CERN openlab

# WLCG benchmarking using HEP experiment workflows

*Extending HEP Benchmark Suite*

David Southwick
*In collaboration with*
WLCG Benchmarking Working Group
CERN, SKAO, GÉANT and PRACE Collaboration
EGI-ACE

26/3/2021

# Introduction

*Benchmarking for Worldwide LHC Computing Grid (WLCG)*


Historically, benchmarking has been CPU focused:

- Matching Experiment requests and site pledges
- Accounting of CPU usage
- Procurement requirements

Subset of SPEC-CPU has been used for past decades,

**HEPspec06** remained in use for 12+ years (after spec17 found to be sub-optimal match)

- Initially designed for WLCG; found widespread use in other communities (not limited to HEP)
- Single number result, definition did not change during lifetime

# Looking for the next benchmark

*Conditions have evolved*

Computing has changed significantly since adoption of HEPspec06:

- 64bit, new compilers, SMP, VMs, containers, compute accelerators
- Reports of scaling deviations of 40% and more

HEPspec06 requires a software license from SPEC

- Strong desire to consider license-free successor

Not Representative of full machine potential, nor improved experiment workloads

Working Group formed to develop the next benchmarking tool; to be based on **experiment workloads**

# HEP Benchmarks Project

*Comprising of three primary components*

**HEP Benchmarking Suite:**

- A benchmark orchestrator, metadata collector, and reporting tool
- Allows a top-level entry point for automated benchmarking sets
- Orchestrates HEPscore, HS06, SPEC CPU2017, and more…

**HEP Score:**

- Orchestrates & tabulates running of a series of HEP Workloads
- Compute HEPscore value (similar to HS06 score)
- Generate reports of whole set execution

**HEP Workloads**

- Individual reference HEP workloads (eg CMS RECO).
- Common build infrastructure & containment

Find more at:
https://gitlab.cern.ch/hep-benchmarks

# HEP Benchmark Suite 1.0

*Developed over 2+ years*

Build on success in providing an array of benchmarks, including HS06 and newly developed **HEPscore.** Initially:

- Designed for WLCG compute environment
- Intended for procurement teams, site administrators
- First with VM containment, later nested docker images

***None of these approaches are compatible with HPC!***

- Refactored & re-tooled for execution outside WLCG sites during 2020

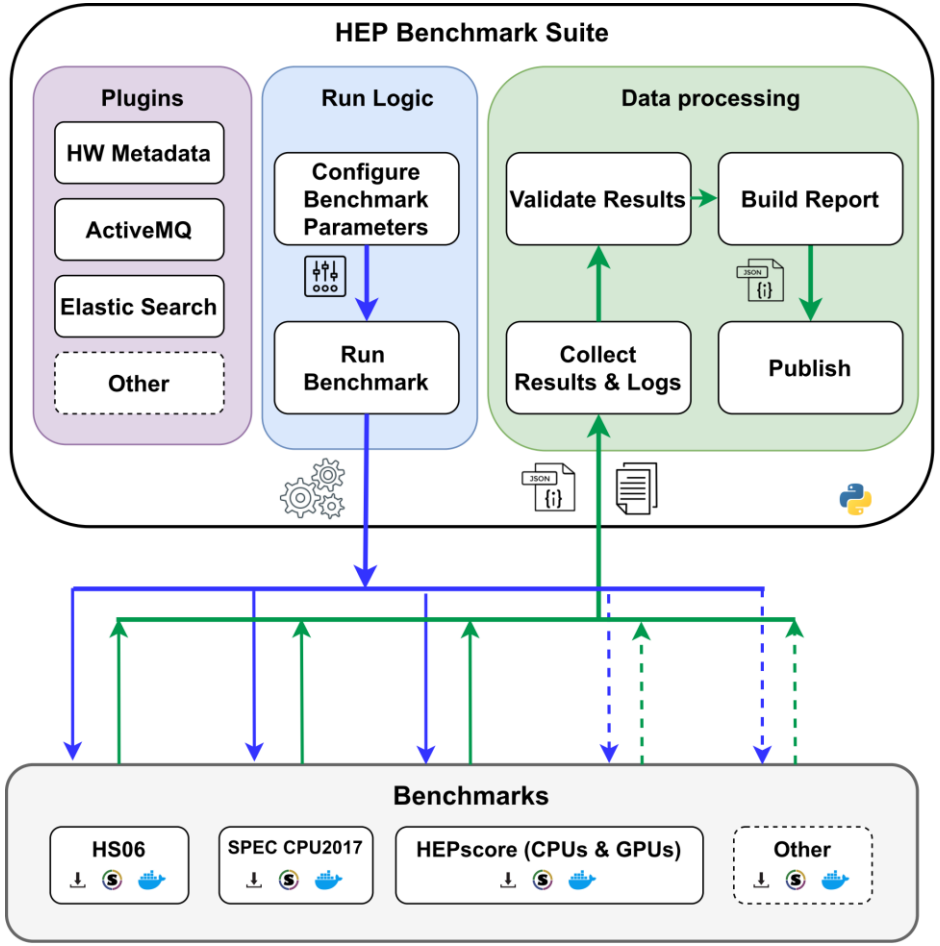
# HEP Benchmark Suite 2.0

*Improvements since 2020*

Minimal Dependencies
***Python3.6 + container choice***

Variety of containment choices
***Singularity (incl. CVMFS Unpacked), Docker, Podman***

Repeatable & Verifiable
***Declarative YAML config***




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

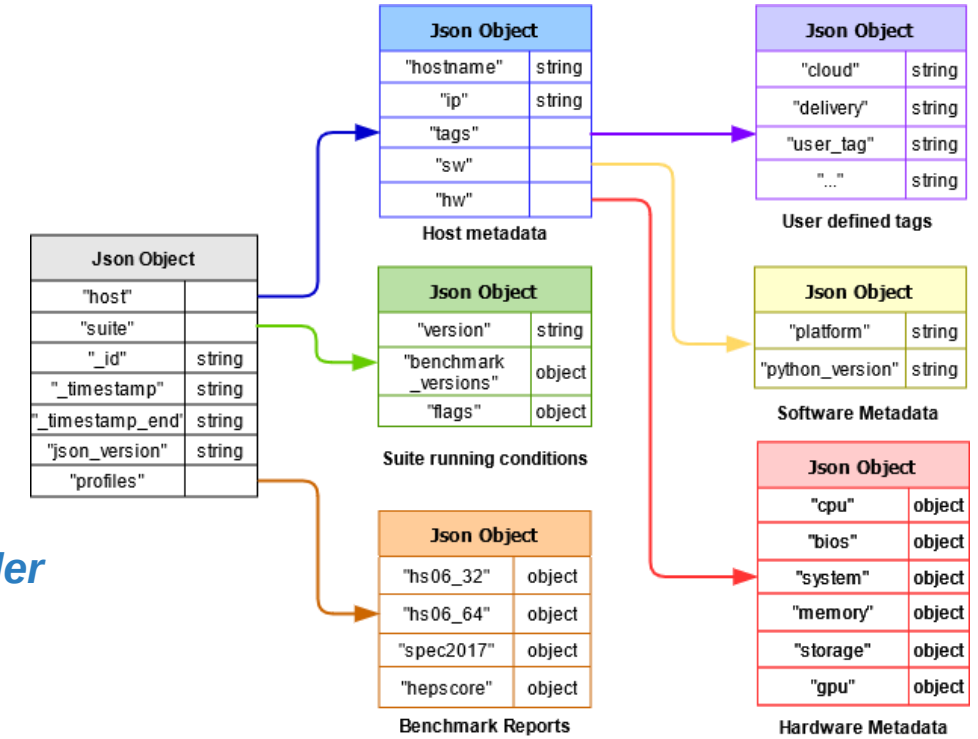
# HEP Benchmark Suite 2.0

*Features (cont.)*

Modular Design
***Snap-in workloads & modules***

Designed for Ease-of-Use
***Simple integration with any job scheduler***

Metadata + Analytics
***Automated Reporting via AMQ***




Hardware Metadata snap-in module API example
https://gitlab.cern.ch/hep-benchmarks/hep-benchmark-suite

# HEP Score

*Candidate replacement for HS06*

- Modular python3 “microservice” approach
- Importable / Extendable, architecture agnostic
- Executes set of containerized workloads (Singularity, Docker, Podman)
- Produced single configurable **weighted geometric mean** of resulting workload scores
- Detailed report delivered in JSON/YAML report

**Summary of currently supported HEP workloads**

| Experiment | Name | Description | Experiment license | Latest Container | Readiness | Pipeline status |
|---|---|---|---|---|---|---|
| Alice | gen-sim | link | GNU GPL v3 | docker | w.i.p. | pipeline passed |
| Atlas | gen | link | Apache v2 | docker | Y | pipeline passed |
| Atlas | sim | link | Apache v2 | docker | Y | pipeline passed |
| Atlas | digi-reco | link | Apache v2 | docker | w.i.p. | pipeline passed |
| CMS | gen-sim | link | Apache v2 | docker | Y | pipeline passed |
| CMS | digi | link | Apache v2 | docker | Y | pipeline passed |
| CMS | reco | link | Apache v2 | docker | Y | pipeline passed |
| LHCb | gen-sim | link | GNU GPL v3 | docker | Y | pipeline passed |
| Belle2 | gen-sim-reco | link | GNU GPL v3 | docker | Y | pipeline passed |

https://gitlab.cern.ch/hep-benchmarks/hep-workloads

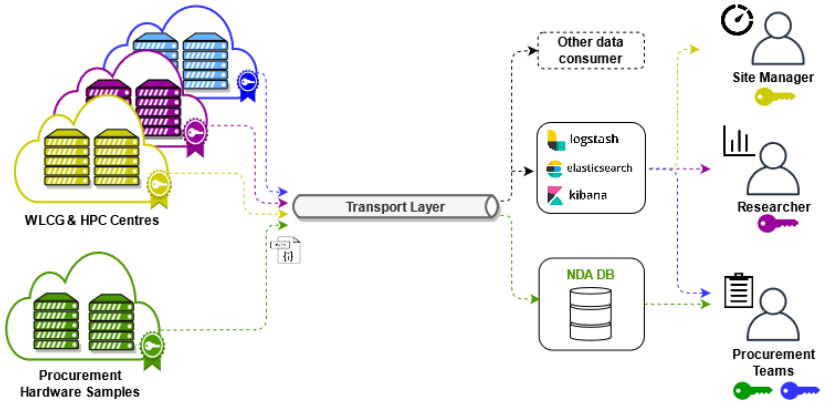
# Features on HPC

## Benchmarking Heterogeneous architectures

- **Multi-Arch container** workloads (x86, IBM Power, ARM …)
- **Multi-GPU container** workloads (Nvidia, AMD, Intel… )
- Easily extendable to other sciences!

## Simple integration with SLRUM & other job orchestrators

- Single dependency on site Python3.6




slurm workload manager

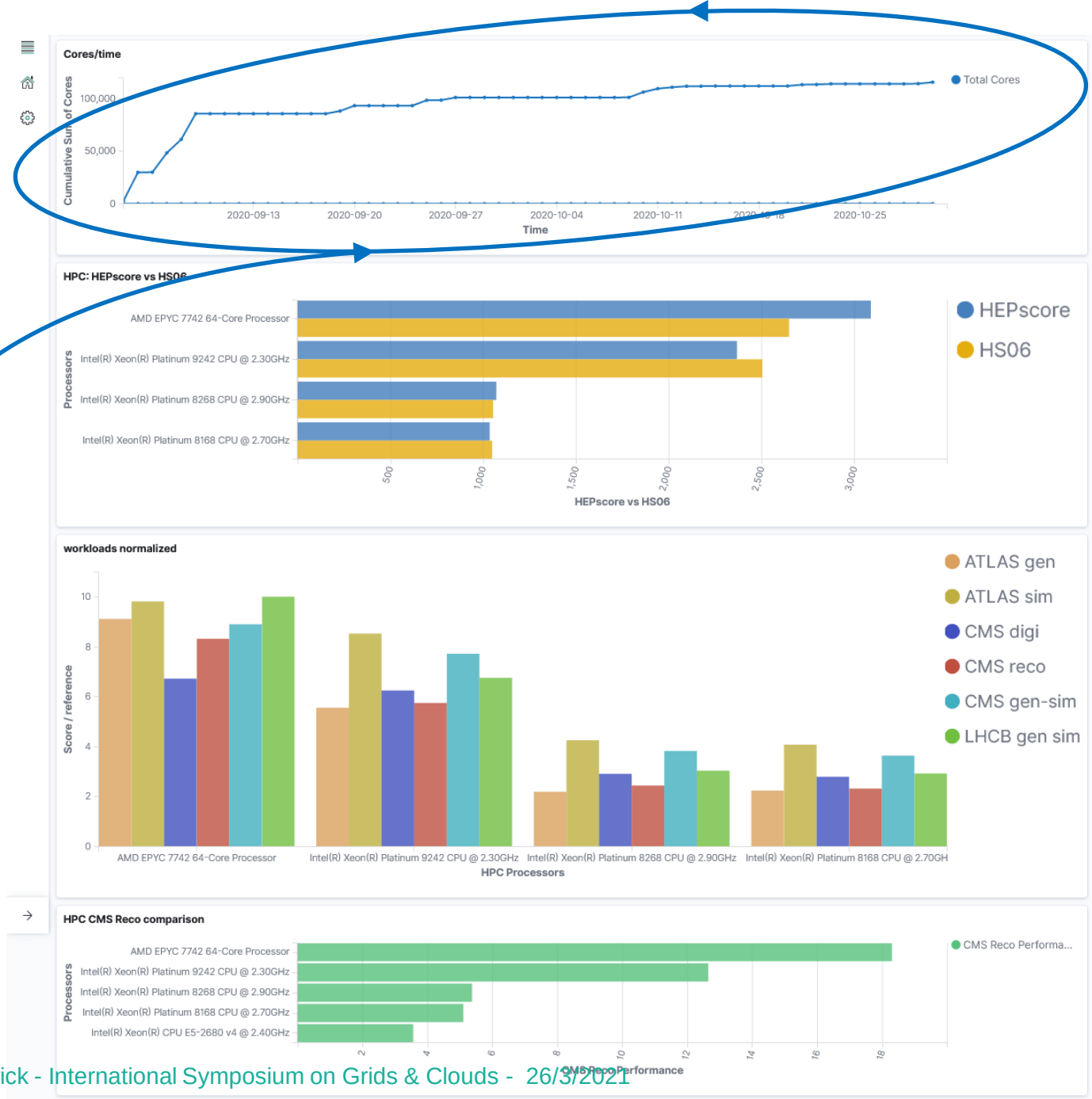
```
# HEP Benchmark Suite requires singularity 3.5.3+, python3.
module load singularity python3
python3 -m pip install --user git+https://gitlab.cern.ch/hep-benchmarks/hep-benchmark-suite.git

echo "Running HEP Benchmark Suite on $SLURM_CPUS_ON_NODE Cores"
srun bmkrun --config default
```

# Aggregation

*Analysis via Kibana*

Short Benchmarking Campaign
~120,000 cores
(Sept – Oct)



# Examples

*Running HEP Benchmark Suite, HEPscore*

Execution is as simple as importing a python module!

```
# Python 3.4+, pip3 19.1+
python3 -m pip install --user
git+https://gitlab.cern.ch/hep-benchmarks/hep-benchmark-suite.git

bmkrun --config default
```

You can override YAML settings via CLI arguments!

```
# Define custom config (even remote!), override it's contents
bmkrun --mode docker --config https://your.server/static_config.yaml \
    --rundir /mnt/scratch/hep --benchmarks hepspec06
```

Run HEPscore in the same manner, without the need for the suite!

```
# Run standalone HEPscore
python3 -m pip install --user
git+https://gitlab.cern.ch/hep-benchmarks/hep-score.git

hep-score /mnt/scratch/hep-score

# Override settings similar to suite
hep-score --conffile /home/dsouthwi/hepscore.yaml -m docker /tmp/hep
```

YAML examples available in backup slide

# Running HS06 on ARM

HEPspec06 multi-arch demonstrated on

AWS Gravitron2 bare-metal server, ARM64

```
$ lscpu
Architecture:                    aarch64
CPU op-mode(s):                  32-bit, 64-bit
Byte Order:                      Little Endian
CPU(s):                          64
On-line CPU(s) list:             0-63
Thread(s) per core:              1
Core(s) per socket:              64
Socket(s):                       1
NUMA node(s):                    1
Vendor ID:                       ARM
Model:                           1
Model name:                      Neoverse-N1
Stepping:                        r3p1
BogoMIPS:                        243.75
L1d cache:                       4 MiB
L1i cache:                       4 MiB
L2 cache:                        64 MiB
L3 cache:                        32 MiB
NUMA node0 CPU(s):               0-63
```

```
$ echo ${SECRET_URL} | /hep-spec/scripts/hep-spec.sh -w $BMK_RUNDIR -b $BMK -m $BMK_OPTION -p ${SPEC_DIR} -u -
############################
        CERN HEPSPEC
Sat Feb 27 17:16:00 UTC 2021
############################
2021-02-27T17:16:00 [/hep-spec/scripts/hep-spec.sh] Variable values:
      HEPSPEC_SOURCEDIR=/hep-spec/scripts
      HEPSPEC_BMK=hs06
      HEPSPEC_NUMPROC=64
      HEPSPEC_PATH=/scratch/HEPSPEC/CI_hs06_ext_g2_bare_12384842
      HEPSPEC_SET=all_cpp
      HEPSPEC_MACHINE_OPTION=default
      HEPSPEC_ITERATIONS=3
      HEPSPEC_WORKDIR=/scratch/jobs/hs06_ext_g2_bare_12384842/hep-spec
      HEPSPEC_DEBUG=0
```


```
{"hs06":{"start":"Sat Feb 27 17:19:26 UTC 2021", "end":"Sat Feb 27 20:10:46 UTC 2021", "copies":64,
          "runcpu_args":"1 runspec: runspec --define machine_option:64 --config=linux_gcc_cern.cfg --action=build all_cpp;64 runs
pec: runspec --define machine_option:64 --config=linux_gcc_cern.cfg --nobuild --noreportable --iterations=3 all_cpp;", "bset":"all
_cpp", "LINK":"  6  g++ -O2 -fPIC -pthread  -DSPEC_CPU_LP64    <objects>     -o options;  1  g++ -O2 -fPIC -pthread  -DSPEC_CPU_LP
64 -DSPEC_CPU_LINUX   <objects>     -o options;", "hash":"7b84bb375cee11731a958a26d6fc155d",
          "score":1170.998, "avg_core_score" : 18.296, "num_bmks":7 ,"bmks":{ "444.namd":[  23.5, 23.5, 23.5, 23.5, 23.5, 23.5, 2
```

# Adoption

*Current status*

Deployed across several HPC sites in the past year, several more currently testing independently

Fully adapted to multi-architectures and variety of container solutions

- Agnostic – same image for all supported modes (ie single HEPspec image using multi-arch tag, single GPU image – determine mode at runtime)

WLCG HEP Score Deployment task force decision deferred until LHC-run3 experiments examined – Benchmark suite framework is ready for widespread use

# Next Steps

*Roadmap tasks for 2021*

- Include LHC Run-3 workloads proposed by WLCG HEP Score task force
- Add more alt-arch workloads as they become available
- Support for FPGA already enabled, add experiments as available
- Continue integration of GPU workloads (MC MadGraph, Atlas O2O)
- Run on several PRACE-Tier0 HPC sites
- Adoption by WLCG as official replacement for HS06

# Conclusions

*HEP Benchmark Suite: Extended to benchmark HPC*

- **Designed & tested for generic job submission WLCG + HPC: Flexible, modular, and unprivileged**
- **Heterogeneous support included (multi-GPU and multi-arch containers) Fault-tolerant, Repeatable & Verifiable**
- **Enables statistical & historical analysis via automated reporting**
- **Verified on several HPC sites, soon on PRACE-tier0 testbeds**

**Suite is ready and available for use today! Please start trying & report issues. Using it is easy!**

**Join our Discourse Forum, or find Ticket tracking in Gitlab**
**Full source and examples at:**
**gitlab.cern.ch/hep-benchmarks/hep-benchmark-suite**

# THANK YOU




# SLURM batch example

```
#!/bin/bash
#SBATCH --exclusive --hint=multithread
#SBATCH --job-name=HEP-Benchmark-suite
#SBATCH --output=HEP-result-%A-%j.out
#SBATCH --mail-type=END,FAIL
#SBATCH --mail-user=david.southwick@cern.ch
#SBATCH --array=1-200

module purge
# HEP suite requires singularity 3.5.3+, python3.
module load singularity python3

export RUNDIR=/tmp/HEP
export BMKSUITE_TAG_SITE="SDSC"

echo "Running HEP Benchmark Suite on $SLURM_CPUS_ON_NODE Cores"
python3 -m pip install --user git+https://gitlab.cern.ch/hep-benchmarks/hep-benchmark-suite.git
# run
srun bmkrun --config default --tags --rundir $RUNDIR

# Copy local JSON & LOG results to $HOME if not reporting via AMQ
find ${RUNDIR} \( -name \*.json -o -name \*.log \) -exec tar -rvf $HOME/results-${SLURM_JOB_ID}-${SLURM_ARRAY_JOB_ID}.tar {} +
```

# Declarative YAML configuration

# Hashed & included in report

```yaml
global:
  # Type of container technology to use: Singularity or Docker
  mode: "singularity"
  # Run directory where all related suite material will be placed.
  rundir: "/tmp/hep-benchmark-suite"
  benchmarks:
    - "hepscore"
    - "db12"
    - "hs06"
    - "spec2017"
  # User defined tags that will show on the metadata file
  tags:
    cloud: “DEEP"
    vo: “DEEP-EST"
    other_tag: “V100"
  # enable AMQ reporting using credentials in activemq
  publish: False

activemq:
  server: 'your-AMQ-server.com'
  port: 61613
  topic: 'hepscore-topic'
  #username: 'user'
  #password: 'pw'
  #key: 'key-file.key'
  #cert: 'cert-file.pem'

hepspec06:
  image: "/cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/hep-benchmarks/hep-spec/hepspec-cc7:v1.0"
  hepspec_volume: "/tmp/SPEC"
  iterations: 3

spec2017:
  image: "/cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/hep-benchmarks/hep-spec/hepspec-cc7:v1.0"
  hepspec_volume: "/tmp/SPEC"
  iterations: 3

hepscore:
  version: "v1.0rc13"
  config: "default"
```

# Collaborating HPC Sites

*...and their hardware*

**San Diego Supercomputer Center**

- Intel Skylake 8168 @ 2.8GHz
- Intel Cascade Lake 8268 @ 2.9GHz
- Intel Skylake 6148 @ 2.4GHz
- AMD EPYC 7742 @ 2.25GHz

**Advania (now atNorth) - Iceland**

- Intel Cascade Lake 9242 @ 2.3GHZ

SDSC SAN DIEGO SUPERCOMPUTER CENTER

atnorth

advania

**We thank you for your support!**

# Backup talks & resources

*For further details see previous presentations below*

HEPspec06 -> HEPscore talk 26/5/20

https://indico.cern.ch/event/917098/contributions/3855129/attachments/2045174/3426154/WLCG-MB-26-05-2020-giordano.pdf

Status Report to HEPiX on benchmarking suite 13/10/20

https://indico.cern.ch/event/898285/contributions/4034096/attachments/2121862/3571531/HEPiX-Workshop-13-10-2020-giordano.pdf

HEPiX Status update 2021

https://indico.cern.ch/event/995485/contributions/4257475/