

HEP Reference Workloads in Containers

D. Giordano (CERN)

domenico.giordano@cern.ch

HEPiX Benchmarking WG

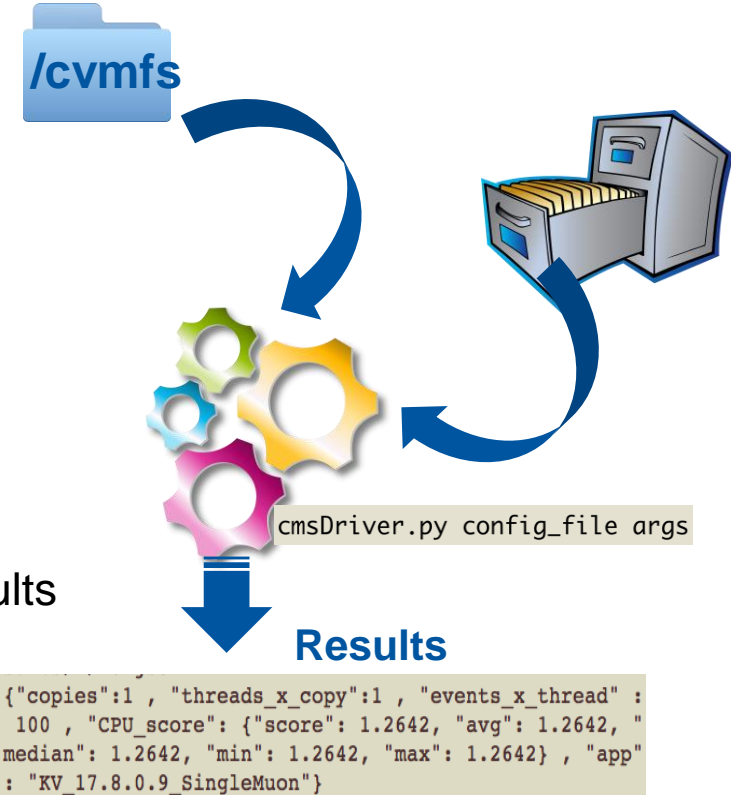
18 Jan 2018

Objective: standalone HEP benchmark workloads

HEP workloads' ingredients:

1. SW repository (cvmfs)
2. Input data (root files and DB conditions)
 - Typically accessed via xrootd
3. An orchestrator script
 - Configure environment
 - Run HEP SW
 - Parse produced output and create score results

We are encapsulating all this in standalone containers



Why adoption of standalone containers?

Requirements for a HEP benchmark suite

(As discussed in previous meetings)

- Must be adopted not only by individuals inside the experiments but also by external **performance experts**, **site procurement** teams and **hardware vendors**
 - Usability:
 - Simple instructions: *Insert disk, run shell script, wait, and read and report score*
 - `docker run --rm -v /tmp/results/:/results $IMAGE`
 - Accessibility:
 - No remote data access from vendor's office
 - *With containers the benchmark can be distributed as full tarball in a drive*
 - Free License:
 - Follow the experiment code license
 - Long term commitment from the experiments
 - Provide the software support for several years
 - *Code is in cvmfs*. Data preservation initiatives can support it

Example: LHCb

- Starting from the **Concezio's** doc [*]
 - and with the help of **Ben Couturier**
- Prepare the orchestrator script [**]
 - Launches a configurable number of parallel processes (and events)
 - Log Parser to extract results
- Prepare the Dockerfile and spec file
- Run the standalone **container builder**
 - The magic copy happens here
- Publish the standalone container in gitlab registry
 - Try it yourself
 - `docker run gitlab-registry.cern.ch/giordano/hep-workloads/lhcb-gen-sim:buildtest`

```
source /cvmfs/lhcb.cern.ch/lib/LbLogin.sh
■ lb-run --use-grid -c x86_64-slc6-gcc48-opt --use="AppConfig v3r335" \
  --use="DecFiles v30r11" --use="ProdConf" Gauss/v49r9 gaudirun.py -T \
  '$APPCONFIGOPTS/Gauss/Beam6500GeV-md100-2016-nu1.6.py' \
  '$APPCONFIGOPTS/Gauss/EnableSpillover-25ns.py' \
  '$APPCONFIGOPTS/Gauss/DataType-2016.py' \
  '$APPCONFIGOPTS/Gauss/RICHRandomHits.py' \
  '$DECFILESROOT/options/27163076.py' '$LBPYTHIA8ROOT/options/Pythia8.py' \
  '$APPCONFIGOPTS/Gauss/G4PL_FTFP_BERT_EmNoCuts.py' \
  '$APPCONFIGOPTS/Persistency/Compression-ZLIB-1.py' \
  'prodConf_Gauss_00071400_00000089_1.py'
```

```
lhcb-bmk.sh
parseResults.sh
prodConf_Gauss_00071400_00000089_1.py
```

```
lhcb-bmk
  Dockerfile
  Dockerfile.template
  lhcb-bmk.spec
```

```
{copies:4 , threads_x_copy:1 , events_x
_thread : 5 , throughput_score: {score:
18.1333, avg: 4.5333, median: 4.5647,
min: 4.4347, max: 4.5693} , app: LHCb f
ull sim chain}
```

```
./run_build.sh /root/hep-workloads/lhcb/gen-sim/lhcb-bmk.spec
```

giordano/hep-workloads/lhcb-gen-sim

Tag	Tag ID	Size	Created
buildtest	f9670e820	709.33 MiB	2 hours ago

Status of inclusion of HEP Workloads

	Stage	Release	Orchestr	Run w/ cvmfs	Standalone container	Validation standalone container
ALICE	Full sim	v5-09-XX-15 v5-09-09-01-1	✓	✓	✓ 4.37GB	✓ IMAGE=gilab-registry.cern.ch/giordano/hep-workloads/alice-gen-sim:latest
ATLAS	Gen	19.2.5.5	✓	⚠	✓ 1.32GB	✓ IMAGE=gilab-registry.cern.ch/giordano/hep-workloads/atlas-gen-bmk:latest
	Sim	21.0.15	✓	□		
	Digi-Reco	21.0.23	✓	□		
CMS	Gen-Sim	10_2_9	✓	⚠	✓ 2.82GB	✓ IMAGE=gilab-registry.cern.ch/giordano/hep-workloads/cms-gen-sim:latest
	Digi		□			
	Reco		□			
LHCb	Full sim chain		✓	✓	✓ 2.31GB	✓ IMAGE=gilab-registry.cern.ch/giordano/hep-workloads/lhcb-gen-sim:latest

⚠ □ some files (conditions/configuration) still from Frontier

To run a HEP standalone container: `docker run --rm --network=host -v /tmp/results/:/results $IMAGE`

To Do List: Infrastructure

#	Description	Status
1	Implement a fully automated procedure to build a standalone container image for each HEP reference workloads	✓
2	Create containers starting from Experiments' recipes (see previous slides)	✓ GEN-SIM □ DIGI-RECO
3	Implement Gitlab Continuous Integration approach for long term maintainability (see https://gitlab.cern.ch/giordano/hep-workloads/pipelines)	✓
4	Consolidate the CI approach	□
5	Integration in the benchmarking suite	□
6	Test migration to singularity containers	□

□ ↔ To be done. Looking for volunteers

To Do List: Workloads

– Here we expect the feedback of the Experiments

#	Description	Alice	Atlas	CMS	LHCb
1	Confirm that running workload is the desired/updated one for benchmarking	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	Can it run without any assumption, but /cvmfs and local input files? Can it run without internet WAN access?	<input type="checkbox"/>	<input type="checkbox"/> ⚠	<input type="checkbox"/> ⚠	<input type="checkbox"/>
3	Identification of the benchmarking metrics from log files	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Define running conditions (#threads, #events), “equalize” job duration, normalize scores	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	Study reproducibility of results	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	Can the container image size be reduced with smaller input data?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

↔ To be done. Looking for volunteers

⚠ some files (conditions/configuration) still from Frontier

Conclusions

- Very good progress
 - Gen-Sim HEP benchmark workloads are in containers
 - Accessible by everyone
 - Current limitation: need to use `--network=host` in order to access Frontier (Atlas and CMS)
 - Looking for volunteers to try and provide feedback

- Long To Do List
 - Integration area & Workloads (benchmarking) area
 - Looking for volunteers and experts from the experiments
 - Contact me directly or the WG mailing list hepixon-cpu-benchmark@hepixon.org



Image credits

- <https://pixabay.com/en/server-small-rack-computer-grey-297721/>
- <https://pixabay.com/en/stick-usb-usb-stick-flash-drive-157819/>
- http://www.publicdomainfiles.com/show_file.php?id=13939197614279
- <https://pixabay.com/en/cabinet-data-file-icon-information-1293245/>
- <https://pixabay.com/en/gears-machine-thinking-teamwork-2135326/>