CLAS12 Data Processing & Computing at JLab

N. Baltzell - CLAS Collaboration Meeting - November 10, 2020

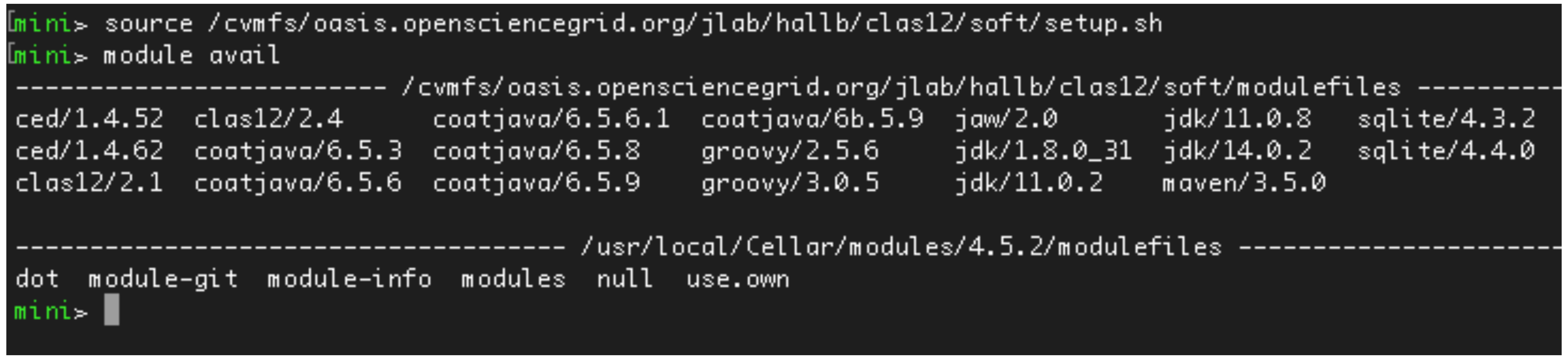
Miscellaneous Scicomp/IT News

- clasweb.jlab.org upgrade is in progress
  - Some short outages have been requested to facilitate the upgrade
    - Maybe 2 or 3 over the course of a month or two, ~30 minutes each, around 5 PM
    - Would send at least 1-day (?) advance notice to the collaboration
- Batch farm
  - The task that kills jobs for going over memory was switched to “cgroups”, which is stricter and faster to prevent single jobs from crashing others. This means memory requests may need to be adjusted again.
  - Some people like using SLURM to request a (large) number of cores for interactive use, and there’s 2 dedicated nodes for that, but remember to relinquish your allocation when done:
    - https://scicomp.jlab.org/docs/farm_slurm_batch_interactive_jobs
- Disk
  - Still expecting a significant increase in our /cache, e.g. pin quota up to 500 TB, similar to what Hall D currently has. Was delayed over the past year’s new Lustre upgrades.
  - A more performant /work filesystem is being pursued, longer term project
- Tape
  - Last week the system was unified to a single tape library (i.e. all drives/movers have access to all tapes), and additional drives added for ultimately ~50% increased throughput

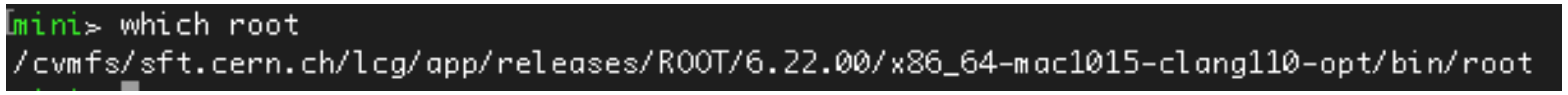
CVMFS & XRootD

- This summer JLab added support for **CVMFS**, which is good for smallish data that doesn't change frequently and is read in its entirety, e.g. software, databases, maps
- Software-wise, currently only our Java-based stuff is fully supported on CVMFS
  - currently we use it, plus sqlite database snapshots and magnetic fields, in jobs on OSG
  - the C(++) side of things is a work in progress, but we'll likely never be able to support many OS/compiler versions
    - need to take a survey on what everyone uses (ubuntu18, centos8, gcc8.9, etc.) or is running a container good enough?
    - and scicomp is working on a better package and build system "spack" to replace /site/12gev_phys, centrally managed for all the halls, so we'll want to leverage that before pursuing
- Meanwhile, you can already run CLAS12's OS-independent software from CVMFS on your personal computer:
  - See #5: https://clasweb.jlab.org/wiki/index.php/CLAS12_Software_Center#tab=HOWTOs

- JLab's scicomp now also supports **XRootD**, which is good for streaming larger data
  - Currently our only use is for background-merging files, e.g. on OSG
  - Accessing it at JLab is mentioned in the simulation chain HOWTO including background-merging:
    - https://clasweb.jlab.org/wiki/index.php/CLAS12_Software_Center#tab=HOWTOs



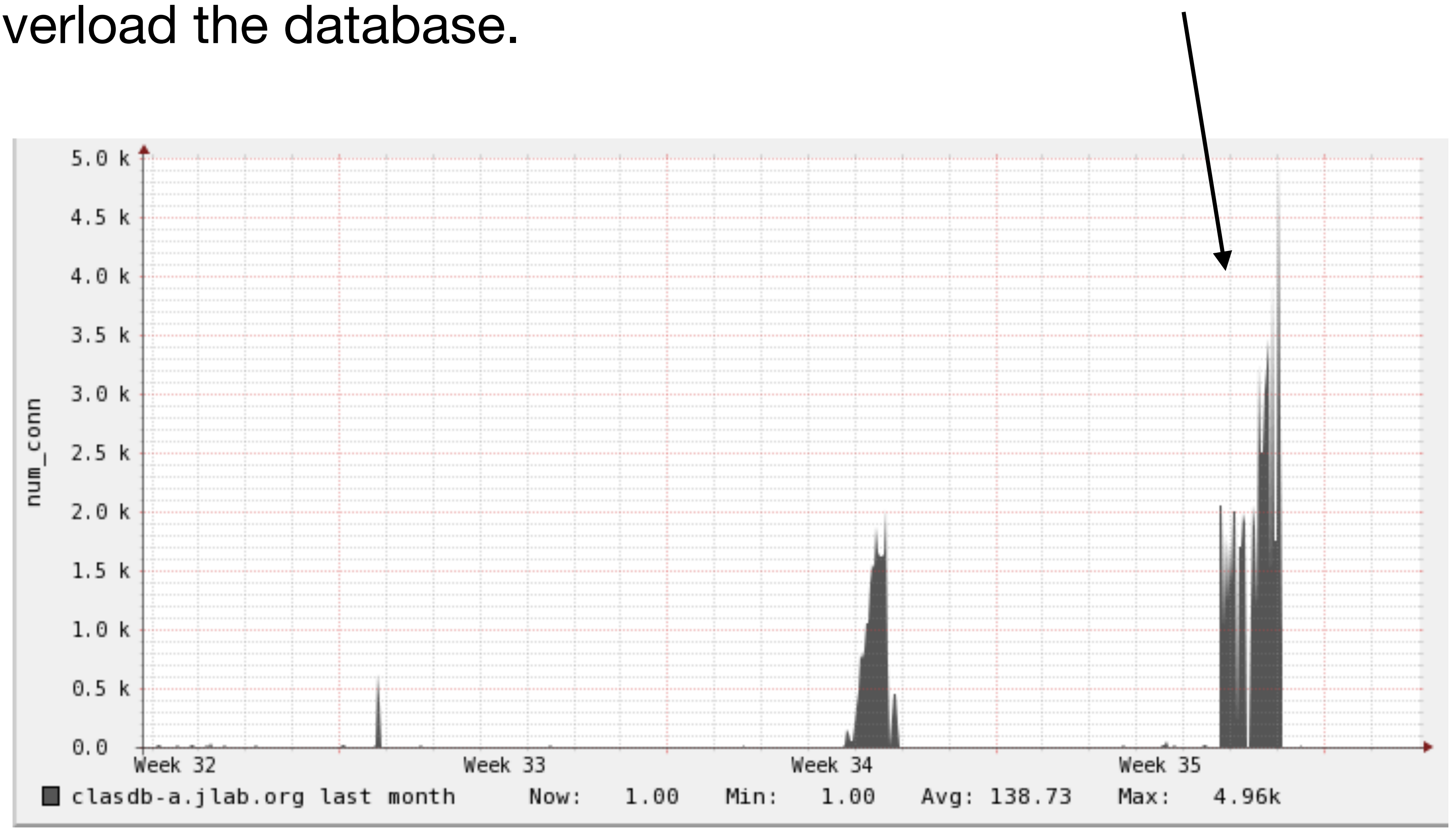
```
mini> source /cvmfs/oasis.opensciencegrid.org/jlab/hallb/clas12/soft/setup.sh
mini> module avail
------------------------- /cvmfs/oasis.opensciencegrid.org/jlab/hallb/clas12/soft/modulefiles ---------
ced/1.4.52  clas12/2.4      coatjava/6.5.6.1  coatjava/6b.5.9  jaw/2.0       jdk/11.0.8   sqlite/4.3.2
ced/1.4.62  coatjava/6.5.3  coatjava/6.5.8    groovy/2.5.6     jdk/1.8.0_31  jdk/14.0.2   sqlite/4.4.0
clas12/2.1  coatjava/6.5.6  coatjava/6.5.9    groovy/3.0.5     jdk/11.0.2    maven/3.5.0

----------------------------------- /usr/local/Cellar/modules/4.5.2/modulefiles ---------------------
dot  module-git  module-info  modules  null  use.own
mini> 
```



```
mini> which root
/cvmfs/sft.cern.ch/lcg/app/releases/ROOT/6.22.00/x86_64-mac1015-clang110-opt/bin/root
```

CCDB/RCDB SQLite snapshots

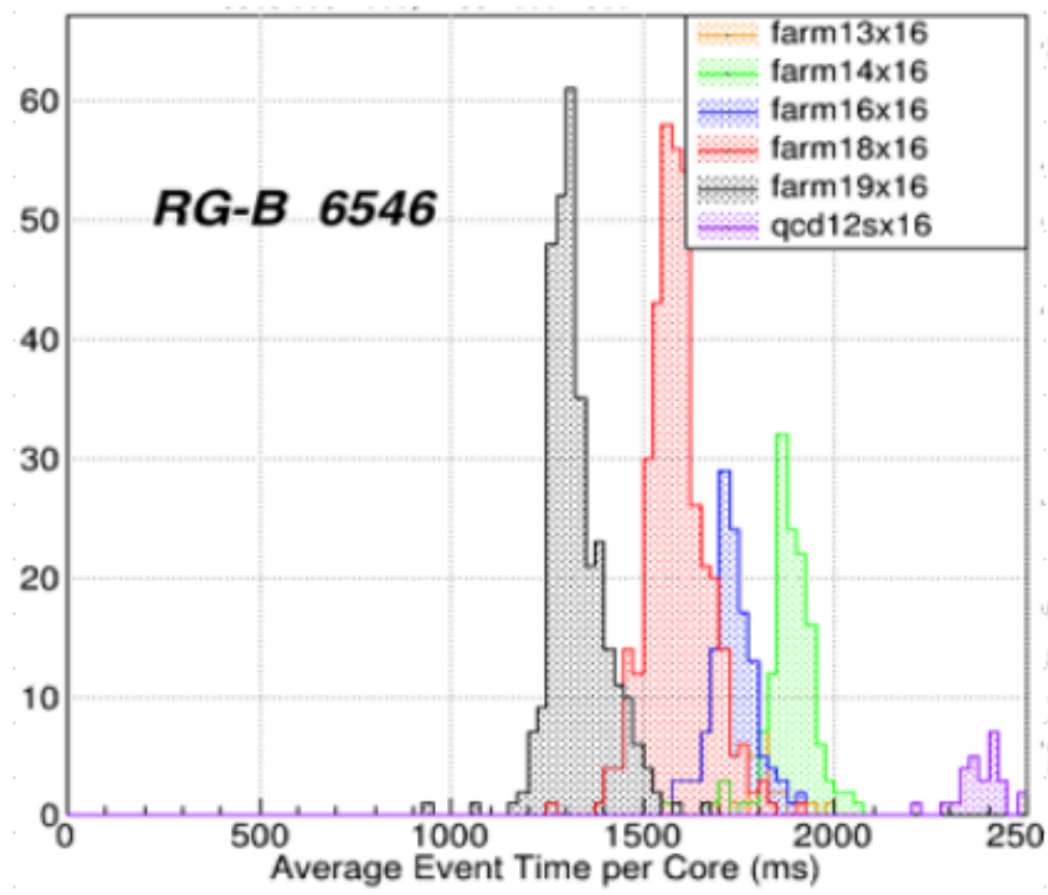
- All main CLAS12 software components query our CCDB database (and some RCDB) for various run-time parameters
  - gemc, evio2hipo, decoder, recon-util, clara
- Their access is reasonably well-optimized, e.g. no persistent nor idle connections, and multi-threaded stuff uses cache managers to be very low overhead, but we occasionally hit new issues to address
- We did have database server upgrades a while back, reported in previous collaboration meetings
- Nonetheless, single-threaded simulation jobs at JLab, when they get lucky and the farm is idle, can start many simultaneously and overload the database.

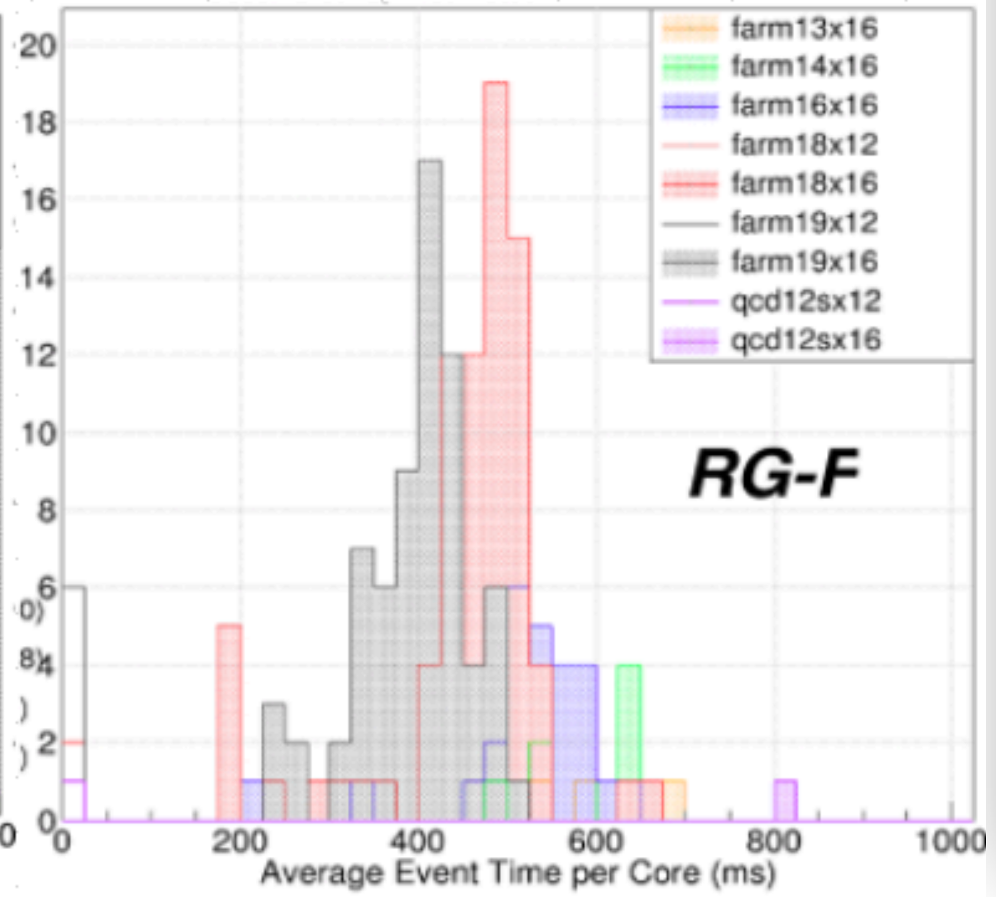


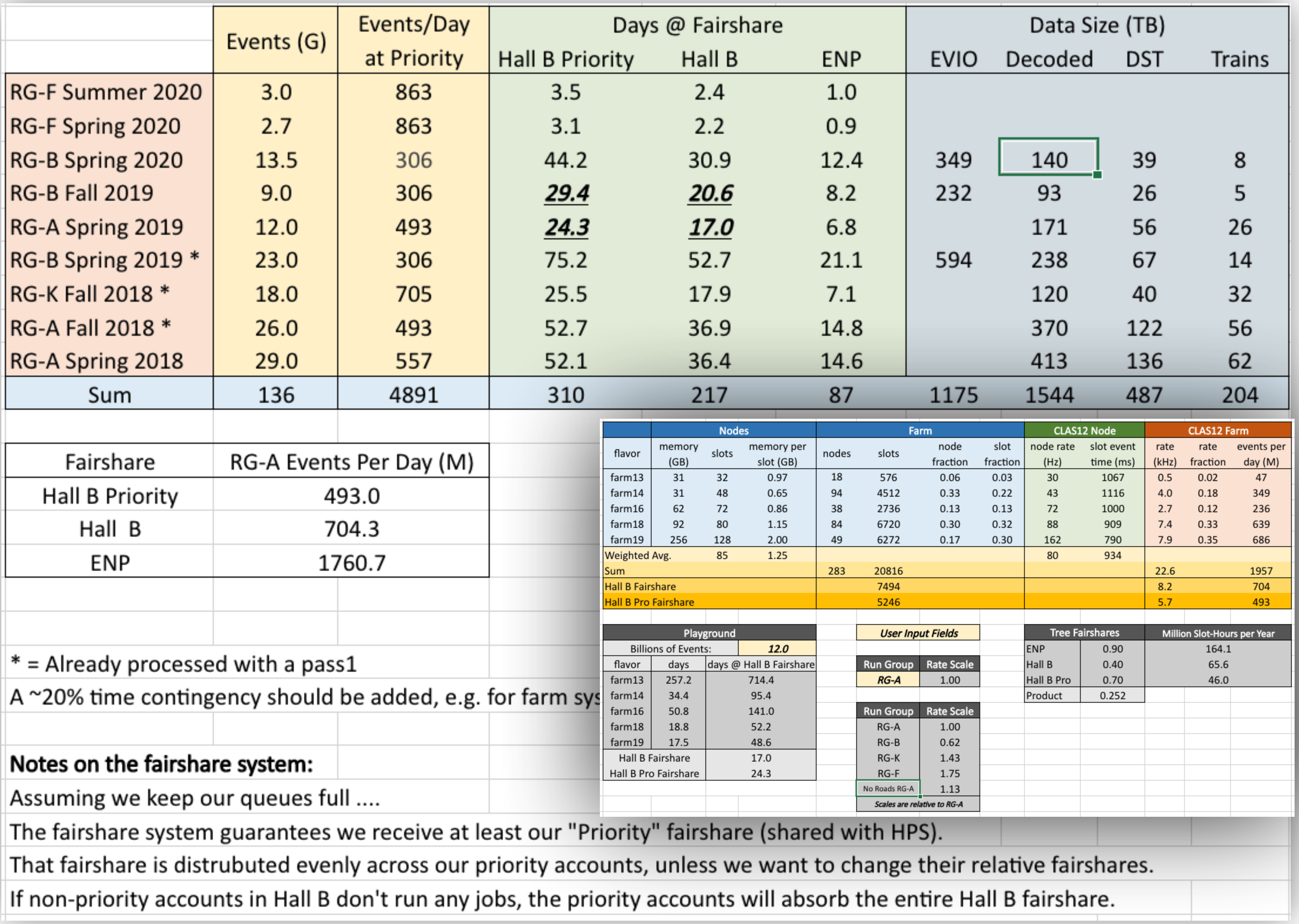
- But, production simulation jobs don't generally need access to the live database. Our OSG jobs already uses sqlite, otherwise we'd have had many more issue. Other large-scale offsite farms should too.
  - If you're running large simulations on the JLab farm, you can pickup the appropriate sqlite snapshots automatically via:
    - `module load sqlite/4.4.0` (the number corresponds to the gemc/clas12tags version, where 4.4.0 is the current production version and 4.3.2 was the previous one)
  - If you're offsite running large simulations, you can either:
    - use CVMFS to access them and set the appropriate environment variables automatically
    - or download them manually (see #6 at https://clasweb.jlab.org/wiki/index.php/CLAS12_Software_Center#tab=FAQ)
      - and set CCDB/RCDB_CONNECTION environment variables to point at your local copies

Run Group Data Processing

- Since the previous collaboration meeting, we processed ~67 billion events (pass1s only) on JLab's farm from 3 CLAS12 run groups (A/B/K)
    - with ultimately hands-free 100% success rate, automated workflows from decoding to trains, and duration close to projections based on benchmarks on the different node flavors and distribution
- Spreadsheets maintained with calculations to provide projections for decisions on future run group processing
    - https://jeffersonlab-my.sharepoint.com/:x:/g/personal/baltzell_jlab_org/EU096WRXcyBLI_ApLfSCuvoBiwsPFfBN_0enCzU3dFV6rw?e=ucRuQc







	Events (G)	Events/Day at Priority	Days @ Fairshare Hall B Priority	Hall B	ENP	Data Size (TB) EVIO	Decoded	DST	Trains
RG-F Summer 2020	3.0	863	3.5	2.4	1.0				
RG-F Spring 2020	2.7	863	3.1	2.2	0.9				
RG-B Spring 2020	13.5	306	44.2	30.9	12.4	349	140	39	8
RG-B Fall 2019	9.0	306	***29.4***	***20.6***	8.2	232	93	26	5
RG-A Spring 2019	12.0	493	***24.3***	***17.0***	6.8		171	56	26
RG-B Spring 2019 *	23.0	306	75.2	52.7	21.1	594	238	67	14
RG-K Fall 2018 *	18.0	705	25.5	17.9	7.1		120	40	32
RG-A Fall 2018 *	26.0	493	52.7	36.9	14.8		370	122	56
RG-A Spring 2018	29.0	557	52.1	36.4	14.6		413	136	62
Sum	136	4891	310	217	87	1175	1544	487	204

Fairshare	RG-A Events Per Day (M)
Hall B Priority	493.0
Hall B	704.3
ENP	1760.7

* = Already processed with a pass1

A ~20% time contingency should be added, e.g. for farm sys

Notes on the fairshare system:

Assuming we keep our queues full ....

The fairshare system guarantees we receive at least our "Priority" fairshare (shared with HPS).

That fairshare is distrubuted evenly across our priority accounts, unless we want to change their relative fairshares.

If non-priority accounts in Hall B don't run any jobs, the priority accounts will absorb the entire Hall B fairshare.

	Nodes			Farm				CLAS12 Node		CLAS12 Farm		
flavor	memory (GB)	slots	memory per slot (GB)	nodes	slots	node fraction	slot fraction	node rate (Hz)	slot event time (ms)	rate (kHz)	rate fraction	events per day (M)
farm13	31	32	0.97	18	576	0.06	0.03	30	1067	0.5	0.02	47
farm14	31	48	0.65	94	4512	0.33	0.22	43	1116	4.0	0.18	349
farm16	62	72	0.86	38	2736	0.13	0.13	72	1000	2.7	0.12	236
farm18	92	80	1.15	84	6720	0.30	0.32	88	909	7.4	0.33	639
farm19	256	128	2.00	49	6272	0.17	0.30	162	790	7.9	0.35	686
Weighted Avg.		85	1.25					80	934			
Sum				283	20816					22.6		1957
Hall B Fairshare					7494					8.2		704
Hall B Pro Fairshare					5246					5.7		493

Playground		
Billions of Events:		*12.0*
flavor	days	days @ Hall B Fairshare
farm13	257.2	714.4
farm14	34.4	95.4
farm16	50.8	141.0
farm18	18.8	52.2
farm19	17.5	48.6
Hall B Fairshare		17.0
Hall B Pro Fairshare		24.3

User Input Fields

Run Group	Rate Scale
RG-A	1.00

Run Group	Rate Scale
RG-A	1.00
RG-B	0.62
RG-K	1.43
RG-F	1.75
No Roads RG-A	1.13
Scales are relative to RG-A	

Tree Fairshares		Million Slot-Hours per Year
ENP	0.90	164.1
Hall B	0.40	65.6
Hall B Pro	0.70	46.0
Product	0.252	

Done