High Performance Computing

What is it used for and why?

Overview

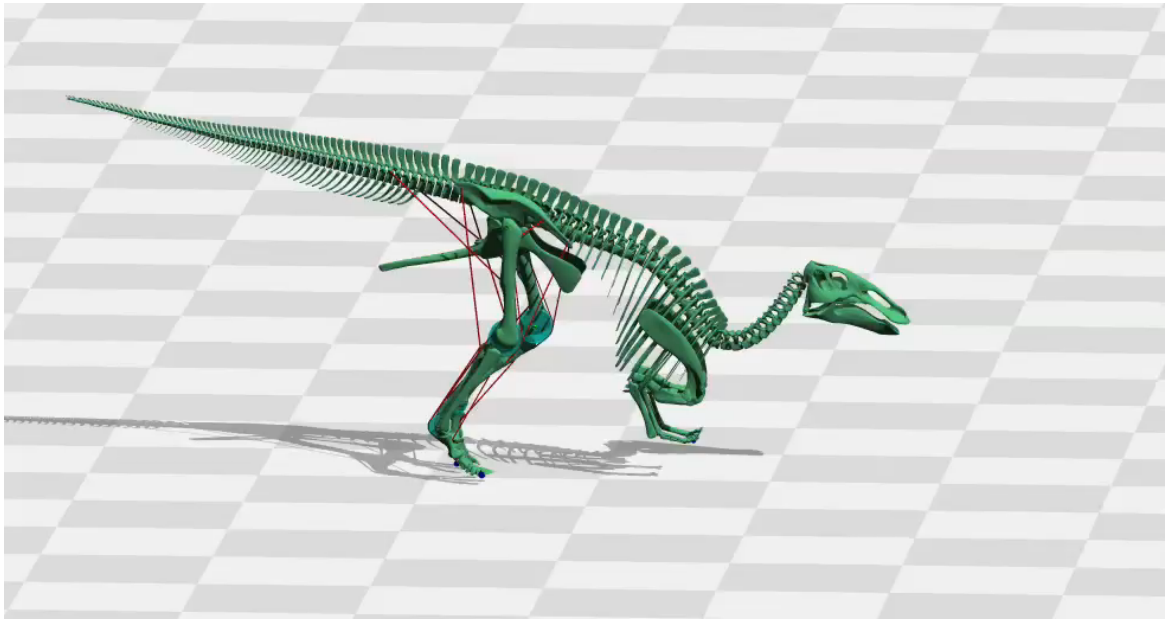
- What is it used for?
  - Drivers for HPC
  - Examples of usage
- Why do you need to learn the basics?
  - Hardware layout and structure matters
  - Serial computing is required for parallel computing
  - Appreciation of fundamentals will help you get more from HPC and scientific computing

What is HPC used for?

Drivers and examples

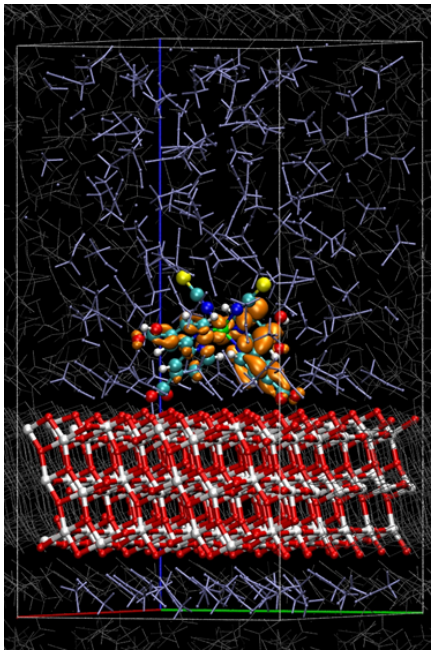
Why HPC?

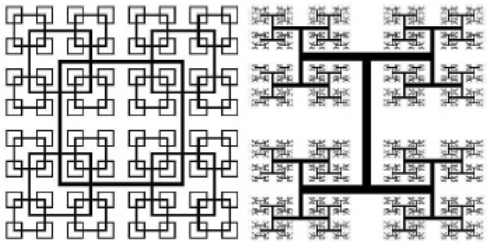
- Scientific simulation and modelling drive the need for greater computing power.
- Single systems could not be made that had enough resource for the simulations needed.
  - Making faster single chip is difficult due to both physical limitations and cost.
  - Adding more memory to single chip is expensive and leads to complexity.
- Solution: parallel computing – divide up the work among numerous linked systems.

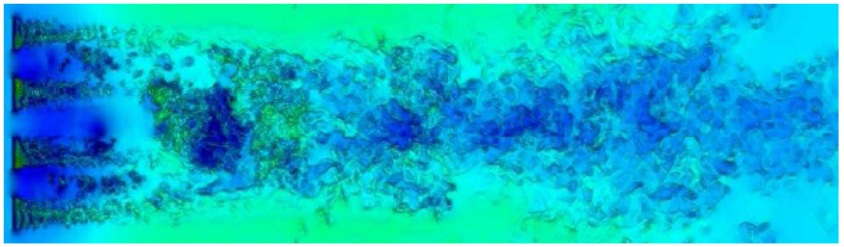


Modelling dinosaur gaits
Dr Bill Sellers, University of Manchester

Dye-sensitised solar cells
F. Schiffmann and J. VandeVondele
University of Zurich







Fractal-based models of turbulent flows
Christos Vassilicos & Sylvain Laizet,
Imperial College

archer

epcc

THE UNIVERSITY OF EDINBURGH

ARCHER

archer

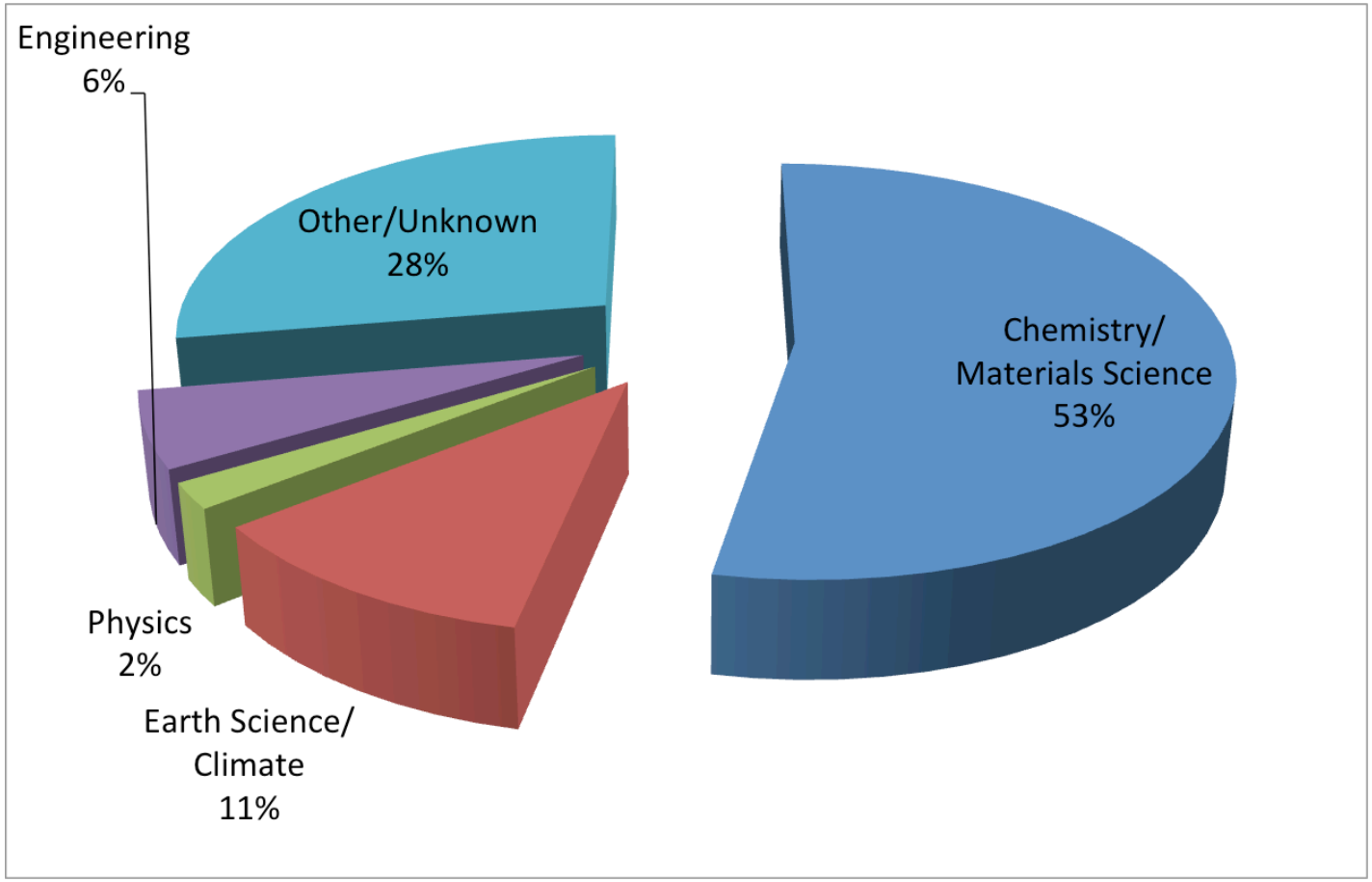
epcc

THE UNIVERSITY OF EDINBURGH

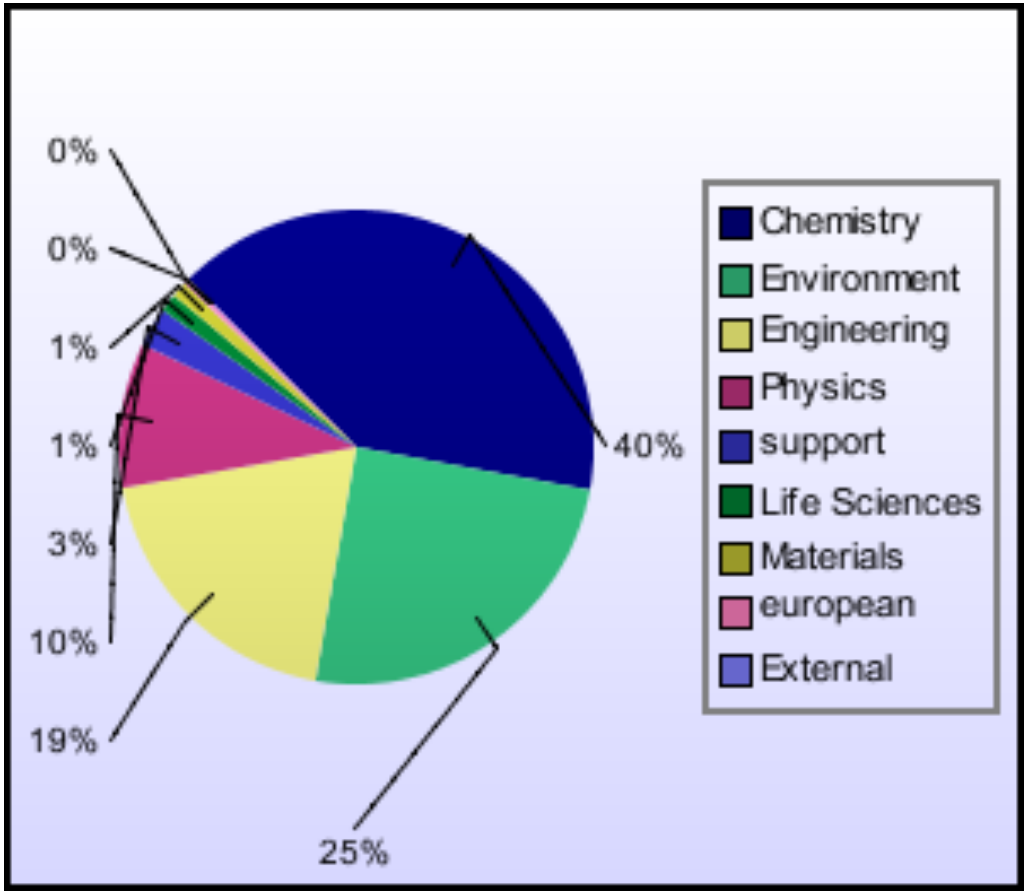
ARCHER in a nutshell

- UK National Supercomputing Service
  - Peak performance of 1.65 PF
  - ARCHER is *not* a Linpack engine
    - #19 in November 2013 top 500 list fastest (known) computer in the UK
  - Designed to provide 3-4 times scientific throughput of HECToR
    - HECToR #50 in top 500 with830 TFlop/s
- Cray XC30 Hardware
  - Nodes based on 2×Intel Ivy Bridge 12-core processors
  - 64GB (or 128GB) memory per node
  - 3008 nodes in total (72,162 cores)
  - Linked by Cray Aries interconnect (dragonfly topology)
- Cray Application Development Environment
  - Cray, Intel, GNU Compilers
  - Cray Parallel Libraries (MPI, SHMEM, PGAS)
  - DDT Debugger, Cray Performance Analysis Tools

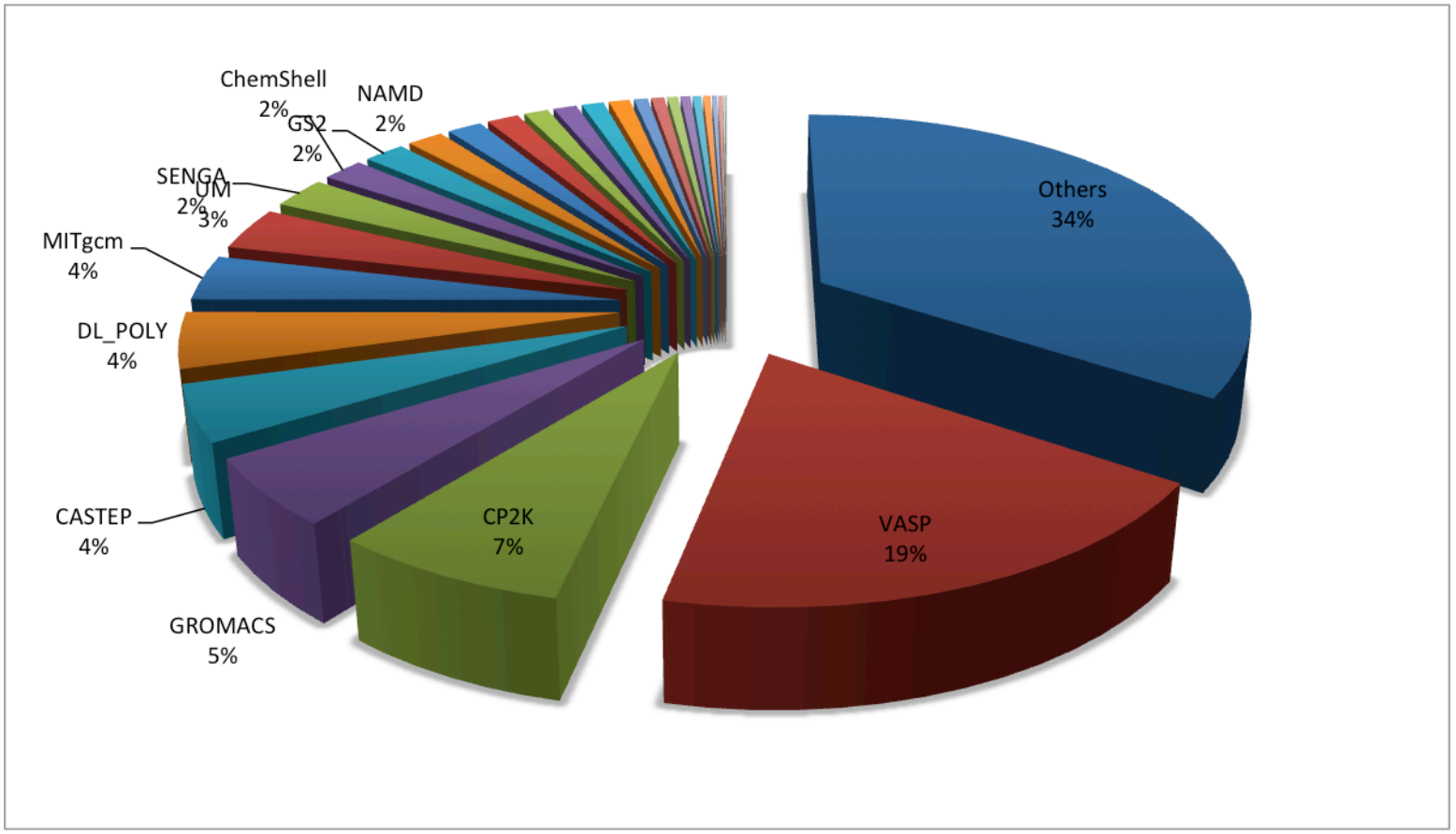
What is it used for?



What is it used for?



Simulation software



The Fundamentals

Why do I need to know this?

Hardware Layout

- Understanding the different types of HPC hardware allows you to understand why some things better on one resource than another
- Allows you to choose the appropriate resource for your application
- Allows you to understand the ways to parallelise your serial application
- Gives you an appreciation of the parts that are important for performance

Serial Computing

- Without an understanding of how serial computing operates it is difficult to understand parallel computing
  - What are the factors that matter for serial computation
  - How does the compiler produce executable code?
  - Which bits are automatic and which parts do I have to worry about

Parallel Computing

- Parallel computing and HPC are intimately related
- Understanding the different parallel programming models allows you to understand how to use HPC resources effectively

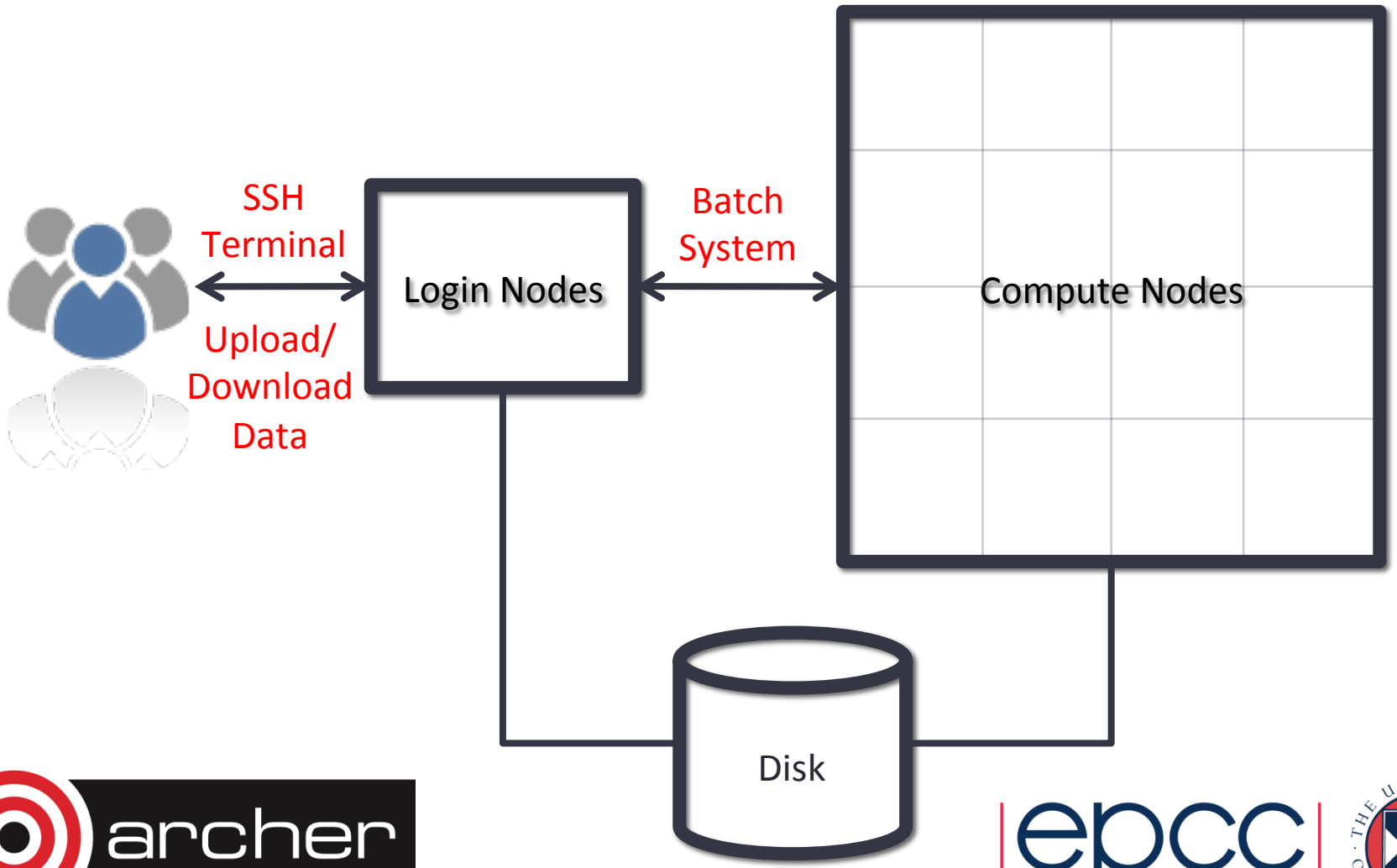
Differences from Cloud computing

- Performance
  - Clouds usually use virtual machines which add an extra layer of software.
  - In cloud you often share hardware resource with other users – HPC access is usually exclusive.
- Tight-coupling
  - HPC parallel programming usually assumes that the separate processes are tightly coupled
  - Requires a low-latency, high-bandwidth communication system between tasks
  - Cloud usually does not usually have this
- Programming models
  - HPC use high-level compiled languages with extensive optimisation.
  - Cloud often based on interpreted/JIT.

HPC Layout and Use

Starting concepts

Typical HPC system layout



Typical Software Usage Flow