

Performance metrics panel

Dan Pryor

IDA/CCS

How we are different

- Integer Use
- Bad data locality
- Extended precision
- Extreme variation in load
 - parallel, vector, scalar, Big-I, Little-O
- Throw-away code
- Almost all code is C

How we are the same

- We want to get closer to peak
- Standards are important
 - portability, parallelization
- We want to cut human time
 - ease of use, model simplicity
- We want more, more, more

Wanted: “FFT” from App space to machine space

- Feed in application load
- Transform to Vendor Space
- Look for performance spikes
- Look for Price performance spikes

Why don't we have a good metric for HPC performance?

- Complexity of:
 - machine space
 - workload space
- Evolving complexity of ``
- We can't even accurately represent our load

Is there a chance to define a single metric?

- Not a single metric, but we can do better
- We need to develop some probes that allow a better decomposition of applications along architectural-feature axes
- Even then, “usability” is elusive

What are the requirements for such metrics?

- Tunability of statement (so that the output is a graph, rather than a number)
- Input variation exposes (clearly) architectural features --- an explanation of the performance can be inferred from output
- Measures degradation from some optimal
- methods to mix and match different metrics in the same benchmark (we can't expect linear behavior)

What needs to be done to get new metrics accepted, if at all?

- Quality of the metric
 - Success in predicting performance
 - range of applicability
 - inertia

Holy Grail or Fata Morgana?

- Holy Grail, I hope
- Fata Morgana, I fear
- It is absolutely worth the attempt