

Performance Panel Topics

HSF Meeting Orsay 2016

Panel

- The short talks by different experiments and communities covered
 - Current practice
 - Plans
 - Problems
- Discussion
 - What does performance mean for your experiment/community?
 - Different areas within experiments
 - Trigger/Physics/Sim/Analysis
 - What are the problems ?

Do we understand the problem?

- Do we have the appropriate metric for performance/efficiency?
 - How does this metric links "physics products" to investments/effort
 - Y Reconstructed-Simulated-analyzed-events within a week for 1Euro ?
- High Performance / High Throughput Computing
 - Do we sufficiently discriminate between the two?
- Different technologies/approaches
 - Benefits
 - Constraints
 - Opportunities
 - Sustainability

Topics

- How do other communities, labs approach performance?
 - Complexity, scale, problems
 - Inviting “other” people
- How to optimize on GPUs
 - what corresponds to hardware counters etc.?
 - could a Hackathon specific to this topic help?
 - For HPC people this should be old news, how do they handle this?

Architectures and Performance

- How to compare different architectures?
- Can we estimate Development effort vs Benefit vs Rate of Change for technologies?
 - FPGA reco has to be how much faster to be worth it?
- Traditional multi-core evolution (now, 5-10 years)
- Many core, can we deal with it?
- GPUs
- HPC
 - HPC = ManyCore + GPUs + low latency network + no external network ??
 - How to make best use of HPC resources ?
- Shift in Commodity Computing?
 - New main platforms based on mass produced devices (Mobile phones etc.)?
 - Is this relevant for HEP?
- Low Power Computing?
 - Can the power question be absorbed by a cost model?
- Specialised Hardware?
 - FPGAs, new memory , etc.....

Concurrency

- Current understanding and plans
 - threading
 - vectorization
 - "MPI"
 - Affinity
 - IPC
 - HEP Frameworks?

I/O

- Do we understand the cost of I/O?
 - Including serialisation/ de-serialisation
 - Impact of latency / bandwidth
- Concurrency
- “disks” (hierarchical persistent storage)
- Network
- Data model (impact on framework?)

Frameworks and their role

- support for heterogeneous computing
- efficient concurrent “task” scheduling
- efficient “jobs” scheduling
- “external” software integration
- Data model (impact on I/O?)

Workflows/Tasks

- Current workflows and the impact on performance
 - Demand on I/O (finale products only, intermediate products and logs)?
 - Demand on CPU/Mem?
 - Do we need to take success/fail rates into account

Tools/Process/Validation

- Tools
 - Existing/Gaps
- Instrumentation
 - At what level, how much?
 - How to handle the data?
- Metrics?

- Process
 - Revolution vs Evolution
 - Adopt, Adapt, Reinvent
 - Task-forces, tiger-teams, daily-work
 - Identify the “Source of all evils”
- Validation:
 - reproducibility among heterogeneous platforms

Support for development and benchmarking

- Access to advanced hardware
- Software installation and support
- Standard "candles"
 - Reference workloads that cover relevant behaviour of our jobs

Knowledgebase

- How to spread and maintain knowledge?
 - Web courses?
 - repository
 - training
 - Hackathons