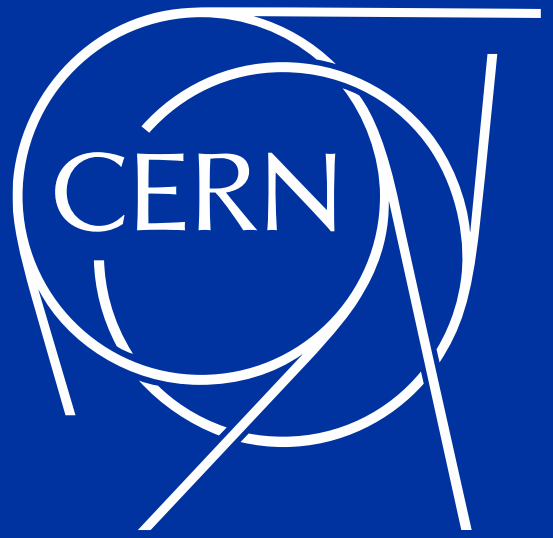
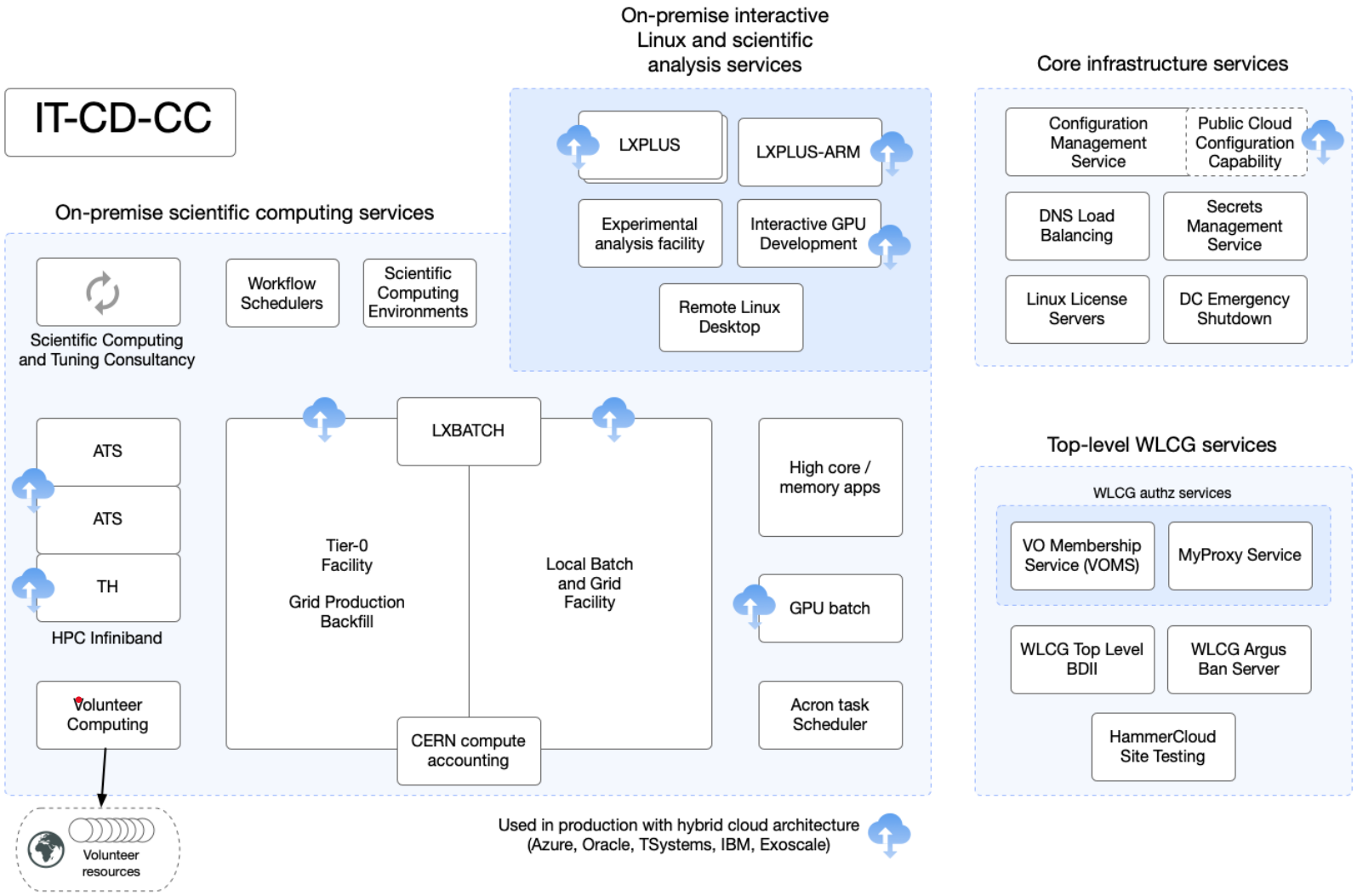

CERN

LxPlus / LxBatch & Analysis

Ben Jones IT-CD

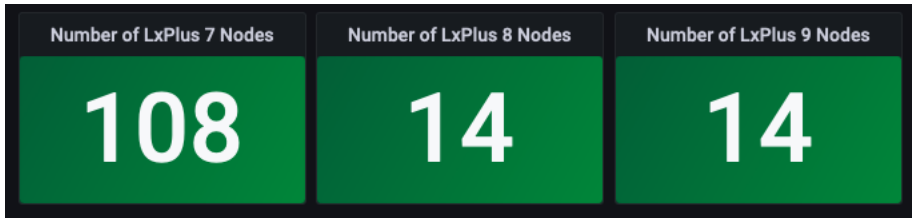
Compute and Config



LxPlus [LinuX Public Login User Service]

- **Interactive linux login service for CERN**
- **What is it used for?**
  - Everything and Anything: general purpose computing facility
  - Batch (remote) submit node
  - Users whose primary desktop/laptop is mac or windows do their physics on lxplus
    - Examine subsets of data, prepare jobs, development, LaTeX
  - Remote desktop for graphical apps (vnc, fastx etc)
  - ansible control of their service, Jenkins CI, etc etc
  - Tunneling, email (mutt, alpine)
- **Reference build: people often ask for ”their own” lxplus**
  - ”contract” lxplus == batch worker node

LxPlus Current Status



- lxplus7 (CERN CentOS7)
- lxplus8 (AlmaLinux 8)
- lxplus9 (AlmaLinux 9)
- lxplus.cern.ch alias -> lxplus7
- lxplus-gpu
  - 5 Nvidia T4 GPUs
- lxplus9-arm
- lxplus node =~ lxbatch node

- Active Users
  - ~1500(day)
  - ~1000(night)

Other LxPlus Flavours

- **CMS, ATLAS, TH, ML & IT also have high-performance LxPlus-like machines**
  - 1Tb Memory. 256 core, local nvme storage
  - Access controlled via community-managed egroups
  - Unlike normal lxplus.cern.ch, no cgroup managed memory / process restrictions
  - Usage much lower, local scratch storage convenient but CEPH often faster
- **lxplus-gpu**
  - Useful for compiles, short debugging
  - Less useful in current configuration for analysis as multi user aspect can result in crashes
- **lxplus8-arm & lxplus9-arm**
  - New(ish) onsite nodes (previously had OCS)

LxPlus Usage, Outlook, Limitations

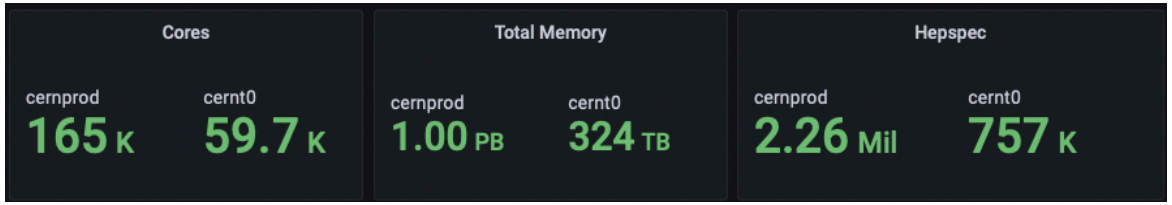
- **Migration to AlmaLinux 9: top level alias change on 7/12/2023**
- **Podman / apptainer available (and heavily used)**
- **Analysis is one use case of LxPlus**
  - General purpose compute service
  - Service managed with cgroups, with admin follow up for abuse etc – but many people want their own
- **Open to all experiment members***
  - *except clearly the self-managed ”lxplus-like” analysis machines
- **Scale out to batch / other systems**
  - The boundary between small scale interactive analysis and batch / other systems clearly down to user navigating the limitations
  - Dual use of shared filesystems on lxplus & lxbatch can cause performance issues – trad problem AFS misuse ruins perf of other home directories

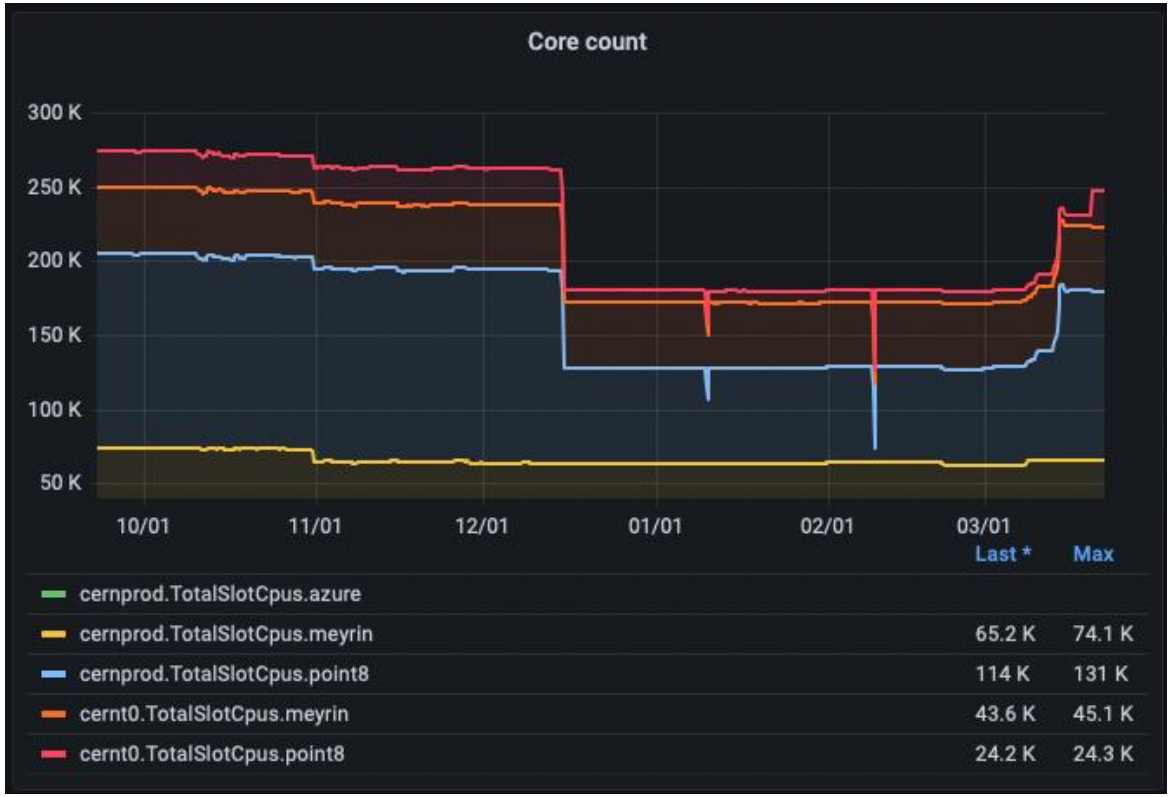
LxBatch

- **Current batch system HTCondor from CHTC Wisconsin**
- **Used for both Grid and ”Local” submission**
  - Grid means submitted to a ”Compute Element” (CE) which more or less means WLCG
  - ”Local” means any user submitting at CERN, authenticated with kerberos
    - Quotas vary (often dramatically) by experiment / group
- **High Throughput Computing**
  - “Embarrassingly parallel” (or “pleasantly parallel)
  - Primary platform for a batch process that can fit on one computer
- **Non-homogeneous resource types**
  - BigMem and BigMCore
  - GPUs (A100, V100, T4)
  - aarch64

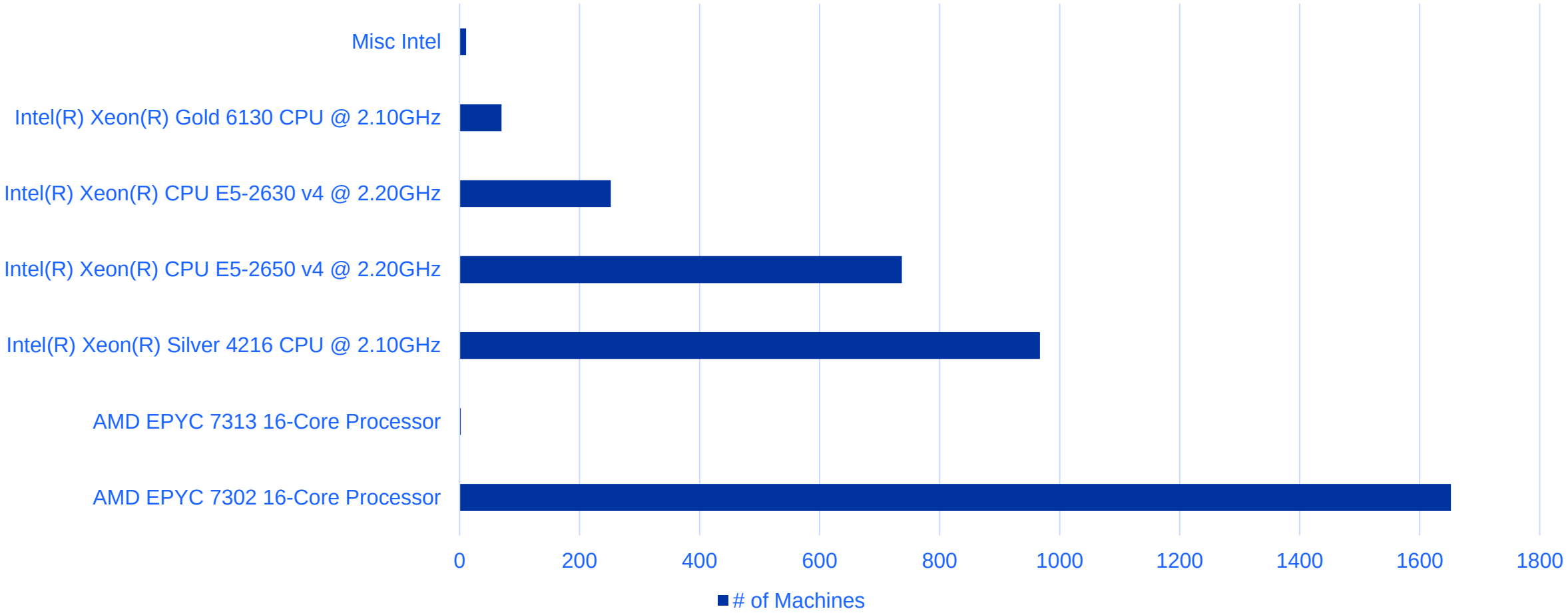
LxBatch Current Status

- **Capacity split between dedicated “cernt0” and shared “cernprod”**
- **Overwhelming majority CentOS7**
- **AlmaLinux9 beginning to roll out, target 50% by 7/12/23**
- **“Slot”**
  - Traditional: 1 core, 2Gb RAM, 20GiB scratch
  - Currently: 3Gb RAM 30GiB scratch
  - PCC: 4Gb RAM
  - mcore sweet spot: 8 core (4-7 easy, 9-16 possible, >16 small pool)
- **EOS, AFS available on nodes**
  - Direct usage via POSIX is a bit problematic

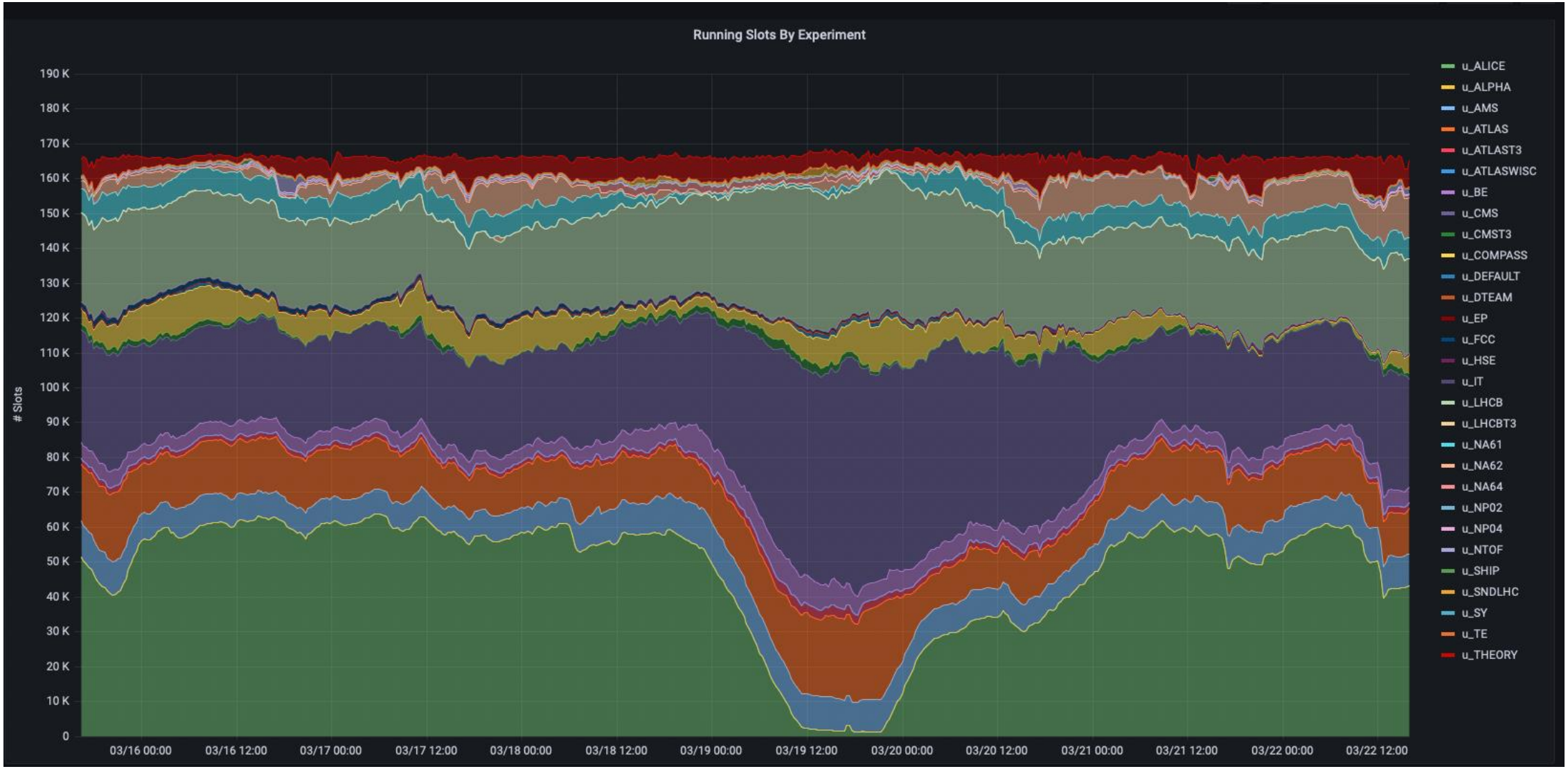




CPU type in LxBatch

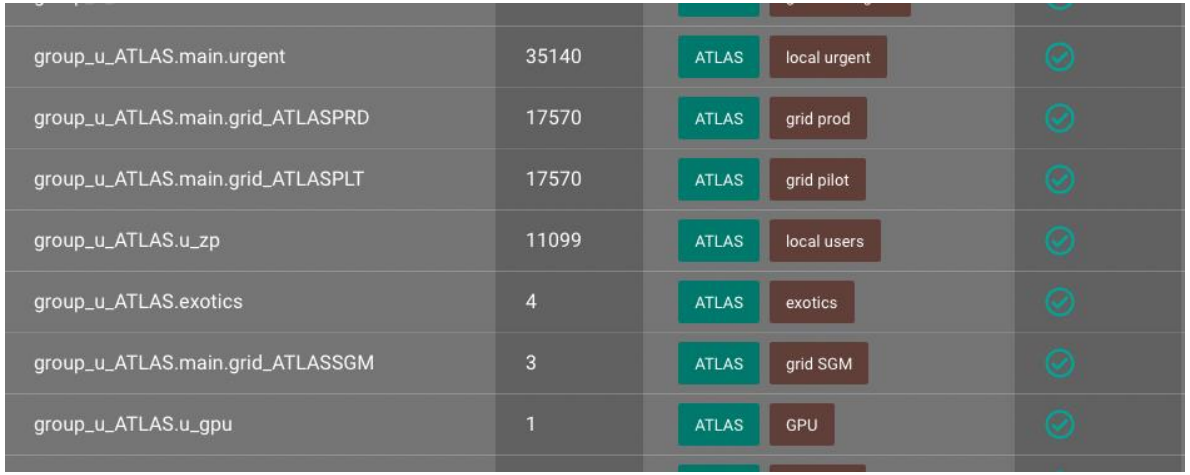


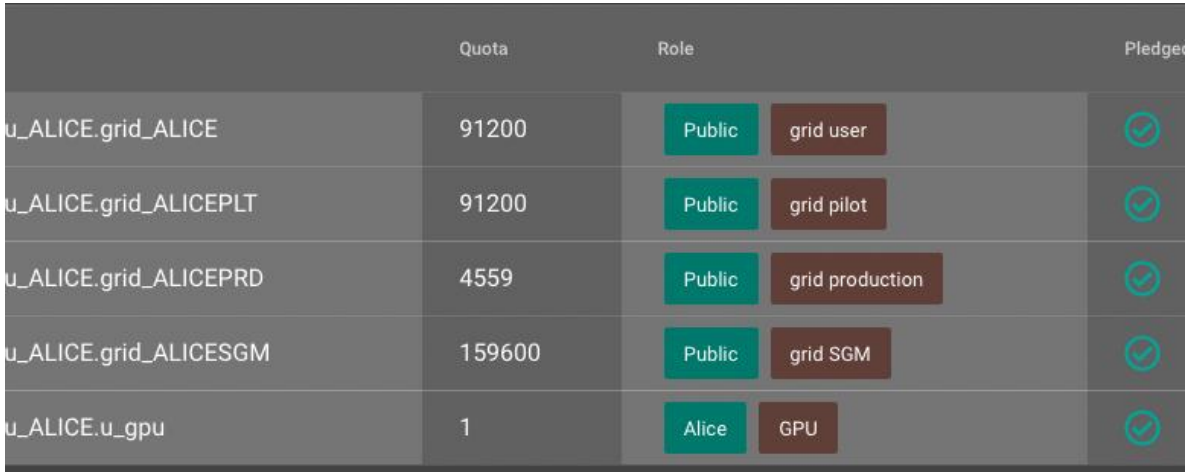
Usage (shared)



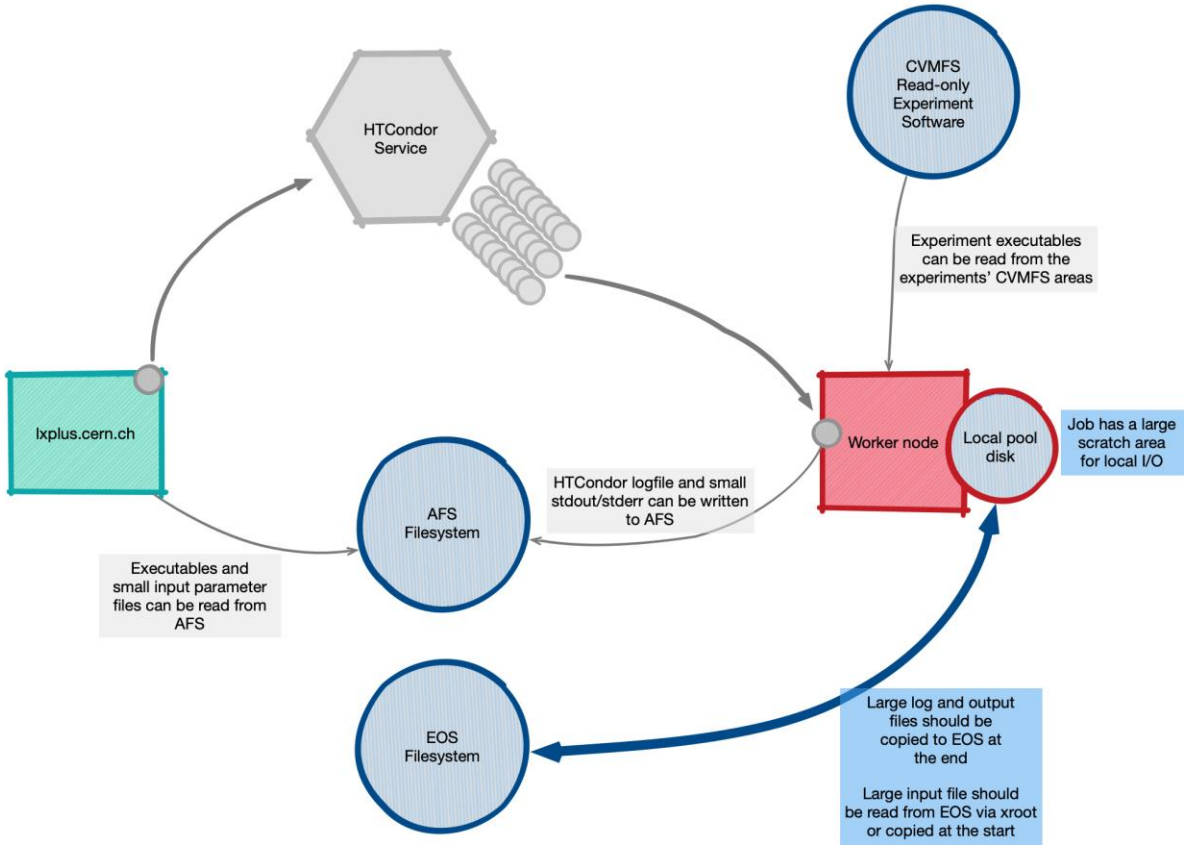
Accounting Groups and Share

- User are assigned to ”Accounting Groups” which are awarded relative share
  - Top level pledges to VOs are expressed in absolute terms (ie hepspec), converted to relative share
  - Within hierarchical Accounting Groups, share is assigned by the VO
  - Experiments take the decisions on share between for eg production, grid, local analysis
  - For some experiments, all analysis via the grid
- ”Unused” share, or “surplus” travels back up the accounting group hierarchy
  - This means that lucky submitters may be able to receive large spikes in capacity





Interacting with storage



- **LxPlus + LxBatch workers have /eos /cvms and /afs mounted. AFS is (still) $HOME**
- **Schedds do not have /eos mounted**
- **Best workflow is to stage in / out from shared filesystems, but use local storage for intermediate i/o**
- **File transfer plugin to use xrootd to stage in / out from EOS:** https://batchdocs.web.cern.ch/local/file_xfer_plugin.html
  - POSIX is hard at scale

Containers

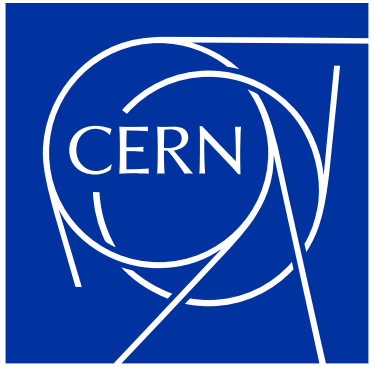
- **On the LxPlus side both ~~singularity~~ apptainer and podman are supported**
- **LxBatch can directly run containers, either via apptainer or docker universe**
  - Preferred method is apptainer with an image dumped into /cvmfs/unpacked.cern.ch
  - Details are documented here: https://batchdocs.web.cern.ch/containers/singularity.html
  - “Recipe” file for sync process controlled via MR to here: https://gitlab.cern.ch/unpacked/sync
  - CVMFS means that we gain from cache etc for job execution
- **Generic containers also available to run in “el7, el8 or el9”**
  - https://batchdocs.web.cern.ch/local/submit.html#os-selection-via-containers
- **No ability currently to run containers from private repositories**

Submit tools

- Aside from HTCondor tools / APIs used directly from LxPlus / elsewhere, other tools work with LxBatch
- reana can use HTCondor as a backend
  - https://docs.reana.io/advanced-usage/compute-backends/htcondor/
- DASK (as is often the case) needs a wrapper for its jobqueue to work nicely with LxBatch
  - https://batchdocs.web.cern.ch/specialpayload/dask.html
  - https://pypi.org/project/dask-lxplus/
- SWAN integration works, but coming soon will be a jupyterhub version including DASK (cf Enric Tejedor)
- Some technical challenges to solve (mostly around auth tokens) but most friction is around interactivity
  - Quotas can make eg dask worker instantiation time unpredictable
  - How do higher level tools help with interactivity ie display of partial results when scaling out

LxBatch constraints / advantages

- **Analysis isn’t the primary use case**
- **However, there’s a large potential for capacity, especially in terms of surplus**
  - This does affect expectations / interactivity “why isn’t my job running - I got 5k cores instantly last week?”
  - Technically feasible to use buffer capacity or preemption to help – many constraints aren’t technical
- **Open to all, but UX experiment dependent**
- **Can be used as infrastructure for other systems (DASK, SWAN, Reana etc etc)**
- **DAG workflows supported natively by HTCondor**
- **Parallel not a strong point, we don’t support it (we have SLURM for HPC)**
- **Support for non-x86 or non-mainstream resource sizes (if limited)**



home.cern