Concurrency Kung-fu: Python Style

Eilif Muller

Blue Brain Project

neural ensemble .org

Reserve your place …

- Login to the head node:

  $ ssh student<1..15>@asppcluster.in.waw.pl

- Schedule an interactive session on a compute node:

  $ qrsh -pe smp 18 -now no

…for the concurrency revolution!

Why Concurrency? (parallelism)

The brain is a parallel machine

Asynchronous

Distributed memory

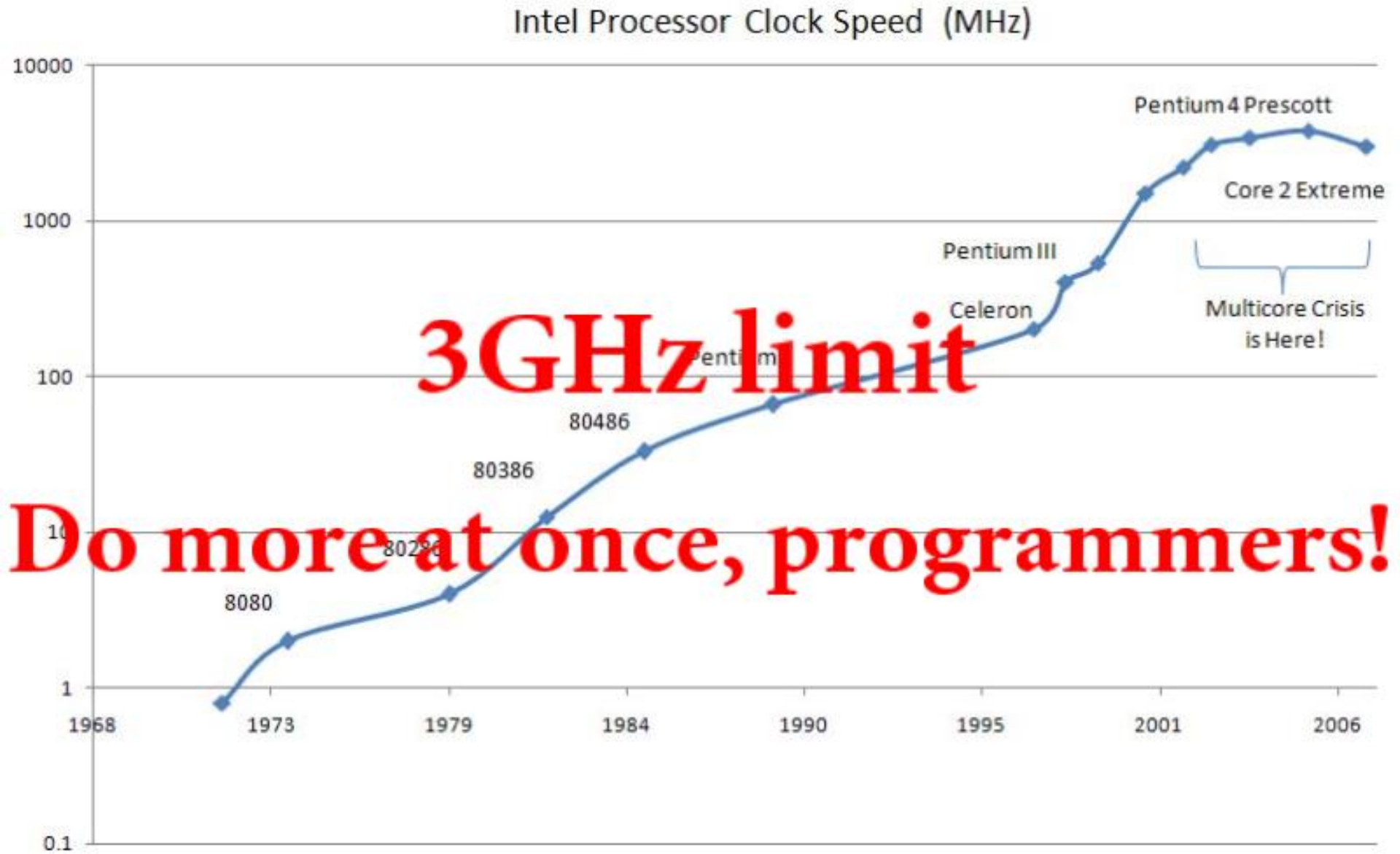
Simple messages

Agnostic to component failure

Connectivity:
dense local, sparse global

Works with long latencies

Why Concurrency? (parallelism)


Intel Processor Clock Speed (MHz)
10000
1000
100
10
1
0.1
1968
1973
1979
1984
1990
1995
2001
2006
8080
80386
80486
Pentium III
Celeron
Pentium 4 Prescott
Core 2 Extreme
Multicore Crisis is Here!
3GHz limit
Do more at once, programmers!

The free lunch is over

"Concurrency is the next major revolution in how we write software [after OOP]."

Herb Sutter, *The Free Lunch is Over: A Fundamental Turn Towards Concurrency in Software*, Dr.Dobb's Journal, 30(3) March 2005.

At least 3 types of Concurrency

SMP	Message passing	Stream
Shared mem. Multi-thread	MPI, sockets Linux Clusters	GPU, Cell
<8 Threads Or $	1000's of processes over network	SIMD Stream: Kernel over arrays
threads multiprocessing	mpi4py, ipython Parallel Python	PyCUDA, PyOpenCL

Others: SSEx ...

Don't Panic!

- Writing parallel programs is easy!
  - Small and simple APIs
- Designing parallel algorithms can be easy or hard.
  - Easy: "embarassingly parallel"
  - Hard: to find the parallelism
  - Hardware: e.g. the starving CPU

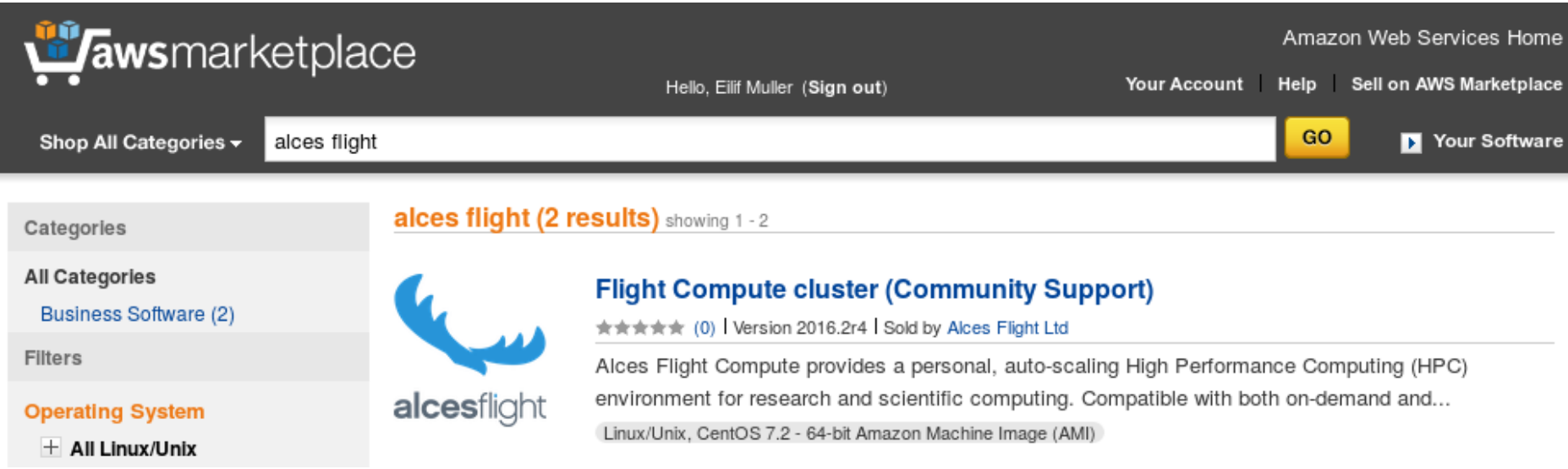
Don't Panic!

- Scientific parallel programs are easy!
  - Parallelism: Number crunching over large datasets
  - Prior-art: Many algorithms already exist
  - Hardware: HPC is traditionally academic

Objectives

- Hands on experience using a real HPC cluster
- Basic APIs
- Speed-ups motivate …

And we have a toy to play with!

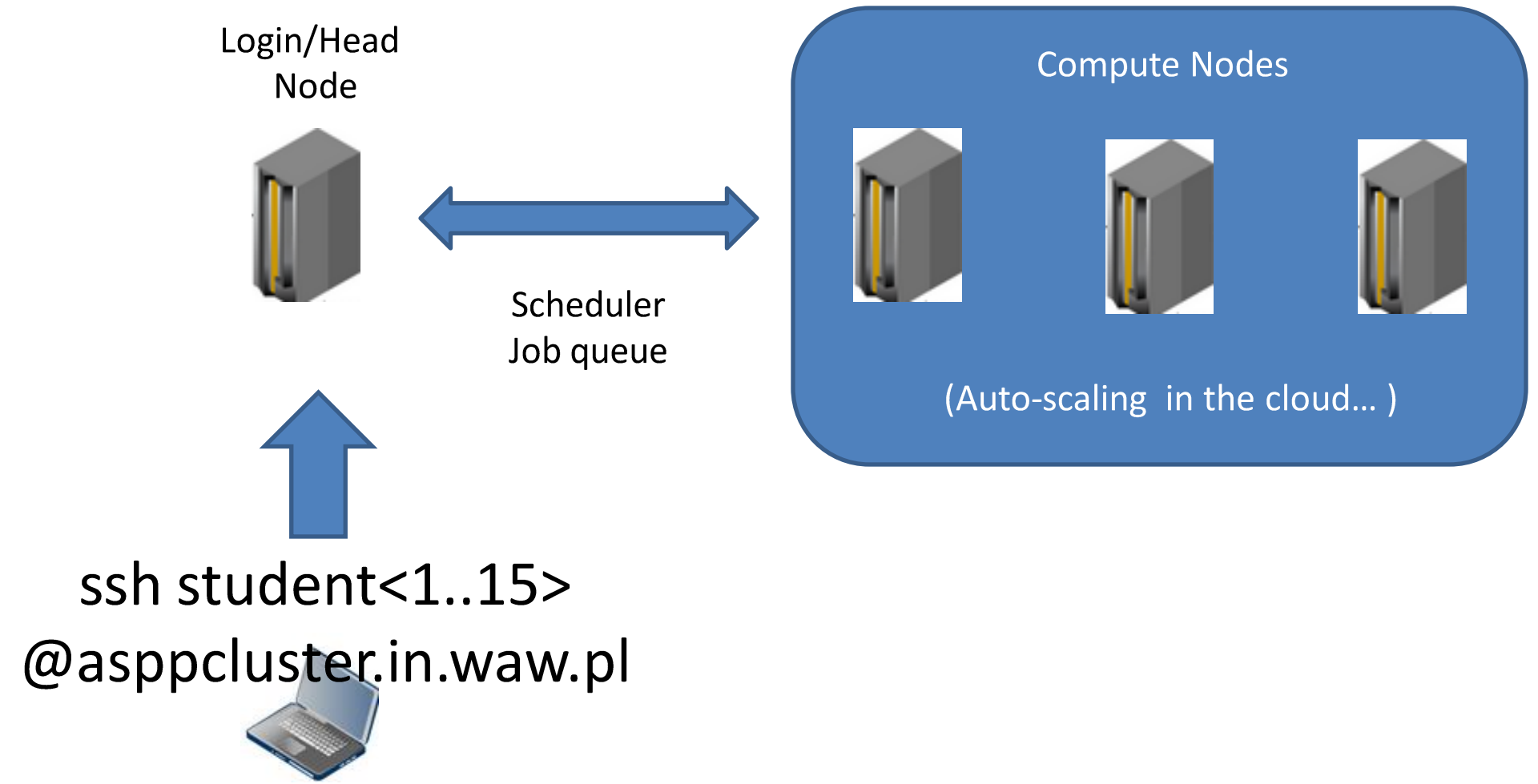


amazon web services

AWS Cloud Credits for Research

ssh student<1..15>@asppcluster.in.waw.pl

Architecture of a cluster




Part 1:
Easy Concurrency with IPython