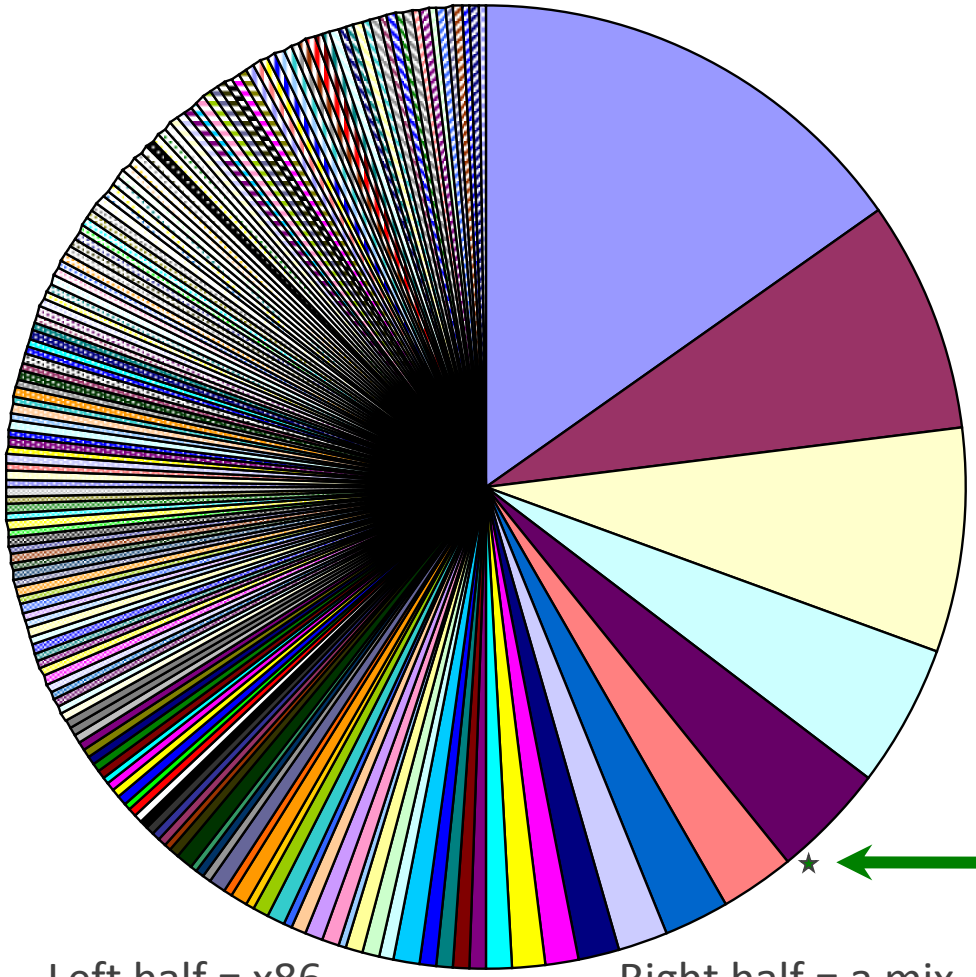
Argonne NATIONAL LABORATORY

High Performance Computing Nine Random Facts

Tom LeCompte
High Energy Physics Division
Argonne National Laboratory

U.S. DEPARTMENT OF ENERGY

1. There is A Lot of Capacity Out There



Left half = x86

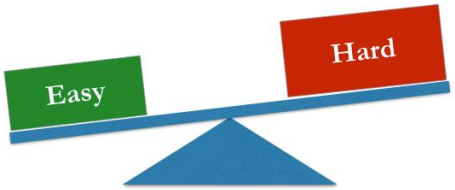
Right half = a mix

- Most of the computational power is concentrated in a small number of machines
  - Half the total power is in the top dozen computers
- Equivalent ATLAS Grid use is about the size of the little green star
- Equivalent ATLAS Grid would be around #27 on this chart

2. Running Athena is non-Trivial

- These machines, particularly the right half, require a recompile
  - Sometimes one gets lucky and copying the binaries works
  - Sometimes it doesn't
  - Sometimes it works, but performance gets worse
  - Sometimes you don't even have an x86 architecture
- There is an implicit assumption that you're running on the grid or a Linux desktop
  - It's OK to ship a compiler out with Athena
  - It's OK to assume CVMFS
- People have spent a lot of effort to size jobs that take ~half a day on a grid node
  - Not helpful when the minimum partition size is 1000x larger
  - Hopefully JEDI will consign this to history

3. Generators are (Sometimes) Easier



- Pythia is trivial – but not very CPU-intensive
- Alpgen (~5% of ATLAS computing) will run up to a few thousand threads with minimal changes
  - With more substantial changes, it will run a quarter-milliion
- Sherpa 2.1 runs with no source modifications to several hundred threads
  - It's not, however, validated by ATLAS
  - Sherpa 1.4 is 10% of ATLAS computing

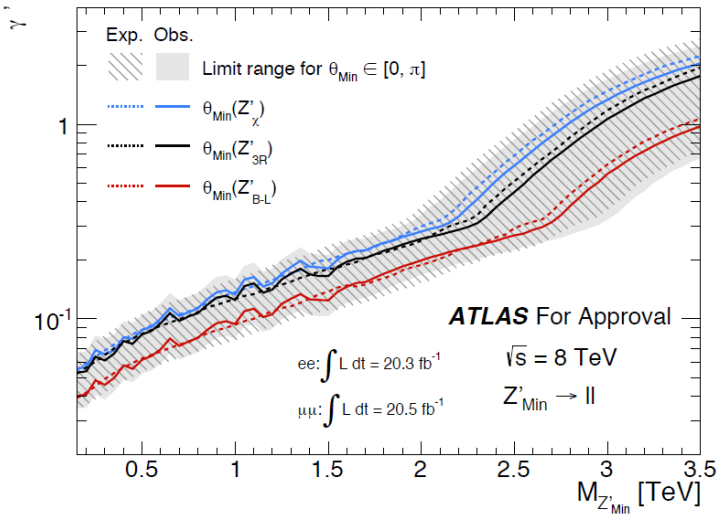
- MadGraph will be a nightmare
  - It compiles as it goes – worker nodes may not have access to compilers (and even when they do, the wisdom of launching 1000 compile jobs is questionable)
  - MadEvent might be OK
- Other generators are small pieces of the overall computing budget

4. Validation is a Political Problem, not a Technical One

- You can never prove a generator/process/dataset is valid
- You can only prove it isn't.
- Consequently, nobody feels empowered to say "Yes, this is OK"
  - But somebody has to

- There are some technical aspects
  - Example: it is possible to speed up a BG/Q floating point calculation and simultaneously improve its accuracy – but this breaks bitwise compatibility
  - Example: if your validation dataset is 100,000 events, and the first 100,000 HPC events are identical, what do you conclude about the next million or ten million?

5. ROOT…ah, ROOT



- The Good:
  - ATLAS has already published a result based on supercomputer-enabled results
    - Z-prime search in a 2-D parameter space
    - Used > 1M hours – saved >6 weeks
- The Bad:
  - Have only done this twice
  - Many people who say “I need more computer time!” actually need more people time.
  - Others started too late.
- The Ugly:
  - ROOT5 (CINT) had a PPC64 bug
    - Makes Mira use impossible
  - ROOT6 (Cling) has an entirely different PPC64 bug
    - Fixed in the next release of LLVM

Has anyone looked at Xeon Phis?

6. Squeezing In Is Key

- Something like 10-20% of the cycles of these machines goes unused
  - "Draining" state – the scheduler needs to make room for a big job.
  - This observation caused a lot of interest in opportunistic running
- There is no single ideal size or duration on any system
  - Sometimes small jobs go in quickly
  - Sometimes it's the larger jobs
- Sometimes availability changes quickly – like minutes
  - Often triggered by a premature end of a big job
  - An argument for late binding

6B. More On Squeezing

- Experience has shown variation in duration when running identical jobs
  - Makes it hard to fit a job into a spot
  - The theory is that it depends on what else is running at the same time.

- We have experience with unreliable resources. Killable jobs might be an option for getting more time.

- These are all ways of getting more time
- The other way of getting more time is getting it allocated to you

7. Getting More Time Is Possible

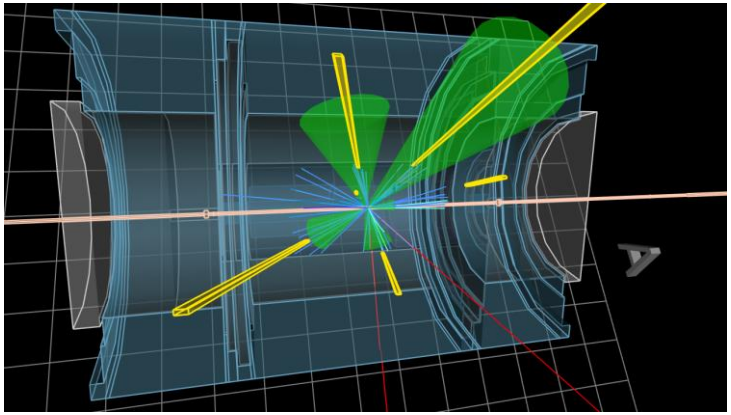
- This round we did well
  - 52 million hours through the ALCC process
  - 13 million hours through Directors Discretionary

- What about next year?
  - We have to demonstrate that we have used this year's time effectively
  - This year, Cosmic Frontier received 100M hours to be shared
  - Energy Frontier could make a similar proposal
    - Such a proposal requires discussion of governance as well as science (how is it decided who gets what?)
    - Personally, I'd be interested in being a co-I on this, but less interested in being the PI (we should rotate)

8. Running PanDA Jobs Is Vital to The Long-Term Future

- 50 million hours of work can be submitted by hand
- 100 million hours probably can't.
- To use these resources effectively, we need to connect them to PanDA
  - Technical issues: there are more states to an HPC job that need to be signaled back
  - Political issues: very mixed messages from DC on identities and authentication
- Many pieces of a solution exist – but don't underestimate the time and effort in integrating them.

9. HPCs Have Unexpected Benefits



- HPCs have been able to run jobs that fail on the Grid
  - This is a Z → ττ + 5 jets event, highly filtered
  - Grid jobs fail because sometimes zero events pass the filters
  - HPCs don't have that problem
- HPC workflows can save computing time
  - Warmups can be done once and reused – saves 10's of thousands of hours
- HPC workflows allow more physics
  - Because they are in one place, it's easy to shower the Alpgen events multiple ways
- Every cycle recovered from the Grid is a cycle we are free to spend any way we want.