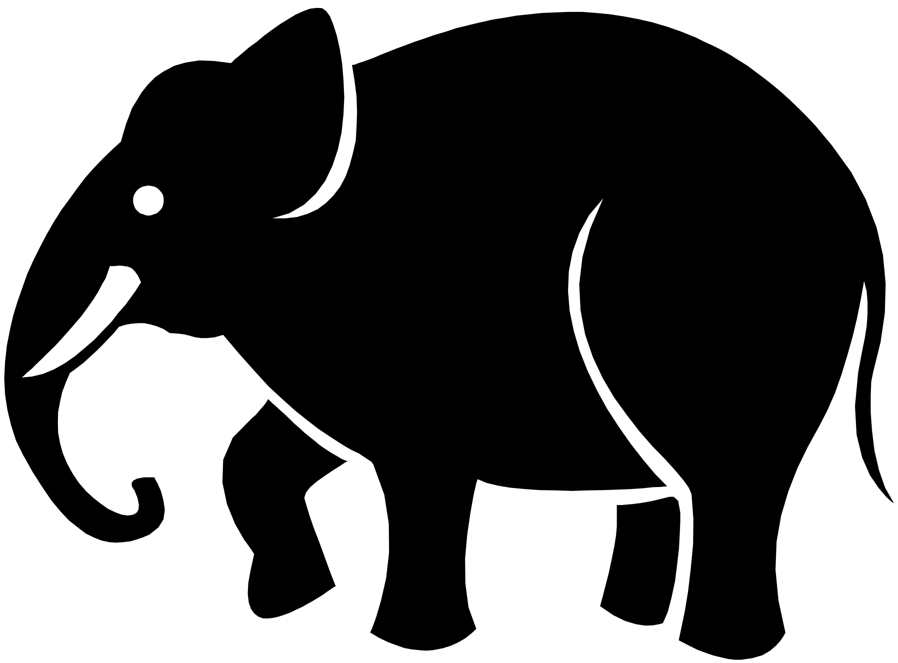
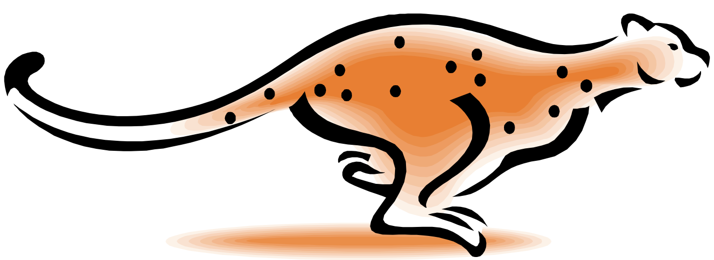
Big Data and Big Compute

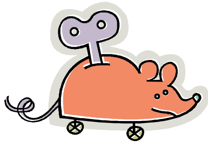
What is Supercomputing About?

Size



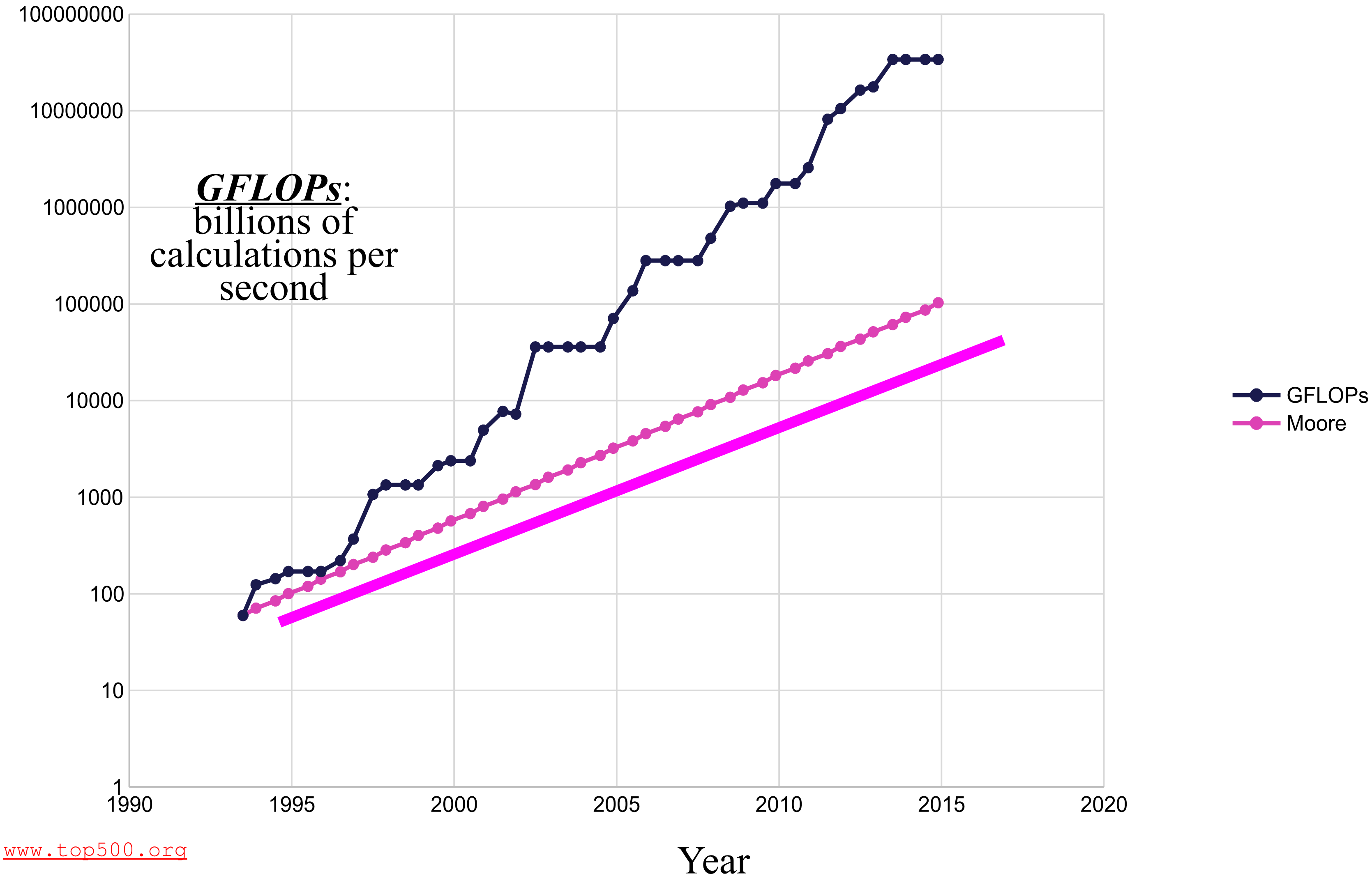
Speed





Laptop

Fastest Supercomputer vs. Moore



www.top500.org

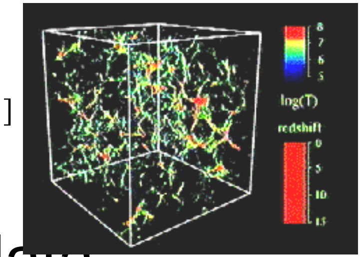
What is Supercomputing?

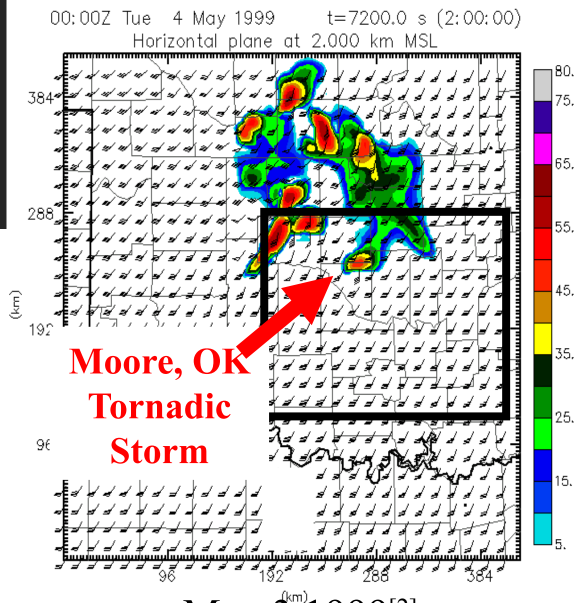
- ***Supercomputing*** is the **biggest, fastest computing** **right this minute**.
- Likewise, a ***supercomputer*** is one of the biggest, fastest computers right this minute.
- So, the definition of supercomputing is **constantly changing**.
- **Rule of Thumb**: A supercomputer is typically at least 100 times as powerful as a PC.
- **Jargon**: Supercomputing is also known as ***High Performance Computing*** **(HPC)** or ***High End Computing*** **(HEC)** or ***Cyberinfrastructure*** **(CI).**

What Is HPC Used For?

- ***Simulation*** of physical phenomena, such as
  - Weather forecasting
  - Galaxy formation
  - Oil reservoir management
- ***Data mining***: finding **needles** of information in a **haystack** of data, such as
  - Gene sequencing
  - Signal processing
  - Detecting storms that might produce
- ***Visualization***: turning a vast sea of **data** into pictures that a scientist can understand

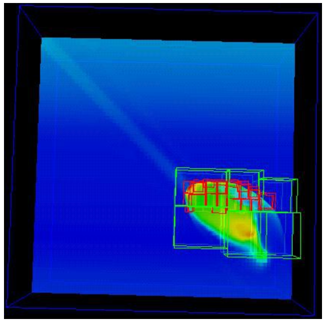
[1]



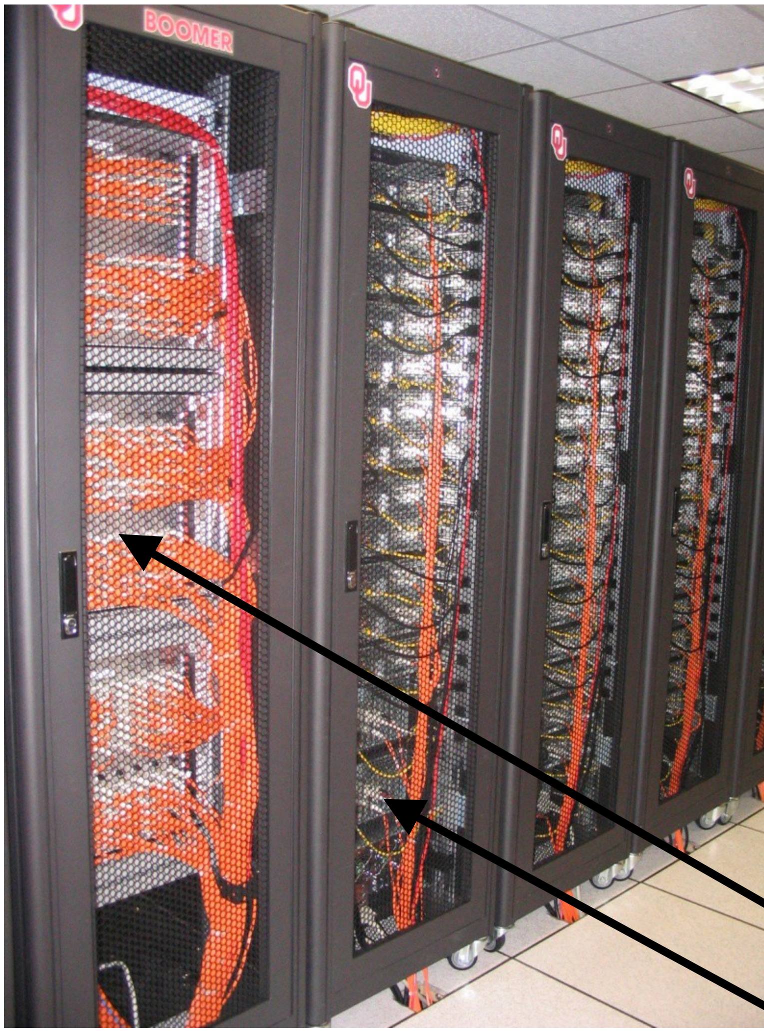


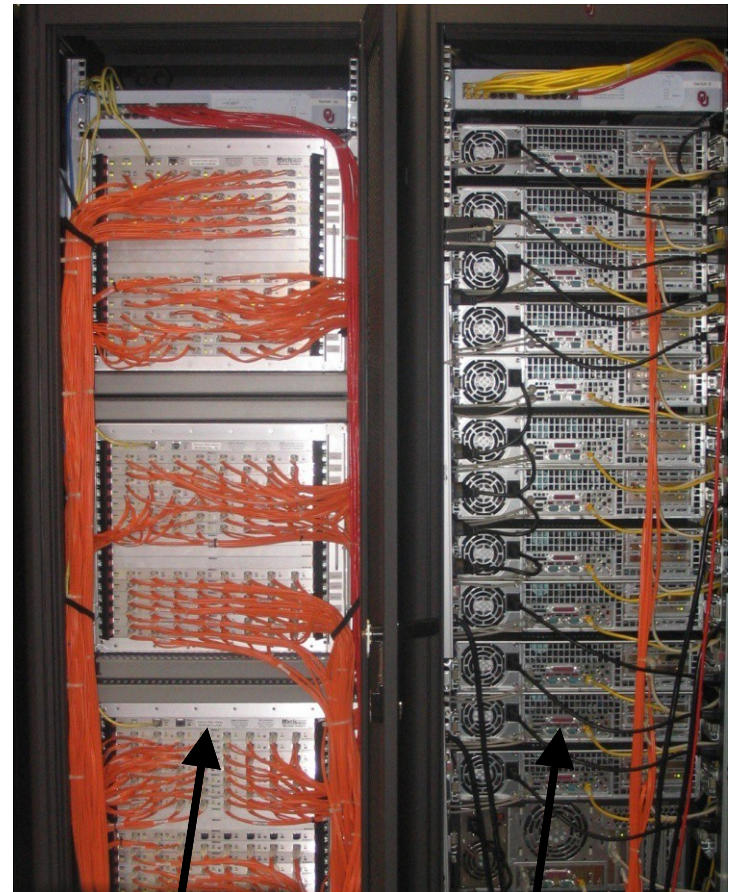
May 3 1999[2]

[3]



An Actual Cluster





Interconnect

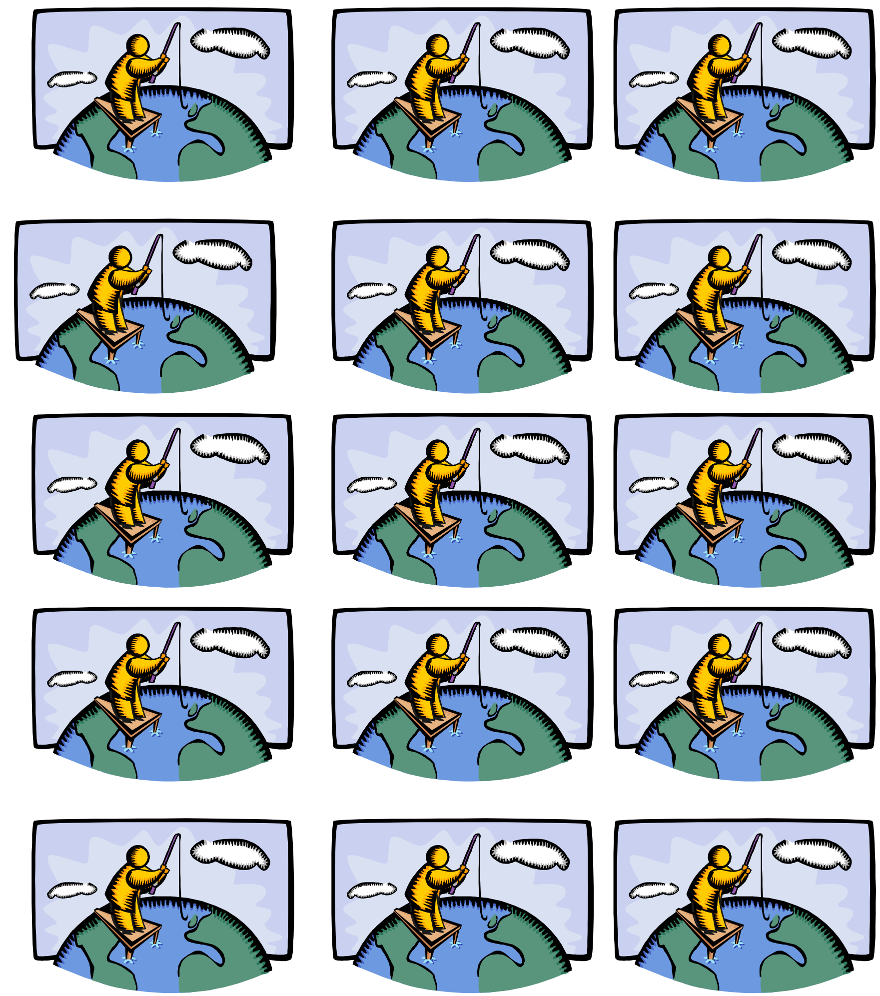
Nodes

Parallelism

Parallelism

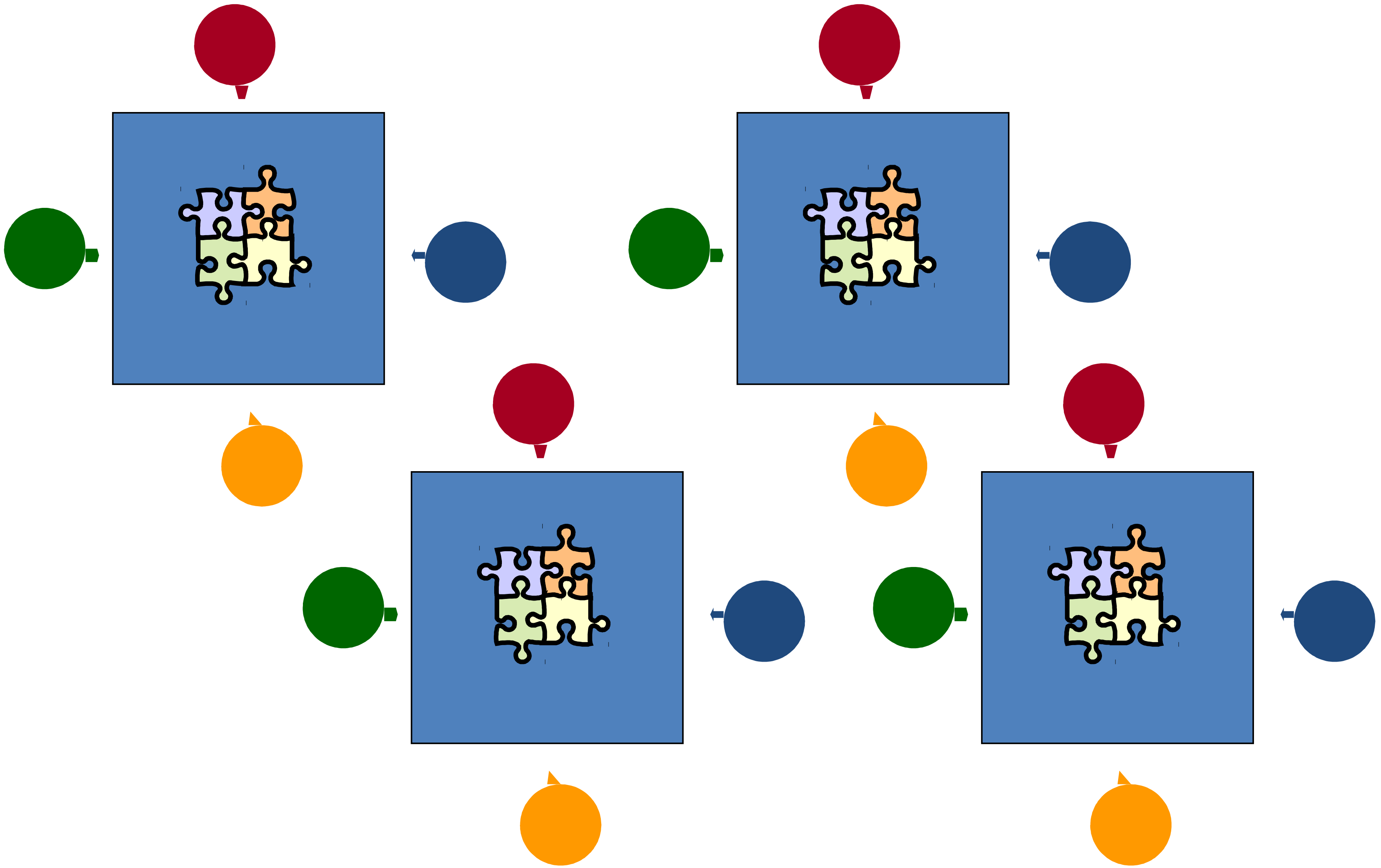
Parallelism means doing multiple things at the same time: you can get more work done in the same time.

Less fish …

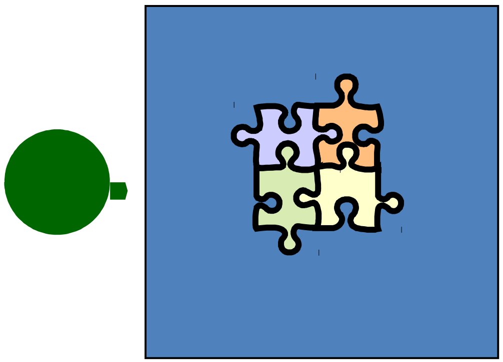


More fish!

The Jigsaw Puzzle Analogy



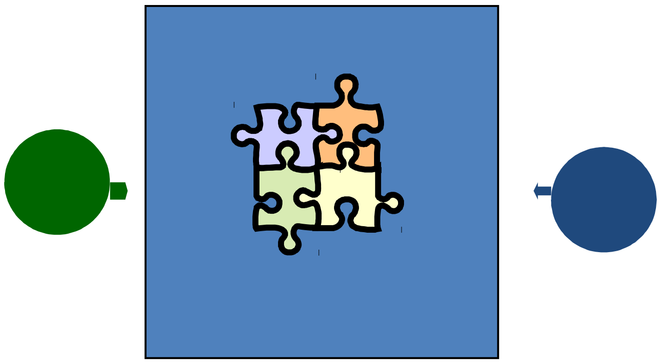
Serial Computing



Suppose you want to do a jigsaw puzzle that has, say, a thousand pieces.

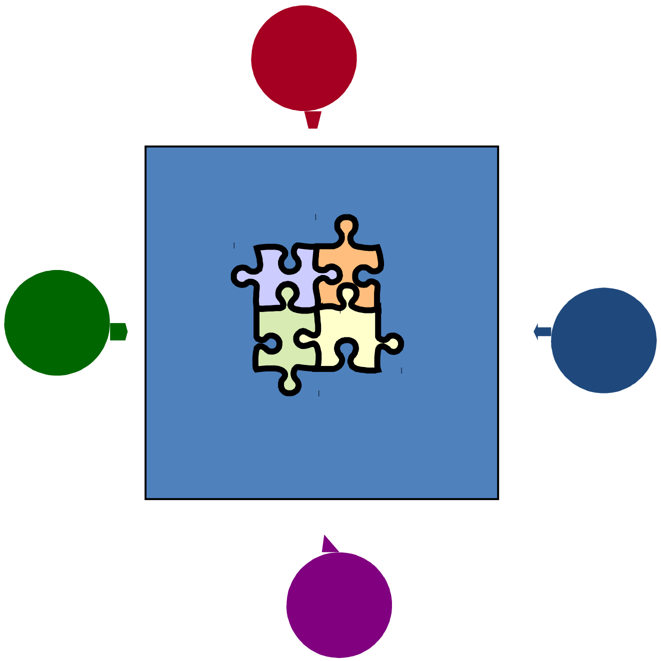
We can imagine that it'll take you a certain amount of time. Let's say that you can put the puzzle together in an hour.

Shared Memory Parallelism



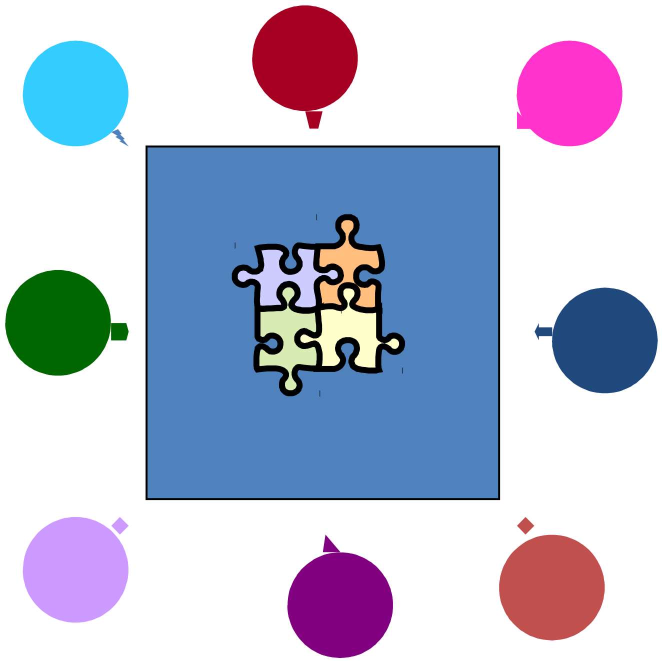
If Scott sits across the table from you, then he can work on his half of the puzzle and you can work on yours. Once in a while, you'll both reach into the pile of pieces at the same time (you'll ***contend*** for the same resource), which will cause a little bit of slowdown. And from time to time you'll have to work together (***communicate***) at the interface between his half and yours. The speedup will be nearly 2-to-1: you might take 35 minutes instead of 30.

The More the Merrier?



Now let's put Paul and Charlie on the other two sides of the table. Each of you can work on a part of the puzzle, but there'll be a lot more contention for the shared resource (the pile of puzzle pieces) and a lot more communication at the interfaces. So you will get noticeably less than a 4-to-1 speedup, but you'll still have an improvement, maybe something like 3-to-1: the four of you can get it done in 20 minutes instead of an hour.

Diminishing Returns



If we now put Dave and Tom and Imran and Julia on the corners of the table, there's going to be a lot of contention for the shared resource, and a lot of communication at the many interfaces.

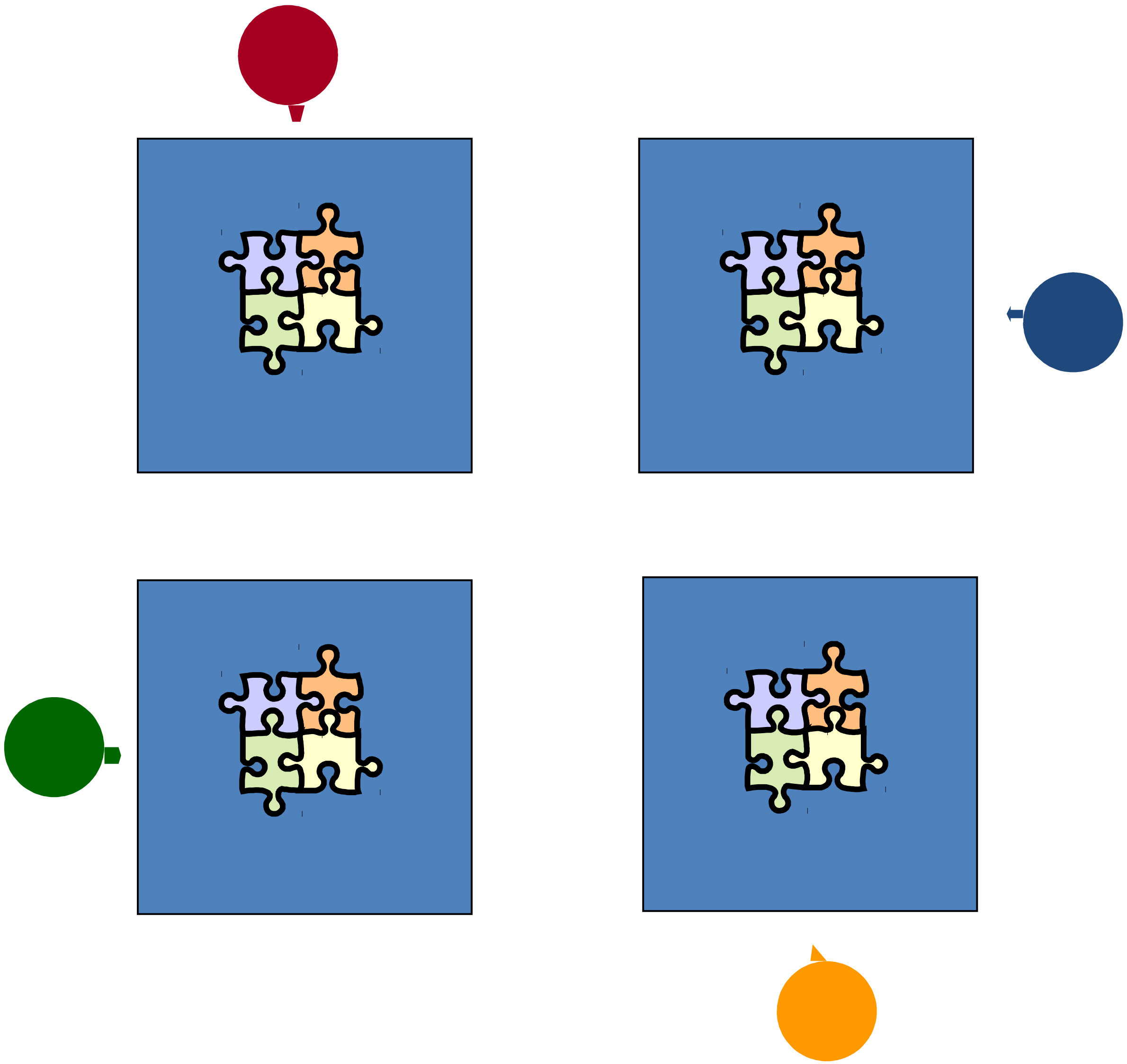
So the speedup you get will be much less than we'd like; you'll be lucky to get 5-to-1.

So we can see that adding more and more workers onto a shared resource is eventually going to have a diminishing return.

Distributed Parallelism

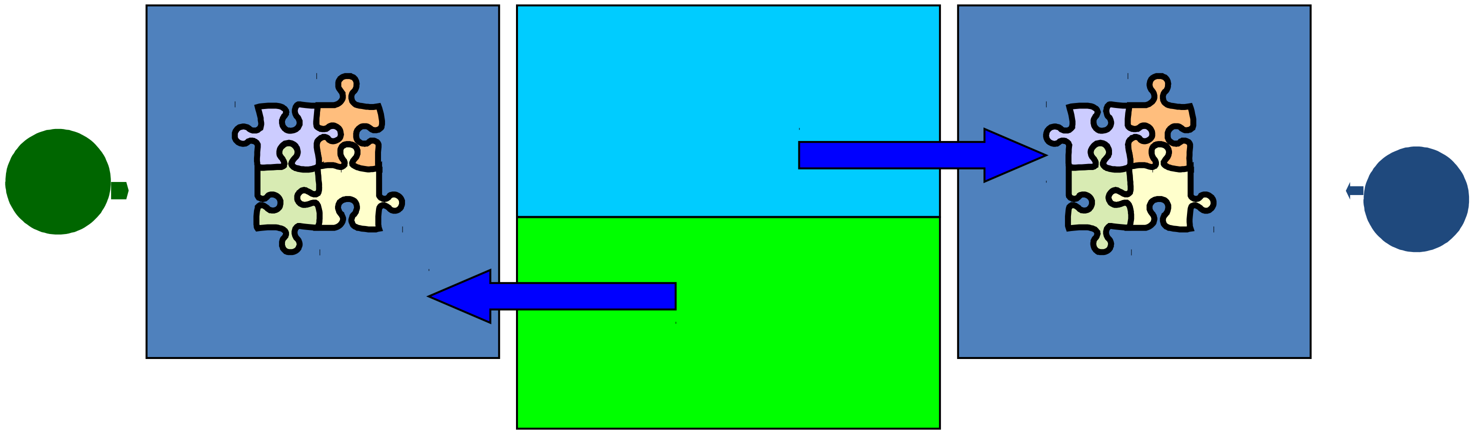
Now let's try something a little different. Let's set up two tables, and let's put you at one of them and Imran at the other. Let's put half of the puzzle pieces on your table and the other half of the pieces on Imran's. Now you can work completely independently, without any contention for a shared resource. **<u>BUT</u>**, the cost per communication is **<u>MUCH</u>** higher (you have to put your tables together), and you need the ability to split up (***<u>decompose</u>***) the puzzle pieces reasonably evenly, which may be tricky to do for some puzzles.

More Distributed Processors



It’s a lot easier to add more processors in distributed parallelism. But, you always have to be aware of the need to decompose the problem and to communicate among the processors. Also, as you add more processors, it may be harder to ***load balance*** the amount of work that each processor gets.

Load Balancing



Load balancing means ensuring that everyone completes their workload at roughly the same time.

For example, if the jigsaw puzzle is half grass and half sky, then you can do the grass and Imran can do the sky, and then you only have to communicate at the horizon – and the amount of work that each of you does on your own is roughly equal. So you'll get pretty good speedup.