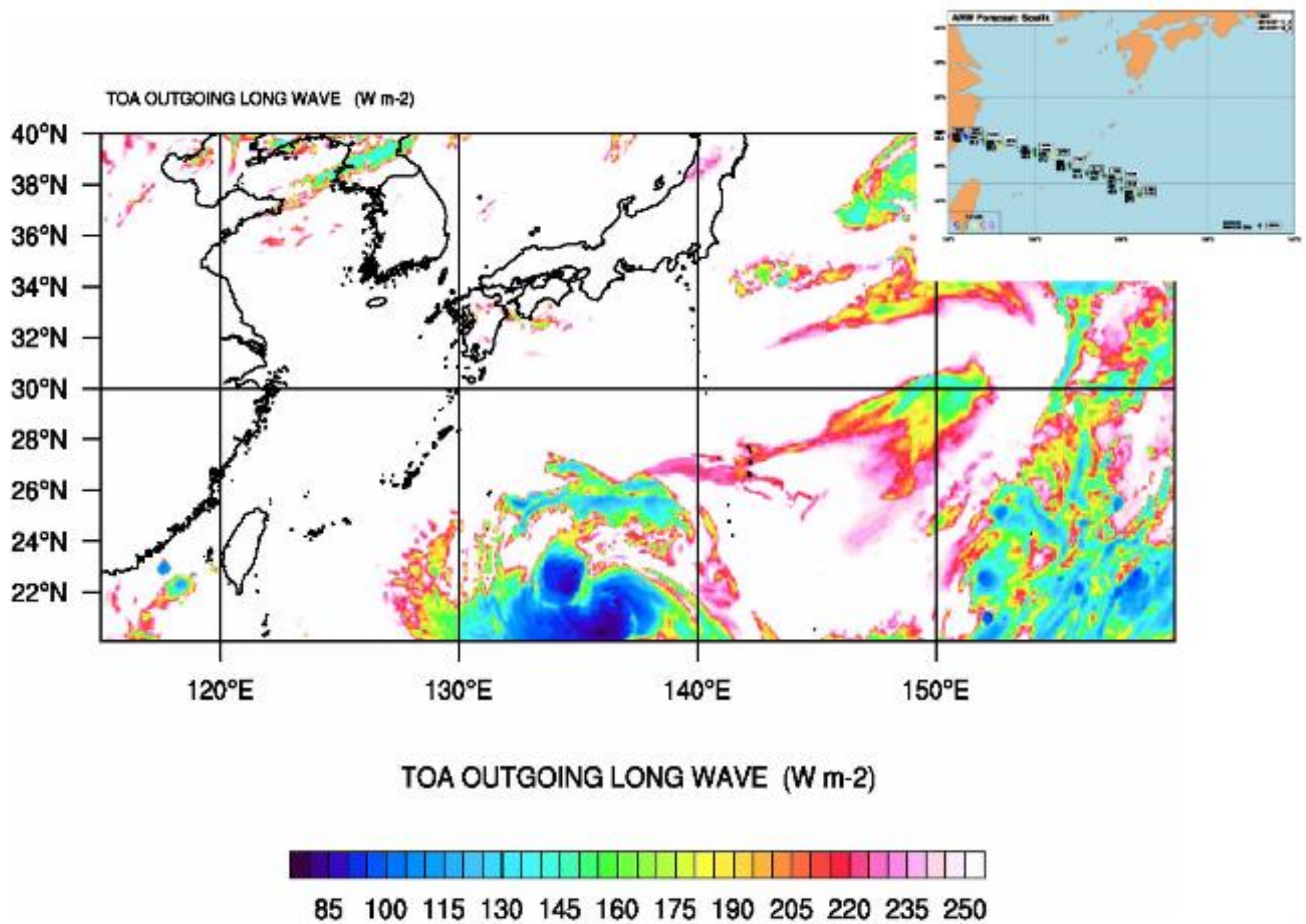
- ~ 12 PB Storage

  ~ 370 GB/s measured total BW

Usage numbers & issues

- One file system for general usage
  - 3.5 mio files
  - ~ 500 TB usage (out of ~700 TB)
- Other file systems by invitation only
  - Power users (capacity, throughput)
  - Industry
- Issues
  - Small files
  - I/O performance of application is rarely looked into

Typhoon Soulik



UNIVERSITY OF HOHENHEIM

The issues

- I/O was and is a problem of this code
- 1st shot:
  - 1 GB/s throughput
- After optimization
  - 7.5 GB/s throughput
  - 2 days of calculation.
  - 1.5 days of I/O
- File System potential: 75 GB/s (measured !!!)
- Software: netcdf4

Outline

- HLRS
- What our users do
- What we currently provide
- What our users (really) want
- What we (the HPC community) plan to provide
- What we probably should and potentially can provide

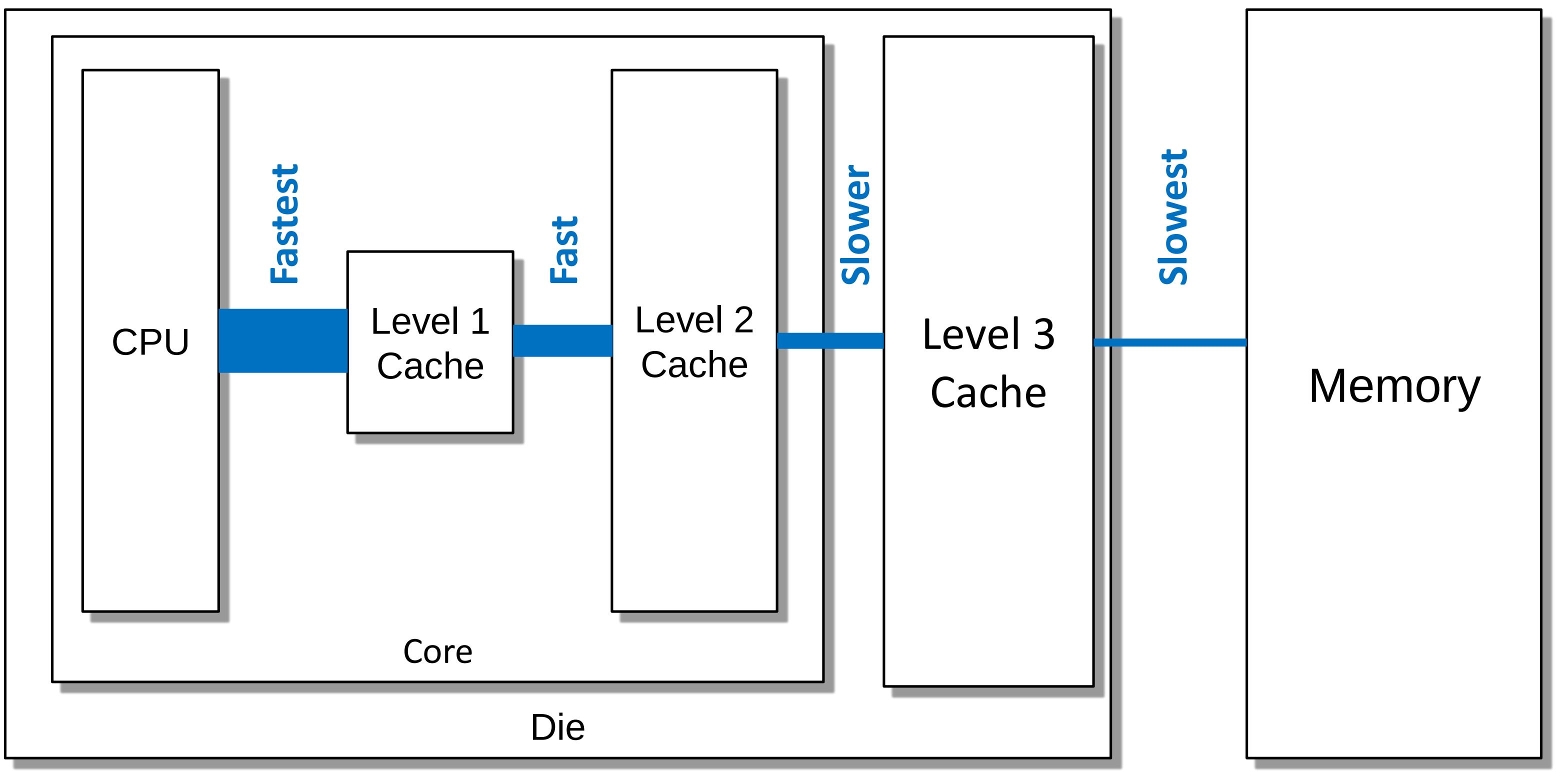
Users say: „I just want to …“

- do my science
- run my application
- Do not care (too much) about the system
- Not interested in HPC in principle

Outline

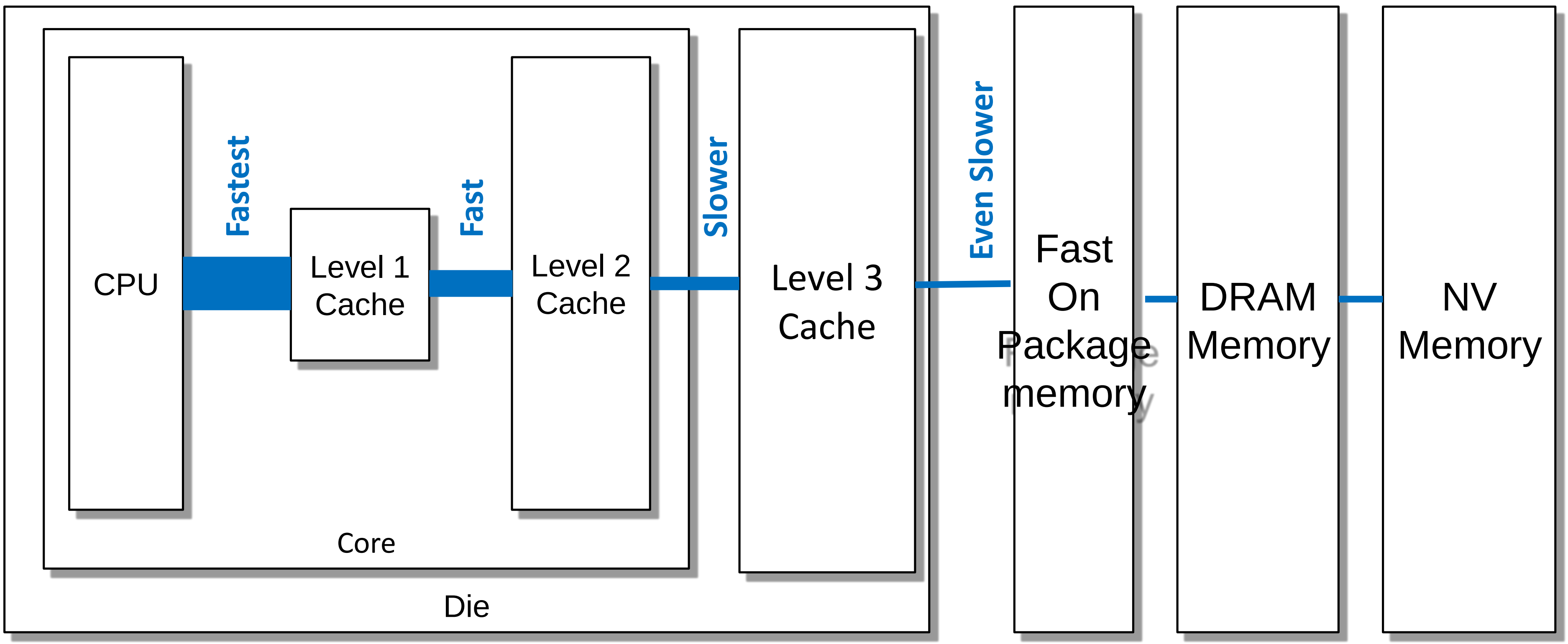
- HLRS
- What our users do
- What we currently provide
- What our users (really) want
- What we (the HPC (I/O) community) plan to provide
- What we probably should and potentially can provide

Node view

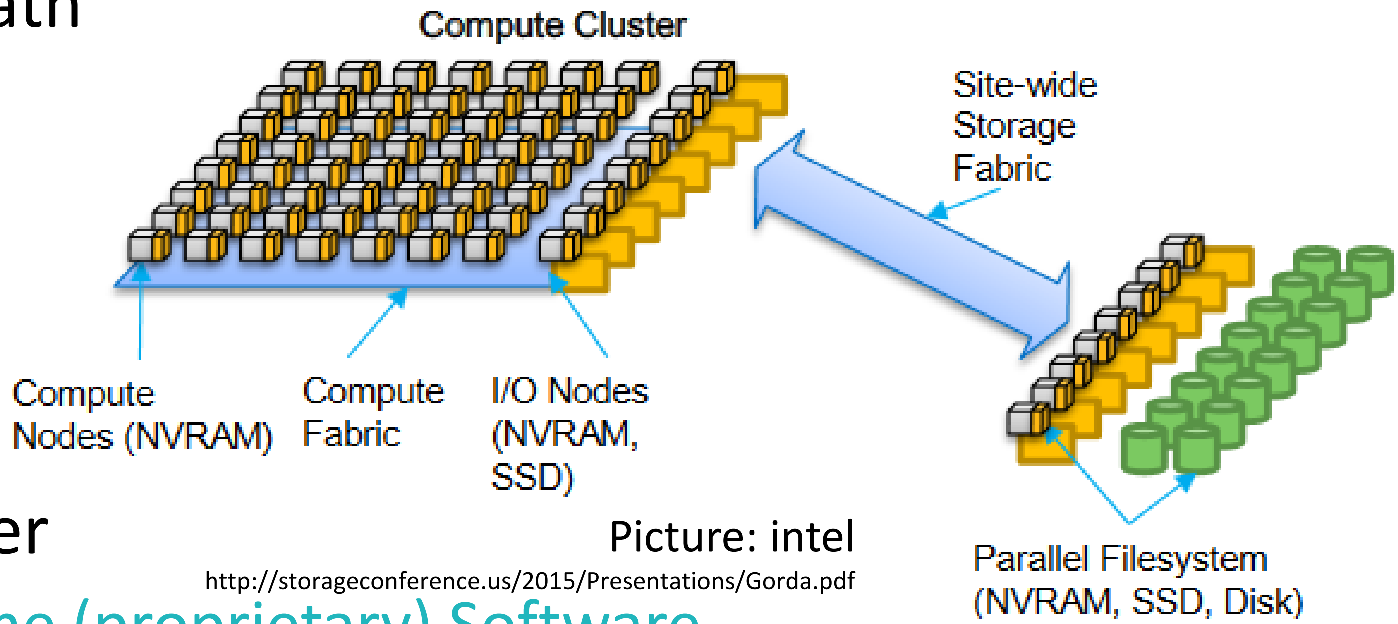


CPU
Fastest
Level 1 Cache
Fast
Level 2 Cache
Slower
Level 3 Cache
Slowest
Memory
Core
Die

Future node view



Future I/O Path



Picture: intel
http://storageconference.us/2015/Presentations/Gorda.pdf

- Burst Buffer
  - Plus some (proprietary) Software
- Flash Pools
  - Plus some (proprietary) Software
- Parallel File System
  - Plus some (proprietary) Interfaces to optimize

Software

- Vectorization
- Cache Optimization
- OpenMP (~300 pages)
- MPI (~800 pages)
- MPI-IO or HDF5 or NetCDF or … (??? Pages)
- I/O Optimization
  - Proprietary libraries
  - New nice libraries
  - Probably some directives
  - API of the FS

Outline

- HLRS
- What our users do
- What we currently provide
- What our users (really) want
- What we (the HPC community) plan to provide
- What we probably should and potentially can provide

Why not

- Use NV-Memory really as a persistent memory
  - Byte adressable (SCM → MCS, Memory Class Storage)
- Give it an easy Interface
  - like malloc, free, added information about persistency
- Allow for some structure and naming
  - Allow putting things logically together which belong together (e.g. like HDF5 structure does)
- Use todays storage as easy to use back end
  - Mainly automatic pre stage-in, and post stage-out

Prospects

- This could become a
  game changer
- Working methods will change
- Life for users will become (much) easier
- Life for admins, too
- Costs? (less HW in file system ←→ NVRAM costs)
- Quite some research and development necessary