Hall A & C Computing Tips and Tricks

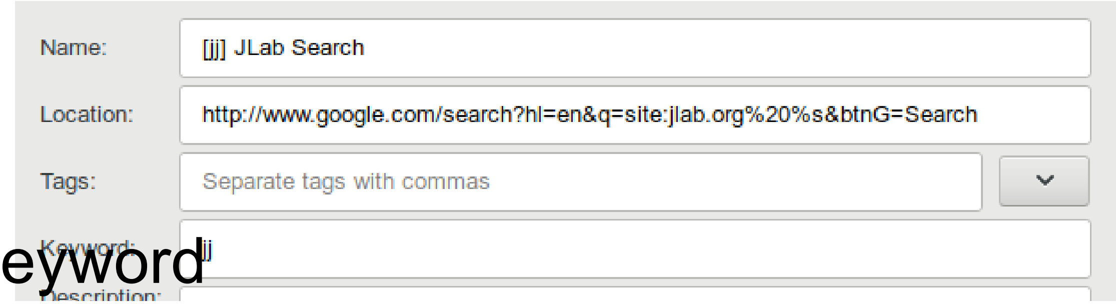
Brad Sawatzky

First up:
A Couple Quick Tricks to make your Computing Work Suck Less

How to find information

- JLab's web search is ... weak ...
  →So, do this in Firefox:
  - » Go to www.google.com and search for 'site:jlab.org foo'
  - » Right click on the bookmark and choose 'Properties'
    - – Give it a good name
    - – Give it a short 'keyword' like 'jj'
    - – Clean up the Location as shown, replace 'foo' with %s

  → Now type 'jj jget' in URL bar
  - » %s in 'Location' string is replaced with text following Keyword



Name: [jj] JLab Search

Location: http://www.google.com/search?hl=en&q=site:jlab.org%20%s&btnG=Search

Tags: Separate tags with commas

Keyword: jj

Description:

  - » 'site:jlab.org' is google-fu to restrict search to jlab.org domain

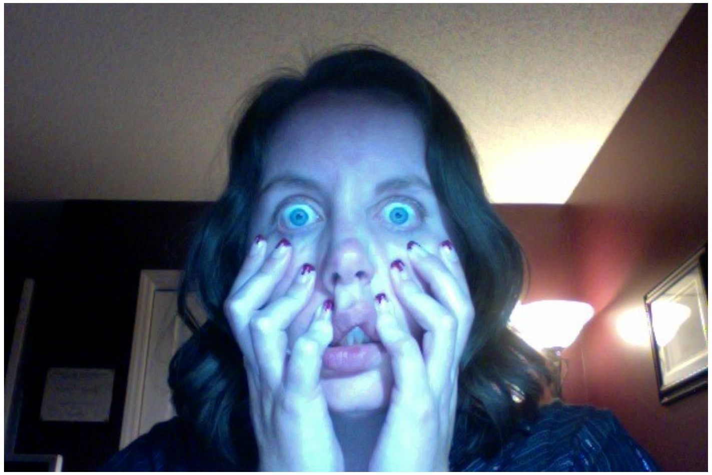
How to find information

- Trick works great for many things
  - →JLab staff page (https://misportal.jlab.org/mis/staff/staff.cfm)
    - » Keyword: 'page'
    - » Location (can extract from search on 'smith' above):
    - » `https://misportal.jlab.org/mis/staff/staff.cfm?field=all&name=%s&Search.x=36&Search.y=11&Search=Search&field=all`
  - →ROOT / G4
    - » Keyword: 'gr'
    - » Location:

      `https://www.google.com/search?hl=en&btnG=Search&q=site:cern.ch%20%s`
  - →Stackoverflow.com
  - →JLab Logbook (a little trickier, but you can work it out)
  - →...

How to work from Offsite

- How to work from offsite without tearing your eyes out because, holy hell, the graphics and menus are just so slow...



- VNC + ssh tunnel to the rescue
  - →VNC: Virtual Network Computing
  - →ssh used to securely move VNC traffic through jlab firewall
- Computer Center How-to
  - → https://cc.jlab.org/accessingvnc
- Old 'howto' I wrote for my collaboration
  - →adapt to machine you use
  - →Search: 'jj vnc session'
  - → https://hallaweb.jlab.org/wiki/index.php/How_to_connect_to_a_d2n_VNC_Session

Offline Analysis
Farm Usage / General JLab Computing

Nuts to the Farm, I analyze on my Desktop

- Simple tasks, some analysis OK on the desktop, BUT!!
  - → Thou shalt backup your code!
  - → Thou shalt backup your results!
  - → Who among us has done

    % rm -rf stuff/

    - » Followed by !@#$?
- Don't keep only copies on your laptop
- Don't keep only copies on your desktop's hard drive
- Do use git for all code and scripts!
  - → Commit early, commit often
  - → 'git push' often too!
    - » It's a backup!

- Hard drives die and the data are gone.
  - → Drives are large and cheap
  - → But reliability on consumer drives is worse than it used to be!
  - → SSDs are (weirdly) no better!

- IF your hard drive died today, how long would it take to recover?
  - » a day, a week,
  - » a month???

JLab Systems can help!

- /home, /group are automatically backed up
  - →They are snapshotted hourly!

    ```
    % cd .snapshot/
    % ls -lrt
    ```

  - →Longer term backups are on tape

- /work, /volatile are on heavily redundant filesystems
  - →NOT backed up
    - » Use tape
  - →More on this later...

- NOTE: Your JLab RHEL system on site can mount these directories if needed
  - →Talk to me if this would help

The JLab Farm • Power at your Fingertips

- Farm has many pieces
  - →~14500 compute nodes
  - →~3 PB of online storage
  - →~20+ PB of Tape

- Growing fairly quickly
  - →More disk before Fall
    - » ~500 TB
    - » Focus on Lustre stability
  - →More nodes before Fall
    - » ~7000 more nodes

(*) If I'm reading these charts correctly...

The JLab Farm • Batch Computing

- The Farm: Batch Computing
  - →No direct access to actual Farm nodes
    - » Use “Interactive” farm nodes for testing
      - – ie. ifarm1402
  - →DB and other network access (git, http, etc) generally constrained
  - →Jobs controlled by automated system called ~~“PBS”~~ “Slurm/Auger”
  - →You submit a job to Slurm, and Slurm schedules it to run
- All about trade offs:
  - →“Latency” can be high (hours+ from submission to job execution)
    - » BUT!
  - →Throughput is enormous
    - » 100s (1000s) of *your* jobs can run at once
    - » High bandwidth access to fast storage
  - →A full replay (100s of runs) can be completed in the time it would take 2–3 runs to complete in series on your desktop.

PBS → Slurm transition

- PBS (with Auger) managed/scheduled jobs on the farm
  - →Decides when/where jobs can run
  - →Manages staging file from tape, etc
  - →Getting pretty old...
- Replaced by Slurm
  - →as of June 4, 2019

- Only a few user-visible changes:
  - →Old swif/jsub/jkill tools still work
  - →stderr/stdout goes to /farm_out/<user>/
  - →Slurm uses real / used memory to schedule now!
    - » You may be able to reduce your memory request!
  - → https://scicomp.jlab.org/docs/auger_slurm

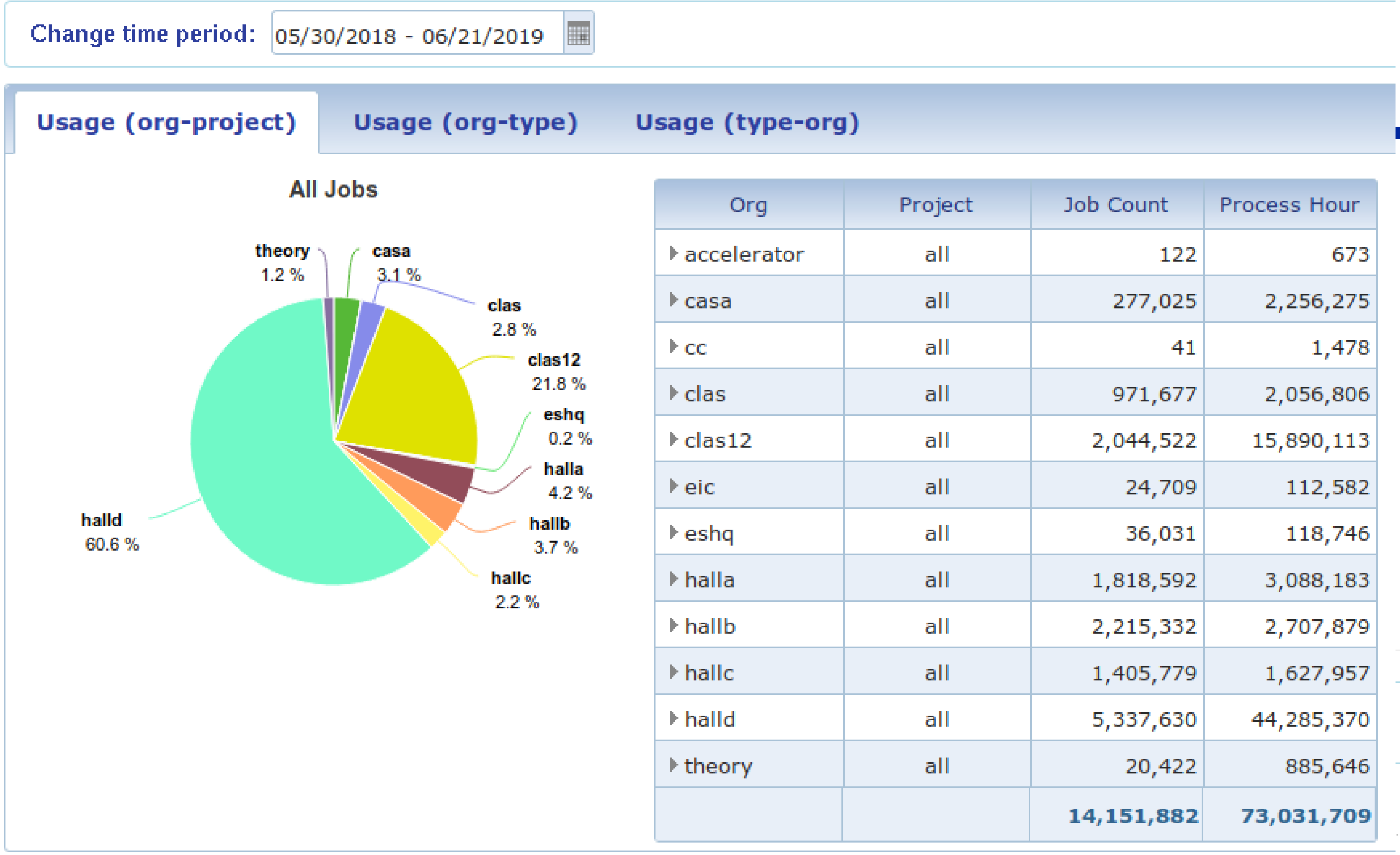
The JLab Farm • Scheduling

- The Farm is a Lab-wide shared resource
  - →Each Hall's budget includes $$$ to support their usage
  - →*Rough* allocation:
    - » A: 10%, C: 10%
    - » B: 35%, D: 35%
    - » Misc: 10%

- Ruled by “Maui”/Fair Share
  - →Allocations *not* written in stone and are adjusted based on needs

- The balance is trickier to manage than you may think...
  - →Jobs take time to run (system doesn't know how long beforehand)
  - →Upcoming job load is hard to predict
  - →System balances allocations over a few days, not hours

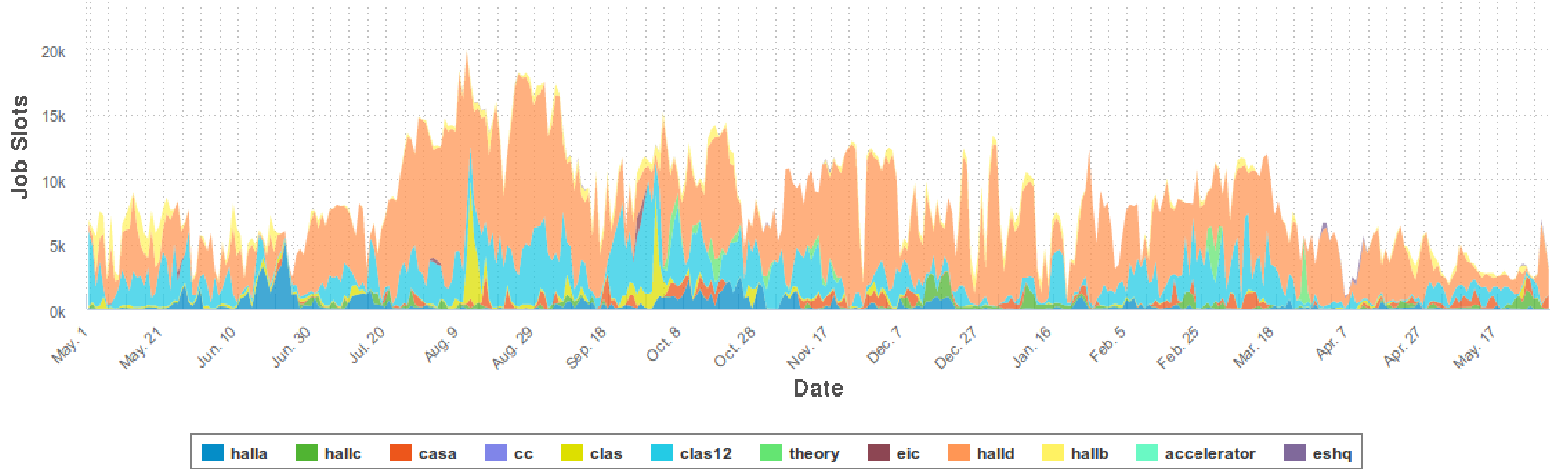
- More documentation here:
  - →https://scicomp.jlab.org/
  - →https://data.jlab.org/



Change time period: 05/30/2018 - 06/21/2019

Org	Project	Job Count	Process Hour
▸ accelerator	all	122	673
▸ casa	all	277,025	2,256,275
▸ cc	all	41	1,478
▸ clas	all	971,677	2,056,806
▸ clas12	all	2,044,522	15,890,113
▸ eic	all	24,709	112,582
▸ eshq	all	36,031	118,746
▸ halla	all	1,818,592	3,088,183
▸ hallb	all	2,215,332	2,707,879
▸ hallc	all	1,405,779	1,627,957
▸ halld	all	5,337,630	44,285,370
▸ theory	all	20,422	885,646
		14,151,882	**73,031,709**

05/01/20



Do use the Farm!

- The Farm is not your desktop
  - →Need to plan a little and fire off groups of jobs
- Test your job first!
  - →Can it run reliably?
    - » If it doesn't run on ifarm140x, it won't run on the farm!
  - →Is the output what you want?
    - » Check before firing off 100 jobs

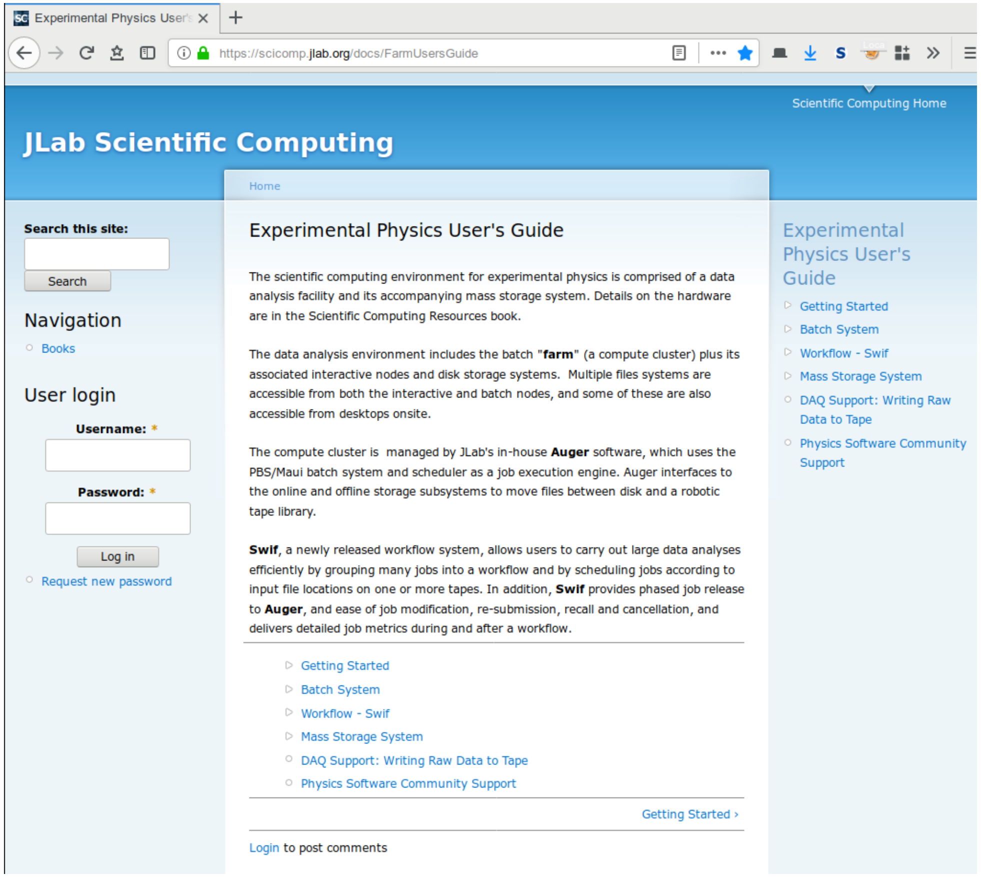
- Simple tasks, some types of analysis can be done on small systems, BUT!!
  - →Thou shalt back up your code!
  - →Thou shalt back up your results!
  - →IF your hard drive died today, how long would it take to recover?
- Don't keep only copies on your laptop

  Don't keep only copies on your desktop's hard drive

What's a “Job”?

- A 'Job' often maps to a shell script that runs your code
    - →It can do multiple things, but usually it executes a single instance of your software
        - » Analyze one run, or
        - » Simulate “1M” events,
        - » *etc...*
- *NOTE: Output that would normally go to a terminal goes to*
    - */farm_out/<user>/*
        - » *NEW (Slurm change)*

https://scicomp.jlab.org/docs/FarmUsersGuide


Experimental Physics User's
https://scicomp.jlab.org/docs/FarmUsersGuide
Scientific Computing Home
JLab Scientific Computing
Home
Search this site:
Search
Navigation
Books
User login
Username: *
Password: *
Log in
Request new password

Experimental Physics User's Guide

The scientific computing environment for experimental physics is comprised of a data analysis facility and its accompanying mass storage system. Details on the hardware are in the Scientific Computing Resources book.

The data analysis environment includes the batch "**farm**" (a compute cluster) plus its associated interactive nodes and disk storage systems. Multiple files systems are accessible from both the interactive and batch nodes, and some of these are also accessible from desktops onsite.

The compute cluster is managed by JLab's in-house **Auger** software, which uses the PBS/Maui batch system and scheduler as a job execution engine. Auger interfaces to the online and offline storage subsystems to move files between disk and a robotic tape library.

Swif, a newly released workflow system, allows users to carry out large data analyses efficiently by grouping many jobs into a workflow and by scheduling jobs according to input file locations on one or more tapes. In addition, **Swif** provides phased job release to **Auger**, and ease of job modification, re-submission, recall and cancellation, and delivers detailed job metrics during and after a workflow.

- Getting Started
- Batch System
- Workflow - Swif
- Mass Storage System
- DAQ Support: Writing Raw Data to Tape
- Physics Software Community Support

Getting Started ›

Login to post comments

Experimental Physics User's Guide

- Getting Started
- Batch System
- Workflow - Swif
- Mass Storage System
- DAQ Support: Writing Raw Data to Tape
- Physics Software Community Support

Debugging a job

- Generally want a single script that does everything!
  - → Set up full environment
  - → Use full paths
    - » `/group/myExp/myscript.sh`
    - » ~~`./myscript.sh`~~
- Testing your script:
  - → 1st: Run on ifarm140x
  - → 2nd: Submit job to Farm
- Test with the 'debug' Farm track
  - → Max priority, fast sched.
  - → Limited 4 hour runtime
  - → Limited jobs/user

- Test on ifarm140x

```
% ssh you@ifarm1402
% /group/myExp/myscript.sh
```

  - → Make sure it worked!
    - » check histos, report files
- Quick Test on Farm

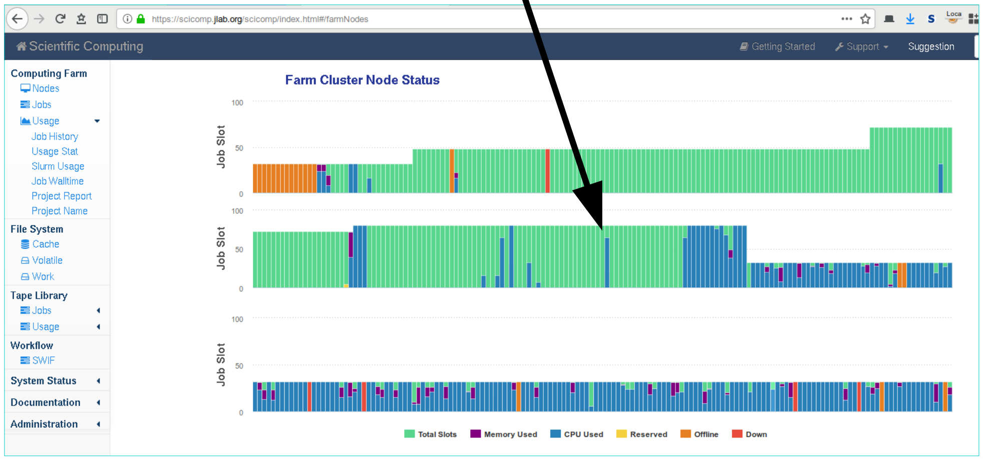
```
% swif add-job -create \
  -track 'debug' \
  <other options> ... \
  /group/myExp/myscript.sh
```

  - → Make sure it worked!
    - » check histos, files
    - » check `/farm_out/<you>/`
- Then submit full set!
  - → SWIF!

Make your jobs run faster/better!

- Scheduling jobs takes many things into account
  - →File availability from tape
  - →Memory request
  - →CPU/core request
    - » >1 is useless for podd/hcana
  - →'Fairshare' metric
    - » Average Hall utilization
    - » 'Project' usage
      - – ie. experiment
- Details
  - → [Fairshare Web Page]

- If a Hall / Project is not using 'their' fraction, then those Farm resources are available to anyone on a first-come, first-serve, basis!
  - →If the Farm is idle, you can take advantage!
    - » Like now!



Make your jobs run faster/better!

- Common Bottlenecks/ Mistakes
  - →CPU count
    - » use 1 core only (for now)
  - →Memory allocation
    - » < 2GB is best!
    - » Smaller → Faster scheduling!
  - →Insufficient debugging/cross checks
    - » Fire off 100s of jobs with bad config, buggy code

Self-Inflicted Wound 1

Sorcery

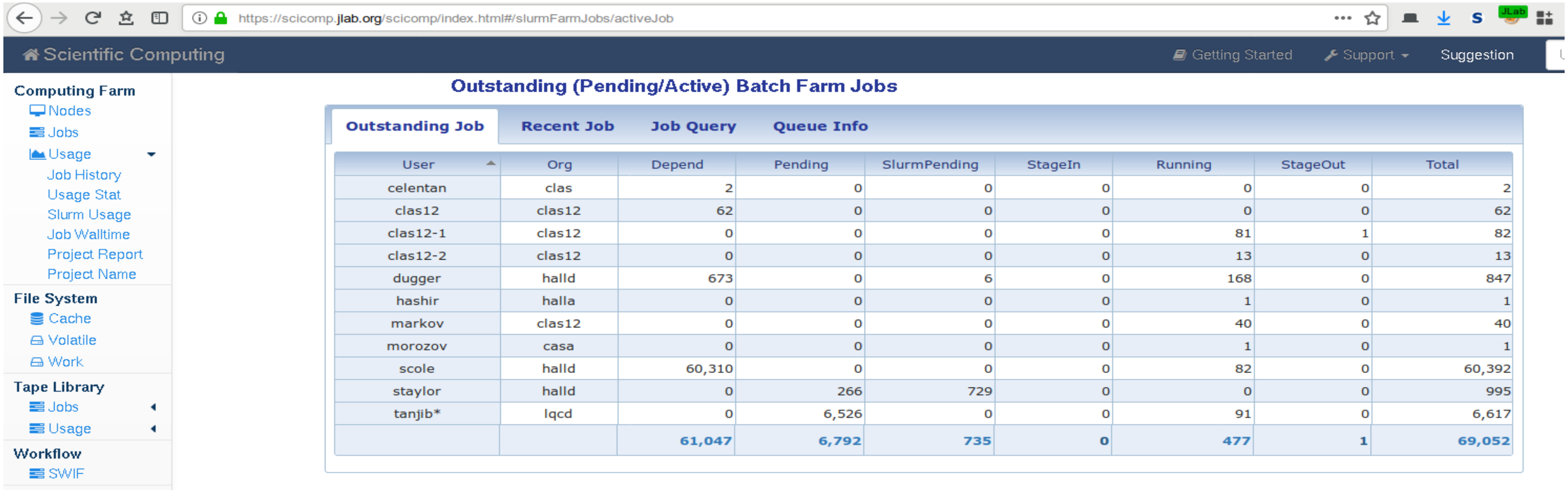
Target opponent sacrifices a green or white creature. If that player does, he or she loses 2 life.

"Worse than watching the cruelest deed is watching it done by your own hand."
—Baihir, Dromoka mage

117/264 U
DTK • EN Mathias Kollros

™ & © 2015 Wizards of the Coast

Check Job Status



Outstanding (Pending/Active) Batch Farm Jobs

User	Org	Depend	Pending	SlurmPending	StageIn	Running	StageOut	Total
celentan	clas	2	0	0	0	0	0	2
clas12	clas12	62	0	0	0	0	0	62
clas12-1	clas12	0	0	0	0	81	1	82
clas12-2	clas12	0	0	0	0	13	0	13
dugger	halld	673	0	6	0	168	0	847
hashir	halla	0	0	0	0	1	0	1
markov	clas12	0	0	0	0	40	0	40
morozov	casa	0	0	0	0	1	0	1
scole	halld	60,310	0	0	0	82	0	60,392
staylor	halld	0	266	729	0	0	0	995
tanjib*	lqcd	0	6,526	0	0	91	0	6,617
		61,047	6,792	735	0	477	1	69,052

- https://scicomp.jlab.org/scicomp/index.html#/slurmFarmJobs/activeJob
- Job Priority tab disappeared with Slurm...
- Recent Job tab can help you find information how jobs ran (or didn't run...)
  - → ie. Memory usage!
  - → See also: `/farm_out/<user>/...`

Small I/O Problems

- Small read/write operations are *very* inefficient
  - → Old/legacy code defaults can be very small (~4kB)
  - → Should be closer to 4MB chunks for decent performance
  - → Buffered IO can bridge the gap if needed
    - » Common errors:
      - – 'Debugging' output
        - » stderr << “got here” << endl;
        - » fprintf(stderr, “event %d\n”, eventNum);
      - – Opening/closing files very frequently
      - – Frequent random I/O
        - » ie. searching through a file for a parameter every event
- Workflows / procedures that may work on desktops or older systems don't scale well on modern systems (100s or 1000s of simultaneous jobs)
  - → Can take down / degrade system-wide filesystems
  - → “Lustre” gets a bad rap for this at JLab, but all filesystems would have big problems under many of the observed loads!
    - » (IT/CNI are still “on the hook” to improve reliability though!)

File Systems: Where do I put my stuff?

- CNI/IT provides
  - →/group - a space for groups to put software and some files, backup up by CNI
  - →/home - your home directory, backed up by CNI
  - →Cache - write through cache
  - →Volatile - acts as a scratch space
  - →Work – unmanaged outside of quotas / reservations

File Systems: Where do I put my stuff?

- CNI/IT provides
  - →/group - a space for groups to put software and some files, backup up by CNI
  - →/home - your home directory, backed up by CNI
  - →Cache - write through cache
  - →Volatile - acts as a scratch space
  - →Work – unmanaged outside of quotas / reservations

- No really, that doesn't help. Where do I put my stuff so Brad won't hassle me and I can get my work done in peace!

Where do I put my JLab stuff?

- /home/<user>/
  - →hourly snapshots
    - » `cd .snapshot/`
  - →personal, non-analysis files
    - » papers, notes, thesis, etc...
  - →analysis scripts: ~OK
    - » use git!
  - →source code: ~OK
    - » /work better
  - →NEVER store ROOT files or CODA files in /home

- Your laptop / desktop
  - →Should really be just a front-end for working on JLab systems
  - →Everybody wants to do backups, but almost no one actually does backups until after they've lost data...

This wasn't supposed to happen to me!!!!!!!!! I really was going to start backing up tomorrow.
IT'S ALL GONE!!!!!

Where do I put my stuff?

- /group
  - → Think "/home" for work groups
    - » papers, thesis, etc
  - → hourly snapshots
    - » `cd .snapshot/`
  - → analysis scripts: YES
    - » use git!
  - → source code: ~OK
    - » /work is better
  - → papers, thesis, etc in user subdirs is great

- /work
  - → Tuned for speed, small files
    - » ie. source code, compilation, etc.
  - → NOT backed up
    - » but is resilient
    - » `rm` can still bite you
  - → Source code: YES
    - » use git!
  - → ROOT output: ~ick
  - → CODA data: No
  - → YOU must backup to tape
    - » tar + jput *(more on this soon)*

Where do I put my stuff?

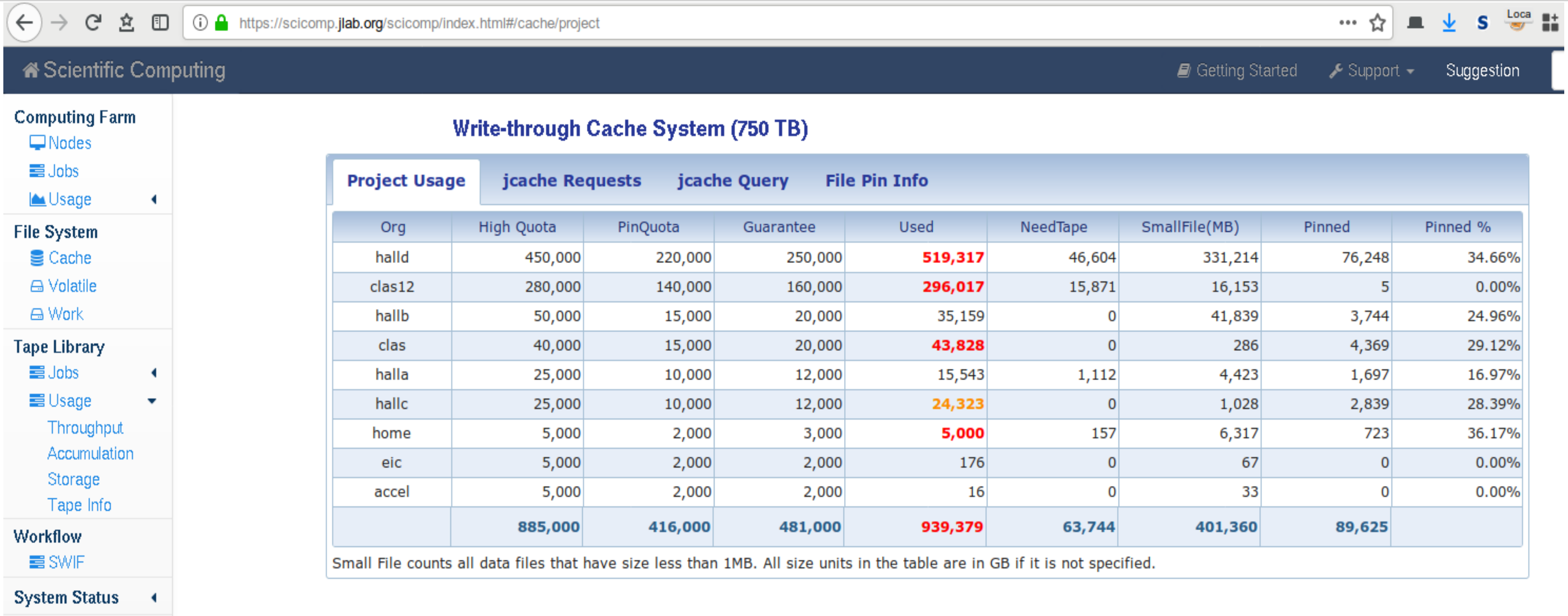
- /volatile
  - →Huge file system
    - » ~ 180 TB
  - →High performance for large files
    - » ie. ROOT output
  - →NOT backed up
  - →Files auto-cleaned based on quota/ reservation/ and filesystem pressure
    - » https://scicomp.jlab.org/scicomp/index.html#/volatile
  - →Analysis output goes here!
    - » Check, then push to tape if good!

- Tape System
  - →Really huge
    - » 43 PB and growing
  - →/mss/hallX/...
    - » Stubs: shows what is in the tape system!
    - » not the actual files
  - →/cache/hallX/...
    - » actual files
    - » auto-clean up in play
      - – next slide

Accessing files from Tape

- Retrieving files from tape
  - →jcache get /mss/.../foo.dat
    - » Manual pull from tape to /cache/.../foo.dat
    - » Never call this (or jget) in a farm script!
      - – Let Auger/SWIF do it!
        - » List needed files on <Input> tag
        - » Auger will prestage them for you in advance
  - →jget /mss/.../foo.dat $PWD/
    - » pull file from tape to any filesystem
    - » generally not the right tool

File duration in /cache



Write-through Cache System (750 TB)

Org	High Quota	PinQuota	Guarantee	Used	NeedTape	SmallFile(MB)	Pinned	Pinned %
halld	450,000	220,000	250,000	519,317	46,604	331,214	76,248	34.66%
clas12	280,000	140,000	160,000	296,017	15,871	16,153	5	0.00%
hallb	50,000	15,000	20,000	35,159	0	41,839	3,744	24.96%
clas	40,000	15,000	20,000	43,828	0	286	4,369	29.12%
halla	25,000	10,000	12,000	15,543	1,112	4,423	1,697	16.97%
hallc	25,000	10,000	12,000	24,323	0	1,028	2,839	28.39%
home	5,000	2,000	3,000	5,000	157	6,317	723	36.17%
eic	5,000	2,000	2,000	176	0	67	0	0.00%
accel	5,000	2,000	2,000	16	0	33	0	0.00%
	885,000	416,000	481,000	939,379	63,744	401,360	89,625	

Small File counts all data files that have size less than 1MB. All size units in the table are in GB if it is not specified.

- Files auto-cleaned based on quota and system pressure on /cache
  - →Clean up least-recently-used files first
  - →Can 'pin' files to keep them stable
    - » Shared resource, don't abuse!

Copying files to Tape

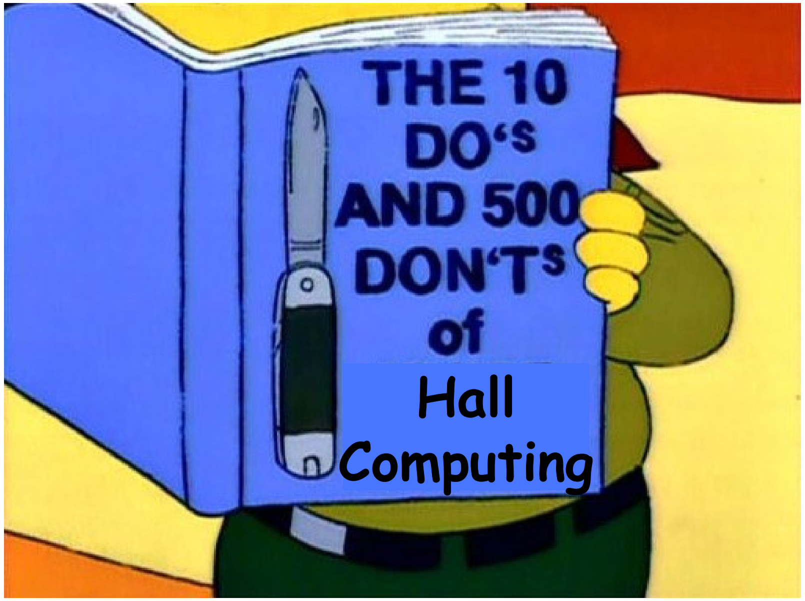
- Storing files on tape
  - → jput file /mss/.../
    - » 'jput -h'
    - » Online Docs
  - → 'write-through cache' (Online Docs)
    - » write large file output directly to /cache/hallX/...
      - – no 'staging' on /volatile
    - » automagically backed up to tape after a few days
      - – guaranteed to be safe on tape before /cache auto-removal kicks in
    - » Gotchas:
      - – small files (<1MB) not backed up to tape
      - – avoid pathname collisions with files already on tape
        - » ie. 'overwriting' files with same name, etc

Quick Breather

Any Questions on that bit?

Hall Computing while Running / Online Analysis

Hall Computing Do's and Don'ts!



Rules to live by so your colleagues
don't curse your name

Donny Don't!

- **Don't** copy or write large files to any "home" directory. This includes
  - →CODA files
  - →ROOT files
  - →tar or zip archives
- When these shared filesystems fill, *many things break*
  - →processes fail with corrupt output (not just yours!)
  - →machines may require rebooting (disruptive!)
  - →DAQ can die!

- **Do use** destinations like:
  - →/chafs/work*, /chafs2/work*
  - →/net/cdaq/*
  - →Use symlinks in local dirs
- **Do use** the tape silo:
  - →/mss/cache/hallX/...

Donny Don't!

- **Don't** change global config files in common accounts (adaq, cdaq, a-onl, cvxwrks, coda, *etc*)
  - » ie .bashrc, .cshrc
- **Don't** change environment variables (*ie.* `'setenv ...'`)
- **Don't** edit or save files in the global replay directories
- **Don't** run analysis on the DAQ machines
  - →All can have unexpected, and difficult to debug impacts on processes running behind the scenes

- **Do use** the appropriate machines and accounts
- **Do use** the JLab Farm:

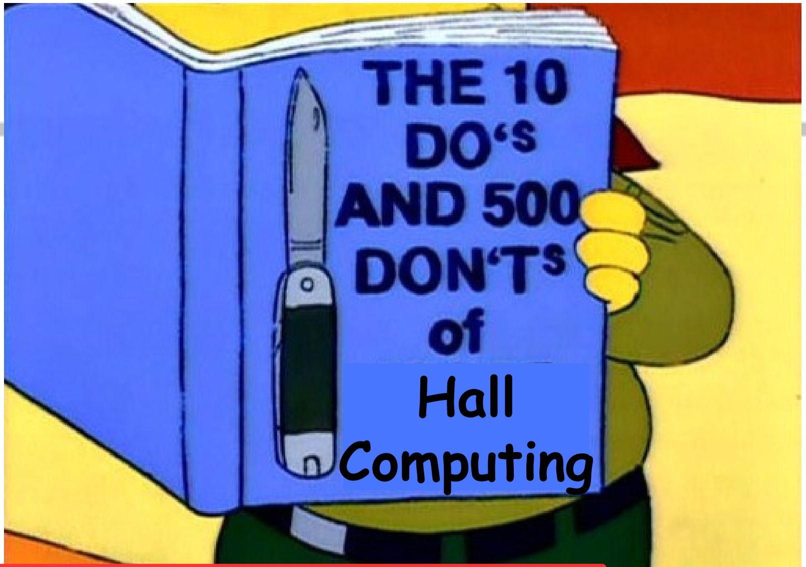
Donny Don't!

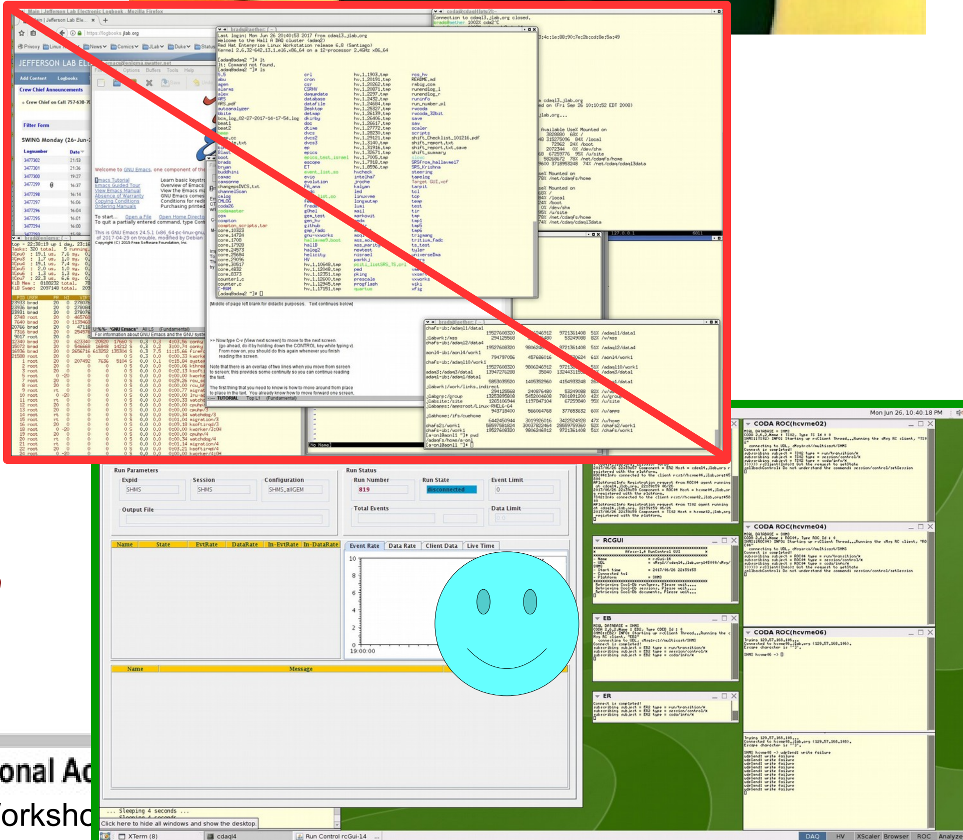
- Don't use buggy/untested code in production environments
  - →If your code is:
    - » generating enormous 'log' files
    - » generating core.NNNN dumps
    - » littering hv.1.NNNN.tmp files
    - » pegging a core at 100% without good reason
    - » spewing warnings / debug info
  - →*Then it needs fixing before the experiment*
- Don't ignore warning in your replay scripts
  - →*NEVER* disable/hide warnings

- We've lost way too many beam hours ($$$ + data) to such problems.
- Poor saps have to come in at crazy hours to fix things, and hassle folks to clean up.
  - →Have mercy on us :-)

Donny Don't!

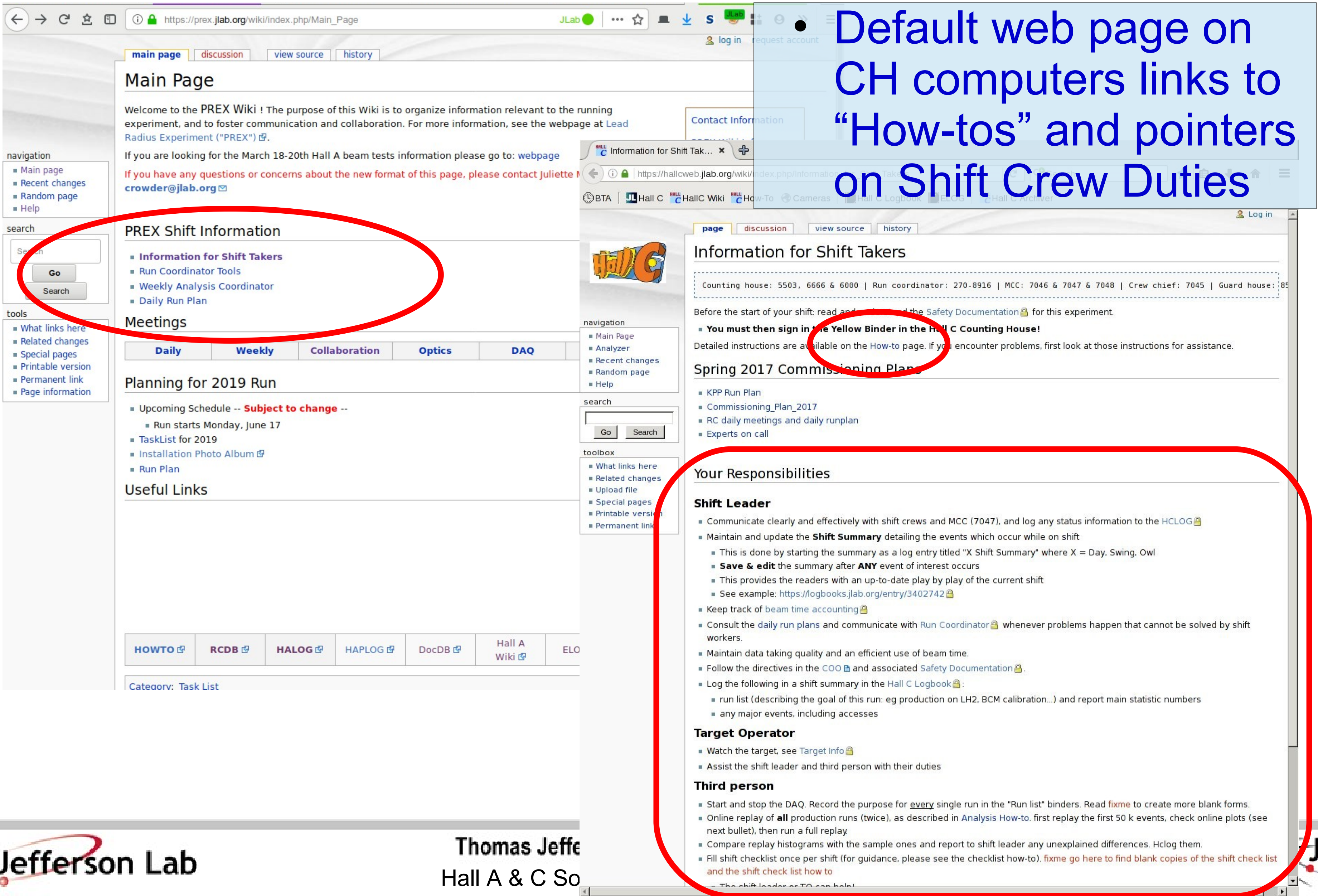
- Remember the Hall Computers are shared machines
  - →**Don't** clutter CH screens with 100s of windows
  - →**Don't** shuffle windows around on Shift Crew Machines
  - →**Clean up** after yourselves
    - » Copy to tape, and/or remove obsolete files
    - » *NEVER move or touch CODA data files though*
      - – *talk to Hall expert*





Online Help is Available

- Default web page on CH computers links to "How-tos" and pointers on Shift Crew Duties



https://prex.jlab.org/wiki/index.php/Main_Page

main page | discussion | view source | history

Main Page

Welcome to the PREX Wiki ! The purpose of this Wiki is to organize information relevant to the running experiment, and to foster communication and collaboration. For more information, see the webpage at Lead Radius Experiment ("PREX").

If you are looking for the March 18-20th Hall A beam tests information please go to: webpage

If you have any questions or concerns about the new format of this page, please contact Juliette M crowder@jlab.org

PREX Shift Information

- **Information for Shift Takers**
- Run Coordinator Tools
- Weekly Analysis Coordinator
- Daily Run Plan

Meetings

Daily	Weekly	Collaboration	Optics	DAQ

Planning for 2019 Run

- Upcoming Schedule -- **Subject to change** --
  - Run starts Monday, June 17
- TaskList for 2019
- Installation Photo Album
- Run Plan

Useful Links

HOWTO	RCDB	HALOG	HAPLOG	DocDB	Hall A Wiki	ELO

Category: Task List

Contact Information

Information for Shift Tak...

https://hallcweb.jlab.org/wiki/index.php/Information_for_Shift_Takers

Information for Shift Takers

```
Counting house: 5503, 6666 & 6000 | Run coordinator: 270-8916 | MCC: 7046 & 7047 & 7048 | Crew chief: 7045 | Guard house: 85
```

Before the start of your shift: read and understand the Safety Documentation for this experiment.

- **You must then sign in the Yellow Binder in the Hall C Counting House!**

Detailed instructions are available on the How-to page. If you encounter problems, first look at those instructions for assistance.

Spring 2017 Commissioning Plans

- KPP Run Plan
- Commissioning_Plan_2017
- RC daily meetings and daily runplan
- Experts on call

Your Responsibilities

Shift Leader

- Communicate clearly and effectively with shift crews and MCC (7047), and log any status information to the HCLOG
- Maintain and update the **Shift Summary** detailing the events which occur while on shift
  - This is done by starting the summary as a log entry titled "X Shift Summary" where X = Day, Swing, Owl
  - **Save & edit** the summary after **ANY** event of interest occurs
  - This provides the readers with an up-to-date play by play of the current shift
  - See example: https://logbooks.jlab.org/entry/3402742
- Keep track of beam time accounting
- Consult the daily run plans and communicate with Run Coordinator whenever problems happen that cannot be solved by shift workers.
- Maintain data taking quality and an efficient use of beam time.
- Follow the directives in the COO and associated Safety Documentation.
- Log the following in a shift summary in the Hall C Logbook:
  - run list (describing the goal of this run: eg production on LH2, BCM calibration...) and report main statistic numbers
  - any major events, including accesses

Target Operator

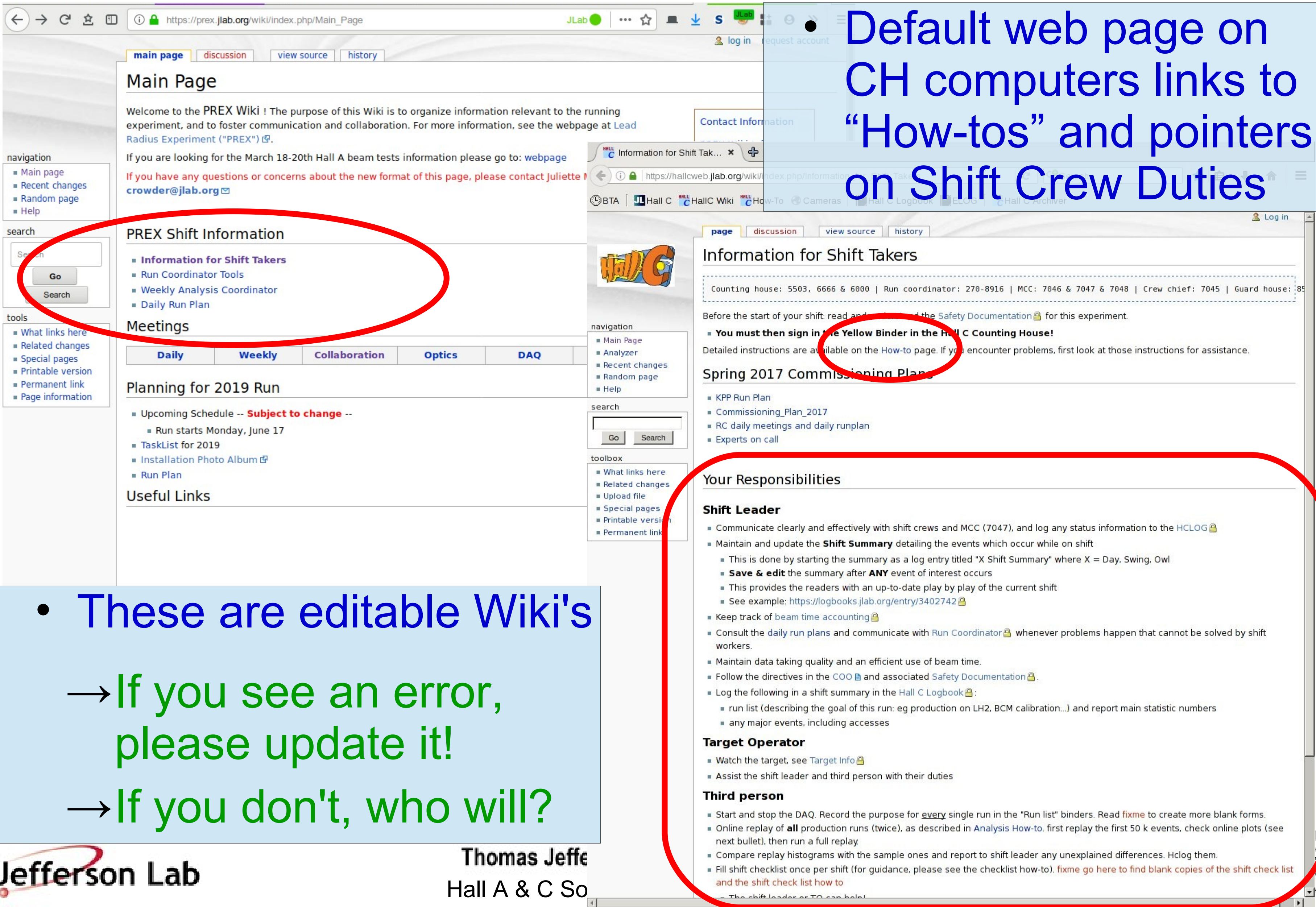
- Watch the target, see Target Info
- Assist the shift leader and third person with their duties

Third person

- Start and stop the DAQ. Record the purpose for every single run in the "Run list" binders. Read fixme to create more blank forms.
- Online replay of **all** production runs (twice), as described in Analysis How-to. first replay the first 50 k events, check online plots (see next bullet), then run a full replay.
- Compare replay histograms with the sample ones and report to shift leader any unexplained differences. Hclog them.
- Fill shift checklist once per shift (for guidance, please see the checklist how-to). fixme go here to find blank copies of the shift check list and the shift check list how to
  - The shift leader or TO can help!

Online Help is Available

- Default web page on CH computers links to “How-tos” and pointers on Shift Crew Duties



https://prex.jlab.org/wiki/index.php/Main_Page

log in request account

main page | discussion | view source | history

Main Page

Welcome to the PREX Wiki ! The purpose of this Wiki is to organize information relevant to the running experiment, and to foster communication and collaboration. For more information, see the webpage at Lead Radius Experiment ("PREX").

If you are looking for the March 18-20th Hall A beam tests information please go to: webpage

If you have any questions or concerns about the new format of this page, please contact Juliette M crowder@jlab.org

Contact Information

navigation
- Main page
- Recent changes
- Random page
- Help

search

Go | Search

tools
- What links here
- Related changes
- Special pages
- Printable version
- Permanent link
- Page information

PREX Shift Information

- **Information for Shift Takers**
- Run Coordinator Tools
- Weekly Analysis Coordinator
- Daily Run Plan

Meetings

Daily	Weekly	Collaboration	Optics	DAQ

Planning for 2019 Run

- Upcoming Schedule -- **Subject to change** --
  - Run starts Monday, June 17
- TaskList for 2019
- Installation Photo Album
- Run Plan

Useful Links

Information for Shift Tak...

https://hallcweb.jlab.org/wiki/index.php/Information_for_Shift_Takers

BTA | Hall C | HallC Wiki | How-To | Cameras | Hall C Logbook | ELOG | Hall C Archiver

Log in

page | discussion | view source | history

navigation
- Main Page
- Analyzer
- Recent changes
- Random page
- Help

search

Go | Search

toolbox
- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link

Information for Shift Takers

```
Counting house: 5503, 6666 & 6000 | Run coordinator: 270-8916 | MCC: 7046 & 7047 & 7048 | Crew chief: 7045 | Guard house: 85
```

Before the start of your shift: read and understand the Safety Documentation for this experiment.

- **You must then sign in the Yellow Binder in the Hall C Counting House!**

Detailed instructions are available on the How-to page. If you encounter problems, first look at those instructions for assistance.

Spring 2017 Commissioning Plans

- KPP Run Plan
- Commissioning_Plan_2017
- RC daily meetings and daily runplan
- Experts on call

Your Responsibilities

Shift Leader

- Communicate clearly and effectively with shift crews and MCC (7047), and log any status information to the HCLOG
- Maintain and update the **Shift Summary** detailing the events which occur while on shift
  - This is done by starting the summary as a log entry titled "X Shift Summary" where X = Day, Swing, Owl
  - **Save & edit** the summary after **ANY** event of interest occurs
  - This provides the readers with an up-to-date play by play of the current shift
  - See example: https://logbooks.jlab.org/entry/3402742
- Keep track of beam time accounting
- Consult the daily run plans and communicate with Run Coordinator whenever problems happen that cannot be solved by shift workers.
- Maintain data taking quality and an efficient use of beam time.
- Follow the directives in the COO and associated Safety Documentation.
- Log the following in a shift summary in the Hall C Logbook:
  - run list (describing the goal of this run: eg production on LH2, BCM calibration...) and report main statistic numbers
  - any major events, including accesses

Target Operator

- Watch the target, see Target Info
- Assist the shift leader and third person with their duties

Third person

- Start and stop the DAQ. Record the purpose for every single run in the "Run list" binders. Read fixme to create more blank forms.
- Online replay of **all** production runs (twice), as described in Analysis How-to. first replay the first 50 k events, check online plots (see next bullet), then run a full replay.
- Compare replay histograms with the sample ones and report to shift leader any unexplained differences. Hclog them.
- Fill shift checklist once per shift (for guidance, please see the checklist how-to). fixme go here to find blank copies of the shift check list and the shift check list how to
  - The shift leader or TO can help!

- These are editable Wiki's

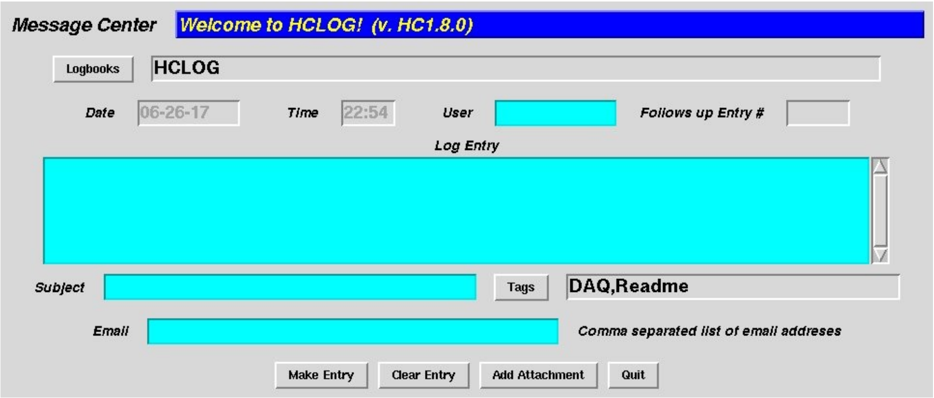
→If you see an error, please update it!

→If you don't, who will?

Logbook Tips

- Always use concise “Subject”
    - » No log entries titled “problem” please...
    - → Make clear entries that don't rely on being there to make sense
        - » We need to under-stand the entries months/years later
    - → Clean up the “`Re: Follow-up: Follow-up: Re: Follow-up: ...`” garbage
        - » it is just noise
    - → Manually link to relevant older entries
    - → Use Tags fields
        - » DAQ, Analysis, Techs, ...

- HALOG, HCLOG are web based:
    - → https://logbooks.jlab.org/book/halog
    - → https://logbooks.jlab.org/book/hclog
- On CH computers, can also make entries via standalone tool:
    - » halog / hclog
    - →Simpler snapshots!


Message Center: Welcome to HCLOG! (v. HC1.8.0)

Logbooks: HCLOG

Date: 06-26-17 Time: 22:54 User: Follows up Entry #:

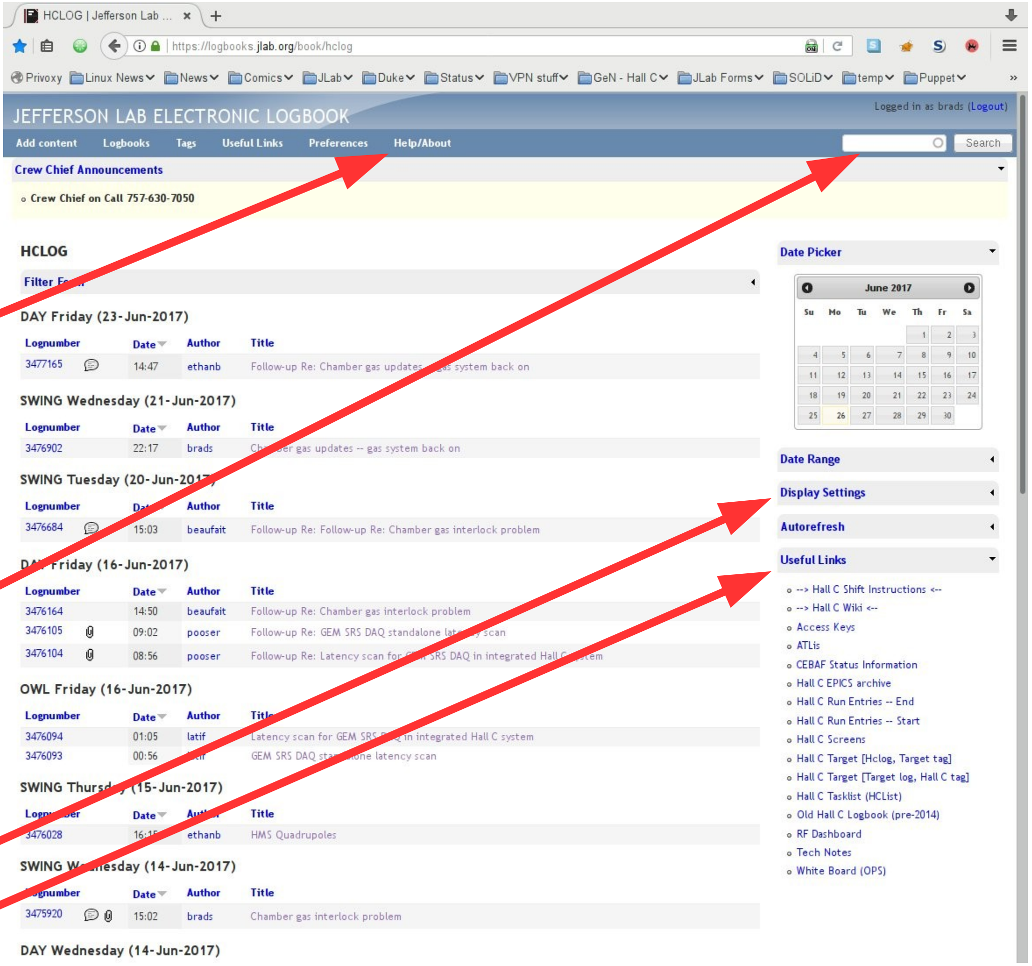
Log Entry:

Subject: Tags: DAQ,Readme

Email: Comma separated list of email addreses

Logbook Tips

- HALOG, HCLOG are web based:
  - → https://logbooks.jlab.org/book/halog
  - → https://logbooks.jlab.org/book/hclog
- Help is worth reading
- Useful 'tricks'
  - →Use UPPER-CASE boolean logic in search
    - » ie. AND *not* and
  - → `Display Settings: Hide Autologs`
  - → `Useful Links`



Crew Chief Announcements

- Crew Chief on Call 757-630-7050

HCLOG

Filter Form

DAY Friday (23-Jun-2017)

Lognumber	Date	Author	Title
3477165 💬	14:47	ethanb	Follow-up Re: Chamber gas updates -- gas system back on

SWING Wednesday (21-Jun-2017)

Lognumber	Date	Author	Title
3476902	22:17	brads	Chamber gas updates -- gas system back on

SWING Tuesday (20-Jun-2017)

Lognumber	Date	Author	Title
3476684 💬	15:03	beaufait	Follow-up Re: Follow-up Re: Chamber gas interlock problem

DAY Friday (16-Jun-2017)

Lognumber	Date	Author	Title
3476164	14:50	beaufait	Follow-up Re: Chamber gas interlock problem
3476105 📎	09:02	pooser	Follow-up Re: GEM SRS DAQ standalone latency scan
3476104 📎	08:56	pooser	Follow-up Re: Latency scan for GEM SRS DAQ in integrated Hall C system

OWL Friday (16-Jun-2017)

Lognumber	Date	Author	Title
3476094	01:05	latif	Latency scan for GEM SRS DAQ in integrated Hall C system
3476093	00:56	latif	GEM SRS DAQ standalone latency scan

SWING Thursday (15-Jun-2017)

Lognumber	Date	Author	Title
3476028	16:15	ethanb	HMS Quadrupoles

SWING Wednesday (14-Jun-2017)

Lognumber	Date	Author	Title
3475920 💬 📎	15:02	brads	Chamber gas interlock problem

DAY Wednesday (14-Jun-2017)

Date Picker

June 2017

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

Date Range

Display Settings

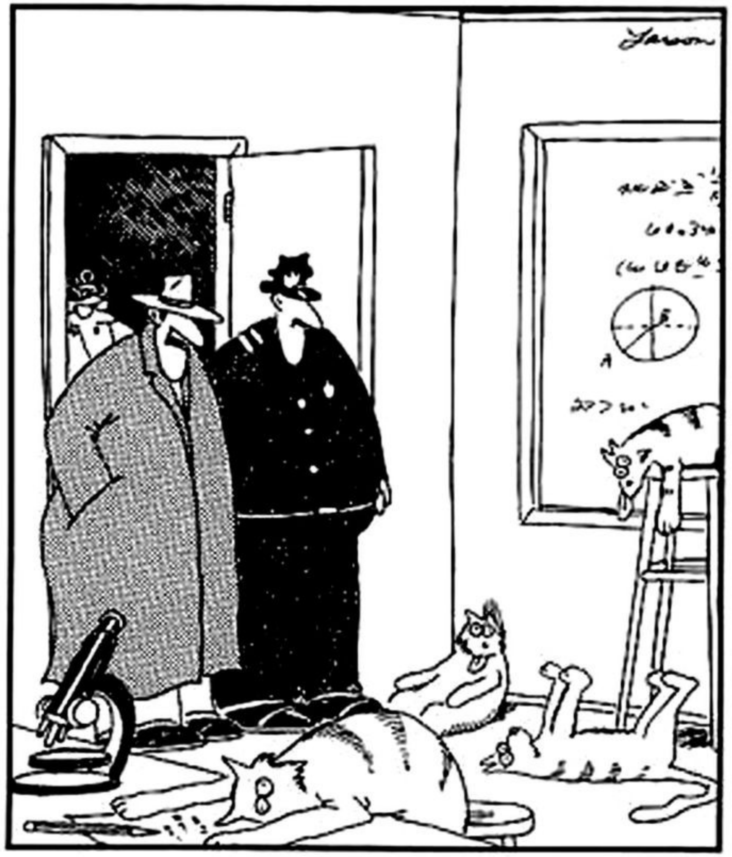
Autorefresh

Useful Links

- --> Hall C Shift Instructions <--
- --> Hall C Wiki <--
- Access Keys
- ATLis
- CEBAF Status Information
- Hall C EPICS archive
- Hall C Run Entries -- End
- Hall C Run Entries -- Start
- Hall C Screens
- Hall C Target [Hclog, Target tag]
- Hall C Target [Target log, Hall C tag]
- Hall C Tasklist (HCList)
- Old Hall C Logbook (pre-2014)
- RF Dashboard
- Tech Notes
- White Board (OPS)

And, Most Importantly

- If you see something you don't understand...
  - →Ask someone
  - →Make a log entry
  - →Dig in and beat on the problem until it makes sense to *you*
    - » You're scientists – understanding weirdness is literally our business!



"Notice all the computations, theoretical scribblings, and lab equipment, Norm. ... Yes, curiosity killed these cats."

Now ask Questions!

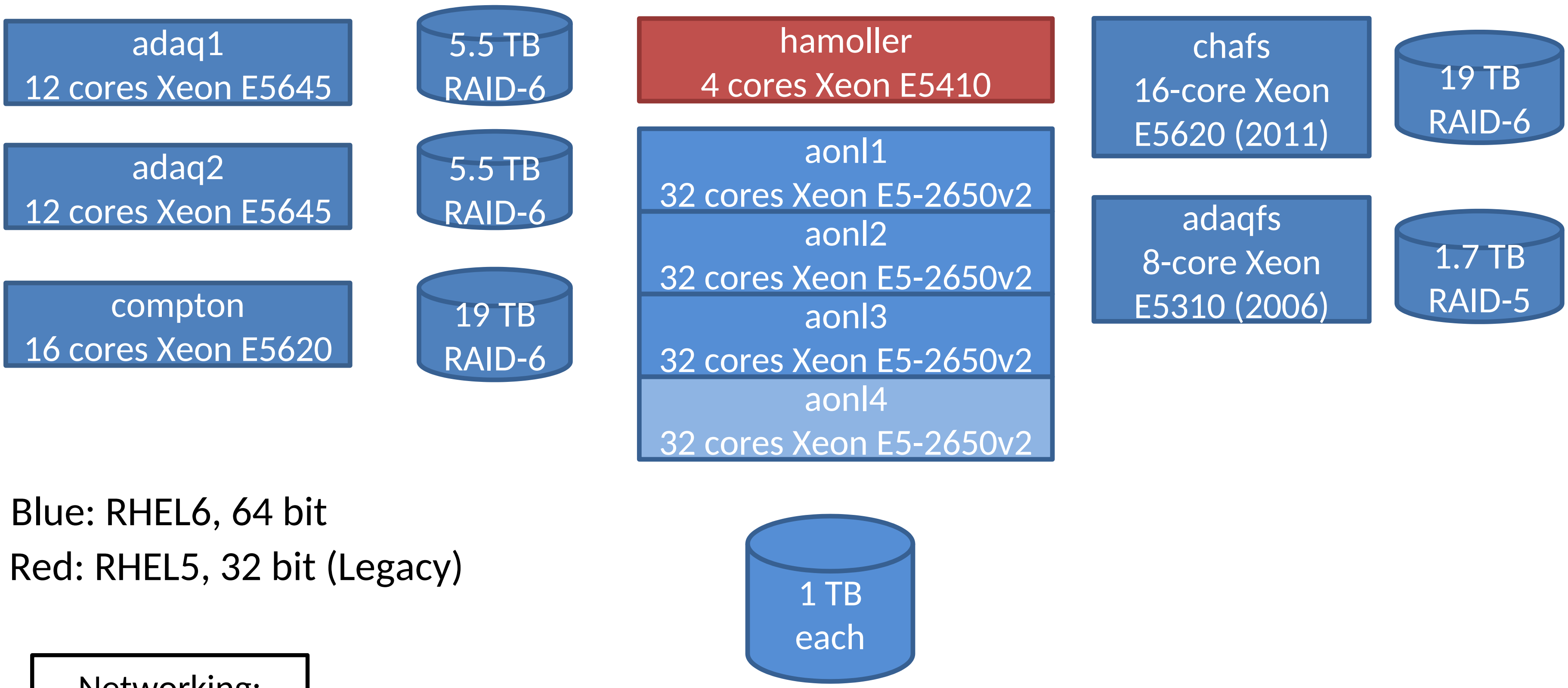
Hall A/C Computer Layout (possibly dated...)

Hall A Counting House Systems

DAQ

Online Analysis

Servers



adaq1
12 cores Xeon E5645

5.5 TB
RAID-6

adaq2
12 cores Xeon E5645

5.5 TB
RAID-6

compton
16 cores Xeon E5620

19 TB
RAID-6

hamoller
4 cores Xeon E5410

aonl1
32 cores Xeon E5-2650v2

aonl2
32 cores Xeon E5-2650v2

aonl3
32 cores Xeon E5-2650v2

aonl4
32 cores Xeon E5-2650v2

1 TB
each

chafs
16-core Xeon
E5620 (2011)

19 TB
RAID-6

adaqfs
8-core Xeon
E5310 (2006)

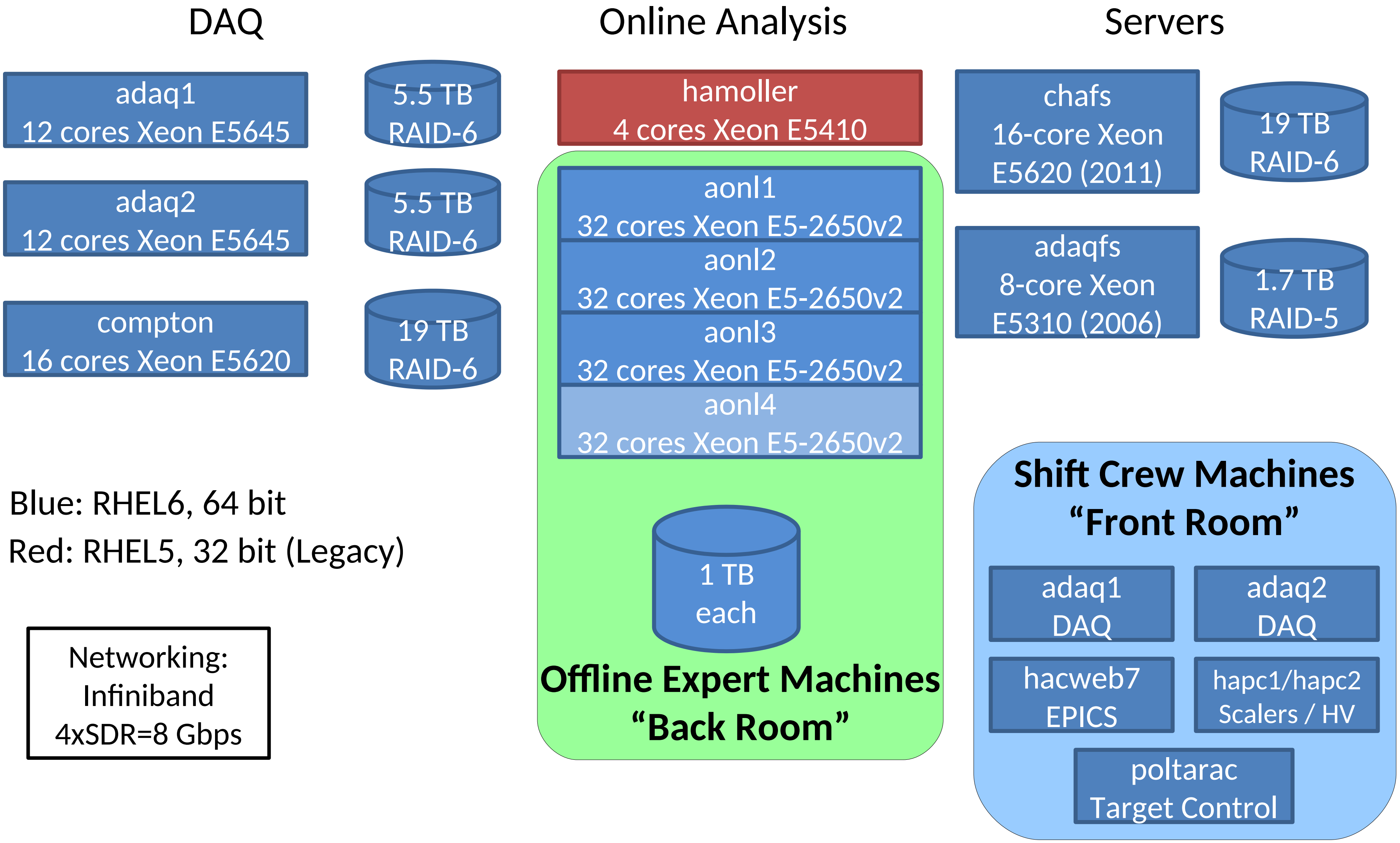
1.7 TB
RAID-5

Blue: RHEL6, 64 bit

Red: RHEL5, 32 bit (Legacy)

Networking:
Infiniband
4xSDR=8 Gbps

Hall A Counting House Systems



DAQ

adaq1
12 cores Xeon E5645

5.5 TB
RAID-6

adaq2
12 cores Xeon E5645

5.5 TB
RAID-6

compton
16 cores Xeon E5620

19 TB
RAID-6

Online Analysis

hamoller
4 cores Xeon E5410

aonl1
32 cores Xeon E5-2650v2

aonl2
32 cores Xeon E5-2650v2

aonl3
32 cores Xeon E5-2650v2

aonl4
32 cores Xeon E5-2650v2

1 TB
each

Offline Expert Machines
“Back Room”

Servers

chafs
16-core Xeon
E5620 (2011)

19 TB
RAID-6

adaqfs
8-core Xeon
E5310 (2006)

1.7 TB
RAID-5

Shift Crew Machines
“Front Room”

adaq1
DAQ

adaq2
DAQ

hacweb7
EPICS

hapc1/hapc2
Scalers / HV

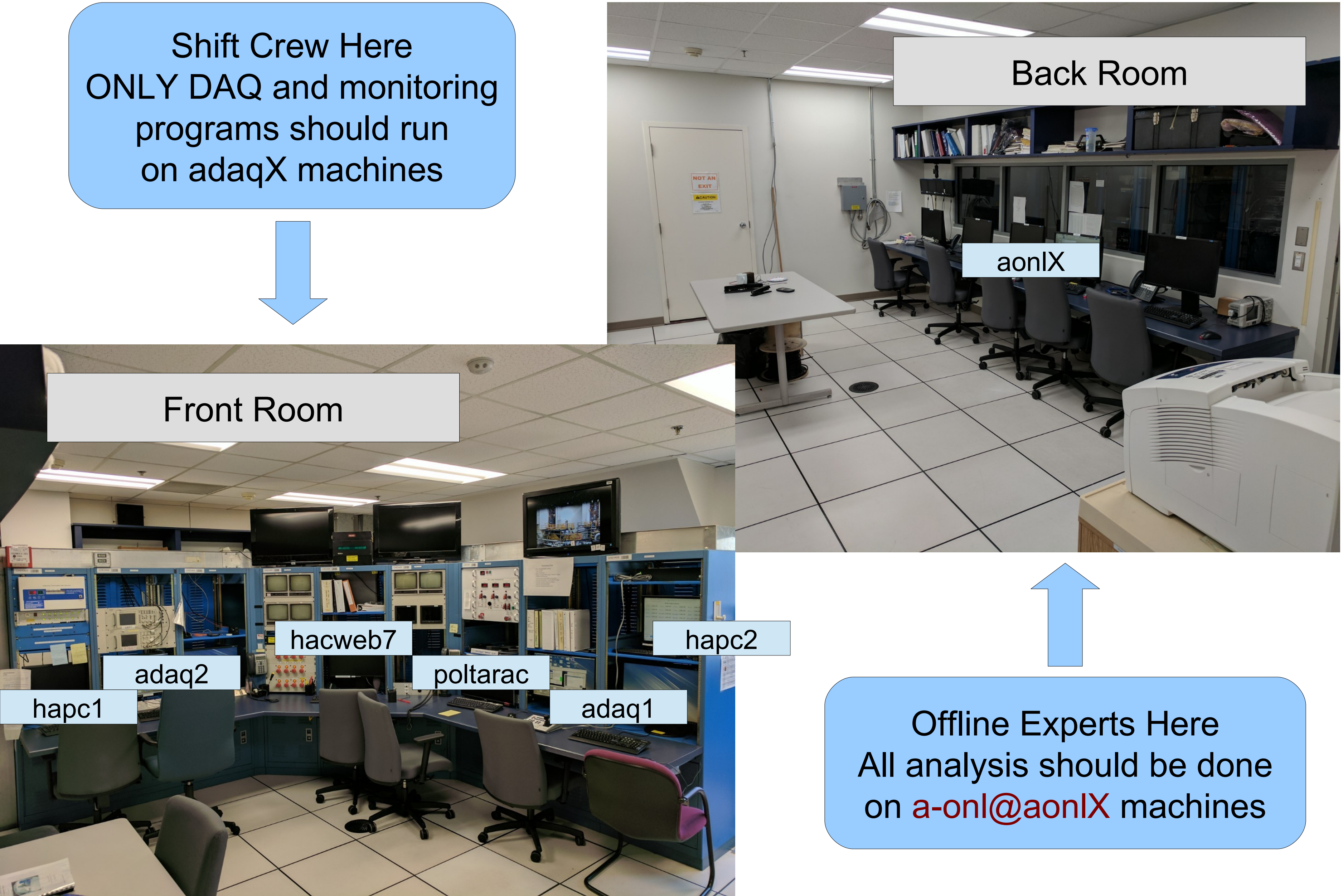
poltarac
Target Control

Blue: RHEL6, 64 bit

Red: RHEL5, 32 bit (Legacy)

Networking:
Infiniband
4xSDR=8 Gbps

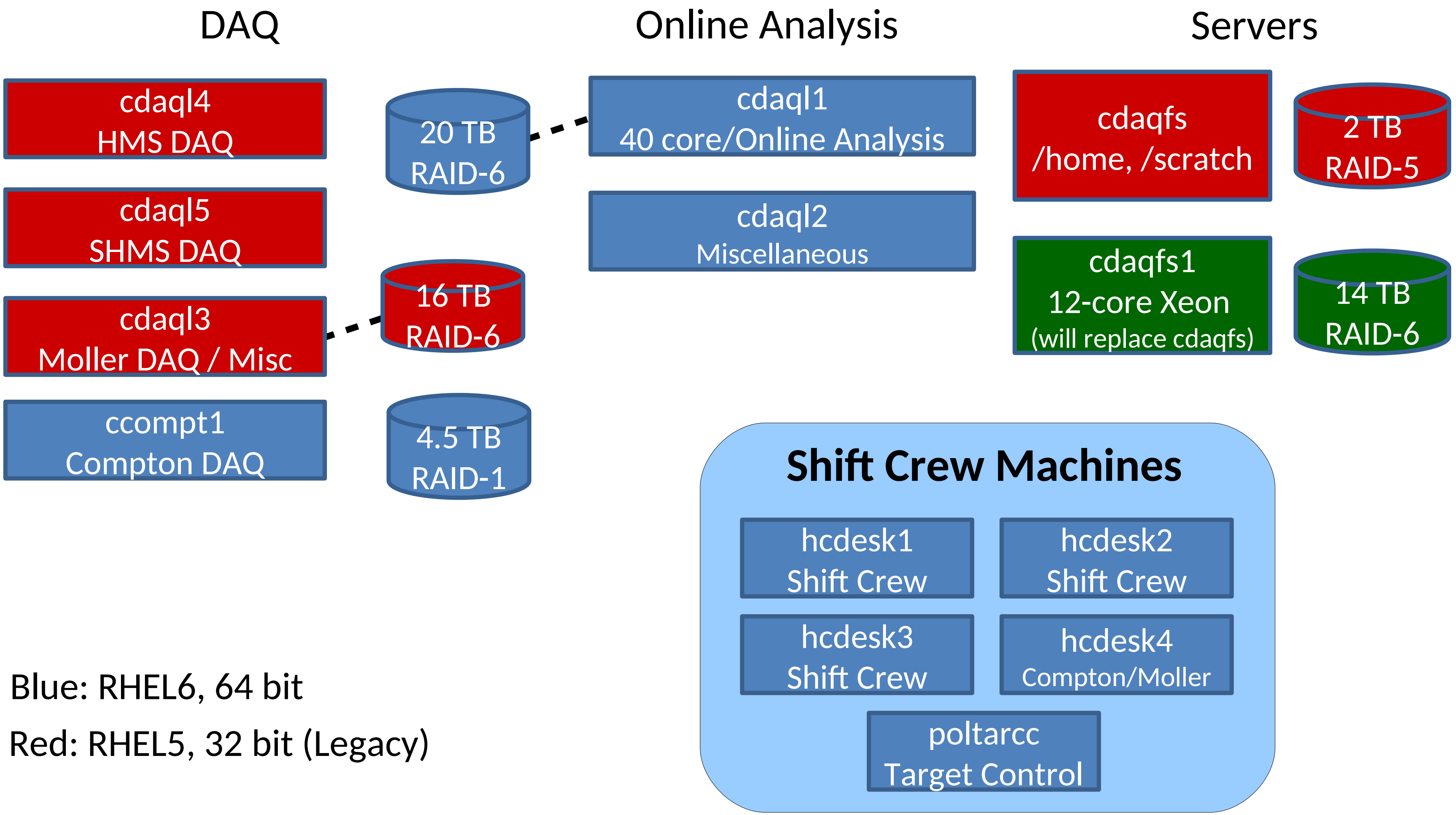
Hall A Counting House Consoles



Shift Crew Here
ONLY DAQ and monitoring programs should run on adaqX machines

Offline Experts Here
All analysis should be done on a-onl@aonlX machines

Hall C Counting House Systems



DAQ

cdaql4
HMS DAQ

cdaql5
SHMS DAQ

cdaql3
Moller DAQ / Misc

ccompt1
Compton DAQ

20 TB
RAID-6

16 TB
RAID-6

4.5 TB
RAID-1

Online Analysis

cdaql1
40 core/Online Analysis

cdaql2
Miscellaneous

Servers

cdaqfs
/home, /scratch

2 TB
RAID-5

cdaqfs1
12-core Xeon
(will replace cdaqfs)

14 TB
RAID-6

Shift Crew Machines

hcdesk1
Shift Crew

hcdesk2
Shift Crew

hcdesk3
Shift Crew

hcdesk4
Compton/Moller

poltarcc
Target Control

Blue: RHEL6, 64 bit

Red: RHEL5, 32 bit (Legacy)

Hall C Counting House Consoles

hcdesk* machines are used as frontends to access the DAQ (cdaql[45]) and analysis machines (cdaql1) using “go_*” scripts and/or ssh