Challenges facing the modelling community

The end of climate modelling as we know it

Bryan Lawrence

NCAS & University of Reading
Departments of Meteorology and Computer Science

Exeter, 17 June 19

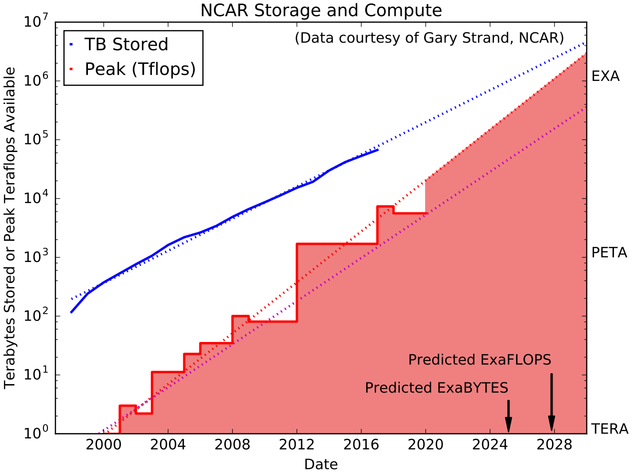
University of Reading

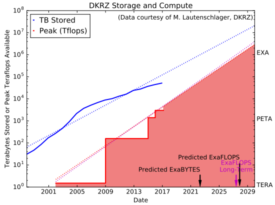
NERC SCIENCE OF THE ENVIRONMENT

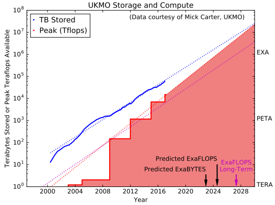
Outline

- Expectation: New science can be done based on ever increasing compute resources
- Moores's Law: Delivered ever increasing compute, but it's nearly over
- Kryder's Law: is failing us too: We have to be smarter!
- Post-Moore's Law: We have to be smarter!
- Avoidance: Documentation to avoid wasted effort (& emissions)

History has given us exponential compute linked to exponential data …

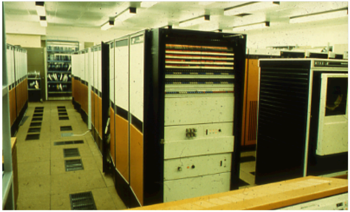






Faster Compute

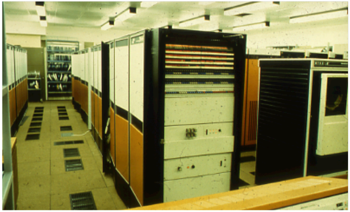
1981: ICL Dist.Array.Proc. (20 MFlops)



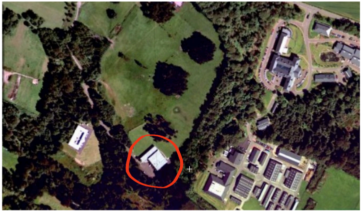
2014: Archer (then 1.4 PFlops)

Faster Compute

1981: ICL Dist.Array.Proc. (20 MFlops)



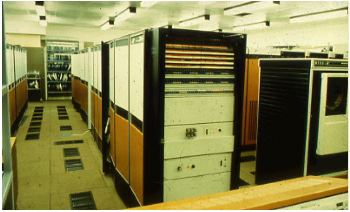
EPCC Advanced Computing Facility, 2014



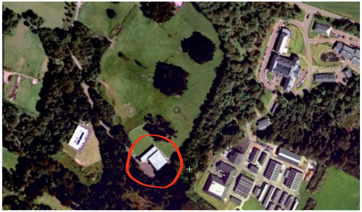
2014: Archer (then 1.4 PFlops)

Faster Compute

1981: ICL Dist.Array.Proc. (20 MFlops)



EPCC Advanced Computing Facility, 2014

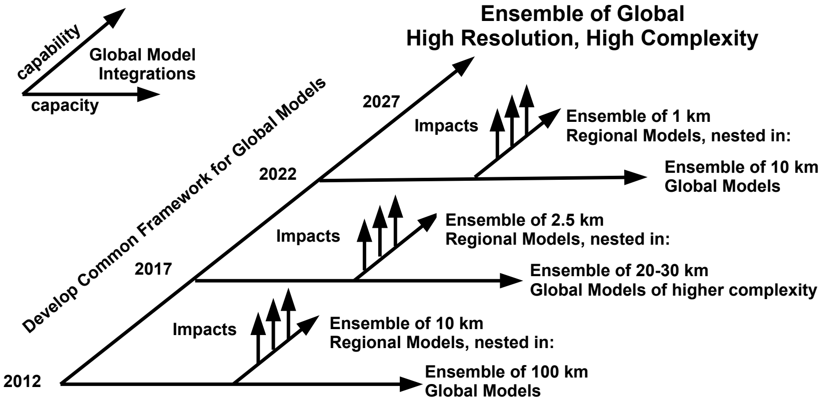


2014: Archer (then 1.4 PFlops)

From 1981, without Moore's Law

Slide content courtesy of Arthur Trew:

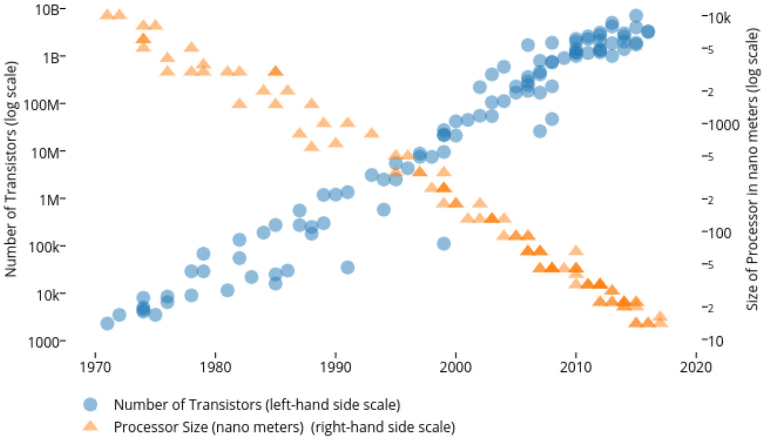
Climate Goals



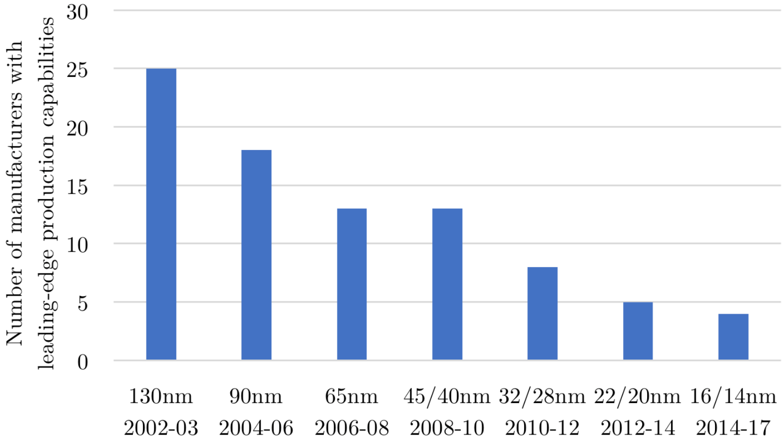
enes EUROPEAN NETWORK FOR EARTH SYSTEM MODELLING

(From “Infrastructure Strategy for the European Earth System Modelling Community” 2012-2022, Mitchell et al, 2012.)

Moores's Law

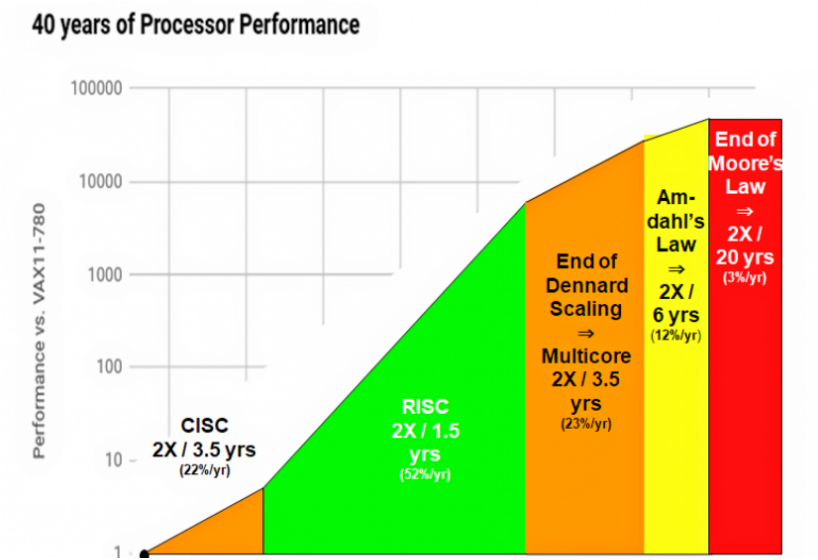


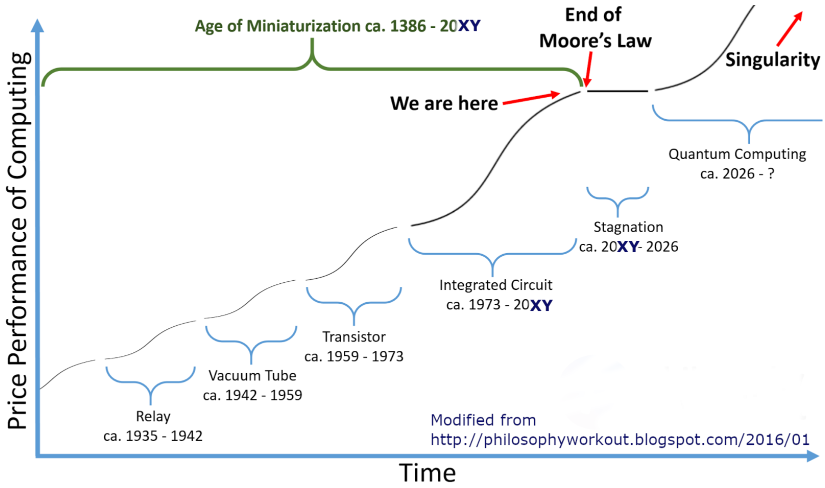
https://www.yaabot.com/31345/quantum-computing-neural-chips-moores-law-future-computing/



https://www.nextplatform.com/2019/02/05/the-era-of-general-purpose-computers-is-ending/

The Evolving Moore's Law



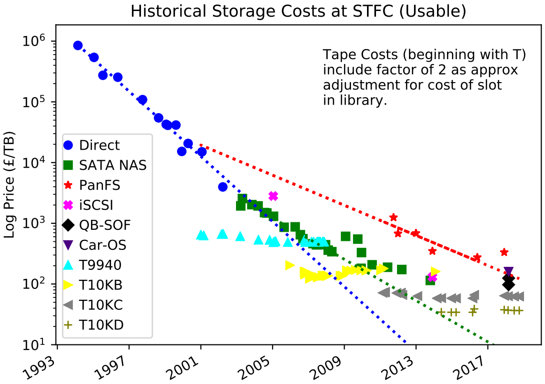


Modified from http://philosophyworkout.blogspot.com/2016/01

Real experience with Kryder's Law!

Kryder's Law

- The assumption that disk drive density, also known as areal density, will double every thirteen months. (Hasn't for some time!)
- The implication of Kryder's Law is that as areal density improves, storage will become cheaper:



- Relative cost of **disk** storage going up: each new generation of disk has a "shallower Kryder rate".
- Each new generation of **tape** is cheaper, and price stable over the lifetime.
- Tape has better technical future prospects than disk!

What now then?

No more advances for free on the back of computer hardware improvements and relatively little pain! Need to “resort” to

Maths

Algorithms

Customised Hardware

Software Solutions for performance, portability, and productivity.

Avoidance and Sharing

No more free lunch, a very different climate modelling world!

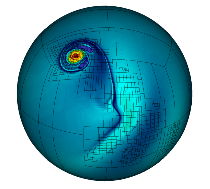
No time to talk about maths and algorithms …these, and more!

Parallel Time-Stepping

$$\mathbf{X}_{t+1}(x,y,z,t) = f(\mathbf{X}_{t-1},\mathbf{X}_t)$$

The function f could involve several steps (iterates) *carried out in parallel*.

Adaptive Grids

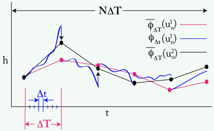


Machine Learning: Optimisation

Using AI/ML to

- Pre-condition solvers.
- Optimise/tune parameters

Parallel in Time Methods



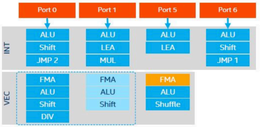
Variable Precision

Not all variables need the same precision (number of bits) in calculation, or in output.

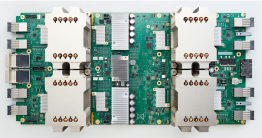
Machine Learning: Emulators

Replace slow "exact" (stationary?) parameterisations with fast "learnt" emulators.

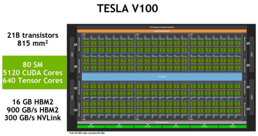
From decades of the same to a Cambrian Explosion



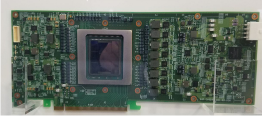
Vector Processors on Intel Zeon



Google's Tensor Programming Unit



GPUs from NVIDIA and AMD



Vector Processing Units from NEC

Server chips based on ARM designs

FPGA from many sources

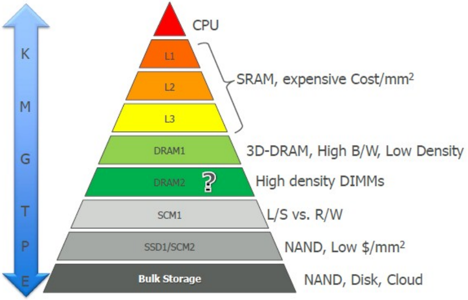
The end of Moore's Law means more specialisation: all with very different programming models!

Too many levels of parallelism

- Vector Units (on chip)
- Parallelism Across Cores
- Shared Memory Concurrency
- Distributed Memory
- Numerical Method Concurrency
- Internal Component Concurrency
- Coupled Component Concurrency
- I/O and Diagnostic Parallelism
- (Storage System Parallelism)

Too many levels of parallelism

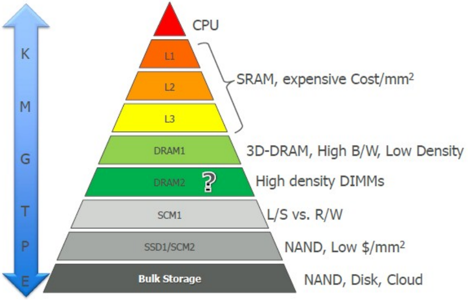
- Vector Units (on chip)
- Parallelism Across Cores
- Shared Memory Concurrency
- Distributed Memory
- Numerical Method Concurrency
- Internal Component Concurrency
- Coupled Component Concurrency
- I/O and Diagnostic Parallelism
- (Storage System Parallelism)



Nearly everything is processor/system dependent!
(except green layers on left).

Too many levels of parallelism

- **Vector Units (on chip)**
- **Parallelism Across Cores**
- **Shared Memory Concurrency**
- **Distributed Memory**
- **Numerical Method Concurrency**
- **Internal Component Concurrency**
- **Coupled Component Concurrency**
- **I/O and Diagnostic Parallelism**
- **(Storage System Parallelism)**

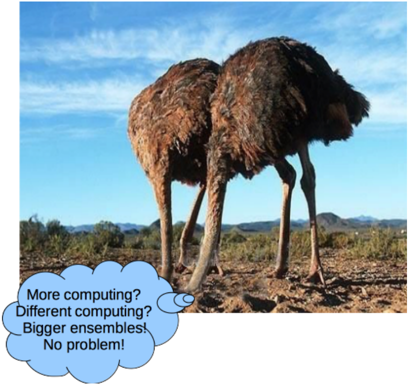


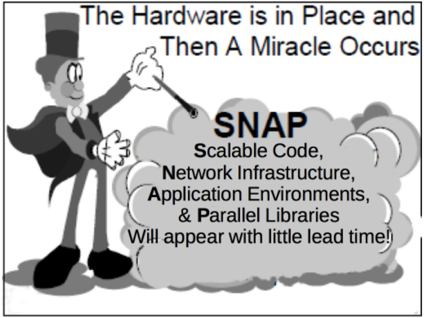
Nearly everything is processor/system dependent! (except green layers on left).

Entirely new programming models are likely to be necessary, with entirely new* constructs such as thread pools and task-based parallelism possible. Memory handling will be crucial!

* New in this context!

What about software?



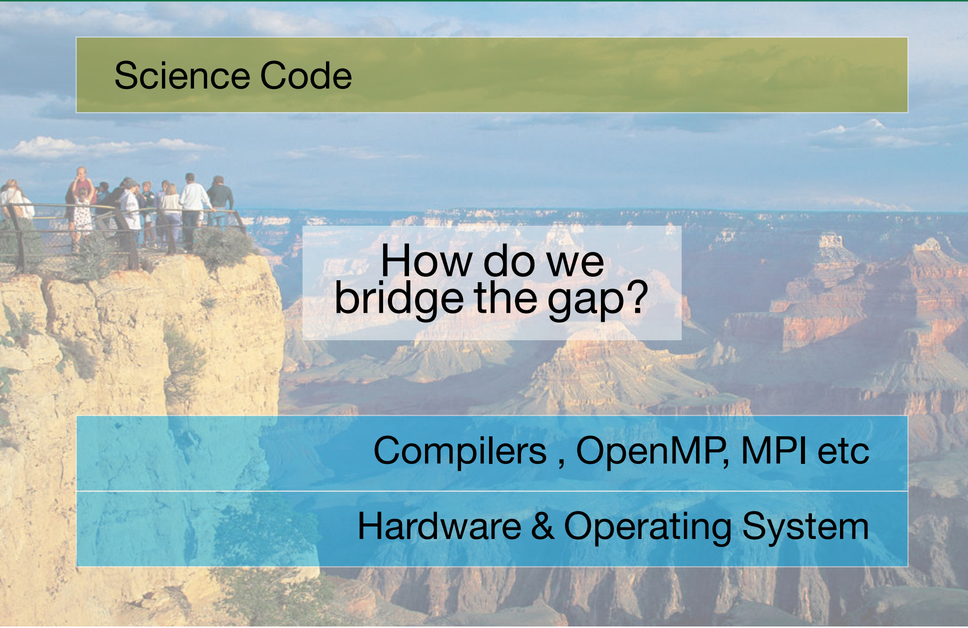


Some people have a very naive idea about the relationship between the hardware and the software!

Software changing slowly & slowing!

Hardware changing rapidly & accelerating!

How far is it between our scientific aspiration and our ability to develop and/or rapidly adapt our codes to the available hardware?



Science Code

How do we bridge the gap?

Compilers , OpenMP, MPI etc

Hardware & Operating System

Crossing the Chasm: How to develop weather and climate models for next generation computers?

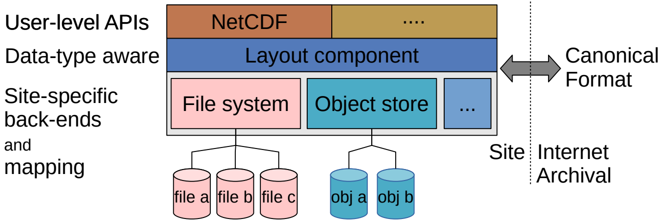
Lawrence, Rezny, Budich, Bauer, Behrens, Carter, Deconinck, Ford, Maynard, Mullerworth, Osuna, Porter, Serradell, Valcke, Wedi, and Wilson

https://doi.org/10.5194/gmd-11-1799-2018

IS-ENES2 Deliverable 3.2

is-enes
INFRASTRUCTURE FOR THE EUROPEAN NETWORK FOR EARTH SYSTEM MODELLING

Earth System Data Middleware



Key Concepts

- Applications work through existing application interfaces (currently: NetCDF library)
- Middleware utilizes layout component to make placement decisions
- Data is then written/read efficiently avoiding file system limitations (e.g. consistency constraints)
- Potential for deploying with an active storage management system.

Work in Progress: In-Flight Parallel Data Analysis

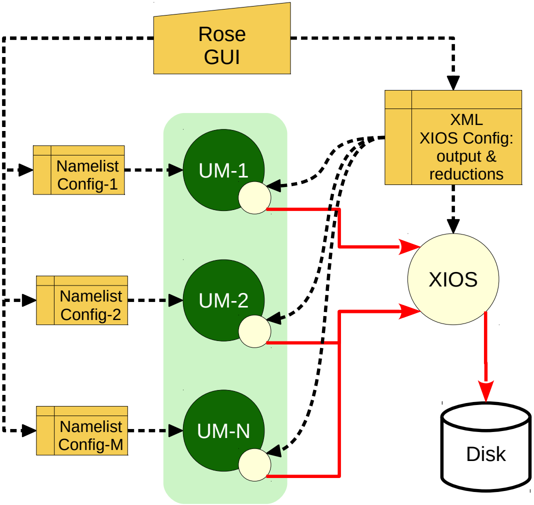
esiwace CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

An ensemble is a set of simulations running different instances of the same numerical experiment. We do this to get information about uncertainty.

Dealing with too much ensemble data

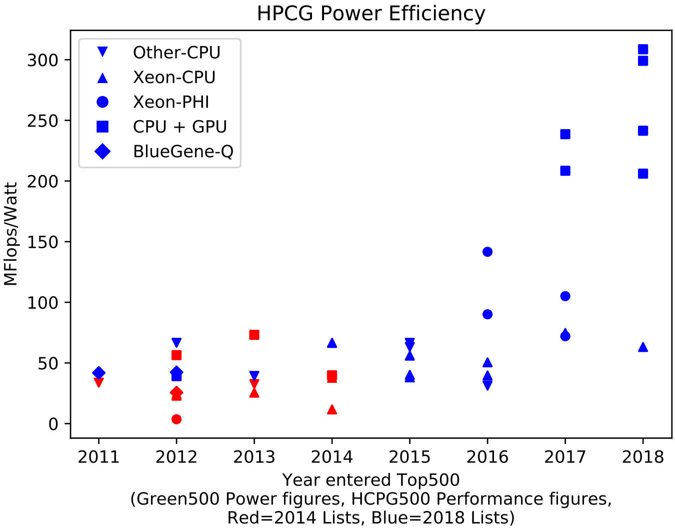
Instead of writing out all ensemble members and doing all the analysis later:

- Calculate ensemble statistics on the fly.
- Only write out some ensemble members.
- (Which ones? A tale for another day, see Daniel Galea's Ph.D work.)

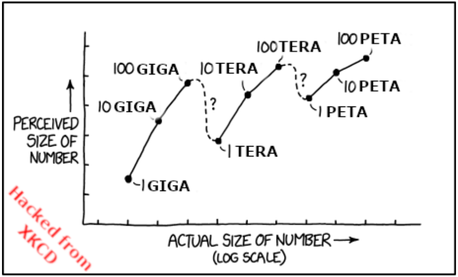


Cole, Lawrence, Lister, Meursdesoif, Nash, Weiland

Power Consumption and Performance



Climate Scientists don't *respect* big numbers!



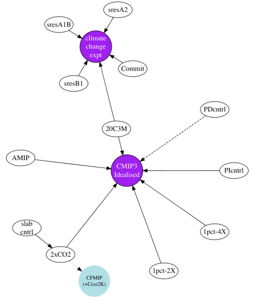
TALKING ABOUT LARGE NUMBERS IS HARD

In modelling, many underestimate:

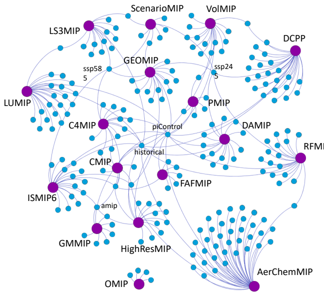
- The energy demands and costs of computing associated with their experiments, and so the need for **energy efficient codes and computational environments**,
- The need for **respecting energy costs in experimental design**, and
- The **difficulty in managing, disseminating, and utilising large volumes of data!**

These issues are only going to become worse unless we do something about it!

The CMIP Evolution: from CMIP3 to CMIP6



to



The Logistics of Collaboration

- In HPC we know that the larger the number of cores, the more the communications cost ...
- these communications costs need to paid for large scale scientific collaboration too!

From experimental design, to the data request, the (ESGF) dissemination infrastructure, and to the analysis systems; we need to invest more in the supporting infrastructure, and respect the constraints — but this is not a popular message!

Simulation Years, Energy, and Planning

Experiments have real physical costs

Most of us do not know the energy costs of our models!

- Balaji et al. (2017) CPMIP: Measurements of Real Computational Performance of Earth System Models in CMIP6. https://doi.org/10.5194/gmd-10-19-2017
- Requires **quantification** of "JPSY" — Joules per simulated year: via direct measurements or estimation (core-hours per simulated year and average system costs per core-hour).

Set real energy budgets for experiments (and MIPs)

Share plans and outputs!

- Exploit ES-DOC tools to define and share experiment plans.
- (See Pascoe et al, 2019, https://doi.org/10.5194/gmd-2019-98 and the published CMIP6 experiment docs: https://documentation.es-doc.org/cmip6/experiments.)
- The larger the experiment in energy terms, the more it needs rigorous justification and/or a big user community!

Summary

- Expectation: We need to recalibrate our expectations of future compute.
- Moores's Law: The end will deliver a Cambrian explosion of hardware.
- Post-Moore's Law: *Being smarter*: Crossing the chasm with better maths and community software such as DSLs.
- Kryder's Law: *Being smarter*: Much going on to help us deal with both avoiding writing data, but if we have to have it, handling it efficiently.
- Avoidance: *Being smarter*: ES-DOC project delivering on methodology to share big experiments from research group scale to CMIP scale — but we have to *design and share*!

We need to be investing in being smarter **NOW** …