Welcome to the PDC Summer School
Introduction to High Performance Computing

PDC Center for
High Performance Computing

Erwin Laure
Director PDC-HPC
Co-Chair PDC Summer School
13. Aug. 2018

Goal of the School

PDC Center for
High Performance Computing

- Introduce you to high-performance computing
- Give you practical knowledge to apply to your own work
- Discuss practical applications and look into the future

What is expected of you?

- Attend the classes
  - Sign up attendance sheet
- Read the handouts and material on the web
- Get lab attendance form signed every lab session
- Submit a project abstract before you leave
  - Then complete the project for credits
- Fill in the online feedback form available on the agenda page:
  - http://agenda.albanova.se/conferenceDisplay.py?confId=6537

PDC Center for High Performance Computing

3

PDC HPC Summer Course DD3258/DD2358 2018
7.5 ETCS

Attendance, two weeks Lectures and labs:
Get Lab attendance sheet signed !

Project, finished Fall '18:

Grade: Grad.: P,F Undergrad. : E... A

Support:
Lab assistant, Project advisor, Examiner

Project:
For some application and HPC architecture of your choice:
- Develop efficient program for non-trivial problem
- Demonstrate and report how efficient it is.

Expected work on the project is **3 weeks** of work ***incl.*** report writing
Deadline for reports: Nov 9, 2018.

4

The project is ***not*** about:

- *Substantial* development of *new* code.
- *Scientific results* obtained with code

So:

Prioritize measurements and analysis/interpretation!
Demonstrate use of tools (profiling, …) ,
and simple performance model.
NO TIME for development of new **significant** *code.*

Examples:

* Parallelize a code you know and/or work with; choose interesting part.
* Write a simple code for key algorithm of bigger solution process
* Write a simple code for a simple problem

5

Now – during lab-afternoons

• Discuss with instructors & course participants, form groups of size *G*.
• *Define project* and choose *tutor*: Michael, Thor, Roman, Stefano, ...
• Write very short synopsis, check with supervisor !
• Submit synopsis to *summer-info@pdc.kth.se* ***before end of the course***

Later -

• Start the work ***ASAP:***
• Finish the work; Get in touch with tutor !!
• Submit report to ***tutor***.

The report will be graded and sent back with comments; you may have to complete some parts and hand in again. ***We need email and paper mail address!***

- KTH students: LADOK
- Other students: Certificate will be sent to you

6

1. Develop initial version of program;
2. Develop approximate Performance model = theoretical prediction:
   time = *f*(problem size *N*, #processors *P*, problem partitioning parameters, …)
   Try to assess the *communication* and *computation* times separately.
3. *Measure* performance, e.g. $t = f(N,P,...)$, for different problem sizes, if relevant
   x = wall clock time start to finish, (*not* CPUtime), …

Size \ # proc	1	2	4 …	*n*
N_1	x	x	x	x
N_2	x	x	x	x
…				
N_M	x	x	x	x

4. If suitable, plot "speedup" and/or "efficiency", MFLOPS?, …
   - Make several measurements to discover variations – discuss sources of variability. (interactive nodes, dedicated,…)
   - Compare w. prediction; Interpret: ***Why these numbers?***
   - Identify "bottlenecks" by profiling tools; find remedy & make changes
   - Check improvement by measurements
   - Write report with description of problem, ***algorithm***, and design decisions, pertinent graphs of measurements and profiling, "before and after".

Single processor performance

Algorithm:
    BLAS etc. library
Memory hierarchy
    Disk - main - cache - register;
Organization of loops
    data layout (cache misses)
    index strides (-"-)
    "unrolling"
Compiler directives ("-O2")

Multi-processor performance

Algorithm: Communication !
    Latency vs. bandwidth
    # messages vs. size

Problem partitioning
    Load balancing

8

Other

- Group size G: $G = 2$ recommended.
- "Standard" grade C. A requires exceptional work
  Requirements for grade ≥ C increase with G.
- Proposed schedule
  - < 18-09-30 First iteration: status report, quick feedback from advisors
  - < 18-10-26 Second (final ?) iteration, results, quick feedback/grading
  - -------------- 18-11-09 --------------
  - > 18-12-7 … evaluation may take a while
  - ***> 2019-01-01 evaluation turnaround time may be very long***
- **Report:**
  - Background, formulas, relevant problem sizes, …:
  - Algorithm, parallelization principle,...
  - "Embarrassingly parallel" OK
  - Performance model and measurements.
  - Graphs, and textual description of what the graphs show, what we learn from them
  - ***Interpretation: WHY these numbers?***

9

Various Information

PDC Center for High Performance Computing

- Certificates will be issued to all successful students
- Tutors will be available for lab sessions
  - Ask them questions
  - But they will also ask you!
- Labs in groups of 2-3 people
- Door access code for lab room: **1425**
- Wireless
  - Eduroam
  - If you don't have eduroam you can use KTHOPEN
    - Passwords will be distributed as needed
- All material available via the course homepage
  http://agenda.albanova.se/conferenceDisplay.py?confId=6537

10

Accounts

PDC Center for
High Performance Computing

- If you do not have a PDC account yet, we will fix this tomorrow during the lab.
- **Tell us NOW!**
- **Have an ID handy**
- Bring your laptop to the session tomorrow afternoon so we can fix any last minute problems!!

11

Rooms

- Lectures in Q2
  - Exceptions:
    - Monday, August 20: Q36
    - Tuesday, August 21: Q34
    - Wednesday, August 22: Q33
    - Thursday, August 23: Q33
- Labs in
  - “Grön” and “Brun” (first week)
  - “Röd” and “Orange” (second week)

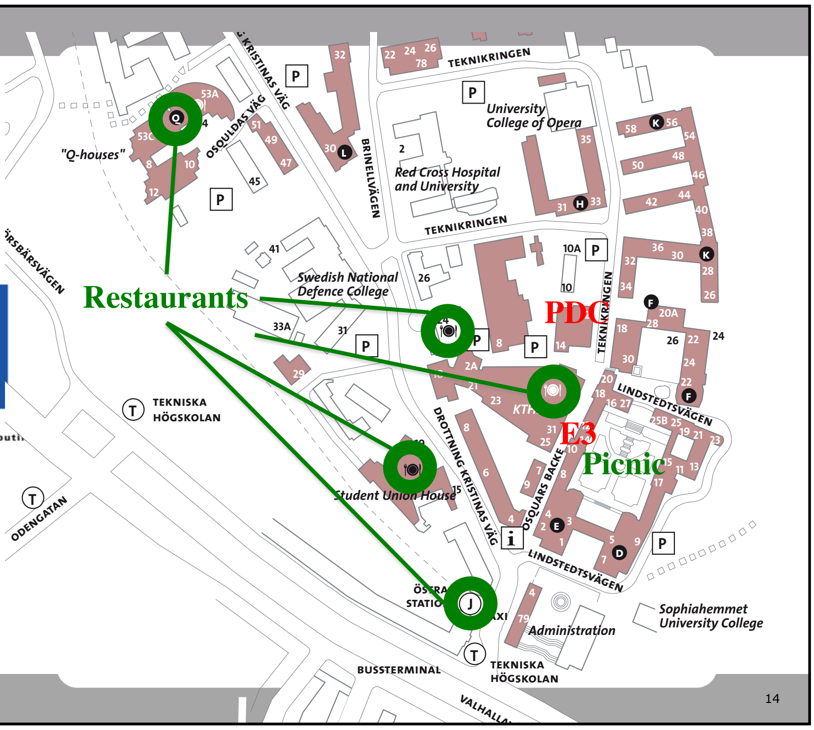
12

Social Program

- Get-together picnic this lunch at 12.15
- Possibility to see the PDC computer room tomorrow afternoon
- Summing-up dinner on Thursday, August 23, at 18:00, Cypern (http://restaurangcypern.se/), Valhallavägen 50

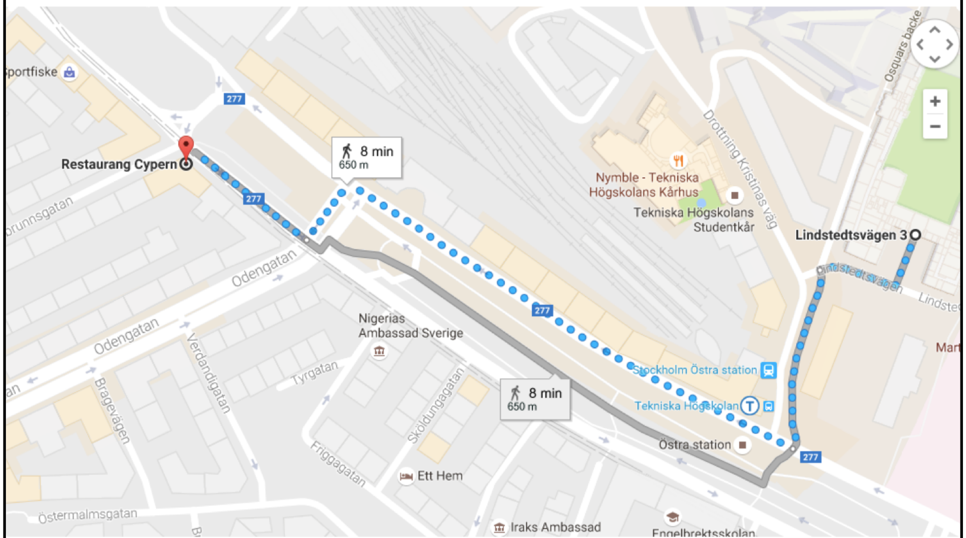
KTH VETENSKAP OCH KONST

PDC Center for High Performance Computing



KTH VETENSKAP OCH KONST

PDC Center for High Performance Computi



What is PDC

- A major Swedish Supercomputing Center supported by SNIC (Swedish National Infrastructure for Computing)
  - Supporting a wide variety of sciences
- Located at the Royal Institute of Technology (KTH), Stockholm
  - Associated to School of Electrical Engineering and Computer Science and communication (EECS)
- Summerschool run jointly with EECS
  - Since 1996

KTH VETENSKAP OCH KONST

PDC Center for High Performance Computing

PDC Center for
High Performance Computing

PDC's Mission

Research

Conduct world-class research and education in parallel and distributed computing methodologies and tools as part of CSC's HPCViz department

Infrastructure (PDC-HPC)

Operation of a world-class ICT infrastructure for Swedish research, including HPC and data services, with associated user support and training

17

PDC Center for
High Performance Computing

PDC HPC Infrastructure

System	System/Processor	TPP (TF)	Cores
Beskow	Cray XC40 Intel Haswell	2,430	67,456
Tegner	SuperMicro Intel Ivy Bridge & Haswell Nvidia K420 & K80	65 + GPU	1,800 + GPU

18

18

Heat Reuse Project

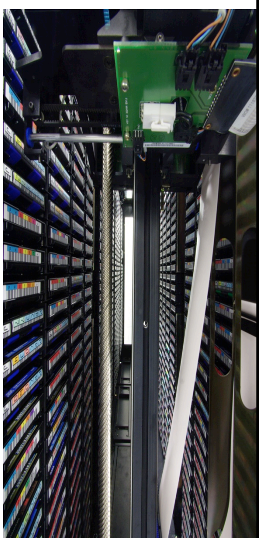
- Background: today around 1.3 MW used at PDC
- Project started 2009 to re-use this energy
- Goals:
  - -Save cooling water for PDC
  - -Save heating costs for KTH
  - -Save the environment
- Use district cooling pipes for heating when no cooling is required
- No heat pumps
- Starting with Cray
- First phase of Cray heats the KTH Chemistry building

KTH VETENSKAP OCH KONST

PDC Center for High Performance Computing

Storage

- ~20 TB disk
  - Accessible via AFS
- ~5000 TB disk
  - Currently attached to individual systems
  - Lustre parallel file system
- ~700 TB LCG disk
- IBM tape robot (~2900 slots, ~2.3 PB)
  - Accessible via HSM, TSM, and dCache

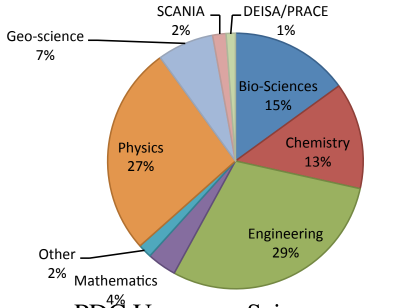


KTH VETENSKAP OCH KONST

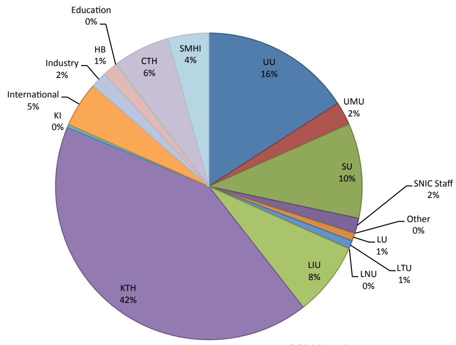
PDC Center for High Performance Computing

PDC Supporting Swedish Research (Data from 2011)

- About 2/3 of the total SNIC computational resources are operated by PDC
- In 2011 some 380 Million core hours have been used on PDC's SNIC machines, that equals to a usage of over 90% of the nominally available compute power



PDC Usage per Science
01/2011-11/2011



PDC Usage per Affiliation
01/2011-11/2011

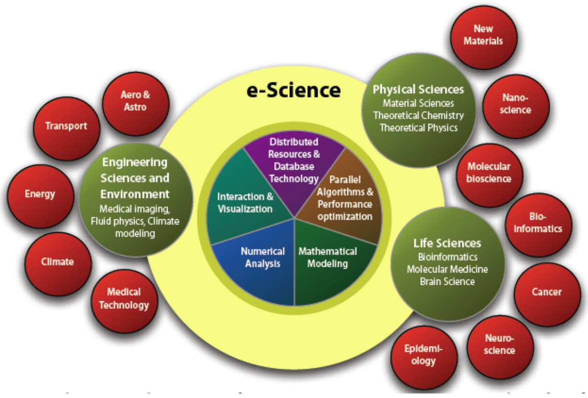
PDC Center for High Performance Computing

Software

- Commercial/Community codes
  - Gaussian
  - Jaguar
  - Gromacs
  - VASP
  - Fluent
  - Blast
  - Edge
  - Starcd
  - Dalton
  - Charmm
  - Numerical libraries
- User codes
  - CFD
  - Ab initio
  - Monte Carlo
  - Bioinformatics
- Programming
  - MPI, OpenMP
  - Fortran/C/C++ compilers
  - Tools
    - acumem
    - totalview
    - papi, papiex
    - jumpshot
    - hpctoolkit
    - Paraver
    - CrayPAT

SeRC - Swedish eScience Research Center

- Collaboration of KTH, SU, KI, and LiU



PDC Center for High Performance Computing

23

Other Activities

- PRACE - Partnership for Advanced Computing in Europe
  Design of future European supercomputing landscape
  - Prototype
    - Focus on energy efficiency and high density packing
  - Tier-1 system
    - Beskow
- EUDAT – European Data Infrastructure
- Advanced software support
  - SeRC
  - EPiGRAM-HS Exascale Programming Models
  - ExaFLOW Fet-HPC Project
  - BioExcel Centre of Excellence
  - Excellerat Centre of Excellence
- Visit program
  - HPC-Europa3

PDC Center for High Performance Computing

24

The PDC Summer School is part of

SeSE

Swedish e-Science Education

KTH VETENSKAP OCH KONST

PDC Center for High Performance Computing

25

What is Parallel Computing/High Performance Computing?

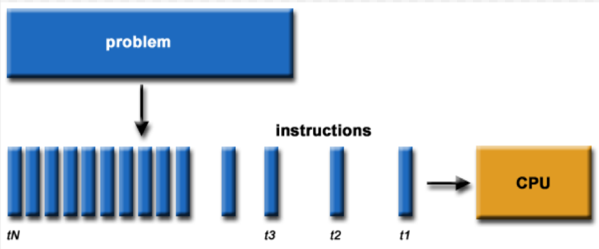
KTH VETENSKAP OCH KONST

PDC Center for High Performance Computing

26

What is Parallel Computing

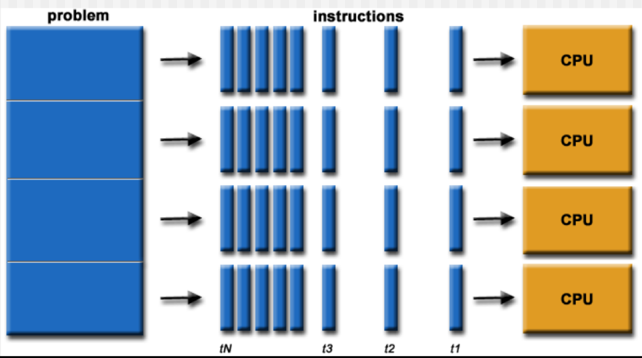
- Traditionally, software has been written for ***serial*** computation:
  - To be run on a single computer having a single Central Processing Unit (CPU);
  - A problem is broken into a discrete series of instructions.
  - Instructions are executed one after another.
  - Only one instruction may execute at any moment in time.



27

Parallel Computing

- In the simplest sense, ***parallel computing*** is the simultaneous use of multiple compute resources to solve a computational problem:
  - A problem is broken into discrete parts that can be solved concurrently
  - Each part is further broken down to a series of instructions



28

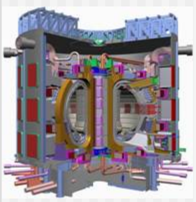
Solve a given problem in less time

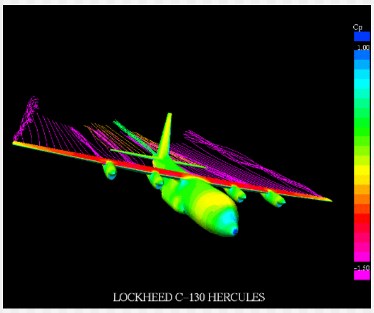
- Increase the productivity, save money

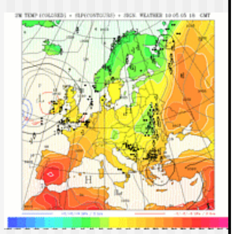
29

Solve a larger problem

- Many problems require more resources than available on a single computer (particularly memory)
  - “Grand Challenge" (en.wikipedia.org/wiki/Grand_Challenge) problems requiring PetaFLOPS and PetaBytes of computing resources.
  - Web search engines/databases processing millions of transactions per second







30

Parallelism on different levels

- CPU
  - Instruction level parallelism, pipelining
  - Vector unit
  - Multiple cores
    - Multiple threads or processes
- Computer
  - Multiple CPUs
  - Co-processors (GPUs, FPGAs, …)
- Network
  - Tightly integrated network of computers (supercomputer)
  - Loosely integrated network of computers (distributed computing)

31

FLOPS

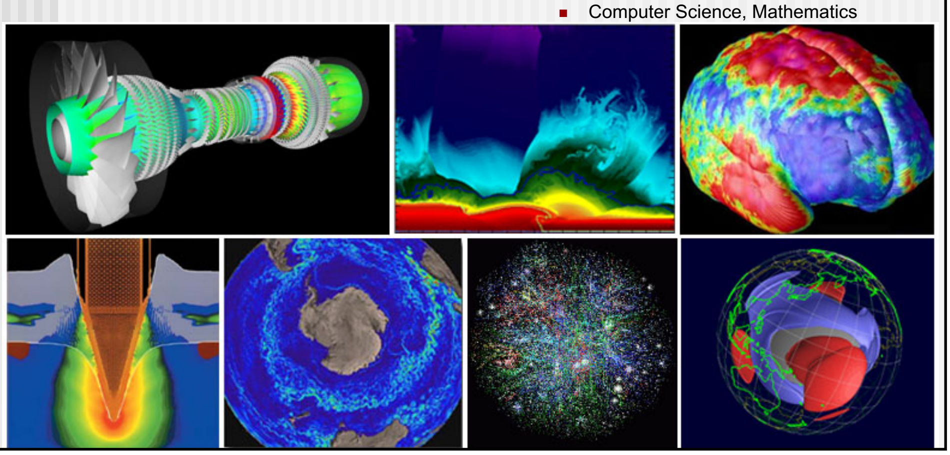
- FLoating Point Operations per Second
- Most commonly used performance indicator for parallel computers
- Typically measured using the Linpack benchmark
- Most useful for scientific applications
- Other benchmarks include SPEC, NAS, stream (memory)

Name	Flops
Yotta	10^{24}
Zetta	10^{21}
Exa	10^{18}
Peta	10^{15}
Tera	10^{12}
Giga	10^{9}
Mega	10^{6}

32

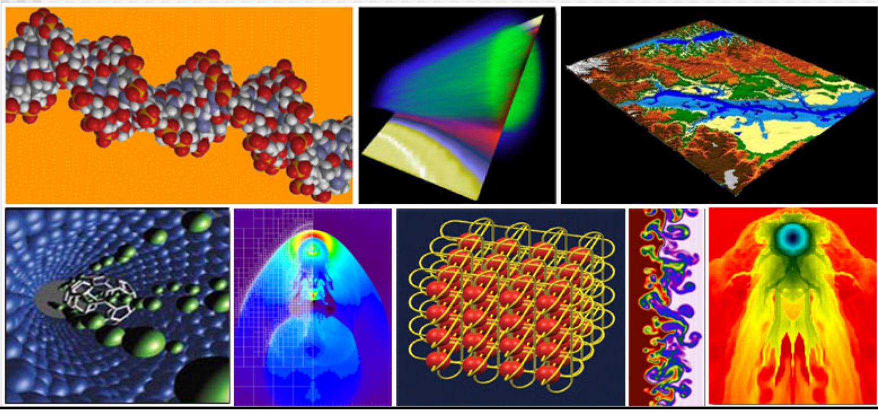
Uses for Parallel Computing

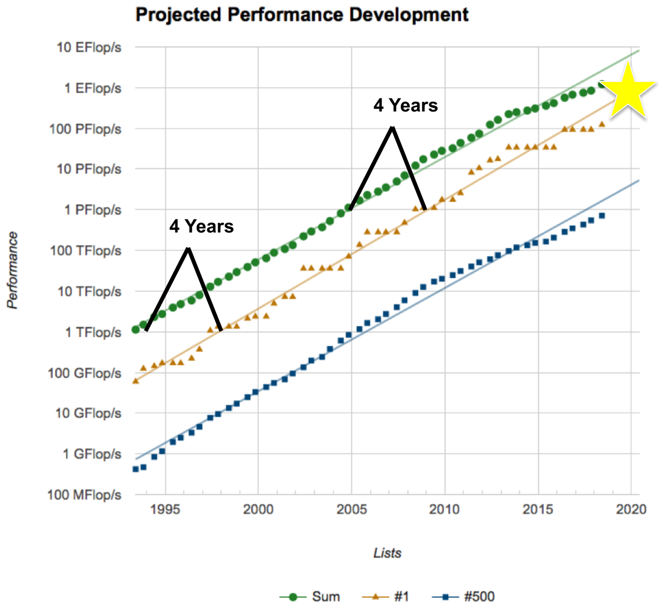
- Historically "the high end of computing"
  - Atmosphere, Earth, Environment
  - Physics - applied, nuclear, particle, condensed matter, high pressure, fusion, photonics
  - Bioscience, Biotechnology, Genetics
- Chemistry, Molecular Sciences
- Geology, Seismology
- Mechanical Engineering - from prosthetics to spacecraft
- Electrical Engineering, Circuit Design, Microelectronics
- Computer Science, Mathematics



Uses for Parallel Computing 2

- Today, commercial applications provide an equal or greater driving force; require processing of large amounts of data in sophisticated ways
  - Databases, data mining
  - Oil exploration
  - Web search engines, web based business services
- Medical imaging and diagnosis
- Pharmaceutical design
- Management of national and multi-national corporations
- Financial and economic modeling
- Advanced graphics and virtual reality, particularly in the entertainment industry
- Networked video and multi-media technologies
- Collaborative work environments





35

Top500 Nr 1: Summit

- Oak Ridge National Lab, USA
- IBM Power9, 22 cores, 3.07GHz & Nvidia Volta GV100
- 2,282,544 cores (2,090,880 GPU cores)
- 122.3 PF Linpack (187.66 PF theoretical peak)
- 8.8 MW

36

The Road to Exascale

- Top500 list projects first Exascale computing around 2019
- DARPA Exascale computing study explored technological trends for exascale
  - Issue number 1 is power:
    - Typical HPC system: ~500 MW for Exascale
    - Summit: ~50 MW for Exascale
    - Realistic limit is around 20 MW ➔ need improvement of factor of 10
    - Several years ago the situation was much worse: >2 GW for Exascale!
  - Data movement is actually the dominating factor (not FPU)
    - Move data on chip is 3 x more expensive
    - Moving data off chip 10 x
  - **Flops cheap are cheap, communication is expensive.**
  - **Exploiting data locality is *critical* for energy efficiency**

- Memory Performance:
  - DRAM density outpaced bandwidth by about 75 times the last 30 years
  - Memory bandwidth is limiting performance of future designs

www.darpa.mil/ipto/personnel/docs/ExaScale_Study_Initial.pdf

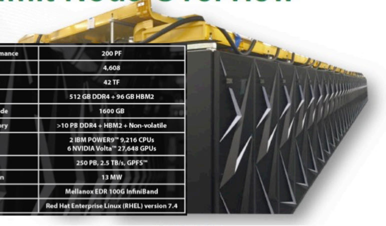
37

Future Computers will be heterogeneous

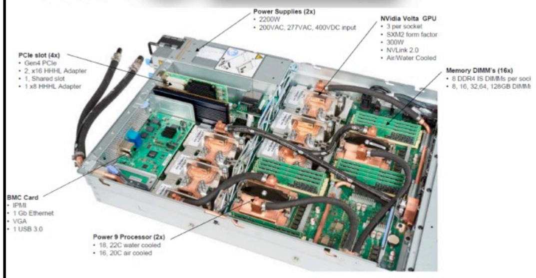
- Heterogeneous node architectures
  - Fast serial threads coupled to many efficient parallel threads
- Deep, explicitly managed memory hierarchy
  - Better exploit locality, improve predictability, and reduce overhead
- Exploit parallelism at all levels of a code
  - Distributed memory, shared memory, vector/SIMD, multithreaded

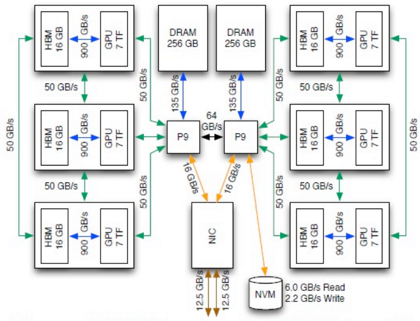
38

Summit Node Overview



Application Performance	200 PF
Number of Nodes	4,608
Node performance	42 TF
Memory per Node	512 GB DDR4 + 96 GB HBM2
NV memory per Node	1600 GB
Total System Memory	>10 PB DDR4 + HBM2 + Non-volatile
Processors	2 IBM POWER9™ 9,216 CPUs 6 NVIDIA Volta™ 27,648 GPUs
File System	250 PB, 2.5 TB/s, GPFS™
Power Consumption	13 MW
Interconnect	Mellanox EDR 100G InfiniBand
Operating System	Red Hat Enterprise Linux (RHEL) version 7.4





TF	42 TF (6x7 TF)
HBM	96 GB (6x16 GB)
DRAM	512 GB (2x16x16 GB)
NET	25 GB/s (2x12.5 GB/s)
MMsg/s	83

- HBM/DRAM Bus (aggregate B/W)
- NVLINK
- X-Bus (SMP)
- PCIe Gen4
- EDR IB

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

OAK RIDGE National Laboratory | LEADERSHIP COMPUTING FACILITY

OAK RIDGE National Laboratory

39

KTH VETENSKAP OCH KONST

PDC Center for
High Performance Computing

Have interesting,
challenging,
fun
two weeks!

40