ECE 574 – Cluster Computing
Lecture 2

Vince Weaver

http://www.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

3 September 2015

Announcements

-

Top500 List – June 2015

#	Name	Country	Arch	Proc	Cores	Max/Peak	Accel	Power
						TFLOPS		kW
1	Tianhe-2	China	x86	IVB	3,120,000	33,862 / 54.902	xeon-phi	17,808
2	Titan	USA/ORNL	x86	Opteron	560,640	17,590 / 27,112	NVD K20	8,209
3	Sequoia	USA/LLNL	Power	BG/Q	1,572,864	17,173 / 20,132	?	7,890
4	RIKEN	Japan	SPARC	VIIIfx	705,024	10,510 / 11,280	?	12,660
5	Mira	USA/Argonne	Power	BG/Q	786,432	8,586 / 10,066	?	3,945
6	Piz Daint	Switzerland	x86	SNB-EP	115,984	6,271 / 7,788	NVD K20	2325
7	Shaheen II	Saudi Arabia	x86	SNB-EP	196,608	5,537 / 7,235	?	2,834
8	Stampede	USA/TACC	x86	SNB-EP	462,462	5,168 / 8,520	XeonPhi	4,510
9	Juqeen	DE/Julich	Power	BG/Q	458,752	5008/5872	?	2,301
10	Vulcan	USA/LLNL	Power	BG/Q	393,216	4293/5033	?	1,972

How long does it take to run LINPACK? How much money does it cost to run LINPACK?

How much RAM? How much cooling?

- 5th time running Tianhe-2.
- Not much turnover.
- 68 systems over a petaflop
- 90 systems use some sort of accelerator
- 87% of nodes have 8 or more cores
- HP, IBM, Cray with most systems

What goes into a top supercomputer?

- Commodity or custom

- Architecture
  x86? SPARC? Power? ARM
  embedded vs high-speed?

- Memory

- Storage
  How much?

Large hadron collider one petabyte of data every day
Shared? If each node wants same data, do you need to replicate it, have a network filesystem, copy it around with jobs, etc? Cluster filesystems?

- Reliability. How long can it stay up without crashing?
  Can you checkpoint/restart jobs?
  Sequoia MTBF 1 day.
  Blue Waters 2 nodes failure per day.
  Titan MTBF less than 1 day

- Power / Cooling
  Big river nearby?

- Accelerator cards / Heterogeneous Systems

- Network
  How fast? Latency? Interconnect? (torus, cube, hypercube, etc)
  Ethernet? Infiniband? Custom?

- Operating System
  Linux? Custom? If just doing FP, do you need overhead

of an OS? Job submission software
Authentication

- Software – how to program?
  Too hard to program can doom you. A lot of interest in the Cell processor. Great performance if programmed well, but hard to do.

- Tools – software that can help you find performance problems

Introduction to Performance Analysis

What is Performance?

- Getting results as quickly as possible?
- Getting *correct* results as quickly as possible?
- What about Budget?
- What about Development Time?
- What about Hardware Usage?
- What about Power Consumption?

Motivation for HPC Optimization

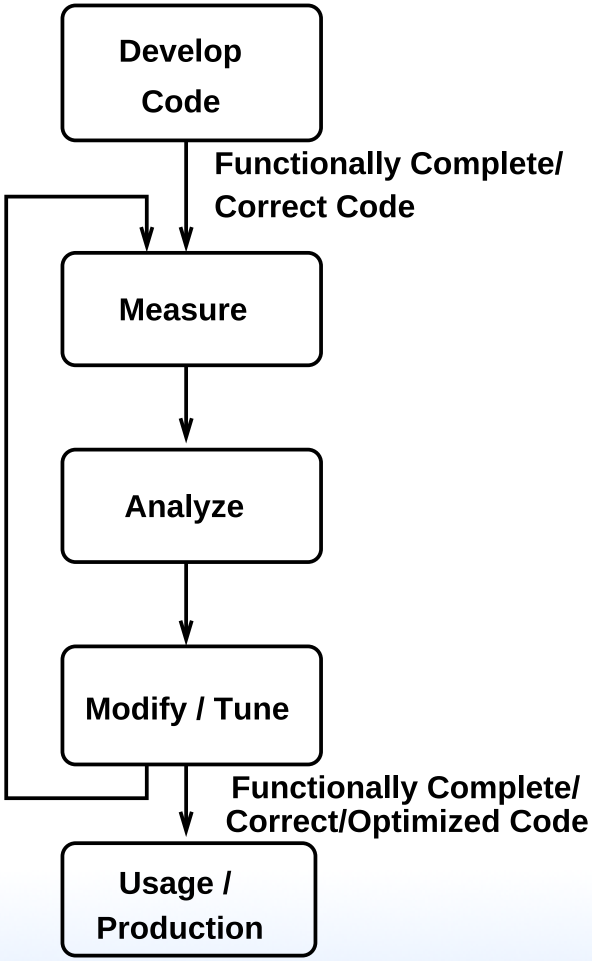
HPC environments are expensive:

- Procurement costs: ~$40 million
- Operational costs: ~$5 million/year
- Electricity costs: 1 MW / year ~$1 million
- Air Conditioning costs: ??

Know Your Limitation

- CPU Constrained
- Memory Constrained (Memory Wall)
- I/O Constrained
- Thermal Constrained
- Energy Constrained

Performance Optimization Cycle


Develop Code
Functionally Complete/ Correct Code
Measure
Analyze
Modify / Tune
Functionally Complete/ Correct/Optimized Code
Usage / Production

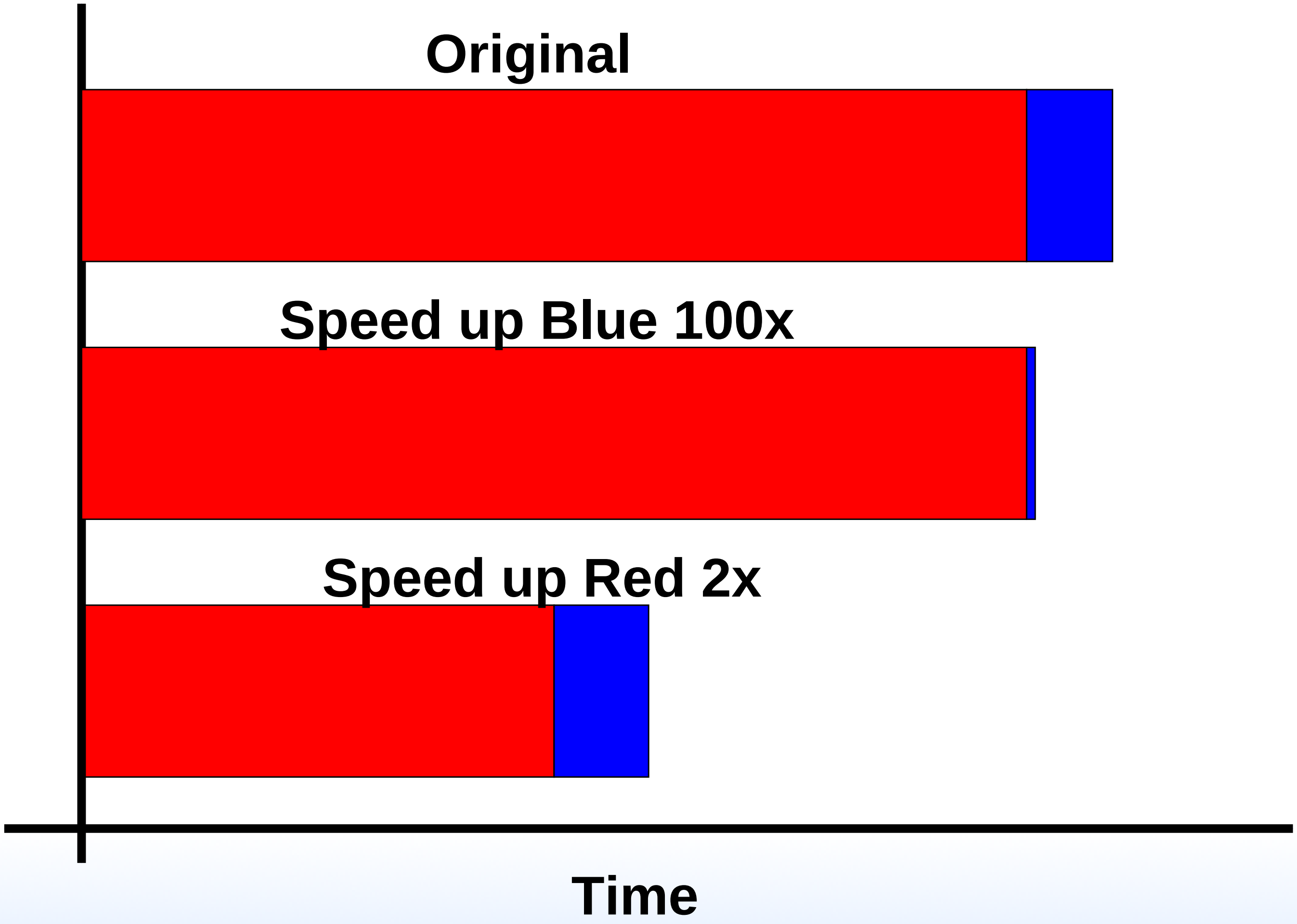
Wisdom from Knuth

“We should forget about small efficiencies, say about 97% of the time:

premature optimization is the root of all evil.

Yet we should not pass up our opportunities in that critical 3%. A good programmer will not be lulled into complacency by such reasoning, he will be wise to look carefully at the critical code; but only after that code has been identified” — Donald Knuth

Amdahl's Law



Speedup

- Speedup for latency $S = \frac{t_{old}}{t_{new}}$ So old took 10s, new took 5s, speedup=2.

Scalability

- How a workload behaves as more processors are added
- Strong Scaling –for fixed program size, how does adding more processors help
- Weak Scaling – how does adding processors help with the same per-processor workload
- Parallel efficiency – $E_p = \frac{S_p}{p} = \frac{T_1}{pT_p}$
- Linear scaling – ideal – $S_p = p$

- Super-linear scaling – possible but unusual