RAISE Center of Excellence

Benchmarking HEP workflows on HPC

David Southwick

In collaboration with HEPiX benchmarking working group

Background

Efficient exploitation of HPC resources presents unique challenges: Scaling workload execution adds layers of complexity not captured in traditional compute environments

- Permissions:
  - Environment (containerization helps)
  - Monitoring (I/O, network, performance bottlenecks, etc)
- Connectivity:
  - isolated worker nodes
  - site connectivity (big data ingress/egress)

To successfully exploit HPC resources we need to understand efficiency both in terms of compute and data access.

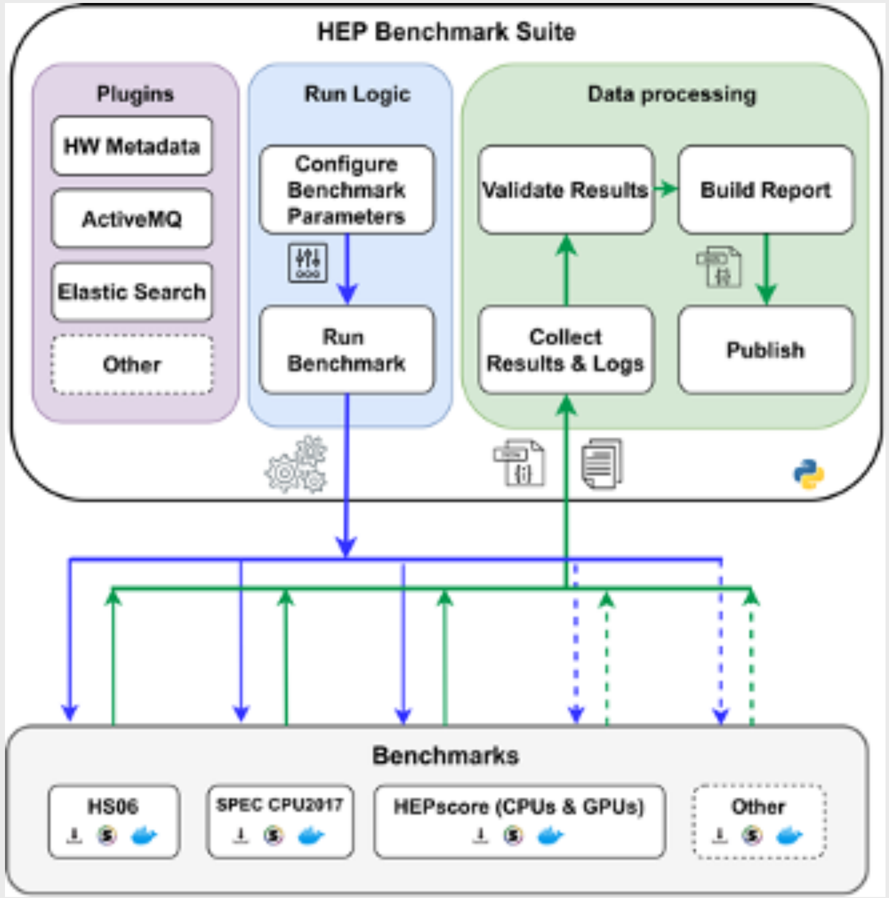
Context: Benchmarking at CERN

HEP Benchmark Suite: A benchmark orchestrator & reporting tool.

Executes an array of user-defined benchmarks & metadata collection

Features that accommodate HPC:

- Minimal dependencies (Python3 + OCI container)
- Automated or batch result reporting (AMQ/Elastic)
- Scheduler agnostic, unprivileged
- Modular, easily extendable

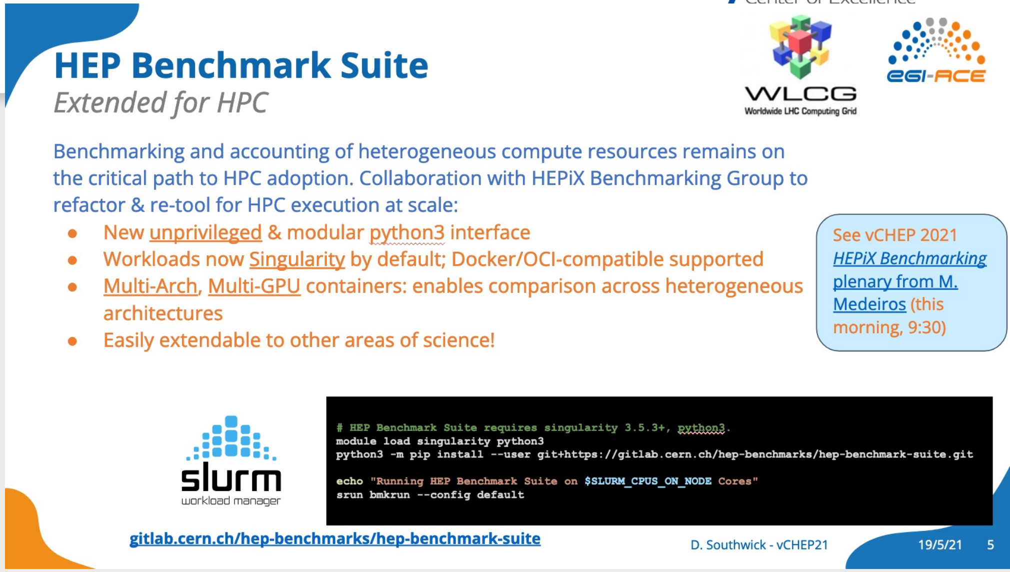


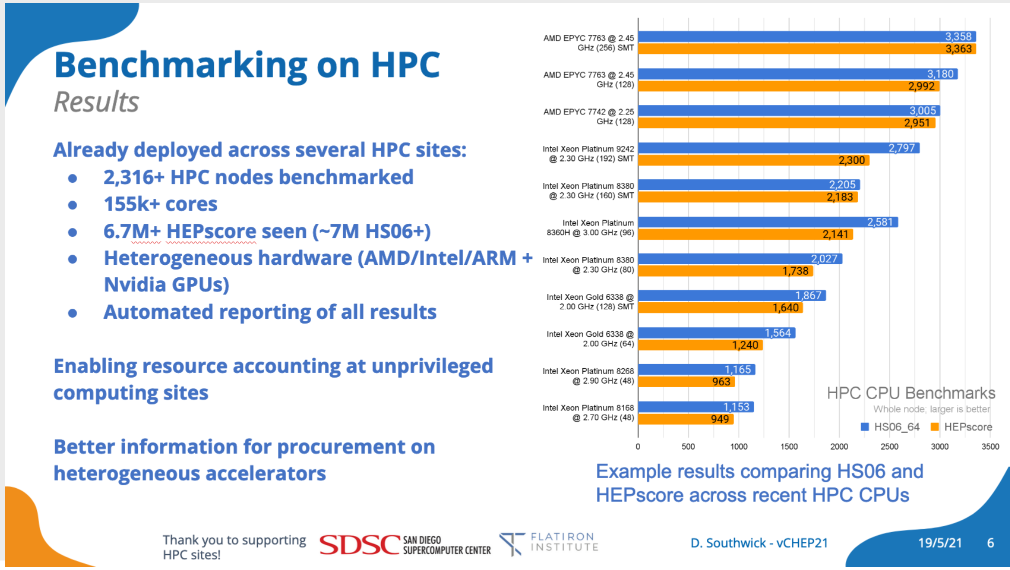
https://gitlab.cern.ch/hep-benchmarks/hep-benchmark-suite

Successes at HPC centers

HEPscore (executed by the HEP-Benchmark-Suite) has already been used for large scale deployments and studies at HPC sites:

- Initial experiences from vCHEP'21
- 200,000-core campaign with Run-2 production WLs
- Scale studies of new/upcoming AMD cpus
- Experiment HPC exploration / adoption





[See vCHEP'21 presentation for full details]

RAISE Center of Excellence

What's new?

First look of run3 workloads - many with heterogenous flavors:

- First ARM, GPU development workloads
- GPU *vs* CPU *vs* GPU+CPU benchmarking studies
- Heterogenous partition studies (ARM+GPU)
- ML / AI workload development (MPI scaled to ~200 GPUs)

Quality-of-Life updates:

- Batch uploading (post-run: supports "secure" worker nodes)
- GPU / accelerator meta-data inclusion
- CVMFS-attached benchmarking campaigns

Thank you to all partner HPC sites enabling this work!

Heterogeneous workloads

Experiments have been hard at work exploiting additional instruction sets outside of traditional x86. These architectures may offer much better energy efficiency, or higher availability at less popular HPC partitions

- ARM: workloads available directly with HEPscore
- POWER: workloads under development (if CMS/others produce)
- Open-like: (OpenCL, python-based, etc) – available directly with HEPscore

GPU workload performance

Considerable percentages of site total computing power increasingly reside in GPUs. HEP workloads with "simple" kernels (*embarassingly parallel)* can profit by orders of magnitude – HEPscore provides workloads that run on both:

Preliminary testing on HPC enables direct comparison of same codebase and same hardware:

Xeon Gold 6148 @ 2.4Ghz, Nvidia V100

PRELIMINARY

Workload	CPU only	GPU only	Speedup	Time(CPU)	Time(GPU)
MadGraph5	0.026(float)	0.744	28x	29m 8s	11m 8s
CMS-HLT	525	9,450	18x	23m 9s	17m 15s
ML particle flow (epoch)	659s	138s *1 GPU	4.8x	33m 36s	8m 29s

Typical HPC single node resources:

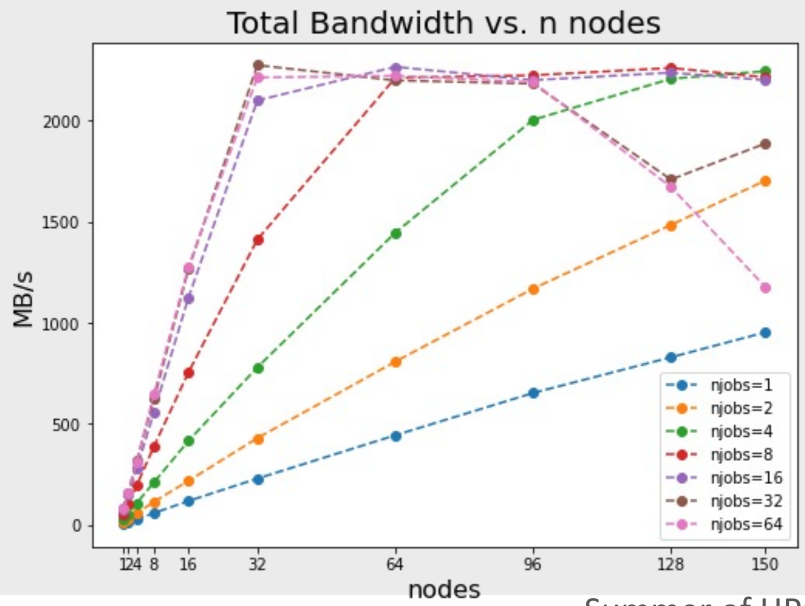
2x AMD EPYC:	256 threads
4x Nvidia V100:	20,480 cuda cores
4x Nvidia A100:	27,648 cuda cores
4x Nvidia H100:	*67,584 cuda cores**

* https://www.nvidia.com/en-us/data-center

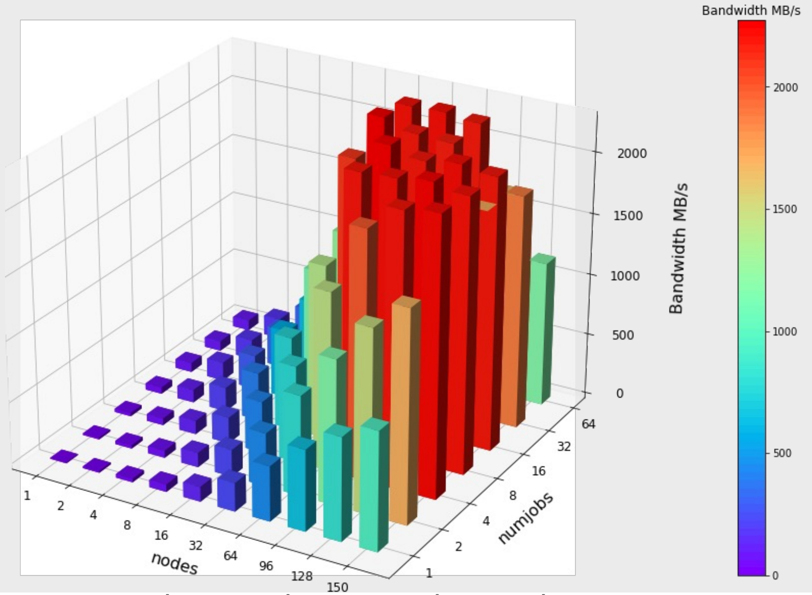
Non-compute benchmarking

Scaling and bottlenecks in Big Data

- Data-driven workloads demand performant storage and connectivity (which are shared!)
- Bottlenecks here significantly throttle job performance
- Capacity, capability, and monitoring not typically advertised by
  - See HEP benchmark WG studies using PRmon for efficiency monitoring



Peak	Bandwidth
16 node	2.2 GB/s



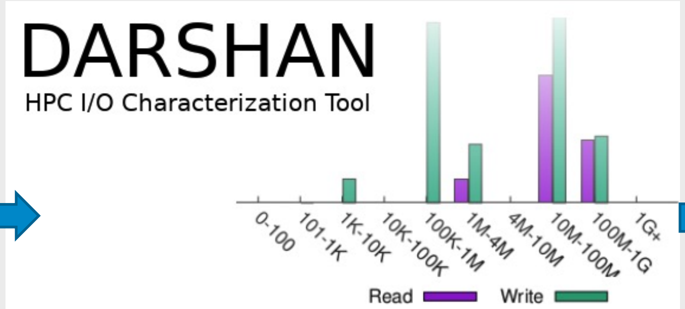
Summer of HPC student HPC I/O investigation C. Cocha, A. Filipcic, V. Khristenko

Workload I/O benchmark

Problem: Unclear how many data-driven workloads a given site may support without bottleneck shared resources

- Development of a *workload I/O benchmark*
- tune to the **I/O patterns of real workloads** to better inform reasonable scaling capabilities at a given HPC site
- More representative than sequential throughput metrics
- Uncover **I/O bottlenecks** (excessive file opens, read patterns, cache issues)

HPC workload

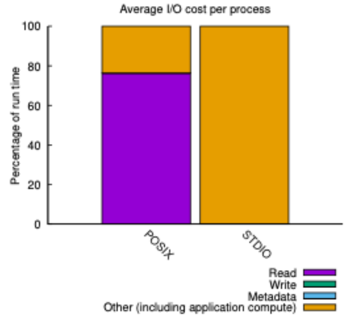


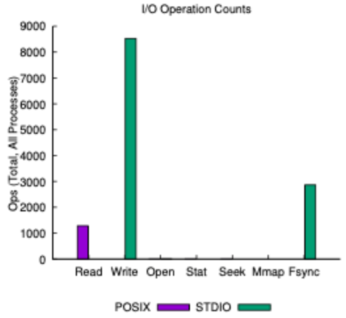
IoR
HPC benchmarks

https://github.com/hpc/ior
https://github.com/darshan-hpc/darshan

jobid: 2190289	uid: 1005	nprocs: 1	runtime: 6 seconds

I/O performance *estimate* (at the POSIX layer): transferred 172.4 MiB at 37.65 MiB/s
I/O performance *estimate* (at the STDIO layer): transferred 0.1 MiB at 63.62 MiB/s





Most Common Access Sizes
(POSIX or MPI-IO)

	access size	count
POSIX	49284	141
	20873	3
	204628	3
	204758	2

File Count Summary
(estimated by POSIX I/O access offsets)

type	number of files	avg. size	max size
total opened	2	950M	1.9G
read-only files	1	1.9G	1.9G
write-only files	1	69K	69K
read/write files	0	0	0
created files	1	69K	69K

Outlook

Lots of development this past year accelerated by HPC, and certainly momentum will only continue!

- Benchmark I/O performance, scaling benchmark for HPC
- First AMD GPU partitions online now, tests underway (LUMI-G)
- New Nvidia H100 testing underway
- ARM partitions coming later this fall
- openMPI workloads (ML, distributed jobs)

drive. enable. innovate.

SDSC SAN DIEGO SUPERCOMPUTER CENTER

FLATIRON INSTITUTE

LUMI

JÜLICH Forschungszentrum

RAISE Center of Excellence

The CoE RAISE project has received funding from
the European Union's Horizon 2020 –
Research and Innovation Framework Programme
H2020-INFRAEDI-2019-1 under grant agreement no. 951733

Follow us:

Understanding workload efficiency

- PRmon plugin to HEP benchmark suite enables profiling of CPU utilization
- Profile both native and containerized workloads
- Identify issues, acceptance testing, verification

PRmon source: https://github.com/HSF/prmon

https://indico.cern.ch/event/1078853/contributions/4576275