

Run3 @CERN with HTCondor

Ben Jones



Photo by courtesy of Romande Energie

LHC Status @lhcstatus2 · Jul 17

Automated

Comments (18:57:49) Problem with IT.L8
leak in the insulation vacuum
No beam until further notice (weeks)

23 24 11.1K

PROTON PHYSICS: STABLE BEAMS

S 12	CM ITR1	CS ITR1	CM MSR1	CS MSR1	CM AR12	CS AR12	CM MSL2	CS MSL2	CM ITL2	CS ITL2
S 23	CM ITR2	CS ITR2	CM MSR2	CS MSR2	CM AML3	CS AML3				
S 34					CM AMR3	CS AMR3	CM MSL4	CS MSL4		
S 45			CM MSR4	CS MSR4	CM AR45	CS AR45	CM MSL5	CS MSL5	CM ITL5	CS ITL5
S 56	CM ITR5	CS ITR5	CM MSR5	CS MSR5	CM AR56	CS AR56	CM MSL6	CS MSL6		
S 67			CM MSR6	CS MSR6	CM AML7	CS AML7				
S 78					CM AMR7	CS AMR7	CM MSL8	CS MSL8	CM ITL8	CS ITL8
S 81	CM ITR8	CS ITR8	CM MSR8	CS MSR8	CM AR81	CS AR81	CM MSL1	CS MSL1	CM ITL1	CS ITL1
60 A	S12	S23	S34	S45	S56	S67	S78	S81		
RF :	CM 1L4	CS 1L4	CM 2L4	CS 2L4	CM 1R4	CS 1R4	CM 2R4	CS 2R4		
EXP :	CR ATLAS TOR	CR ATLAS SOL	CR CMS							

Average Arc Temperatures

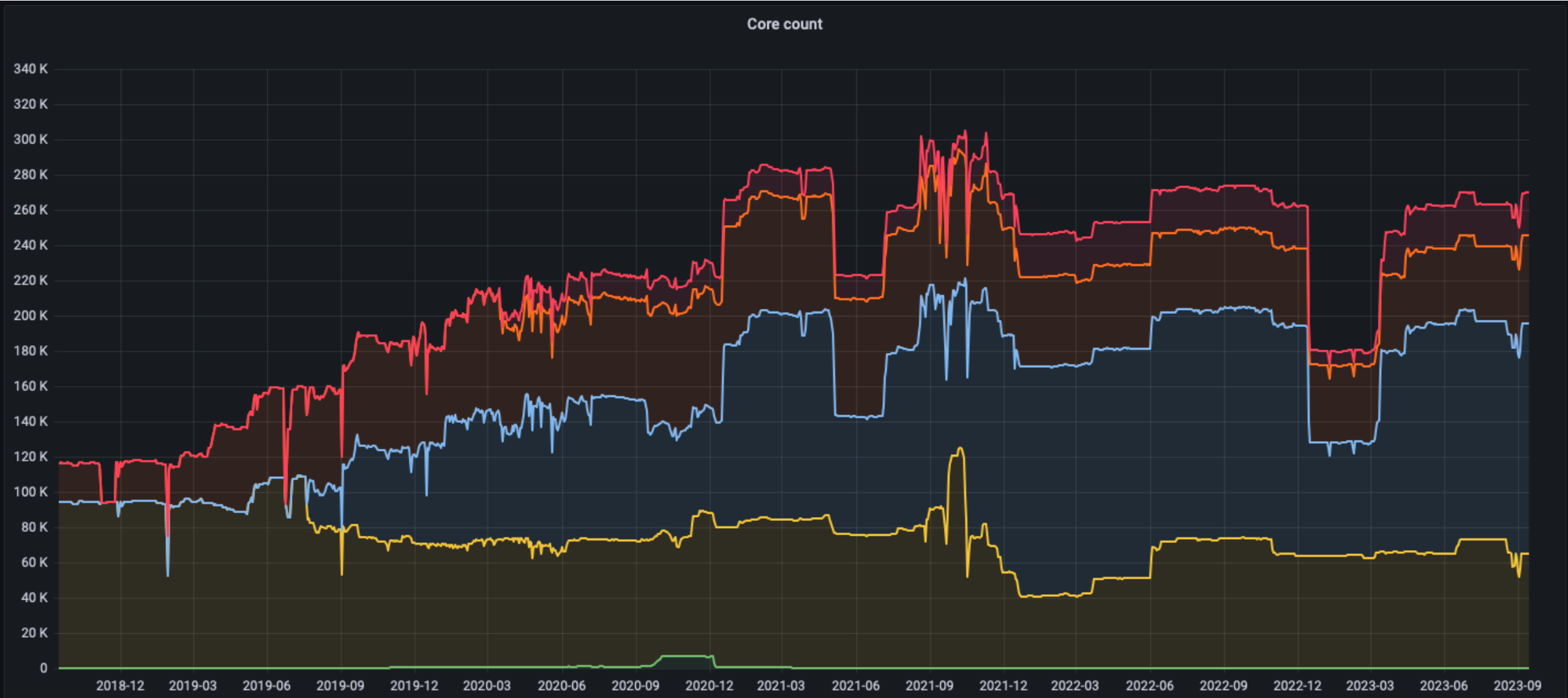
Average Arc Temperatures
Updated: 00:53:13

	T max / K
S12	2.0
S23	2.0
S34	2.0
S45	3.1
S56	2.0
S67	2.0
S78	2.0
S81	3.6

LHC Status @lhcstatus2 · Jul 17

udp://multicast-bev.lhc2:1234

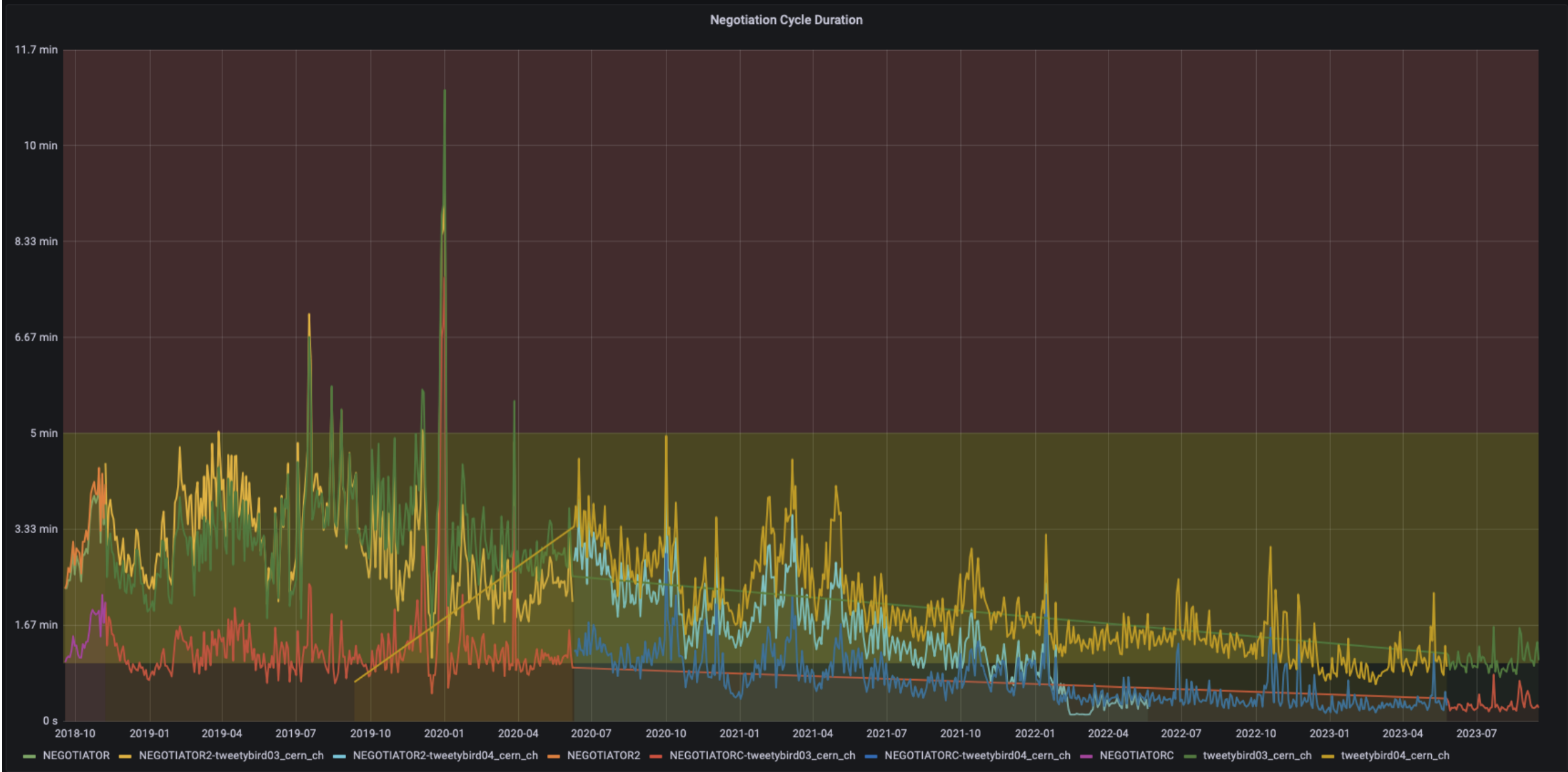
Last 5 years of HTCondor @CERN



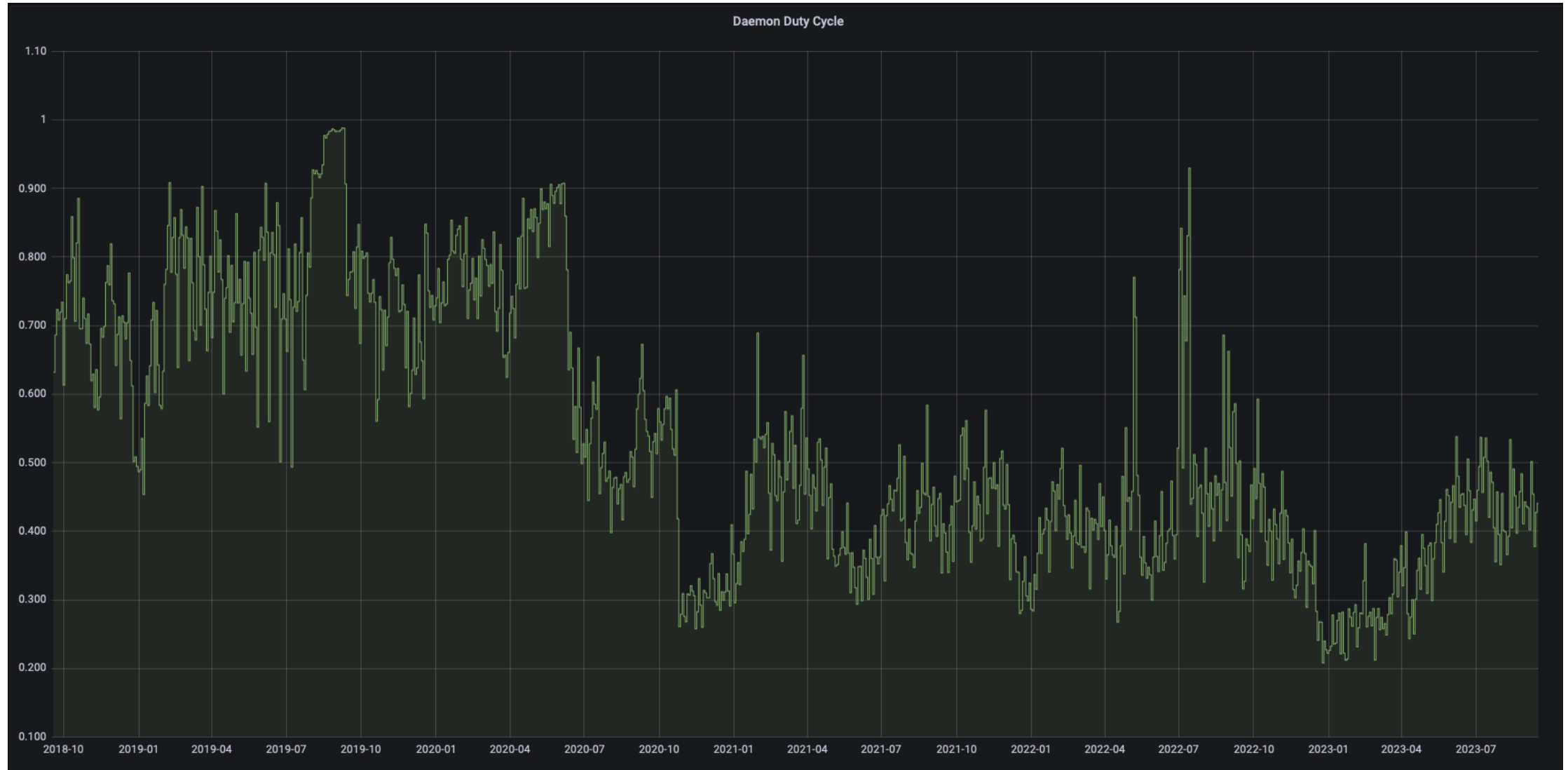
Last 5 years of HTCCondor @CERN



Negotiator happier than ever



Top level collector same story



Continue to be able to scale with LHC requirements

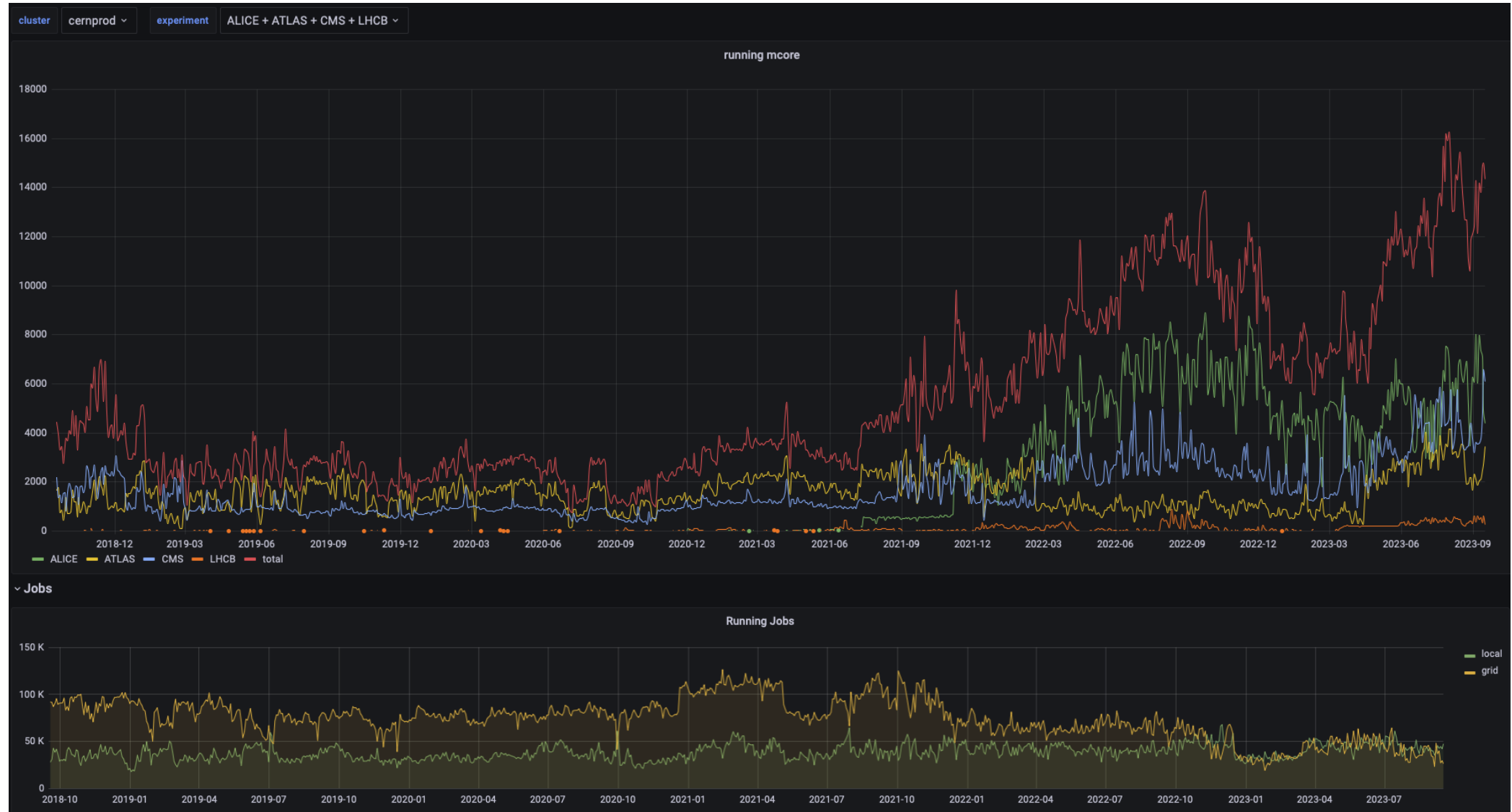
- **Most credit has to go to HTCondor devs**
 - (also CMS global pool!)
- **There are other reasons scaling has been easier too**
 - CERN batch farm has moved from VMs to "Ironic" bare metal. Keeps cloud APIs but with normal bare metal
 - From 8/10/12 core VMS (dependent on NUMA arch) to full nodes
 - Reduced nodes by 75%
 - Everything but defrag gets a bit easier
- **Most Grid pilots are now 8 core**
 - ALICE now also predominantly mcore



IRONIC

an OpenStack Community Project

mcore up, running jobs stable



What is a job at CERN?

- **Physicist with a submit file, sending a job via an AP (all our submission is remote...)**
- **An experiment's production jobs, sent by “submission framework” via an AP**
- **ATLAS sending jobs (real jobs) via Grid to CEs**
- **ALICE / LHCb sending non-condor pilots via Grid to CEs**
- **CMS sending glideins via Grid to CEs**
- **A physicist with a metascheduler sending workers via an AP**

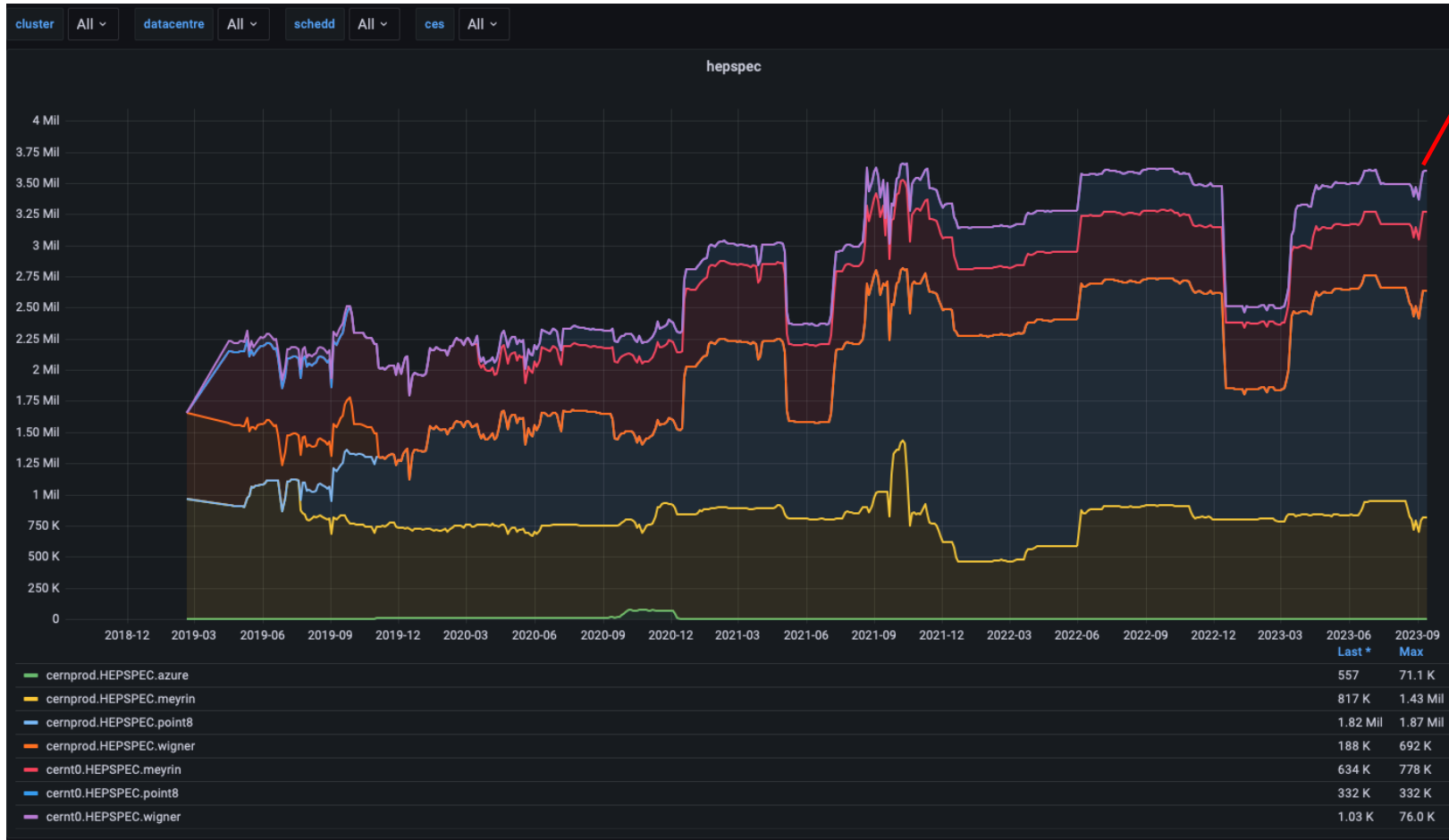
Next scale point: Preveessin Data Centre

Three phases on three floors:

- 4MW – 2nd floor
- 8MW – 1st and 2nd floor
- 12MW ground floor + 1st + 2nd floor

Air cooled racks with hot-aisle containment

Adding 2MHS06



+ 2MHS06
1160 nodes
AMD 7543 (EPYC v3)
[more powerful than extant models so # cpus fewer in proportion]

Managing OS heterogeneity

- **Even before the Red Hat / CentOS announcements, managing OS during Run was going to be a challenge**
 - cc7 still primary platform, EoL June 2024
- **Given timelines skipping 8 and using 9**
- **Most LHC experiments & grid workflow uses containers, and pilots support el9 now**
 - (some fixes needed, especially for multiple 9 variants in use)
- **As ever the long tail of "local" user submitted jobs is 80% of the effort**



355_RHEL_0723

Red Hat strikes a crushing blow against RHEL downstreams

From now on, only CentOS Stream's source code is available to all

[Liam Proven](#)

Fri 23 Jun 2023 // 16:30 UTC

Conditional containerisation of user jobs

```
JOB_TRANSFORM_AutoWrapSingularity @=end
```

```
NAME AutoWrapSingularity
```

```
REQUIREMENTS (JobUniverse =?= 5 && isUndefined(SingularityImage) && !isUndefined(WantOS))
```

```
EVALSET MappedImage userMap("JobImages", toLower(My.WantOS))
```

```
EVALSET MappedOS userMap("JobOS", toLower(My.WantOS))
```

```
SET SingularityImage (MappedOS =?= Undefined) ? Error : ((TARGET.OpSysAndVer != MappedOS) ?  
((MappedImage != Undefined) ? MappedImage : Error) : Undefined)
```

```
@end
```

```
# Submit Requirement: ValidSingularityJob
```

```
SUBMIT_REQUIREMENT_ValidSingularityJob = isUndefined(toLower(WantOS)) || (!  
isUndefined(toLower(WantOS)) && ! isError(SingularityImage) )
```

```
SUBMIT_REQUIREMENT_ValidSingularityJob_REASON = "The requested OS is not valid. The currently  
available operating systems are: e17, e18, e19"
```

Conditional containerisation of user jobs

JobImages :

- * e17 /cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/batch-team/containers/plusbatch/e17-full:latest
- * e18 /cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/batch-team/containers/plusbatch/e18-full:latest
- e19 /cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/batch-team/containers/plusbatch/e19-full:latest

JobOS :

- * e17 CentOS7
- * e18 CentOS8
- * e19 AlmaLinux9

Container creation

- **“Contract” that the login nodes (LxPlus) have to look the same as the batch workers**
- **Containers must maintain pkg list / config of changing puppet managed LxPlus**
- **DUCC to dump container images to /cvmfs/unpacked.cern.ch/**
 - <https://cvmfs.readthedocs.io/en/stable/cpt-ducc.html>
 - <https://gitlab.cern.ch/unpacked/sync>
- **Kapitan & some puppetdb hackery to create and maintain images via Gitlab-CI**
 - <https://kapitan.dev>
 - <https://gitlab.cern.ch/batch-team/containers/plusbatch>

CVMFS_REPOSITORIES='alice-nightlies.cern.ch,alice-ocdb.cern.ch,alice.cern.ch,ams.cern.ch,atlas-condb.cern.ch,atlas-nightlies.cern.ch,atlas-online-nightlies.cern.ch,atlas-pixel-daq.cern.ch,atlas.cern.ch,cms-bril.cern.ch,cms-ci.cern.ch,cms-ib.cern.ch,cms-opendata-conddb.cern.ch,cms.cern.ch,compass-condb.cern.ch,compass-mc.cern.ch,compass.cern.ch,geant4.cern.ch,ilc.desy.de,lhcb-condb.cern.ch,lhcb.cern.ch,lhcbdev.cern.ch,na61.cern.ch,na62.cern.ch,sft-nightlies.cern.ch,sft.cern.ch

- **The above is just a list of the CVMFS mount points that we monitor**
- **Yes, CVMFS, can break, and yet it's essential to Experiment workflow**
 - Containers currently making things worse (zombie mounts etc, though el9 should improve things)
- **We used to very simply monitor to see if there were any broken (monitored) mounts**
 - Condor cron -> NODE_MOUNTS_CVMFS={true,false}
- **The more mounts we have this isn't feasible**
 - ilc.desy.de being down can't prevent an EP taking any jobs
- **Collectd probe: <https://github.com/cvmfs/collectd-cvmfs>**
 - Condor cron -> CVMFS_HEALTHY_REPOS=[list of repos]

CVMFS requirements for jobs

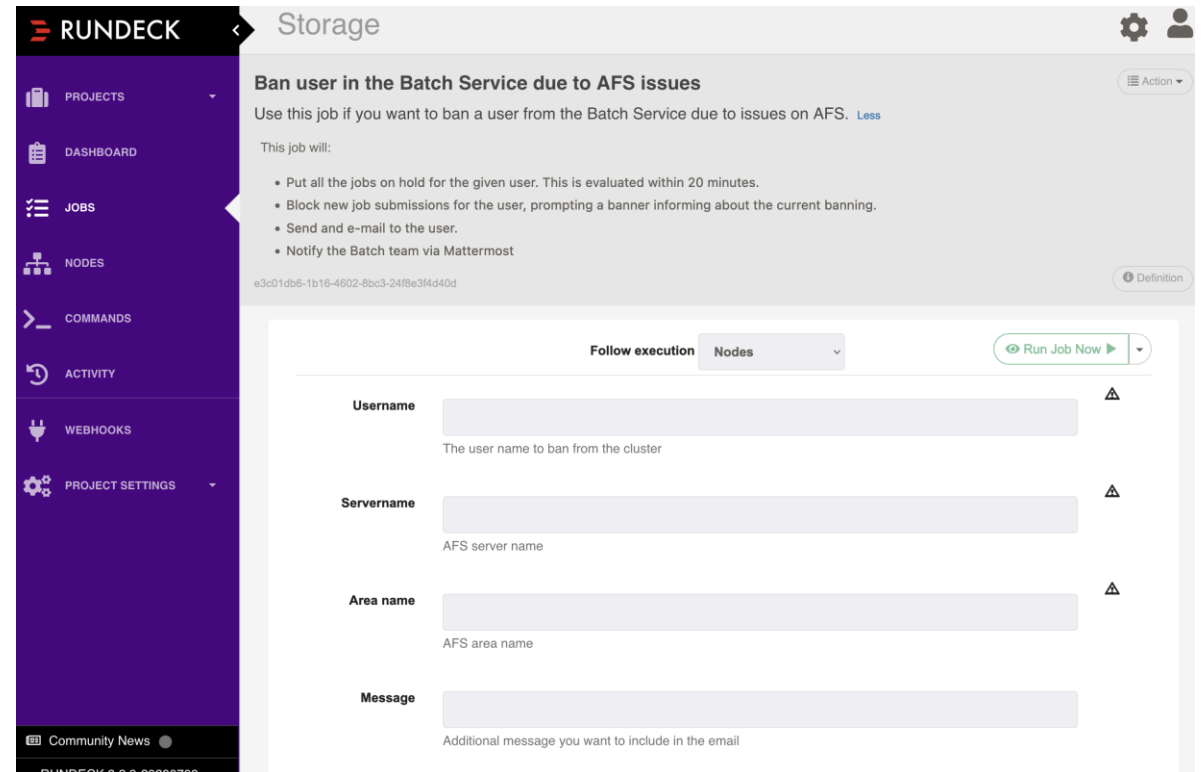
```
JOB_TRANSFORM_LhcbCvmfs @=end
  NAME LhcbCvmfs
  REQUIREMENTS (MY.WantBeer != True) && (x509UserProxyVoName =?= "lhcb")
  COPY Requirements PreCvmfsRequirements
  SET Requirements (NODE_MOUNTS_CVMFS || (stringListMember("lhcb-
condb.cern.ch", CVMFS_HEALTHY_REPOS) && stringListMember("lhcbdev.cern.ch",
CVMFS_HEALTHY_REPOS) && stringListMember("lhcb.cern.ch", CVMFS_HEALTHY_REPOS) &&
stringListMember("sft.cern.ch", CVMFS_HEALTHY_REPOS) &&
stringListMember("grid.cern.ch", CVMFS_HEALTHY_REPOS))) && PreCvmfsRequirements
@end
```

Non-x86

- **GPUs & ARM both becoming more important**
- **GPUs**
 - Still (for our batch farm) fairly niche – though lots of usage for experiments online farms
 - Main use cases:
 - Interactive (shell & notebook)
 - HTC style batch (long simulation + shorter ie cmsbuild)
 - GitLab runners
 - Kubeflow / ML
 - Hard to know how best to manage finite resource
 - Shared k8s pool? If so how best to manage flow from HTCondor
- **ARM**
 - Will have ~1.5k cores by end of year. Real world power etc numbers

Preventing AFS overloads

- **Perennial problem of small subset of users creating AFS overload**
 - Yes, shared filesystems are bad (but convenient)
- **Manual process to ban bad users**
 - Too slow to react
 - AFS admins sad
 - Users with \$HOME on same server sad
- **Automation to “fix” the problem**
 - Still reactive and yes we still need to wean people off shared filesystems

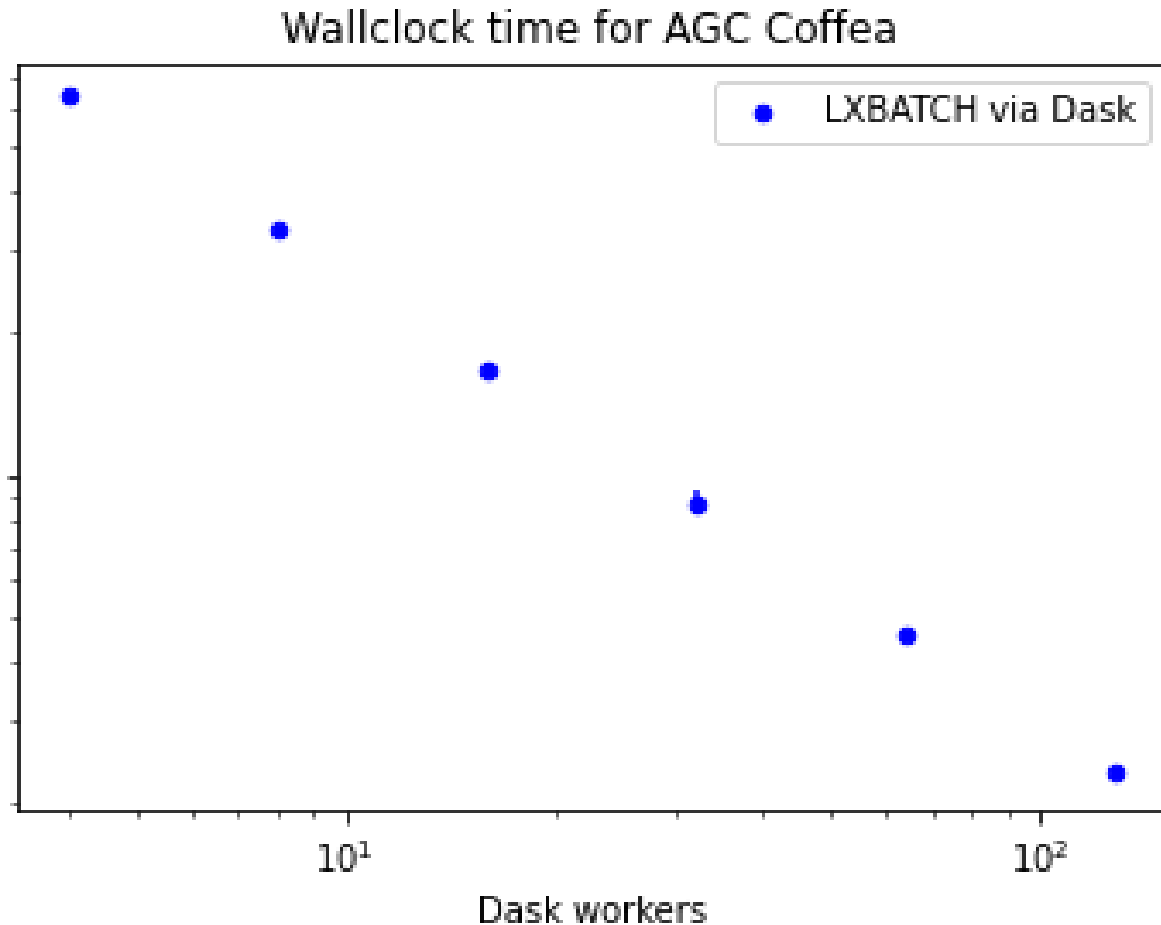


The screenshot displays the RunDeck web interface. On the left is a dark purple sidebar with navigation options: PROJECTS, DASHBOARD, JOBS, NODES, COMMANDS, ACTIVITY, WEBHOOKS, and PROJECT SETTINGS. The main content area is titled 'Storage' and shows a job definition for 'Ban user in the Batch Service due to AFS issues'. The job description includes: 'Use this job if you want to ban a user from the Batch Service due to issues on AFS. Less'. Below this, it states 'This job will:' followed by three bullet points: 'Put all the jobs on hold for the given user. This is evaluated within 20 minutes.', 'Block new job submissions for the user, prompting a banner informing about the current banning.', and 'Send and e-mail to the user.' A fourth bullet point reads 'Notify the Batch team via Mattermost'. The job ID 'e3c01db6-1b16-4602-8bc3-24f8e3f4d40d' is visible. At the bottom of the job definition, there is a 'Follow execution' section with a 'Nodes' dropdown and a 'Run Job Now' button. Below this are four input fields: 'Username' (with a warning icon), 'Servername' (with a warning icon), 'Area name' (with a warning icon), and 'Message'. Each field has a descriptive label below it: 'The user name to ban from the cluster', 'AFS server name', 'AFS area name', and 'Additional message you want to include in the email'.

Stackstorm & “hot volume” API

- **AFS load is known on the file server side, so they publish stats with volume “badness” score**
- **3 levels of “badness” with 3 actions our automation engine (Stackstorm) takes**
 - lvl1 (slow): Set ceiling to 75% of running jobs for user (`condor_userprio $user –setceil N`)
 - lvl2 (bad): Actively reduce running jobs to under ceiling (using `htcondor.JobAction.Vacate`)
 - lvl3 (horrid): Set ceiling to 0, vacate all jobs
- **Per user ceiling controls very handy**
 - Would be useful to have a few more controls – specifically a per user throttle on job starts might be handy
 - The pattern where N jobs start in short period and all request same file on network is tricky for any service
 - Per-schedd limit is an indirect method to prevent this issue

Analysis Facilities



- Everyone agrees that we need AFs even as there's less agreement at what they mean exactly
- We have some of the tools at CERN and continue to improve them
- SWAN (notebook service) now with Jupyterhub extensions -> coffea / RDataFrame -> Dask -> HTCondor
- Successfully run RDataFrame & Coffea examples for Analysis Grand Challenge

SWAN / dask-labextension

CLUSTERS ↻ + NEW

SwanHTCondorCluster 1
 Scheduler Address: <https://10.100.28.153:32004>
 Dashboard URL: <http://10.100.28.153:8787/status>
 Number of Cores: 16
 Memory: 48.00 GiB
 Number of Workers: 16

<> SCALE SHUTDOWN

The screenshot displays the Dask dashboard with several key components:

- Terminal 1:** Shows the execution of `./run.sh` and the output of `analysisManager`, listing various histograms created for different tasks like `ttbar_nominal_4j2b`, `single_top_s_chan_nominal_4j2b`, etc.
- Task Stream:** A horizontal bar chart showing the progress of tasks over time, with a vertical red line indicating the current time.
- Workers:** A table listing 16 worker nodes with their status, address, and event loop interval.

name	address	event_loop_interval
Total (16)		0.3417577954133351
SwanHTCondorCluster-0-0	https://188.184.163.18:10000	0.01998620989384767
SwanHTCondorCluster-0-1	https://188.184.163.18:10001	0.019967718124389647
SwanHTCondorCluster-0-2	https://188.184.163.18:10002	0.01998022965663842
SwanHTCondorCluster-0-3	https://188.184.163.18:10003	0.019982285499572754
SwanHTCondorCluster-1-0	https://188.184.163.13:10000	0.019990239143371582
SwanHTCondorCluster-1-1	https://188.184.163.13:10001	0.0416725627454122
SwanHTCondorCluster-1-2	https://188.184.163.13:10002	0.019991936683654784
- Graph:** A task graph showing the dependencies between tasks, with nodes colored by their state: released (blue), memory (red), processing (green), waiting (grey), and queued (yellow).

AF future work

- **SWAN / Dask / HTCondor integration works well and is interactive enough when user/experiment has enough quota**
 - Stating the obvious, but user / analysis quota is uneven for LHC experiments at CERN
 - Most obstacles are not technical (quotas / priorities / preemption)
 - Some thought on how to provide interactive buffer (temp overcommit slots? dedicated buffer?)
- **How to transition between interactive / dev to batch use case?**
 - Login to SWAN, use notebook on small dataset, when happy and ready to submit to batch...?
 - Currently: interactive services like SWAN timeout.
 - Close laptop, go away, ideally should be able to wait for jobs to complete, reconnect to SWAN and get results – but where should Dask scheduler run?
- **We still have a user `kinit` in a web console**
 - Need to do IDTOKEN via SSO authenticated service (htgettoken type mechanism)



home.cern

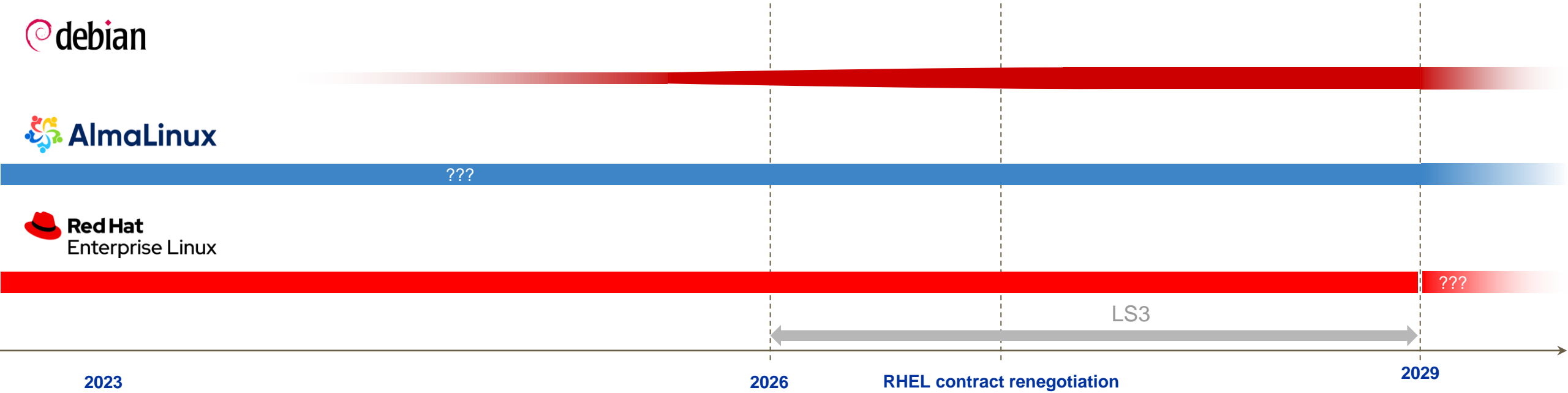
Docker autowrap only for subset of grid jobs

```
JOB_ROUTER_ROUTE_Beer @=end
  NAME Beer
  MaxJobs = 2000
  MaxIdleJobs = 1000
  TargetUniverse = 5
  SET Requirements (BeerMachine =?= True && HasDocker =?= True)
  SET BeerJob true
  SET WantDocker true
  SET CoreSize 0
  COPY Cmd orig_Cmd
  EVALSET Cmd ifThenElse(regex("^\/", orig_Cmd), orig_Cmd, strcat("./",orig_Cmd))
  COPY DockerImage orig_DockerImage
  EVALSET DockerImage ifThenElse(isUndefined(orig_DockerImage), "gitlab-registry.cern.ch/batch-
team/cc7-batchwn", orig_DockerImage)
  REQUIREMENTS (MY.WantBeer =?= True) || (MY.queue =?= "WantBeer") || (MY.queue =?= "beer")
@end
```

Potential Future Scenario: Dual Ecosystem

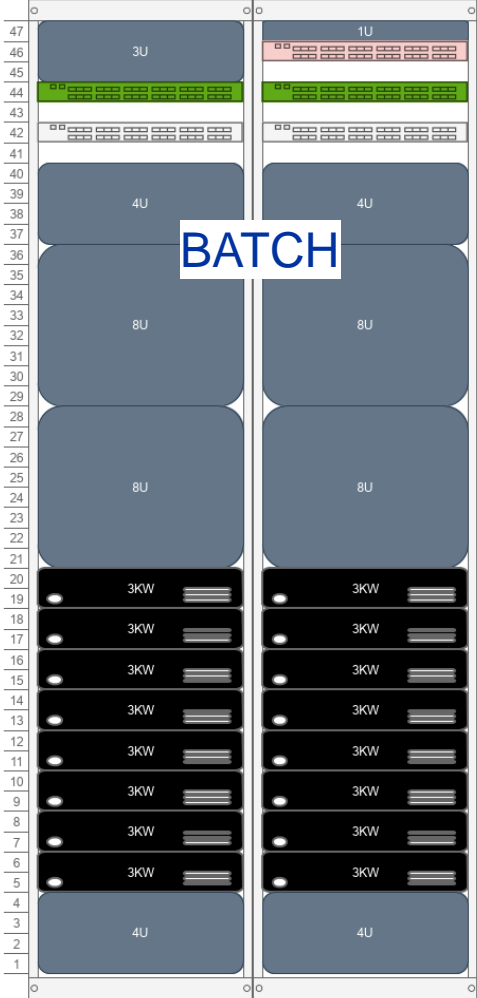
- Ramp up Debian support to match Red Hat / AlmaLinux
 - Credible exit strategy for multiple risk scenarios
- Long-term project, major effort from multiple parties
 - Requires management approval and commitment

End of current RHEL contract

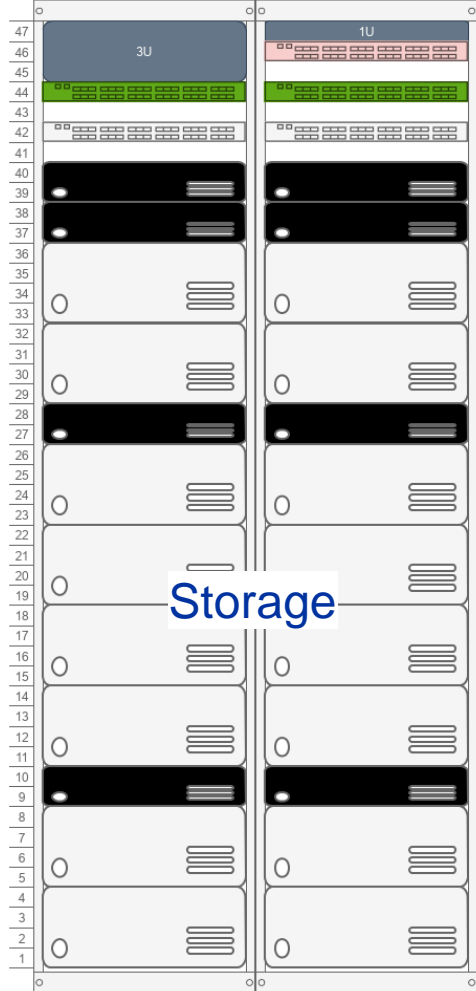


Sample future rack layouts

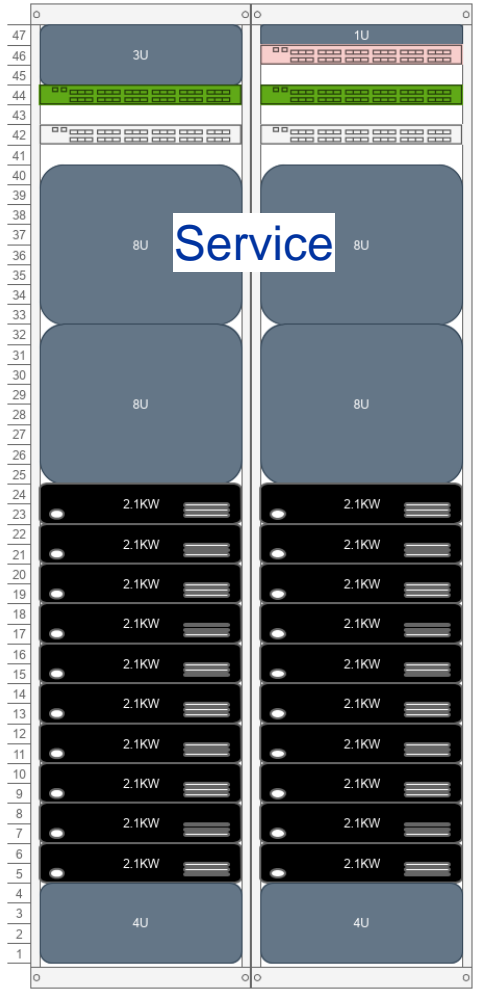
~25 kW - No UPS



~10 kW - UPS



~20 kW - UPS



Fan-wall delivery



