Jefferson Lab

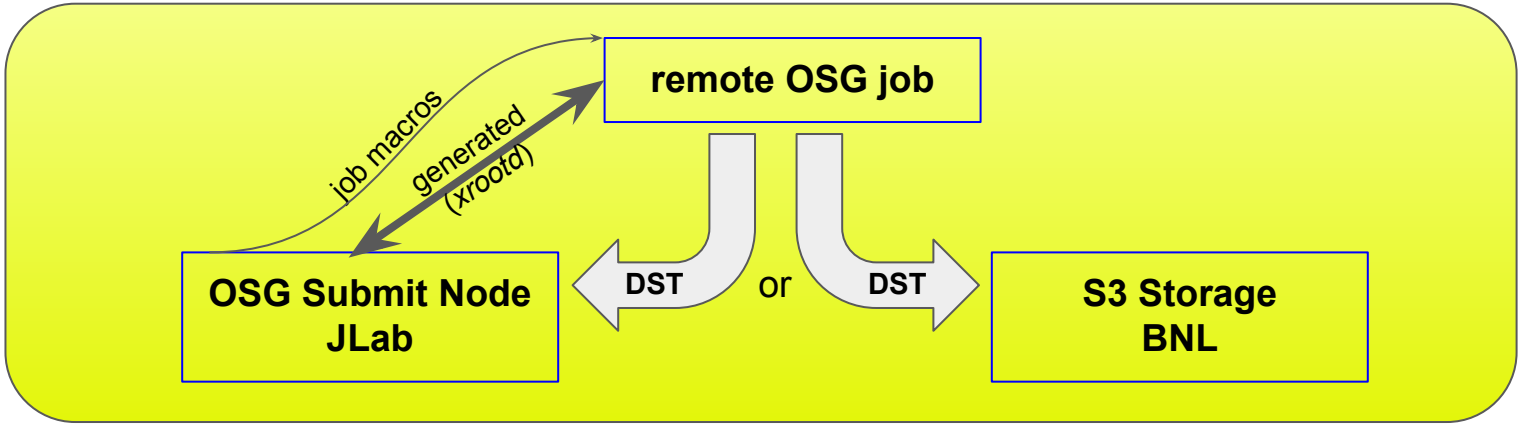
EPSCI

Computing Resources @ JLab and OSG

David Lawrence - Jefferson Lab

ECCE Simulation Workshop
July 8 2021

Multiple sources now configured and available for production



JLab
SLURM
/work/eic2 0.5PB
/work/eic3 0.5PB

BNL
condor
S3 (object store) 1PB

MIT/Bates
opportunistic

S3 Storage
BNL

EIC/ECCE Computing Resources

For more information:
J. Lauret Computing Report at EICUG, 5/20/2021

Storage

- Pledged resources

BNL	1 PB
JLab	1 PB

Availability:

BNL
- physical disk in place, S3 access

JLab
- physical disk in place, local access (+some xrootd)

Compute

- Pledged resources

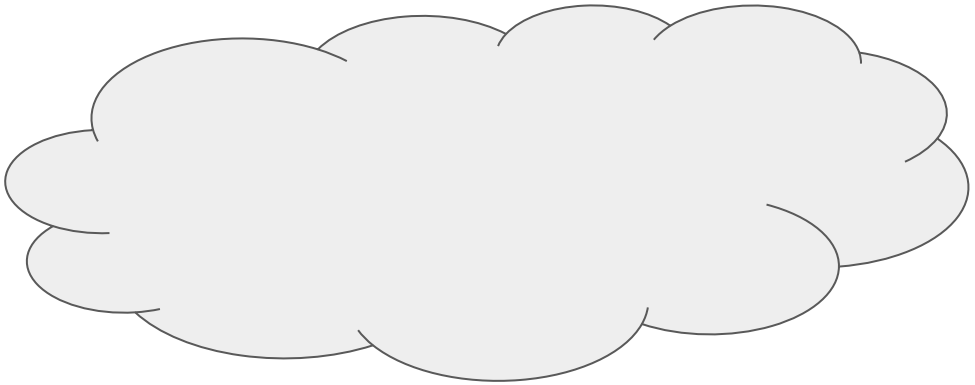
BNL	2k cores - 4 months (75% time)
JLab	2k cores - 4 months
OSG	4k cores

Availability:

now

- ***All pledged resources are for "EIC" and not for specific proto-collaborations***
  - Fairly allocate between ECCE, ATHENA, CORE, …
    - UG formation of Computing Coordination Group (CCG)
      - Committee forming but no word on allocation fractions (may not be needed)
    - Start with administrative controls for quota management

ECCE Compute Estimate



2 jobs run on OSG nodes

Default events from tutorial

https://www.phenix.bnl.gov/WWW/publish/phnxbld/sPHENIX/files/sPHENIX_G4Hits_sHijing_9-11fm_00000_00010.root

Total committed by [(BNL + JLab)*75% + OSG] / 2 = 4k cores x 4 months (x 75%) = **10 Mcore-hrs**

expect ~ ½ for ECCE

102 Events - avg. time : 2,644 sec (single thread)
2 Events - avg. time : 215 sec (single thread)

} time/event: 24.3 sec
overhead: 166 sec

assume 2k events/job

for **2.25B** events:
15.2 Mcore-hrs for event simulation
51.9 kcore-hrs for overhead *(program startup and shutdown)*

for **1.0B** events:
6.8 Mcore-hrs for event simulation
23.1 kcore-hrs for overhead *(program startup and shutdown)*

Original estimate for 2021 ECCE storage requirement

- sPHENIX mock data challenge: 100M pp events -> 130TB
- Estimate 1B events needed for proposal development -> 1.3PB
- Include contingency for larger eA event size, etc … -> 2PB

1M event test sets

	pythia6 - general	**pythia6 - SIDIS**	**pythia6 - HF & Jets**
DST	159GB	220GB	177GB
Evaluator	59GB	78GB	60GB
Total	**218GB**	**298GB**	**237GB**

4.3x smaller than sPHENIX MDC numbers

	factor	units	err on value	err on total
Total ep σ	45	μb	0	0
Integrated luminosity	5	fb-1	0	0
Fraction of total σ/physics chan.	1.00E-02		1.00E-02	2.25E+00
Frac. signal events needed for simulation	2.00E-02		2.00E-02	2.25E+00
Num. physics generators to simulate	25		5	0.45
Num. detector configurations to simulate	2		1	1.125
Total number of events to simulate	**2.25** *Bevents*	billion events		*contingency* **3.4** *Bevents*
Event size	300	kB/event	150	0.34
Total number of events	2.25	billion events	3.4	1.02
Total storage	**0.675** *PB*	PB		*contingency* **1.08** *PB*

Cameron's SIDIS 1M event test

VERY PRELIMINARY

Estimate + 1σ
=
1.7PB

How much do we really need?

Wikipage has table that is most accurate tally right now:

https://wiki.bnl.gov/eicug/index.php/ECCE_Simulations_Working_Group

(as of this morning, it adds up to ~250M events)

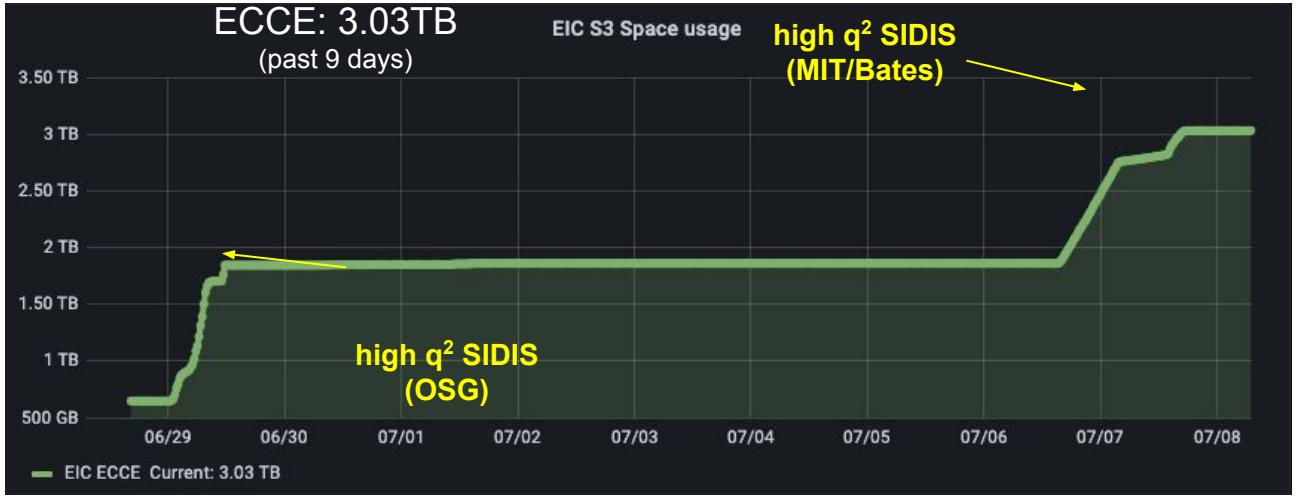
	Number of events	Storage needed (upper limit based on SIDIS)
original guesstimate	1B events	~0.3 PB
very rough estimate	2.25B events	~0.675 PB (+1PB)
wiki table	2x250M = 0.5B events	~0.15PB

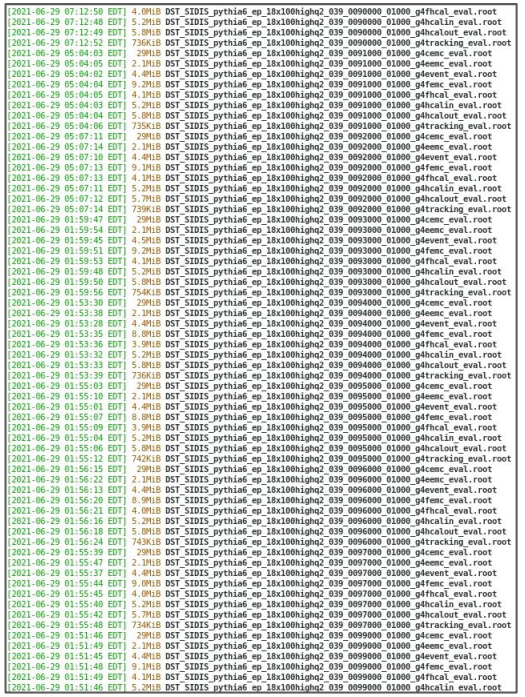
does not account for multiple evaluator passes

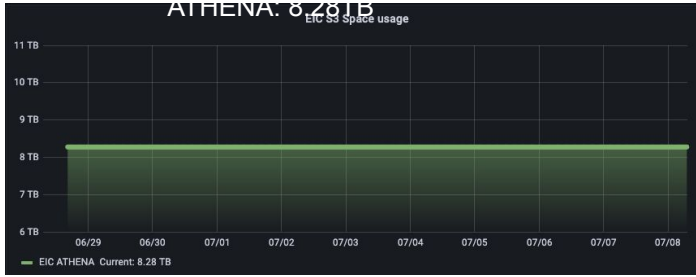
assume 2nd campaign and first are the same

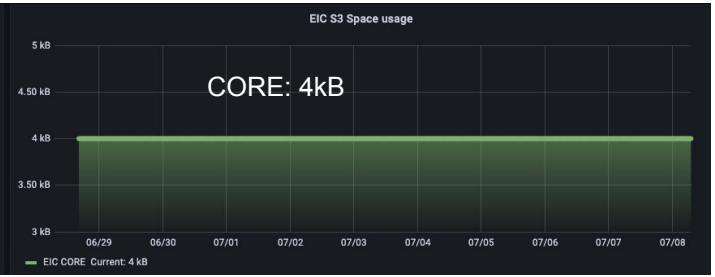
BNL S3 Storage

> alias mc=/cvmfs/eic.opensciencegrid.org/ecce/gcc-8.3/opt/fun4all/utils/bin/mcs3
> mc config host add eic https://dtn01.sdcc.bnl.gov:9000/ eicS3read eicS3read
> mc ls S3/eictest/ECCE/MC/ana.14/5f210c7/SIDIS/pythia6/ep_18x100highq2/eval_00000/









- NOT Amazon S3
- object store (must copy entire file out in order to read

JLab Storage

- 2 new work directories:
  - /work/eic2 *(0.5PB)*
  - /work/eic3 *(0.5PB)*
  - (don’t use /work/eic for large file sets -- limited space)

- Put your stuff in:
  - /work/eic2/ECCE/users

- Production simulations can be found in:
  - /work/eic2/ECCE/MC
  - /work/eic3/ECCE/MC
  - /work/osgpool/eic/ECCE/MC *(select evaluators only)*

xrootd: remote file access

- Install xrootd : https://xrootd.slac.stanford.edu/
  - e.g. yum install xrootd
- Set environment variable:
  - export LD_PRELOAD=/usr/lib64/libXrdPosixPreload.so
  - export LD_PRELOAD=$ROOTSYS/lib/libXrdPosixPreload.so
- Use normal tools
  - ls root://sci-xrootd.jlab.org//osgpool/eic
  - root root://sci-xrootd.jlab.org//osgpool/eic/ECCE/MC/ana.14/5f210c7/SIDIS/pythia6/ep_18x100highq2/eval_00000/DST_SIDIS_pythia6_ep_18x100highq2_027_0037000_01000_g4tracking_eval.root

n.b. double '//' is not a typo!

/work/osgpool = root://sci-xrootd.jlab.org//osgpool/eic

at JLab

everywhere else

Utilizing BNL and JLab storage for OSG jobs

- JLab
  - Use built-in GridFTP mechanism
    - files copied back to **`/work/eic2`** or **`/work/eic3`**
- BNL
  - Write to S3 storage from remote node
  - Minio client installed in `/cvmfs/eic.opensciencegrid.org/ecce/gcc-8.3/opt/fun4all/utils/bin/mcs3`
  - Authentication:
    - transfer script from secure area with job
    - run script using modified HOME to install secrets in cwd on remote node
    - delete *".mcs3"* (aka *".mc"*) directory when job finishes
- https://github.com/ECCE-EIC/productions/tree/master/OSG

ECCE OSG Production Scripts

These scripts are used to submit and run ecce simulation production jobs on the OSG. They were developed at JLab using the scosg16 submit node.

The simulation output can be directed to go either to JLab storage or BNL S3. This is specified by editing the makeOSGJobs.py script and modifying the line in class pars to be something like one of these:

```
simulationsTopDir = 'S3://eictest/ECCE/MC'
simulationsTopDir = '/work/eic2/ECCE/MC'
```

If set to the first value, the output files will be pushed to BNL S3 storage straight from the remote OSG node. The files be placed in a directory tree starting with ''eictest/ECCE/MC''. All subdirectories will be automatically created.

If the second form in the above examples is used then the files will be sent back to JLab where they will be stored in the specified directory.

S3 Write access

The BNL S3 storage can be accessed from anywhere, but requires a username/accesskey pair with write priviliges if one wishes to write to it. These are currently limited to only a few people. The way this works is the copy_to_S3.py file must be copied to the directory where the condor_submit command is being executed from and modified to include the secret information. For example

```
cp OSG/copy_to_S3.py .
chmod 700 copy_to_S3.py
<edit copy_to_S3.py to include write-authorized username/accesskey>
condor_submit path/to/file.job
```

Condor will take care of securely transferring the ''copy_to_S3.py'' file to the remote site. At the end of the job, the script is run where it will copy all of the files back. The ''copy_to_S3.py'' script will set the HOME directory to the local working directory so the minio client will only write the secret infomation locally. It will also remove the configuration just before the script exits to ensure the secret information is not left on the remote job node.

Where is Everything? wiki page

https://wiki.bnl.gov/eicug/index.php/ECCE:Where_is_everything%3F