ATHENA Job Submission

Summary:

- Main platforms: **htcondor** (OSG/BNL) and **slurm** (JLab/Compute Canada)
  - htcondor: 2 hour target job duration, automatic upload to S3 as part of job
  - slurm: 20 hour target job duration, requires post-job mirroring to S3 outside job
- Distribution and deployment: **sandbox singularity containers on cvmfs**, synced by OSG from docker every 6h and mirrored onto OSG site caches
- Continuous integration and testing: **automatic benchmarking** of all jobs to determine optimal running time, artifact is csv file used on submission

Deployment With Containers

Container versioning:

- nightly (ATHENA master branches)
- stable (released ATHENA versions)
- unstable (merge requests)

Container registries:

- eicweb (GitLab): all containers, private
  - actually mostly for singularity containers
- Docker Hub: nightly, stable, stable-$(date)

CVMFS singularity sandboxes:

- using OSG ~6 hour synchronizations, to `/cvmfs/singularity.opensciencegrid.org`
- distribution to clients on OSG, users at large lab facilities, end users with CVMFS

User access to containers:

- goals: quick, transparent to user
- `curl -L get.athena-eic.org | bash`
- `./eic-shell`

If CVMFS found:

- use auto-updating sandbox image

If CVMFS not found:

- singularity pull sandbox image
- `./eic-shell --upgrade`

Other features: use gpfs, automatic bindpath detection, use singularity from `/cvmfs/oasis.opensciencegrid.org`

Deployment With Containers

Synchronization through github.com/opensciencegrid/cvmfs-singularity-sync:

```
$ cat docker_images.txt
      eicweb/jug_xl:*-stable
      eicweb/jug_xl:*-beta
      eicweb/jug_xl:*-alpha
      eicweb/jug_xl:testing
      eicweb/jug_xl:nightly

$ ls -1d /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl*
      /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:3.0-stable
      /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:4.0-acadia-stable
      /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:4.0-canyonlands-stable
      /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:4.0-deathvalley-1.5T-stable
      /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:4.0-deathvalley-stable
      /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:nightly
      /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:testing
```

Deployment With Containers

Each container has multiple geometries:

```
$ ls -1 /opt/detector/
      athena-acadia
      athena-acadia-v2.1
      athena-canyonlands
      athena-canyonlands_old
      athena-canyonlands-v2.1
      athena-canyonlands-v2.2
      athena-deathvalley
      athena-deathvalley-1.5T
      athena-deathvalley-v1.0-1.5T
      athena-deathvalley-v1.1
      athena-nightly
      setup.sh
```

Continuous Integration and Testing

Input:

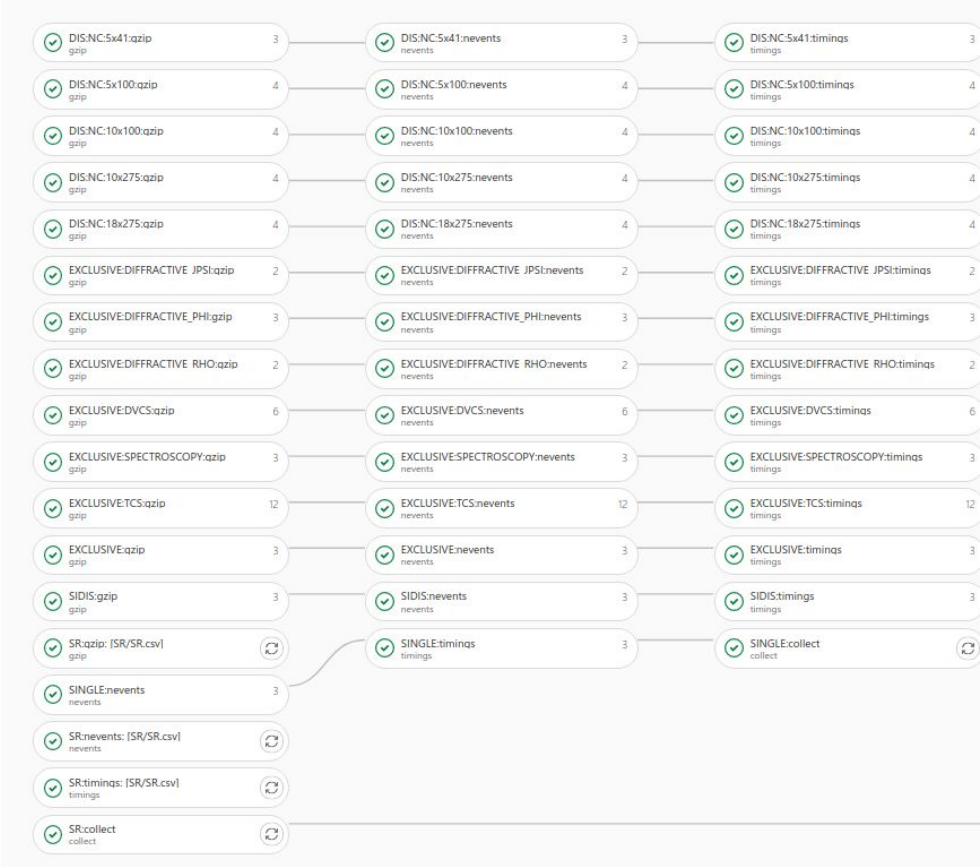
- S3: `mc cp` of HepMC v3 files (gzipped)
- condor: transfer DD4hep gun steer file

Benchmarking on eicweb as part of CI:

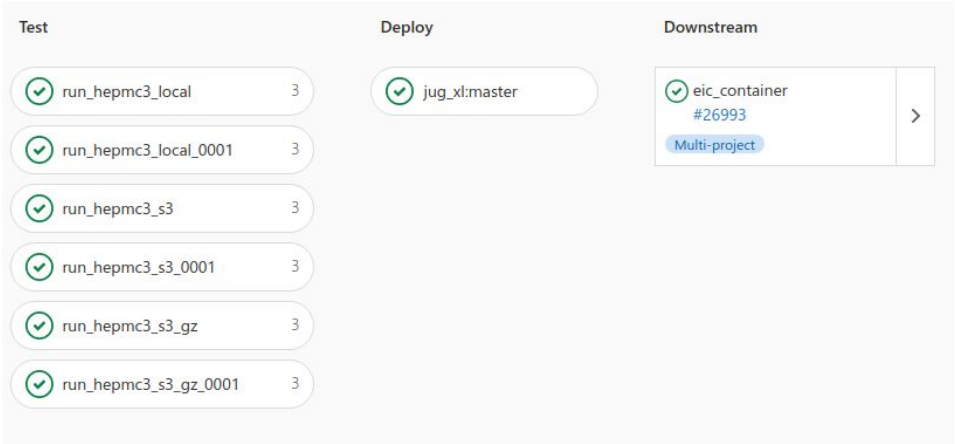
- running test, smoke tests, sanity checks, time-per-event determination into csv
- target time: slurm: ~20 hrs, condor: ~2 hrs

Job submission:

- identical syntax for slurm and condor
  - Automatic retrieval of csv artifacts, automatic job strategy determination
  - No user code is needed: all submission support is available on CVMFS
- memory request: 2 GB, typical use 1.5 GB



Continuous Integration and Testing



Structure on job node:

- S3 retrieval (inside the job, not using OSG transfer_input_file, no pre-signed urls)
- run simulation (artifacts downloaded as needed, or found in container cache)
- S3 upload of full simulation podio output
- run reconstruction (artifacts downloaded as needed, or found in container cache)
- S3 upload of reconstruction eicd output

Interactions with OSG on best practices:

- hold release based on log parsing
- secrets transmission to nodes (env.sh)
- delayed job instantiation (`max_idle`)
- `queue from csv`
- reporting misbehaving nodes

Htcondor Job Submission: Template for BNL & OSG

```
universe   = vanilla
executable = %EXECUTABLE%
arguments  = %ARGUMENTS%

error  = LOG/CONDOR/osg_$(Cluster)_$(Process).err
output = LOG/CONDOR/osg_$(Cluster)_$(Process).out
log    = LOG/CONDOR/osg_$(Cluster)_$(Process).log

transfer_input_files = %ENVIRONMENT%

on_exit_hold = (ExitBySignal == True) || (ExitCode != 0)

+ProjectName="EIC"
+SingularityImage="/cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:%JUGGLER_TAG%"

Requirements = HAS_SINGULARITY == TRUE && HAS_CVMFS_singularity_opensciencegrid_org == TRUE && OSG_HOST_KERNEL_VERSION >= 31000

request_cpus   = 1
request_memory = 2 GB
request_disk   = 2 GB

max_idle = 100

queue file,nevents,ichunk from %CSV_FILE%
```

S3 access secrets

Avoid RHEL6

Delayed job instantiation

Job segmentation

Htcondor Job Submission: Submitting a Campaign

Example:

```
cd /cvmfs/singularity.opensciencegrid.org/eicweb/jug_xl:4.0-deathvalley-1.5T-stable/opt/campaigns

scripts/submit_csv.sh osg_csv hepmc3 SIDIS_Lambda_hiDiv.csv
```

- If local csv file, use it. Otherwise retrieve from CI.
- Submit with target duration of 2 hours based on `osg_csv` template.

Htcondor Job Submission: Lessons Learned

- Some sites do not allow internet access from the node

```
GLIDEIN_ResourceName != "Purdue-Geddes" && GLIDEIN_ResourceName != "TCNJ-ELSA" && GLIDEIN_ResourceName != "UConn-OSG" &&
GLIDEIN_ResourceName != "NWICG_NDCMS" && GLIDEIN_ResourceName != "OSG_US_FSU_HNPGRID" && GLIDEIN_ResourceName !=
"ASU-DELL_M420" && GLIDEIN_ResourceName != "GPN-GP-ARGO-Backfill" && GLIDEIN_ResourceName != "AGLT2" && GLIDEIN_ResourceName
!= "TACC-Jetstream-Backfill" && GLIDEIN_ResourceName != "MWT2" && GLIDEIN_ResourceName != "GLOW" && GLIDEIN_ResourceName !=
"CHTC" && GLIDEIN_ResourceName != "NDSU-Lancium-Backfill"
```

  Working with OSG on better test for internet connectivity

- S3 integration with htcondor is targeted at ‘real’ S3 servers

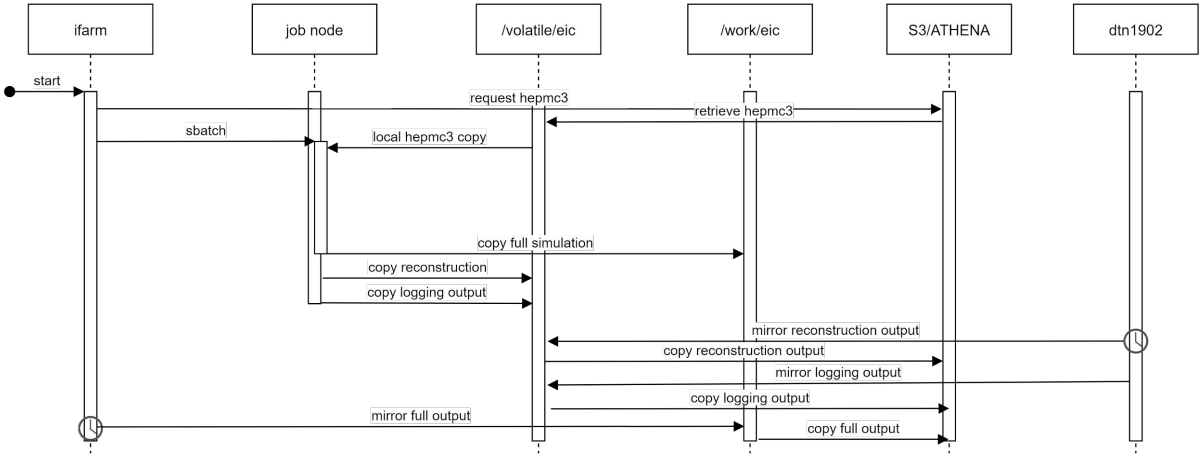
```
input_files = s3:// dtn01.sdcc.bnl.gov:9000/<path>
```

  Working with OSG on enabling job-based S3 access to minio appliances

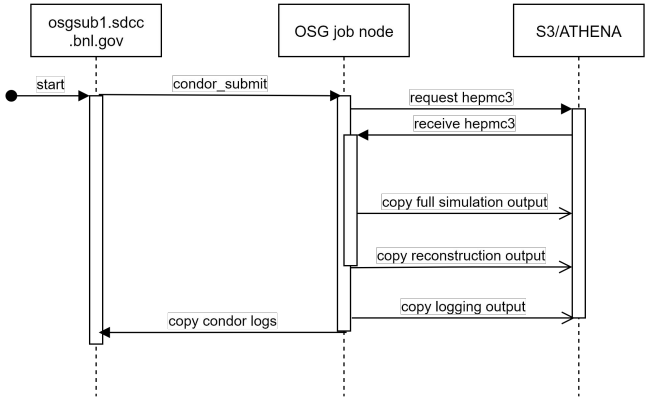
- Big benefit: hold on error, automatic resubmit until successful completion
  - Slurm: figure out which job numbers failed, dissect, resubmit

Benefits of Condor over Slurm: Internet Access

Running at JLab (capacity 25k job slots, 14% for EIC)



Running on OSG (capacity 50k)



Benefits of Slurm over Condor: Partial File Access

- Jobs may run anywhere: no affinity between jobs on the same input file
- Copying larger input file than needed does not result in efficiencies
  - Main reason for using `input_files = s3://<path>` since scheduler can optimize
- For condor on OSG: all jobs ran on an entire input file, defined by 2 hour job duration, typically 10k events
- Results in inefficient S3 storage usage
- Partial reads of event ranges not supported in ascii hepmc files

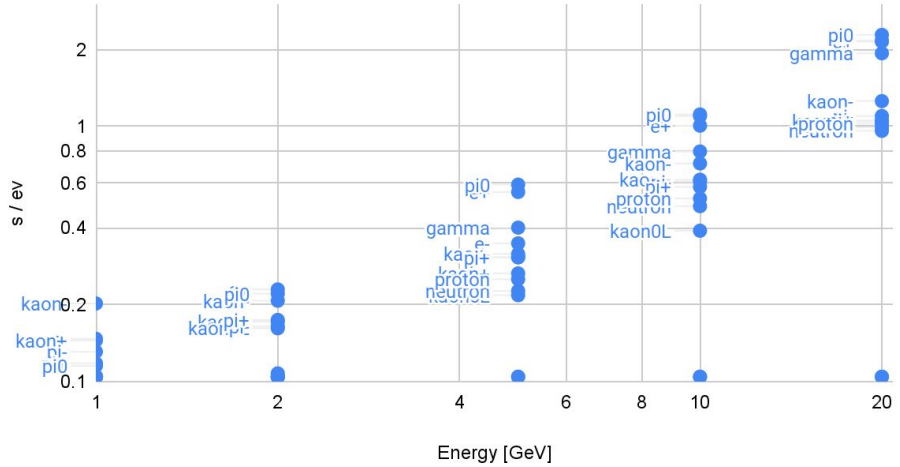
Typical Full Simulation & Reconstruction Timings

Typical memory used: ~1.5 GB/process

Typical fixed overhead: ~60 s/process

Single particle initial states:

Barrel



DIS NC (Q^2>1, Q^2>10, Q^2>100, Q^2>1000)

- 5x41: 1.3 s/ev, 1.8 s/ev, 2.5 s/ev
- 10x100: 2.3 s/ev, 2.9 s/ev, 3.9 s/ev, 5.2 s/ev
- 18x275: 5.6 s/ev, 6.1 s/ev, 7.6 s/ev, 10.2 s/ev

SIDIS

- Lambda 18x275: 7.4 s/ev
- eAu 18x110: 2.7 s/ev

Diffractive J/psi

- eAu: 3.8 s/ev

Dual EPYC 7H12 (256 threads, 512 GB, 2 GB/thread) HEP-SPEC06 ~20/thread (no actual benchmark available, based on NIKHEF Physics Data Processing Facility Dual EPYC 7H12 without hyperthreading)