



# Batch Job Submission Best Practices

*Clarín Newspaper, public domain*

Ken Herner  
 FIFE Workshop  
 22 June 2017

22 June 1986:  
 “Hand of God”  
 and “Goal of the  
 Century”



# Outline

- Behind the scenes in your job: common problems
  - How to avoid said problems
- Best practices in job management (resource requests, script structure)
- Where to go for help

## Girl Scout Law

I will do my best to

...

**use resources wisely,**

...

make the world a better place,

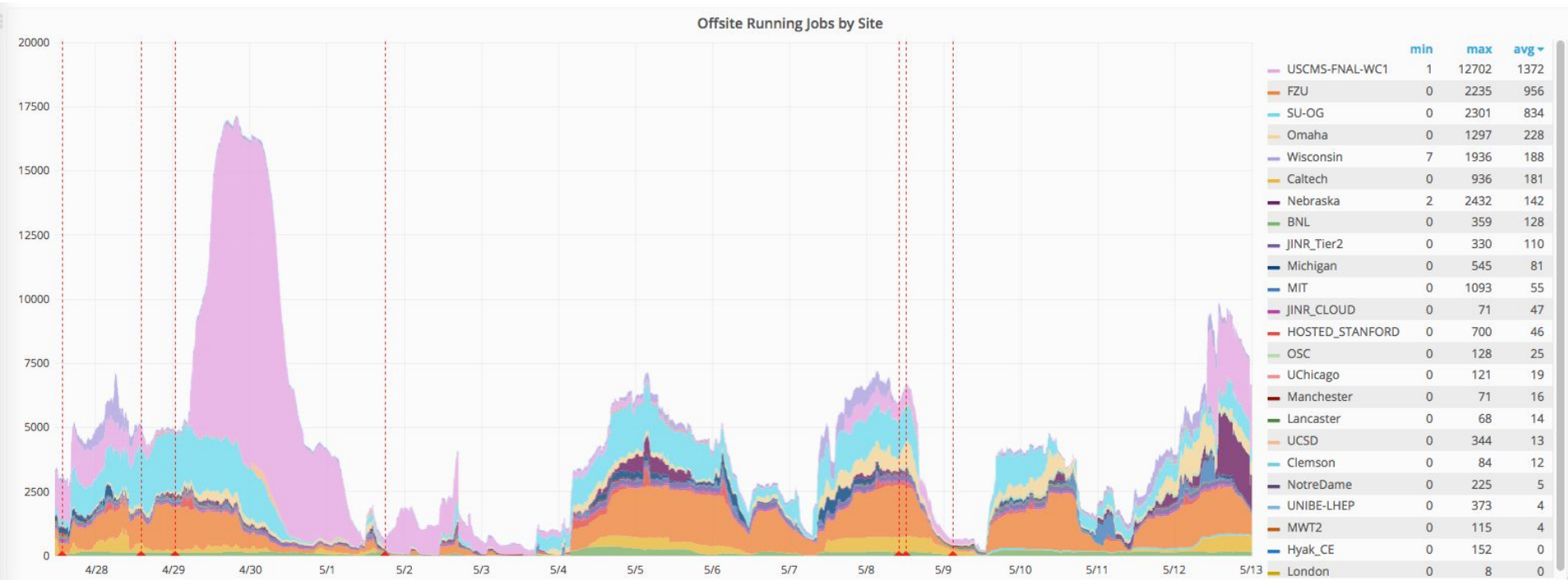
....

# When you submit a job

- You execute `jobsub_submit ... <your_job_spec>`
  - You get credentials on the jobsub server, executables and input files get copied around, and the job is submitted for you behind the scenes
  - You wait in the queue for a while and you start running on a worker node
- Your job doesn't just run via HTCondor on the worker node. It actually runs as part of a **glidein** (part of the [glideinWMS](#) system) on the worker node
- The glideins are **partitionable**: they start out with access to a number of cores with some memory (8 CPUs and 16384MB memory on GPGrid; other sites differ) and they're then carved up into sandboxes for individual jobs like yours.
- Glideins have a finite lifetime (few hours to several days depending on the site.) There is a cutoff time for accepting new jobs. That's why we ask you for an expected lifetime; we don't want to match you to slots that will disappear while you're running.

# Getting more resources offsite

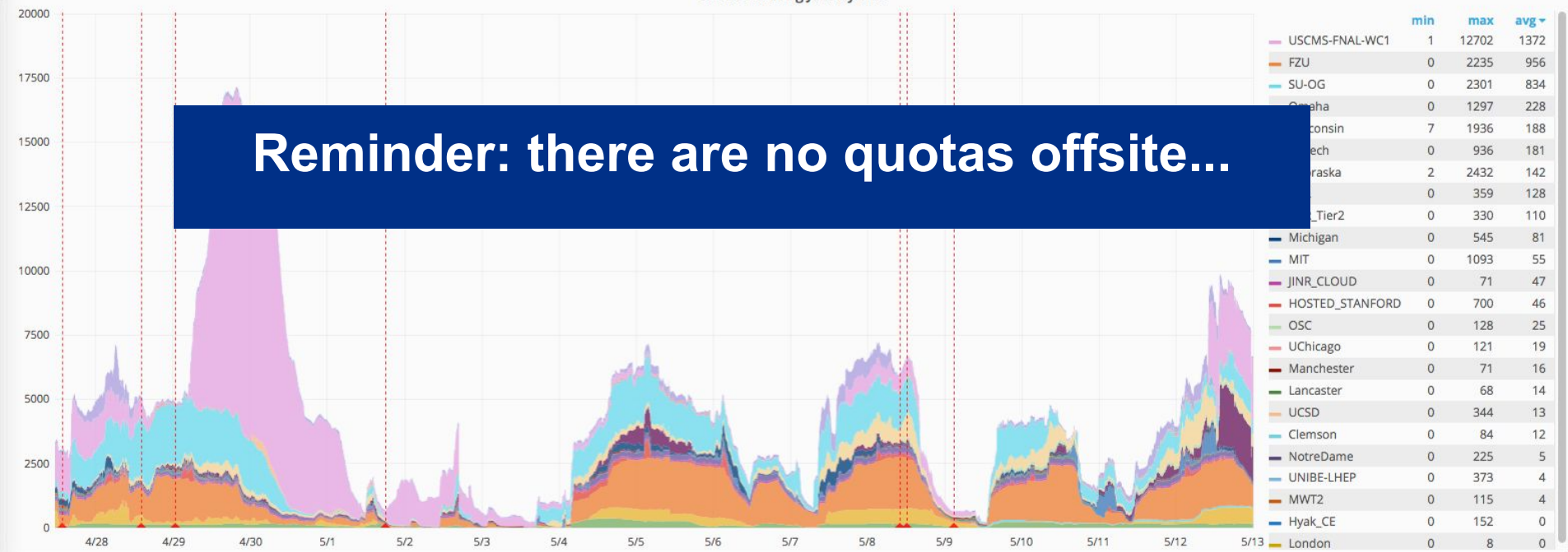
There are potentially thousands of cores available beyond GPGrid.



# Getting more resources offsite

There are potentially thousands of cores available beyond GPGrid.

Offsite Running Jobs by Site



Reminder: there are no quotas offsite...

# Getting more resources offsite



SCD Summary - CMS -



Back to dashboard < Zoom Out > ☉ May 5, 2017 15:23:41 to Jun 15, 2017 15:18:41



Outages



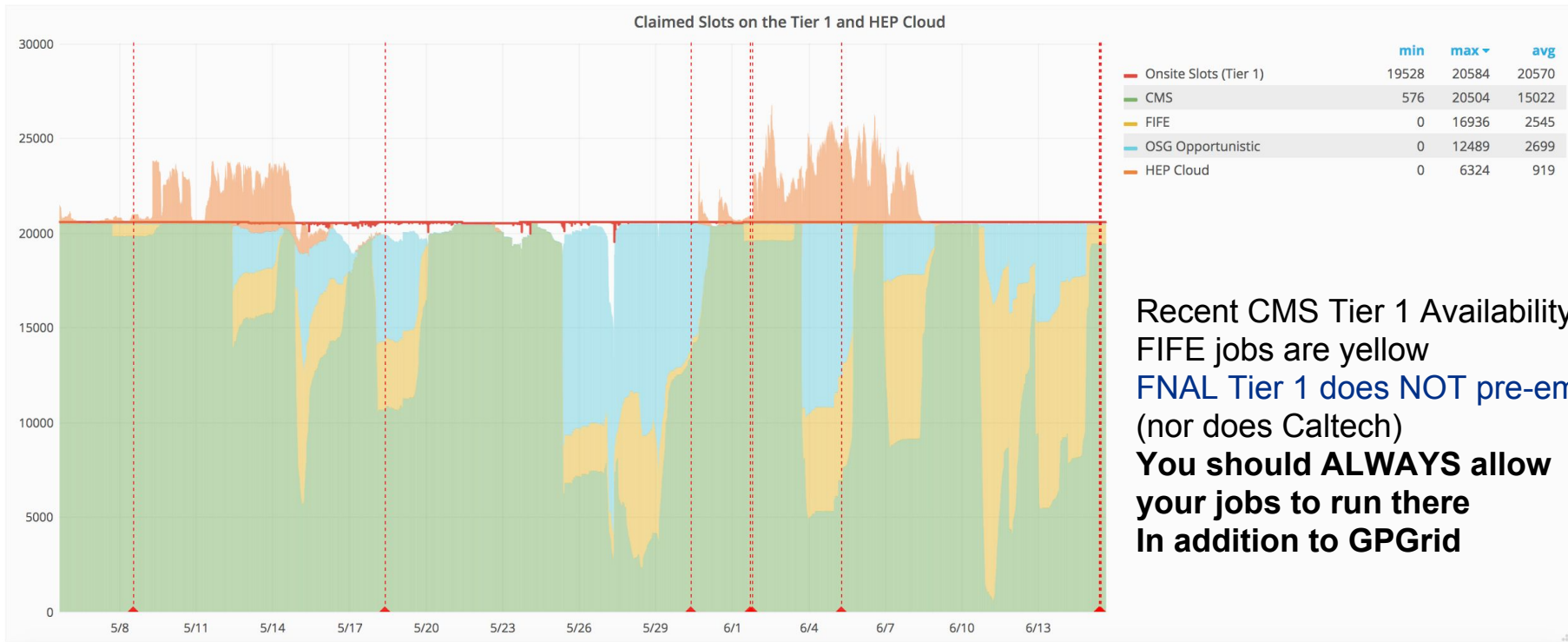
SCD Summary - FIFE

SCD Summary - FIFE 2

SCD Summary - FIFE 3

SCD Summary - GPGGrid

SCD Summary - GPGGrid (No DES Direct)



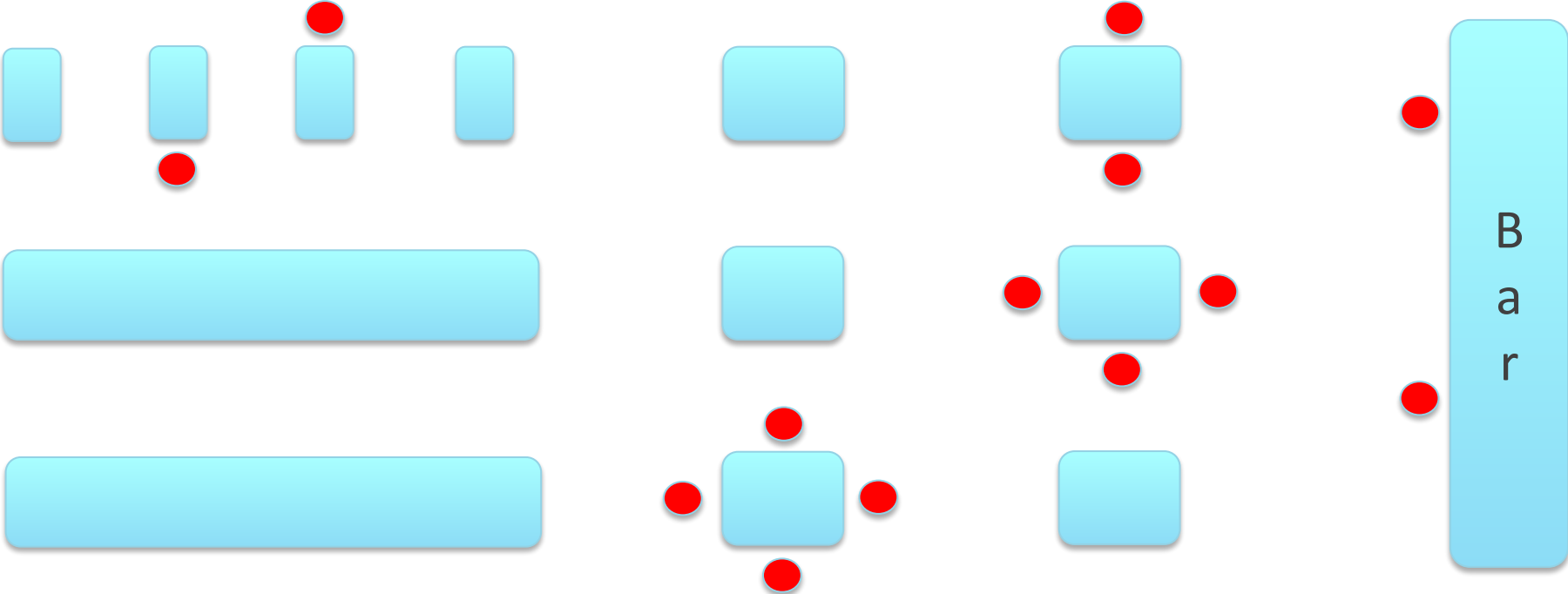
Recent CMS Tier 1 Availability  
FIFE jobs are yellow  
FNAL Tier 1 does NOT pre-empt  
(nor does Caltech)  
**You should ALWAYS allow  
your jobs to run there  
In addition to GPGGrid**

# Best Practice: Accurate resource requests

- Especially check **memory, disk, and run time**
- If you use Art/LArSoft, **use the profiling tools to determine your resource needs**
- Request only what you need. The default might be more than you need. If it is, don't just stick with it. You can gain a lot by requesting *less* than the default.
- Jobs are matched to slots based on resource **requests**.
- Consider the problem of getting a table in a restaurant:

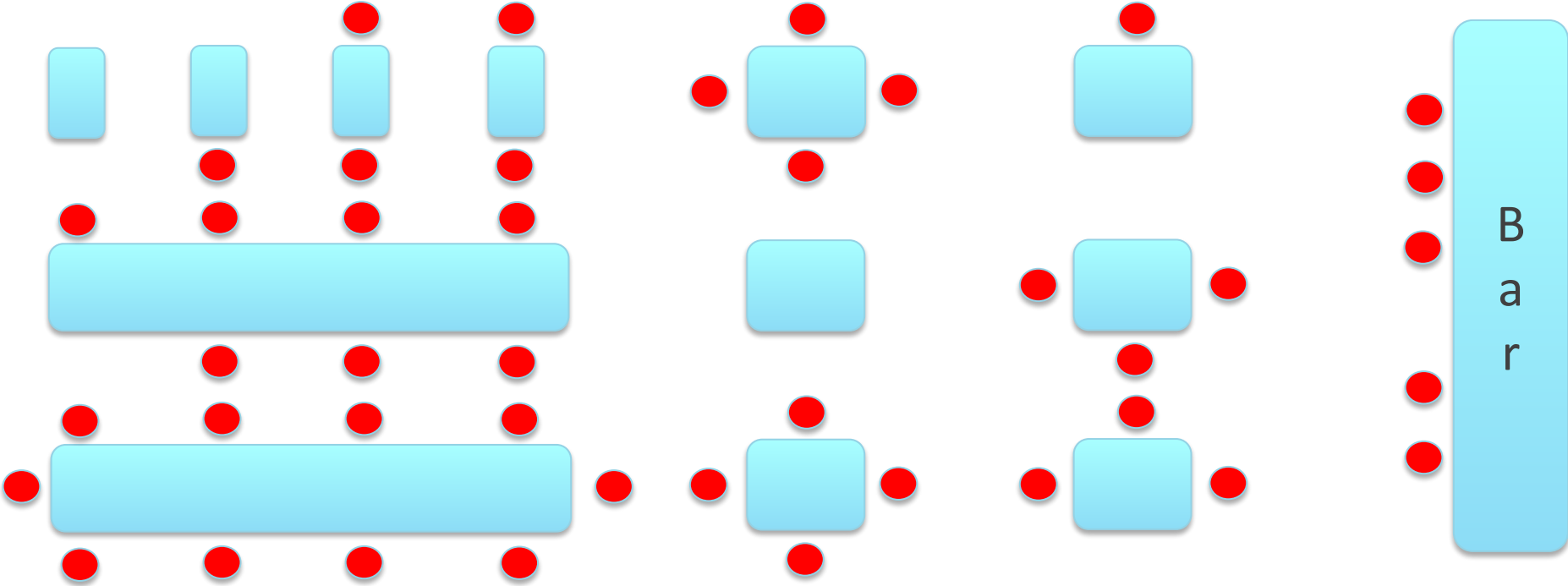
# Lunchtime

- A table seating more than one is no problem right now...



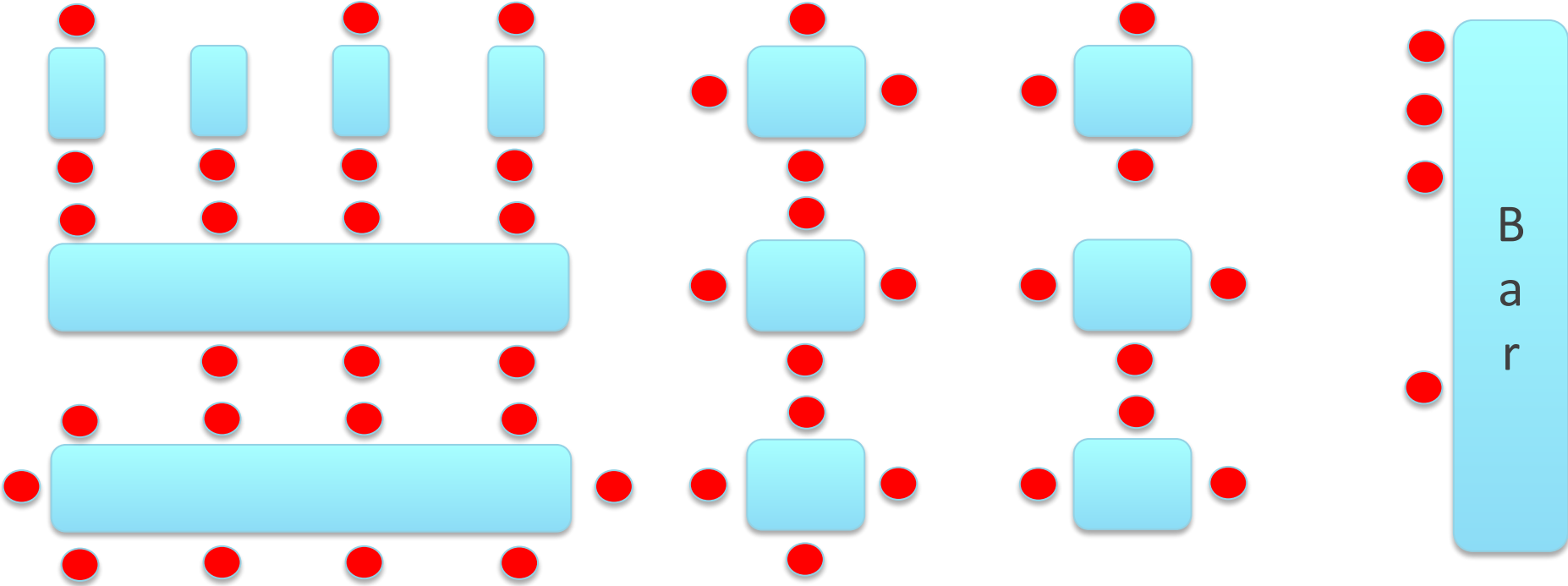


# Lunchtime



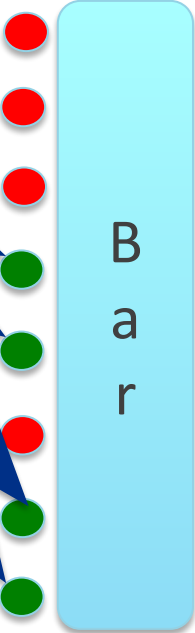
# Lunchtime

- But if you want a table now, you are going to wait



# Lunchtime

But there are plenty of seats at the bar!



# When you ask for more than you need

- Suppose the grid is very busy and for a moment there is only one 8-core 16 GB glidein with some free space. Currently it's running jobs with the following requests:

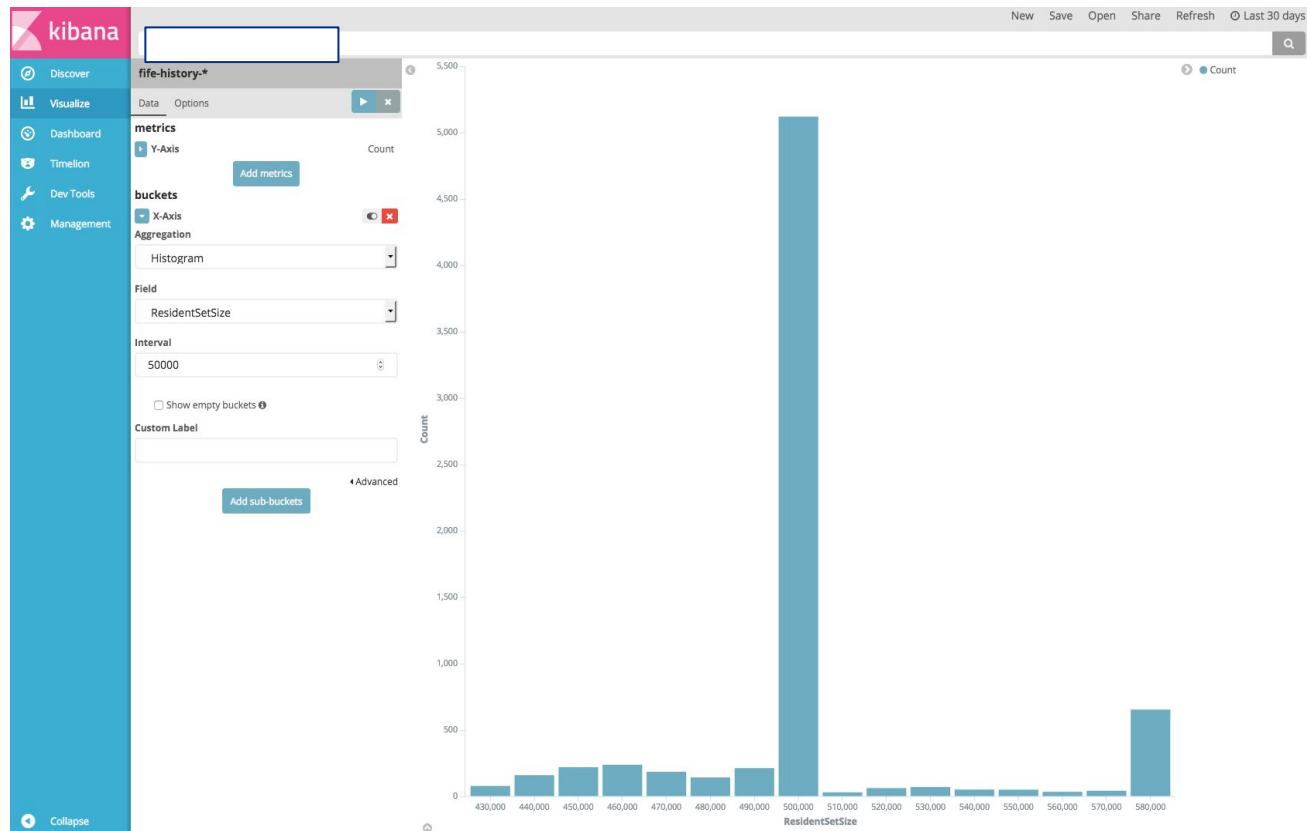
Slots	CPUs	Memory (MB)
Job 1	1	2000
Job 2	1	2000
Job 3	1	1900
Job 4	1	1900
Job 5	1	2500
Job 6	2	4000
Tot	7	14300
Free	1	1700

A job requesting 1 CPU  
and  $\leq 1700$  MB memory  
can run here; all others have to wait...  
We see glideins looking like this  
*all the time* on GPGrid

# A missed opportunity

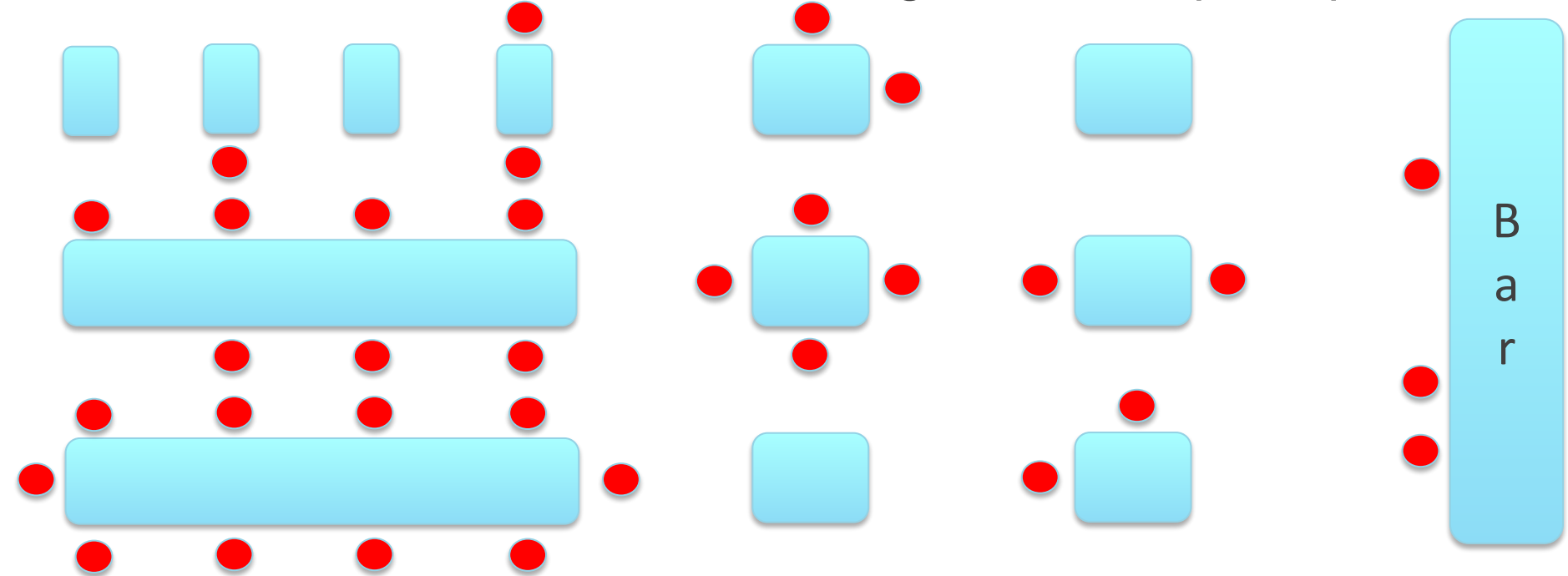
These jobs never used more than **600MB** of memory, but were requesting **2000MB**.

At the time there were **220** free slots with memory between 600 MB and 2000 MB that could have been used



# Another scenario

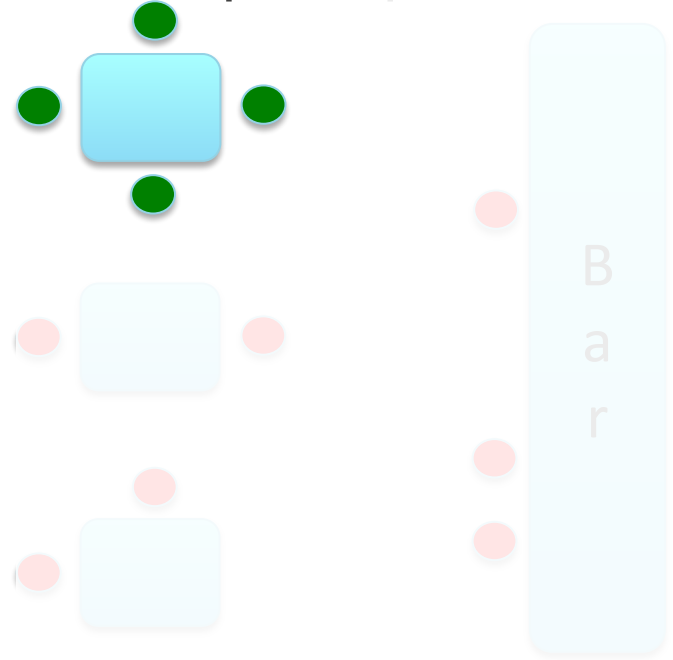
- Let's say we go as a party of 8 (they don't take reservations)
- Here, we will have to wait for the big tables to open up



## Another scenario

- Let's say we go as a party of 8 (they don't take reservations)
- Here, we will have to wait for the big tables to open up

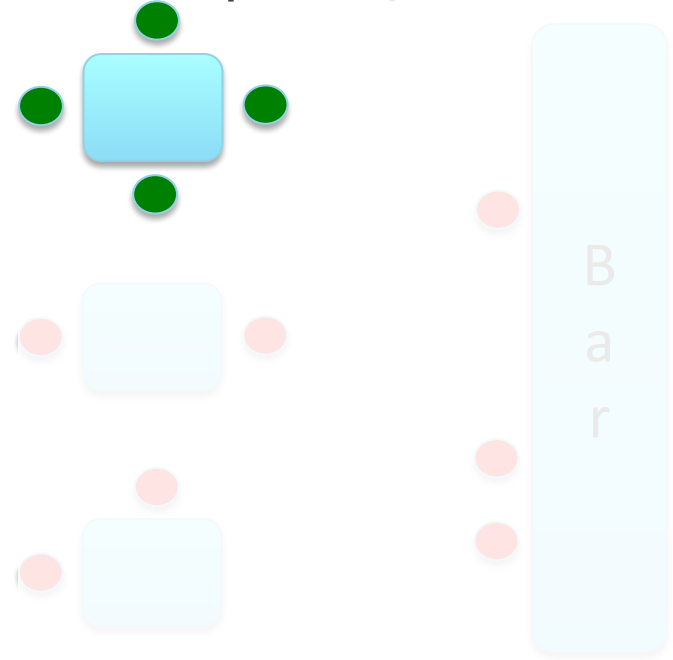
But if we split into  
two parties of 4,  
we can start immediately.



## Another scenario

- Let's say we go as a party of 8 (they don't take reservations)
- Here, we will have to wait for the big tables to open up

But if we split into two parties of 4, we can start immediately. Consider a larger number of smaller jobs if you're not getting resources.





# But requesting too little isn't good either

- As you may know, your job will be automatically **held** if it goes above its memory, local disk, or run time request. Held = job stopped and will restart from the beginning when manually released
- With partitionable slots this is very important
- At some sites if the overall glidein memory usage goes over the max (could be caused by only one job), the *entire* glidein is killed immediately (all jobs within it are lost.) **This can affect other users.**
- Memory/disk usage checks runs every 2 minutes

# Setting up your resource request

- Jobsub has --memory, --disk, --cpu, --expected-lifetime opts
- --memory and disk will take units of KB, MB, GB, TB (default MB for memory and KB for disk)
  - Important note: **Right now 1 GB = 1024 MB, not 1000 MB**
- --cpu takes an integer
- --expected-lifetime can take a number with units of h,m, or s (default s) or a “short”, “medium”, or “long” preset (currently 3, 8, 24 hours)
- A well-formed request might look like:
- --cpu=1 --memory=1800MB --disk=15GB --expected-lifetime=6h
- You might not be using jobsub directly, so consult the documentation of whatever you're using for how to pass resource requests to jobusb\_submit
- **Default resource requests** (you get these when you don't specify the corresponding option):
  - **2,000 MB memory; 1 cpu; 35,000,000KB disk; 8 hours max run time**

# A Common Denominator in problems

What can bring about all of these problems?

Inability to use offsite resources

Slow run times and slow file transfer rates

Jobs failures due to missing mount points

Stuck in idle waiting for jobs to copy

**Using BlueArc areas in your job can do all of these things.**

# Why BlueArc in Grid jobs is Bad

- 1) Scripts that directly depend on BlueArc paths (/grid/fermiapp/foo, /dune/app/foo, etc.) or home areas **will not work offsite (and soon not on GPGrid)**. That alone deprives you of significant resources!
- 2) Scripts that copy things to/from BlueArc (e.g. ifdh cp /dune/app/users/me/myfiles ...) must all go through the BestMan gridftp server...

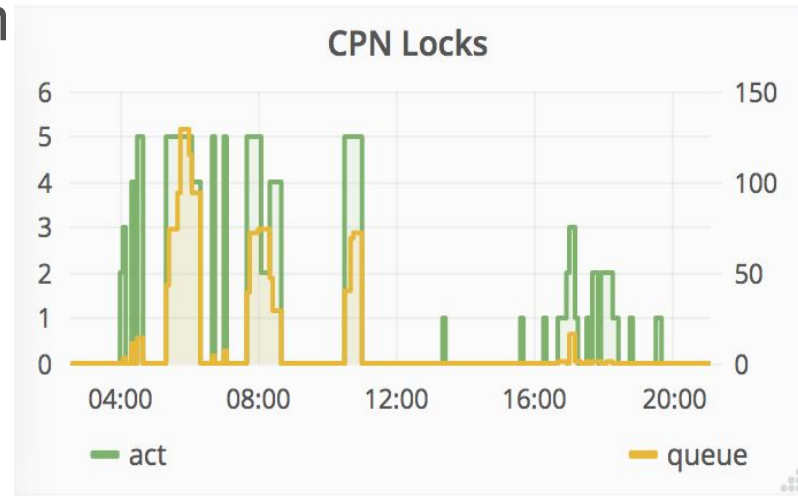
# The BestMan server and CPN locks

This server is the way you get to/from BlueArc in Grid jobs

It is a single machine with a **1 Gb/s** (gigabit, not gigabyte) network connection. Shared by **ALL** experiments across ALL grid jobs.

Only 5 simultaneous transfers are allowed per EXPERIMENT. Anything over that goes into the CPN lock queue

**Off-hours support not guaranteed**



# When there's a lock queue...

Your ifdh cp command will sleep and try again later. This can take a long time if the queue is large. Meanwhile your job is running at 0% efficiency, effectively blocking a slot.

LOCK - Wed Apr 26 21:08:06 UTC 2017 LOCKS/LIMIT/QUEUE 5/5/581 sleeping 581

LOCK - Wed Apr 26 21:08:06 UTC 2017 queue

20170426.21:08:06.fnpc7025.3155713.REDACTED.REDACTED

LOCK - Wed Apr 26 21:17:47 UTC 2017 LOCKS/LIMIT/QUEUE 5/5/1461 sleeping 1461

LOCK - Wed Apr 26 21:17:47 UTC 2017 queue

20170426.21:08:06.fnpc7025.3155713.REDACTED.REDACTED

LOCK - Wed Apr 26 21:42:08 UTC 2017 LOCKS/LIMIT/QUEUE 5/5/1434 sleeping 1434

LOCK - Wed Apr 26 21:42:08 UTC 2017 queue

20170426.21:08:06.fnpc7025.3155713.REDACTED.REDACTED

LOCK - Wed Apr 26 22:06:02 UTC 2017 LOCKS/LIMIT/QUEUE 5/5/1434 sleeping 1434

...

LOCK - Thu Apr 27 01:52:45 UTC 2017 freed

/grid/data/lariat/LOCK/LOCKS/20170427.01:52:44.17078.fnpc7025.3155713.REDACTED.REDACTED

This job took almost **four hours** just to copy some initial input files!!!

# The BestMan server and CPN locks

This server is the way you get to/from BlueArc in Grid jobs

It is

ne

Onl

Any

Q: Is the server going to be upgraded?

Q: Is the transfer limit going to be raised?

# The BestMan server and CPN locks

This server is the way you get to/from BlueArc in Grid jobs

It is

ne

Onl

Any

Q: Is the server going to be upgraded?

NO

Q: Is the transfer limit going to be raised?

NO



# Let me sum up...

Consider **BlueArc use in grid jobs** (even an ifdh cp to bring files in) **deprecated**.

# Let me sum up...

Consider **BlueArc use in grid jobs** (even an ifdh cp to bring files in) **deprecated**.

If you don't use it now, **DON'T START**.

## Let me sum up...

Consider **BlueArc use in grid jobs** (even an ifdh cp to bring files in) **deprecated**.

If you don't use it now, **DON'T START**.

If you do use it now, **STOP. STOP NOW.**

IT IS GOING AWAY ANYWAY LATER THIS YEAR.

## Let me sum up...

Consider **BlueArc use in grid jobs** (even an ifdh cp to bring files in) **deprecated**.

If you don't use it now, **DON'T START**.

If you do use it now, **STOP. STOP NOW.**

IT IS GOING AWAY ANYWAY LATER THIS YEAR.

Seriously, **STOP DOING IT**. It is the **biggest single obstacle** to improving your grid jobs' throughput.

# Let me sum up...

Consider **BlueArc use in grid jobs** (even an ifdh cp to bring files in) **deprecated**.

If you don't use it now, **DON'T START**.

If you do use it now, **STOP. STOP NOW.**

IT IS GOING AWAY ANYWAY LATER THIS YEAR.

Seriously, **STOP DOING IT**. It is the **biggest single obstacle** to improving your grid jobs' throughput.

It is OK to do e.g. *jobsub\_submit <options> file:///dune/app/foo*

But */dune/app/foo* should not use BlueArc inside of it

# Cutting the BlueArc cord

How do you get rid of BlueArc? Consider this simple job script on the worker node:

```
#!/bin/bash
# setup SW
. /grid/fermiapp/products/dune/setup_dune.sh
setup some_packages
lfdh cp -D /pnfs/dune/scratch/users/${GRID_USER}/my_input_file ./
/dune/app/users/${GRID_USER}/my_custom_code/mycode -i my_input_file -o my_output_file
lfdh cp -D my_output_file /pnfs/dune/scratch/users/${GRID_USER}/some_dir/
```

## Bad News:

- 1) Dependence on /grid/fermiapp and /dune/app means no running outside of GPGrid

# Cutting the BlueArc cord

How do you get rid of BlueArc? Consider this simple job script:

```
#!/bin/bash
# setup SW
./cvmfs/dune.opensciencegrid.org/products/dune/setup_dune.sh
setup some_packages
lfdh cp -D /pnfs/dune/scratch/users/${GRID_USER}/my_input_file ./
ifdh cp -D /dune/app/users/${GRID_USER}/my_custom_code.tar.gz ./
tar zmf my_custom_code.tar.gz
./my_custom_code/mycode -i my_input_file -o my_output_file
lfdh cp -D my_output_file /pnfs/dune/scratch/users/${GRID_USER}/some_dir/
```

## Bad News:

2) ifdh cp from /dune/app requires a CPN lock and goes through the BestMan server.

# Cutting the BlueArc cord

How do you get rid of BlueArc? Consider this simple job script:

```
#!/bin/bash
# setup SW
./cvmfs/dune.opensciencegrid.org/products/dune/setup_dune.sh
setup some_packages
lfdh cp -D /pnfs/dune/scratch/users/${GRID_USER}/my_input_file ./
lfdh cp -D /pnfs/dune/scratch/users/${GRID_USER}/my_custom_code.tar.gz ./
tar zmf my_custom_code.tar.gz
./my_custom_code/mycode -i my_input_file -o my_output_file
lfdh cp -D my_output_file /pnfs/dune/scratch/users/${GRID_USER}/some_dir/
```

## Good News:

All file I/O is via dCache (no locks)  
Main experiment software is from CVMFS; we can run anywhere!

## BUT:

**Still lots of bad behavior here (not checking exit codes, not checking dependencies are present, etc.)**



# Options for custom code in jobs

If you have custom code/libraries/scripts you'll need to get that to the worker node (do NOT read it directly, except for CVMFS and StashCache paths)

## Recommended practices (any combination of these is OK):

- 1) Within the job script use `ifdh cp` and `tar` as needed. **All source files should be in dCache**; nothing should be on BlueArc (see previous example)
- 2) As part of your submission, use the `-f` option with `jobsub_submit`. Files will be in a directory stored in the `CONDOR_DIR_INPUT` environment variable. You can move it around locally, `untar`, etc as you like. **All files used with `-f` should be in dCache**. Example on the next slide.
- 3) If StashCache is an option for you, use that and then you can read the files directly (do not need to `ifdh cp` or `-f` them)

## Bad practice:

There is a `jobsub` option called `--tar_file_name`. This will copy the tarball once at submission time, transfer and extract it for you within the job. *However everything goes through fifebatch server and can grind everything (including other users!) to a halt if the file is too big.* This has happened!!!

Consider this option deprecated and **do not use it**. Will be removed from future releases

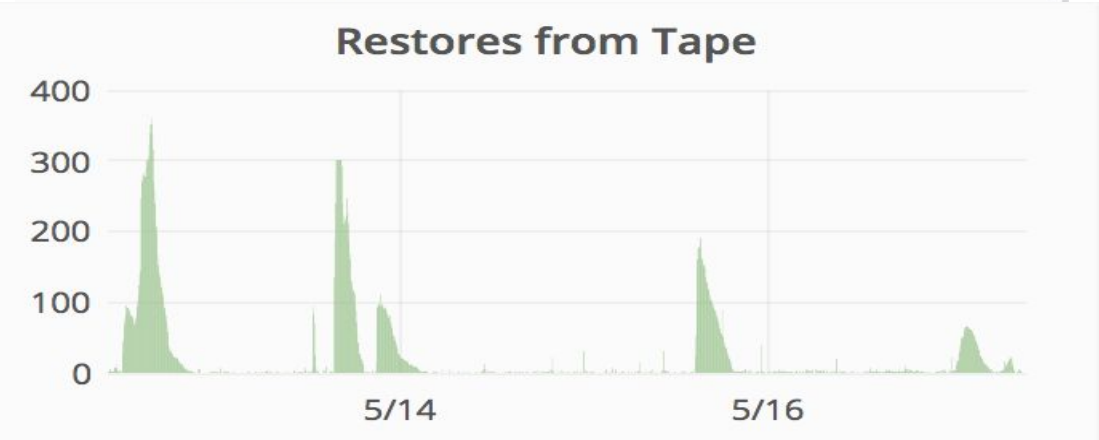
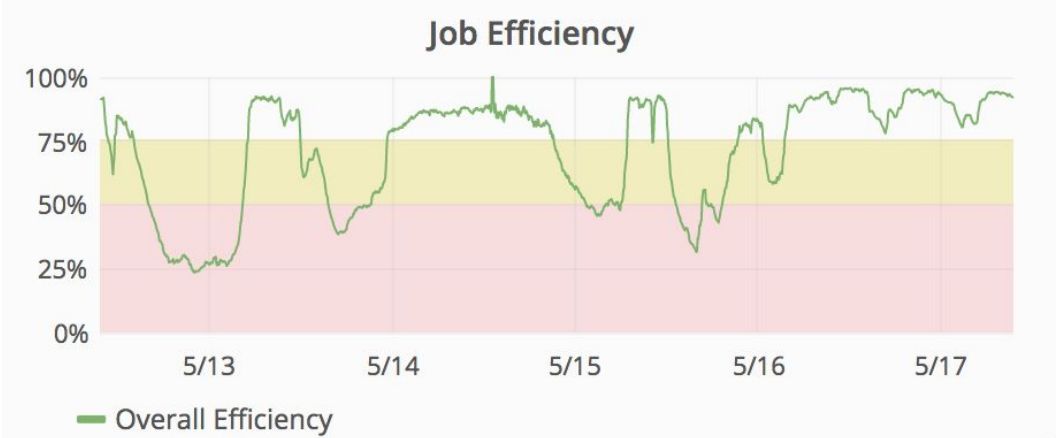
# Other Best Practices

- **No BlueArc dependence.** Get code from CVMFS and/or tarballs copied in from dCache areas via ifdh. **Do not copy to or from BlueArc, period.**
- Avoid hard-coding file paths, especially home areas (CVMFS is OK though)
- Do not count on a certain username when running the job. Use the **GRID\_USER** environment variable (this is set to the “real you” at FNAL, or maybe exptpro when running production jobs.)
- Check for the existence of necessary files before using them
- ***Check exit codes of all major steps in your scripts;*** print useful messages on failure

## Other Best Practices (2)

- Submit some test job(s) before any large submissions that will use a new script or after changing something
  - **Test interactively (a few events/single input file) first**
  - Then send a few test batch jobs. If those work, proceed with the full project. Use `gpgtest` or `fermicloud168` for these tests.
- If you have specific requirements the worker node must meet, put them in the job requirements via the **--append\_condor\_requirements** jobsub option
  - These might include a specific CVMFS/StashCache version, veto of a specific site, running on a specific subcluster, etc.
  - Jobs will not start on worker nodes that don't meet your request
- **Prestage your input files. See data handling best practices.**

# When you don't prestage...



# Adding additional condor requirements

--append\_condor\_requirements takes HTCondor requirement expressions; evaluated before job starts. **Can have more than one.**

Examples (note the quotes escaped with \)

1) Require the worker to have NOvA StashCache version >1234:

```
--append_condor_requirements='(TARGET.HAS_CVMFS_NOVA_OSGSTORAGE_ORG=="true"&&TARGET.CVMFS_NOVA_OSGSTORAGE_ORG_REVISION>1234)'
```

2) Omaha has the Tusker and Crane subclusters. Allow only to run at Crane:

```
--append_condor_requirements='(TARGET.GLIDEIN_ResourceName=="Crane")'
```

# Passing options to your script

You may have arguments to your script (anything after file://foo.sh is treated as an argument)

Sometimes these could be an init script with a specific path

Example: you might be used to doing something like  
**jobsub\_subimt <opts> file://myscript.sh \  
--/experiment/app/users/me/script.sh**

Logical to copy script.sh in fromm dCache with -f and do  
**--source=\$CONDOR\_DIR\_INPUT/script.sh**

**But jobsub will try to expand \$CONDOR\_DIR\_INPUT at submit time, and \$CONDOR\_DIR\_INPUT isn't known until the job starts!** How can we solve this problem?

## Passing options to your script (2)

You can escape the environment variable with a backslash:

```
--source=\$CONDOR_DIR_INPUT/script.sh
```

Then `$CONDOR_DIR_INPUT` won't be evaluated until you're in the job.

In this way you can change from hard-coding BlueArc paths in scripts like this

# Maxima and Large submissions

- What are the effective resource limits, if any?
  - Memory: 16000MB on GPGrid; varies on OSG sites
  - CPU: 8 on GPGrid; typically that or less on OSG
  - Run time: 4 days on GPGrid; varies on OSG sites
- How many jobs can I submit at once?
  - Now: max in a single submission is **10K** (but need to be careful with these.)
  - When doing multiple submission, do not exceed an average rate of **1000 jobs/minute**.  
**If you submit 10K, wait at least 10 min before submitting more.**
- How many jobs can I (a user) have queued?
  - No hard limit, but more than you could run in one week using your experiment's GPGrid quota is a good rule of thumb.
  - Example: Say the quota is 1000. You have jobs that you expect to take 12h to run. So  $1000 * (168 \text{ hours/week}) / (12 \text{ h run time request}) = 14\text{K}$  Max jobs queued.
  - Don't forget to account for slot weight! E.g. for 4 GB memory jobs, again divide by 2.



# Getting Help

- **Rule #1: Never be ashamed to ask for help!**
- Different problems generally have different experts
- For issues with grid job submission or strange errors within jobs (please fetch the log files and try to see if it's a software or config issue first), contact the USDC group (also known as Distributed Computing Support) *via the Service Desk*. We can also answer other more general questions if you're not sure about a category.
- Issues with data movement (ifdh cp stuck, trouble with SAM, dCache) should either be Service Desk tickets with Scientific Data Management or Scientific Data Storage and Access

# Helpful Documentation

Additional Reading and useful places to look:

<https://fermipoint.fnal.gov/project/FIFE/SitePages/Home.aspx>

<https://cdcvs.fnal.gov/redmine/projects/fife/wiki/Wiki>

[https://cdcvs.fnal.gov/redmine/projects/fife/wiki/Introduction\\_to\\_FIFE\\_and\\_Component\\_Services](https://cdcvs.fnal.gov/redmine/projects/fife/wiki/Introduction_to_FIFE_and_Component_Services)

[https://cdcvs.fnal.gov/redmine/projects/fife/wiki/Advanced\\_Computing](https://cdcvs.fnal.gov/redmine/projects/fife/wiki/Advanced_Computing)

<https://cdcvs.fnal.gov/redmine/projects/jobsub/wiki#Client-User-Guide>

<https://cdcvs.fnal.gov/redmine/projects/ifdhc/wiki>

<https://fifemon.fnal.gov/monitor/dashboard/db/experiment-batch-details?from=now-24h&to=now&var-experiment=dune>

[https://cdcvs.fnal.gov/redmine/projects/dune/wiki/Computing\\_How-To\\_Documentation](https://cdcvs.fnal.gov/redmine/projects/dune/wiki/Computing_How-To_Documentation)

# Summary

- We've discussed some common issues and how to avoid them
- Given some hints for best practices in jobs and data mgmt.
- **DON'T USE BLUEARC IN ANY WAY IN GRID JOBS.**
- Calculate the total resource for your project and consult the S&C coordinators on large submissions
- Following these best practices will help everyone get their work done better, **especially you!**
- We've gone over how to get help.
  - Please talk to us if anything comes up, or if you have needs/requests that weren't discussed here.

# BACKUP

# Opening a Distributed Computing Support Ticket

Service Desk page:

<https://fermi.service-now.com/>

Log in with your services (FNAL email) username & password

Start Here

The screenshot shows the Fermilab Service Management interface. The left sidebar contains a menu with the following items: Self-Service, Homepage, Self Service, Service Request Catalog, Core Computing Services, Scientific Computing Services (highlighted with a green box), Scientific Services Outage Calendar, Knowledge, Connect Chat, My Current Requested Items, My Watched Requested Items, My Past Requested Items, My Current Incidents, My Watched Incidents, and My Past Incidents. A blue box with the text 'Start Here' and a white arrow points to the 'Scientific Computing Services' menu item. The main content area shows a dropdown menu for 'My ITIL Homepage' and a list of 'My Group's Current' items, including 'CMS-Tier1-LPC' and 'Distributed Con'. The Fermilab logo is visible in the bottom right corner.

# Opening a Distributed Computing Support Ticket

Filter navigator

★

Self-Service

Homepage

Self Service

Service Request Catalog

Core Computing Services

Scientific Computing Services

Scientific Services Outage Calendar

Knowledge

Connect Chat

My Current Requested Items

My Watched Requested Items

My Past Requested Items

My Current Incidents

My Watched Incidents

My Past Incidents

## Scientific Computing Services at Fermilab

News

No items

Search in  Core Computing  Scientific Computing  All

Select Visibility Type:

[Search tips](#)

Have ideas or suggestions on how we can improve this functionality? Submit a [Feedback Request](#).

### Service Areas

[All detail](#)

DAQ and Engineering

artdaq

DAQ and Engineering Consulting

Electronic Module Support

Distributed Computing

Batch Job Management (jobsub condorsubmit)

Community On-Boarding to Use Distributed Computing

Distributed Resource Accounting (Gratia)

User Jobs Monitoring (fifemon)

High Performance Computing

USQCD Facility Application Support

USQCD Facility File-System Support

Select Distributed Computing

# Opening a Distributed Computing Support Ticket

A word about ticket categories:





General questions: **Ask a Question**

New features, enhancement ideas: **Requests**

Something is not working normally: **Incident**

*NOTE: Things that you want to add are not Incidents.*

## Get Help

-  [Ask a question about this service](#) **Question**
-  [Report a service outage or incident](#) **Incident**
-  [Submit a request to service providers](#) **Request**
-  [Give us feedback about this service](#)

## More Information/Need Help?

Service Desk

Monday - Friday, 8-5

<https://fermi.service-now.com>

or 630.840.2345 or walk-ups (WHGF)

## ITIL Stats

# Other Tickets (SDM, SDSA)

Fermilab Service Management KH Kenneth Herner

Filter navigator

- Self-Service
  - Homepage
  - Self Service
  - Service Request Catalog
  - Core Computing Services
  - Scientific Computing Services
  - Scientific Services Outage Calendar
  - Knowledge
  - Connect Chat
  - My Current Requested Items
  - My Watched Requested Items
  - My Past Requested Items
  - My Current Incidents
  - My Watched Incidents

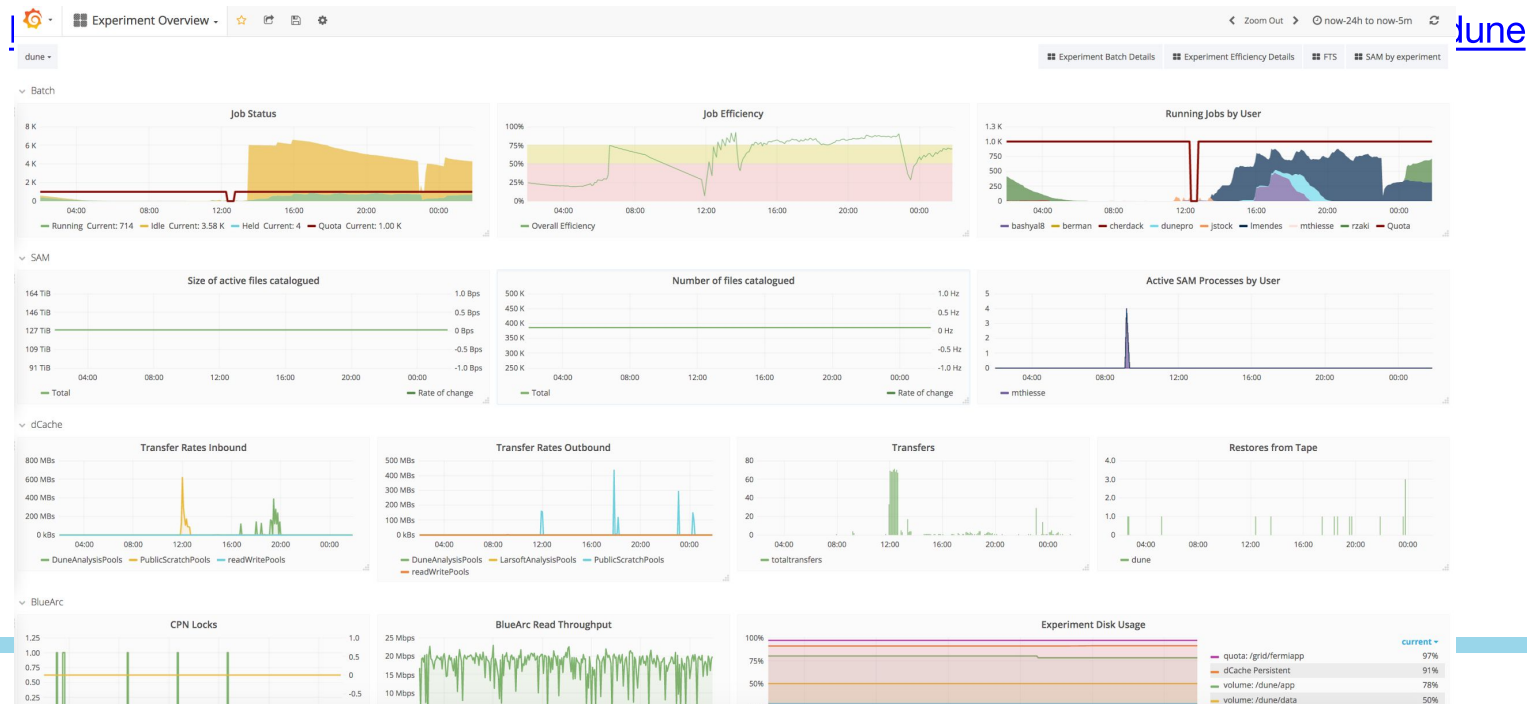
Physics and Detector Simulation	geant4	Genie	Pythia	Synergia								
Scientific Collaboration Tools	DES Members	DES Publication Board	DES Speakers Bureau	Electronic Logbook	Projects	Shift Scheduler	Speakers Bureau	Git	CVS	Redmine	SVN	Electronic Logbook during Data Taking
Scientific Data Management	FTS (File Transfer Service)	IFDHC (Intensity Frontier Data Handling Client)	SAM4Users	UConDB	SAM (Sequential Access via Metadata)							
Scientific Data Storage and Access	Active Archive Facility	dCache Disk Cache Storage	Enstore Tape Storage									
Scientific Database Applications	Conditions Database	DES Alarms Viewer	DES Constants Database	DES Exposures Viewer	DES Telemetry Viewer	Hardware Database	IFBeam Conditions Database (IFBeamDB)	IFBeam Conditions Database (IFBeamDB) (Enhanced)	Query Engine			
Scientific Linux Systems Engineering	Managed Scientific Workstation	Scientific Linux Distribution	Scientific Linux Engineering	Online System Engineering and Lifecycle Management - Enhanced	Control Room System Management - Enhanced	DAQ Infrastructure Operations Engineering - Enhanced	Scientific Test Stand Engineering - Enhanced					



# Monitoring your submissions

FIFEMON has a wealth of information

For example, the DUNE overview:

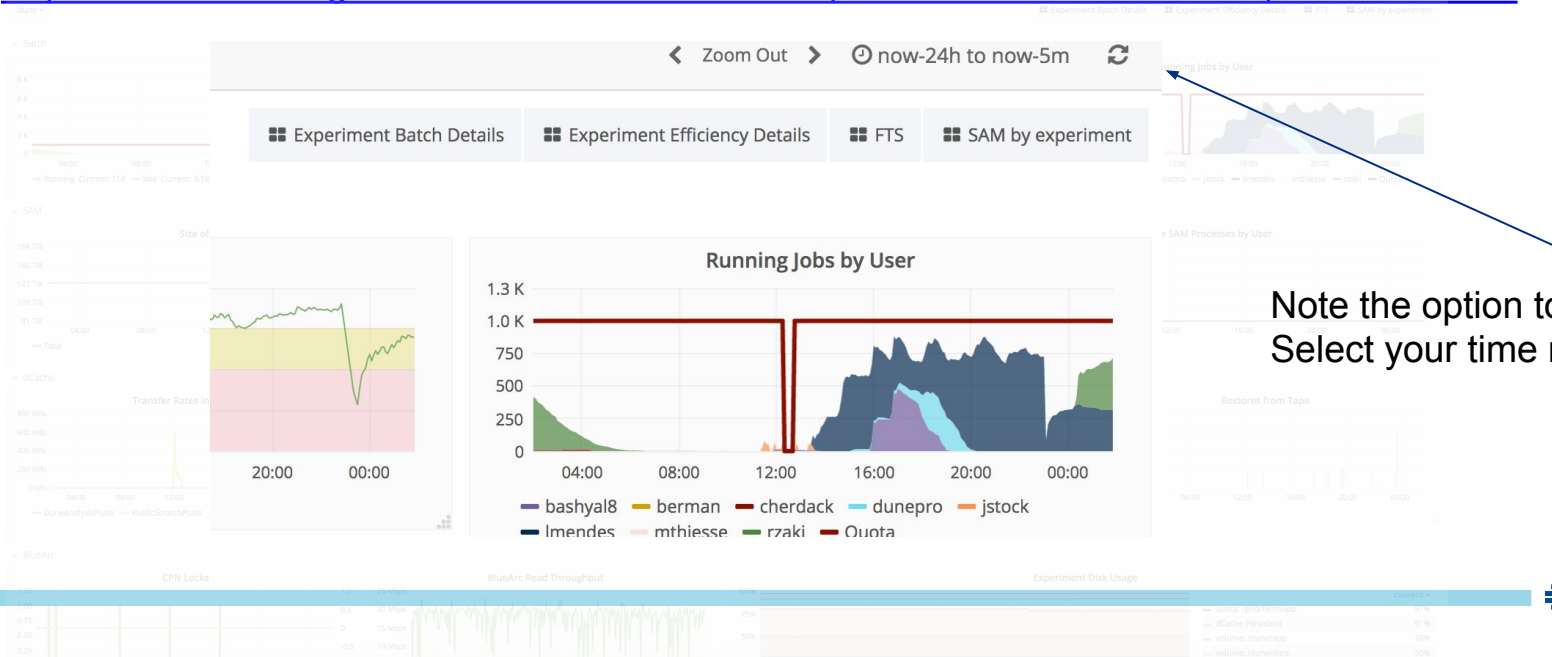


# Monitoring your submissions

FIFEMON has a wealth of information

For example, the DUNE overview:

<https://fifemon.fnal.gov/monitor/dashboard/db/experiment-overview?var-experiment=dune>

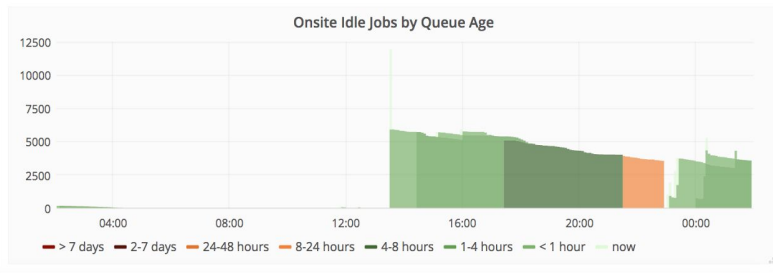
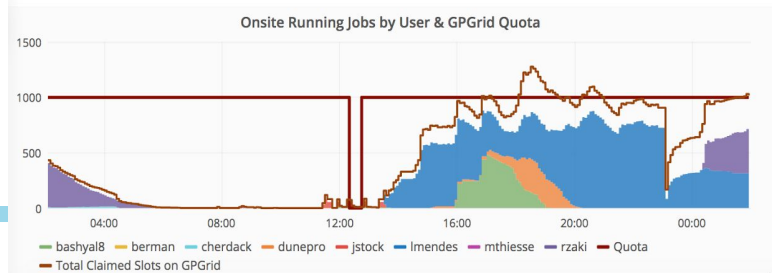
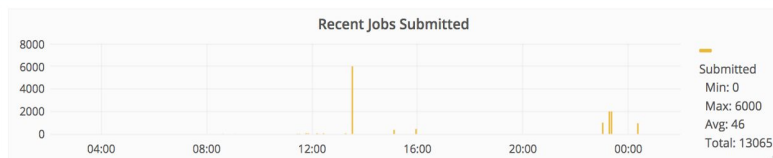


# Monitoring your submissions

See details of DUNE batch jobs:

User jobs

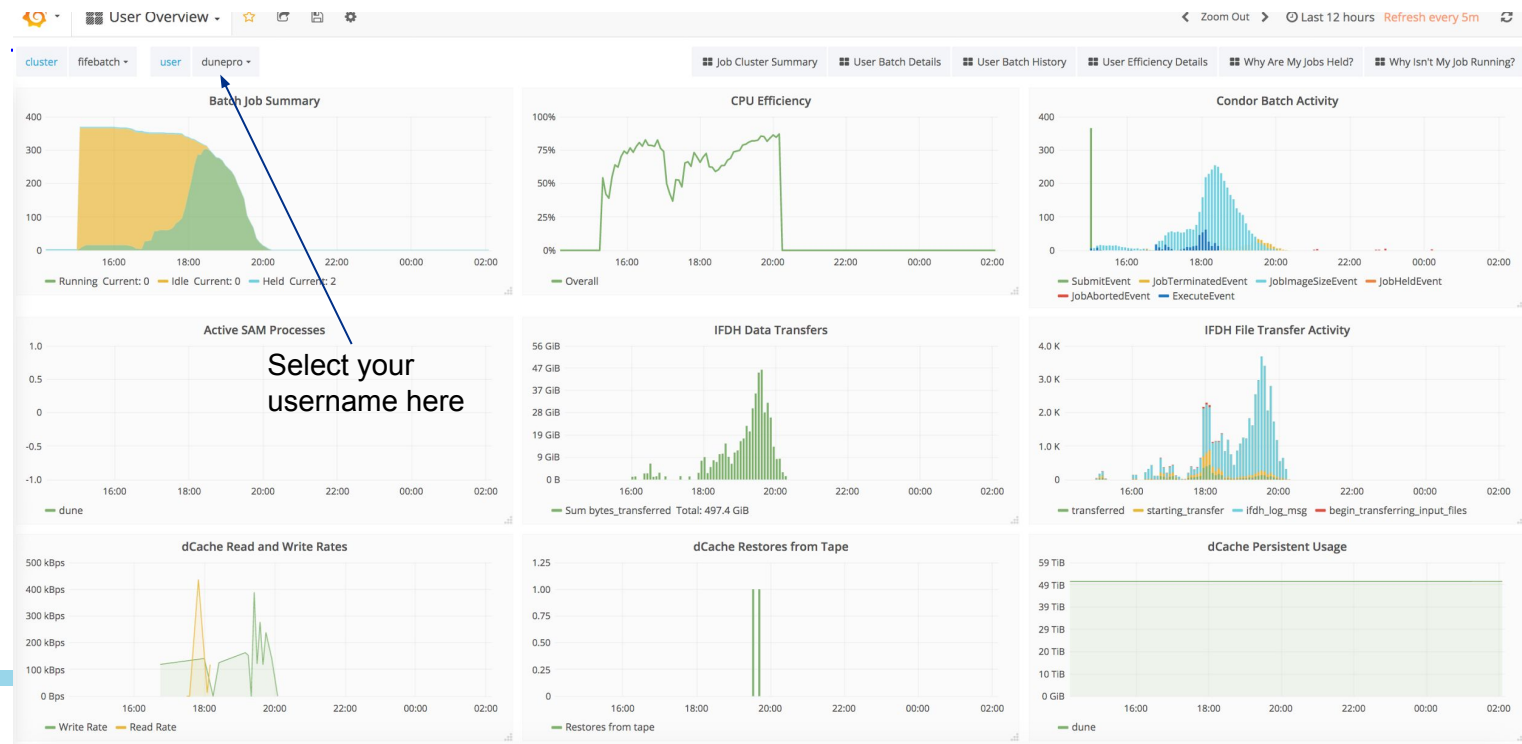
User	Idle	Run	Held	Max Mem Req. (MiB)	Max Disk Req. (GiB)	Max Time Req. (Hr)
<a href="#">dunepro</a>	0	0	20	4000.0	33.4	8.0
<a href="#">kherner</a>	8	0	0	1000.0	33.4	3.0
<a href="#">lmendes</a>	3049	316	0	4000.0	33.4	10.0
<a href="#">radovic</a>	0	0	1	2500.0	33.4	8.0
<a href="#">rzaki</a>	493	386	1	2000.0	33.4	12.0
<a href="#">trj</a>	10	0	0	1700.0	4.0	1.0



# Monitoring your submissions

There are also pages with details for each user

<https://fifemon.fnal.gov/monitor/dashboard/db/user-overview>



# Monitoring your submissions

There are also pages with details for each user

<https://fifemon.fnal.gov/monitor/dashboard/db/user-overview>

