

Biowulf

High performance computing @ NIH

staff@hpc.nih.gov

Wolfgang Resch



Steve Bailey



Steven Fellini, Ph.D.



Susan Chacko,
Ph.D.



Gennady Denisov,
Ph.D.

**Picture
unavailable**

Afif Elghraoui



Ali Erfani



Andrew Fant, Ph.D.



Robert Gallimore



David Godlove, Ph.D.



David Hoover, Ph.D.



Patsy Jones

**Picture
unavailable**

Charles Lehr



Jean Mao, Ph.D.



Tim Miller



Nitish Narula, M.S.



Charlene Osborn



Mark Patkus



Wolfgang Resch,
Ph.D.



Antonio Ulloa, Ph.D.



Qi Yu, Ph.D.

**Picture
unavailable**

Abiola Bankole



Bill Woodford

The Biowulf system


105,500 cores / **211,000 CPUs** / 4095 nodes

920 TB memory

3.3 PB local scratch (lscratch)

38 PB shared high performance storage

5 PB of object storage

2018:  on the top 500 list

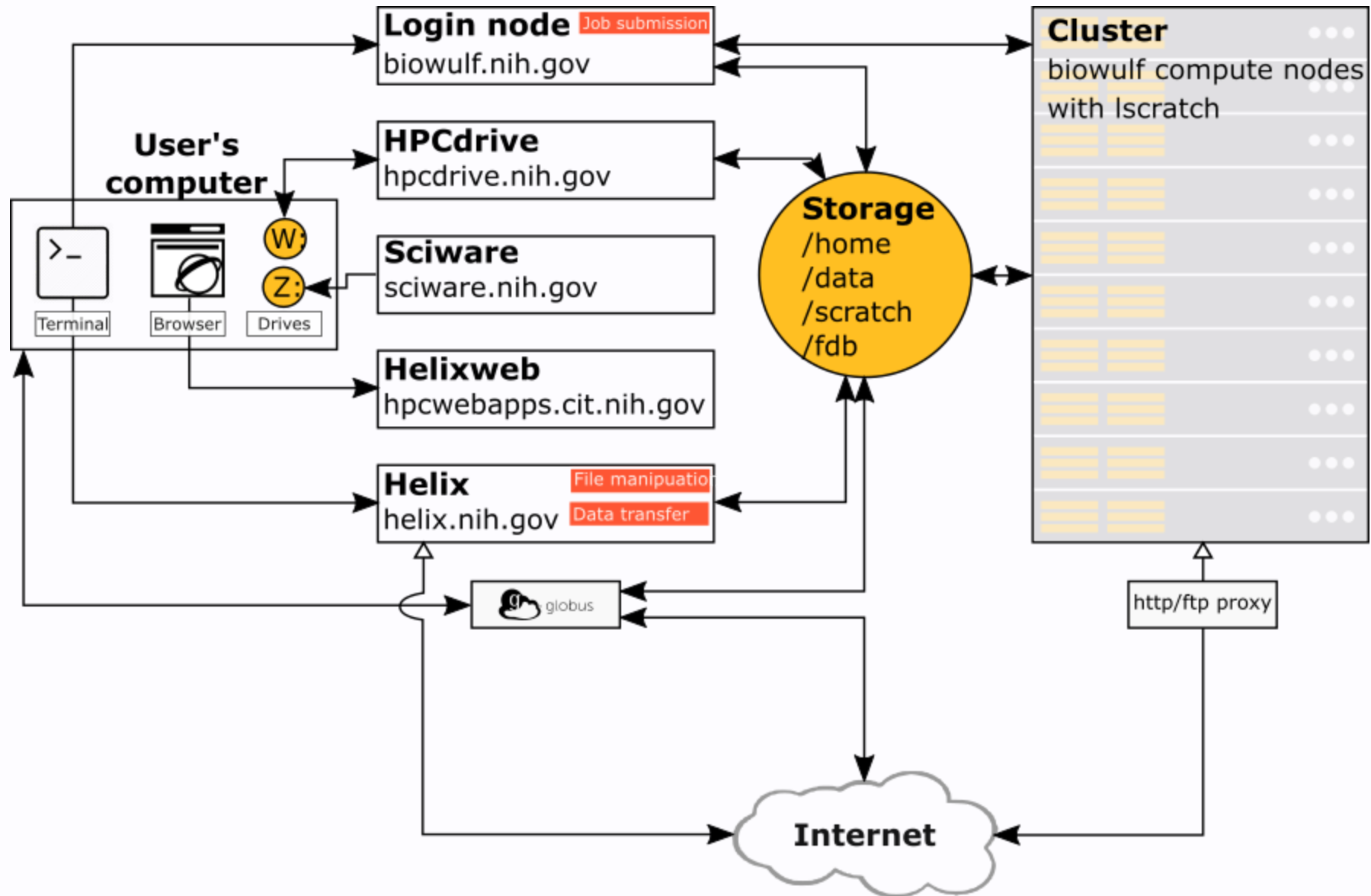
NIH interamural community

Most nodes connected by **FDR infiniband** @ 56 Gb/s.

Newest nodes are HDR (100 Gb/s)

Connected to NIH backbone @ 100Gb/s

Architecture

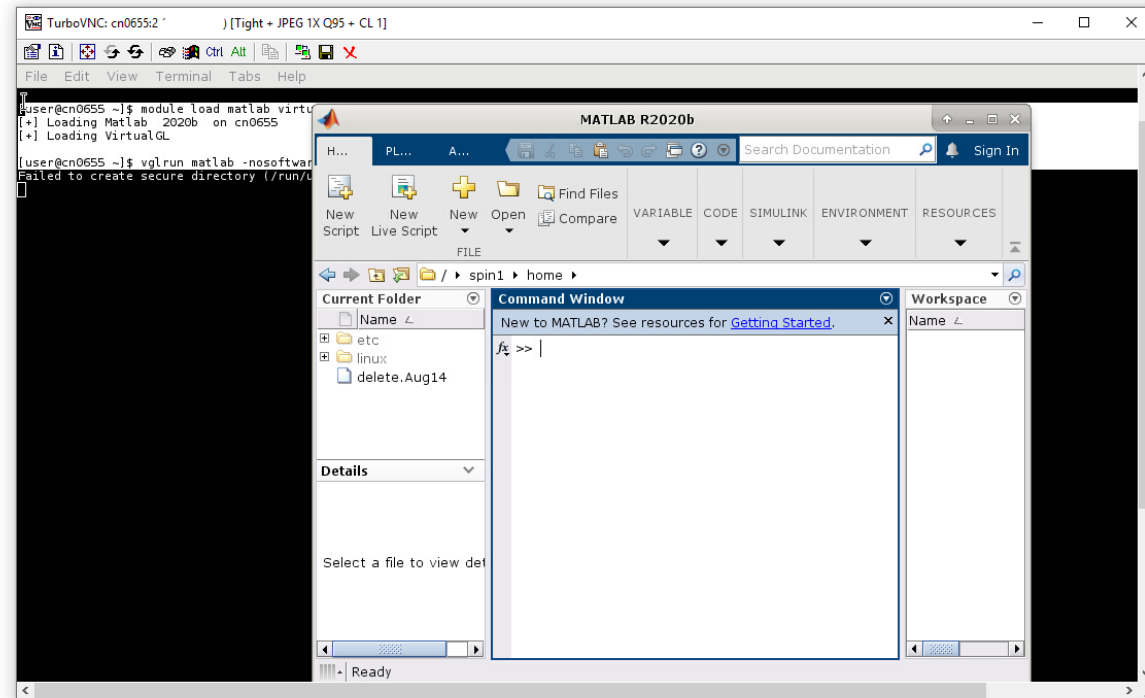


Node classes

- Single node, CPU (32-72 CPUs, 120-370GB)
- Multinode, CPU (32-56 CPUs, 58-240GB)
- GPU (k20x, k80, P100, V100, V100x)
- large memory (1-3TB)
- unlimited
- quick
- interactive, **visual**
- buy in nodes

Visualization partition

<https://hpc.nih.gov/docs/svis.html>



VNC + k20x GPU accelerated graphics

Applications

- multiple versions for ~1000 applications available in ~4000 modules
- Multiple versions of python with ~500 packages each
- Multiple versions of R with about 1600 packages each
- **singularity** for containerization
- jupyter and rstudio (desktop, community edition)

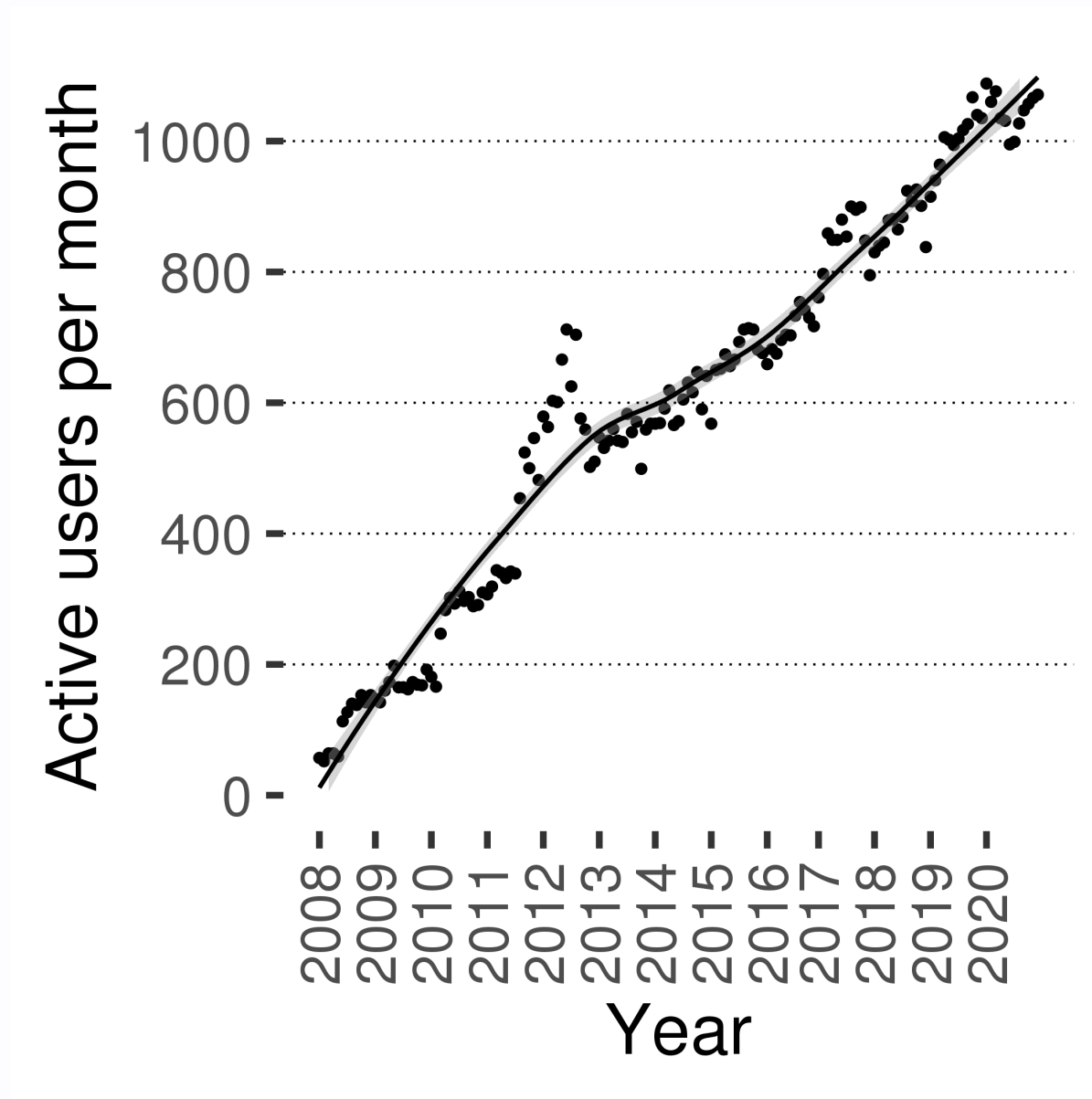
Data transfer

- For lots of data **use globus**
- Interactive download clients and scp/rsync/sftp **use helix**
- For convenience **mount hpcdrive**
- For transfers to/from box and OneDrive **use rclone**

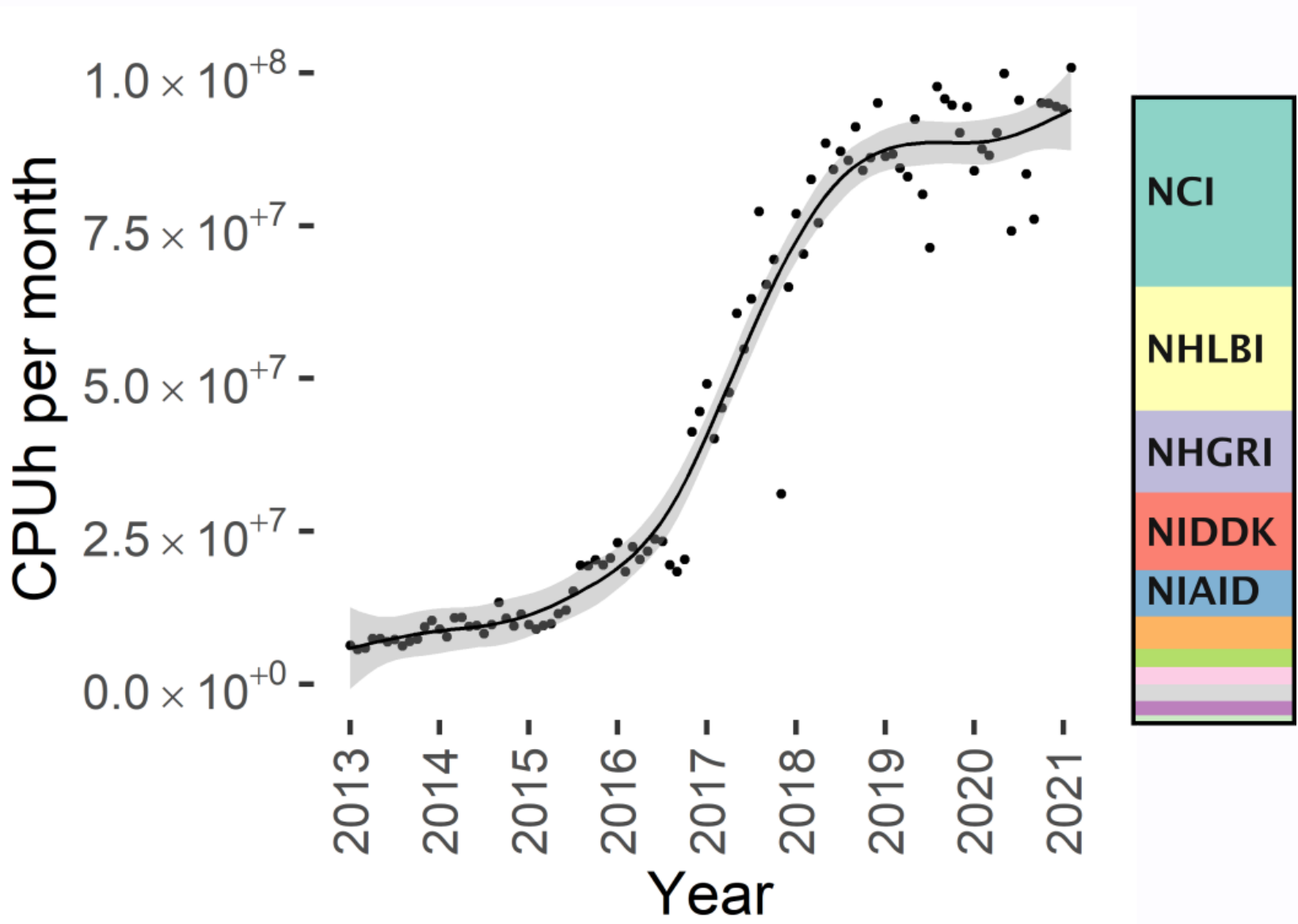
<https://hpc.nih.gov/docs/accounts.html>

Biowulf - utilization

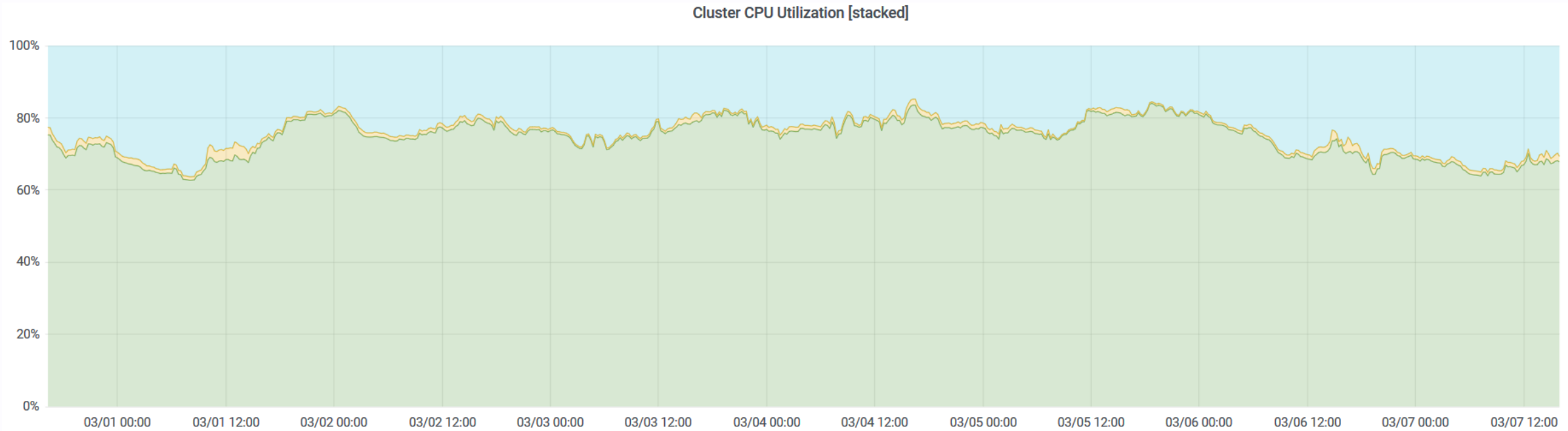
Active users



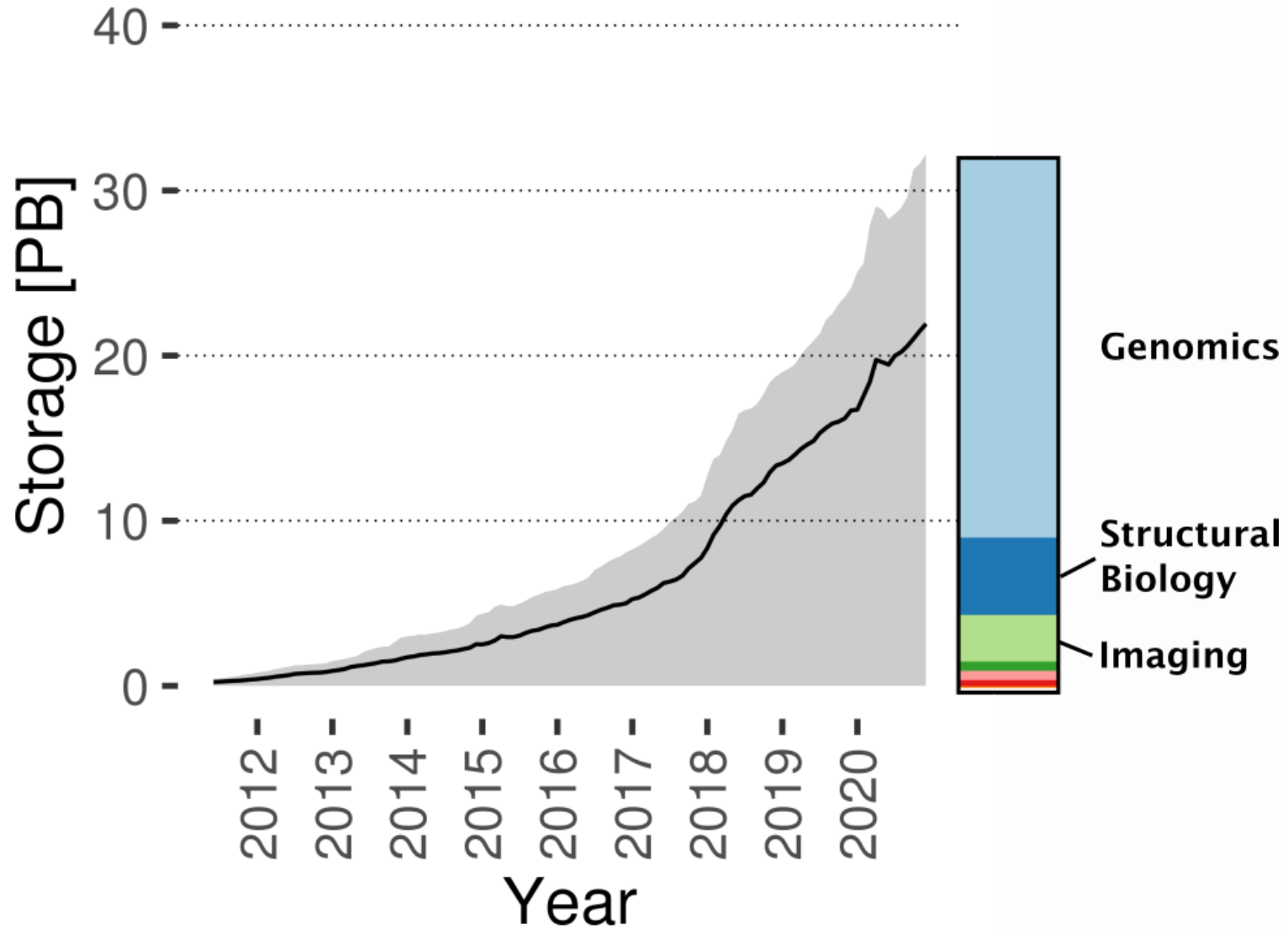
CPU



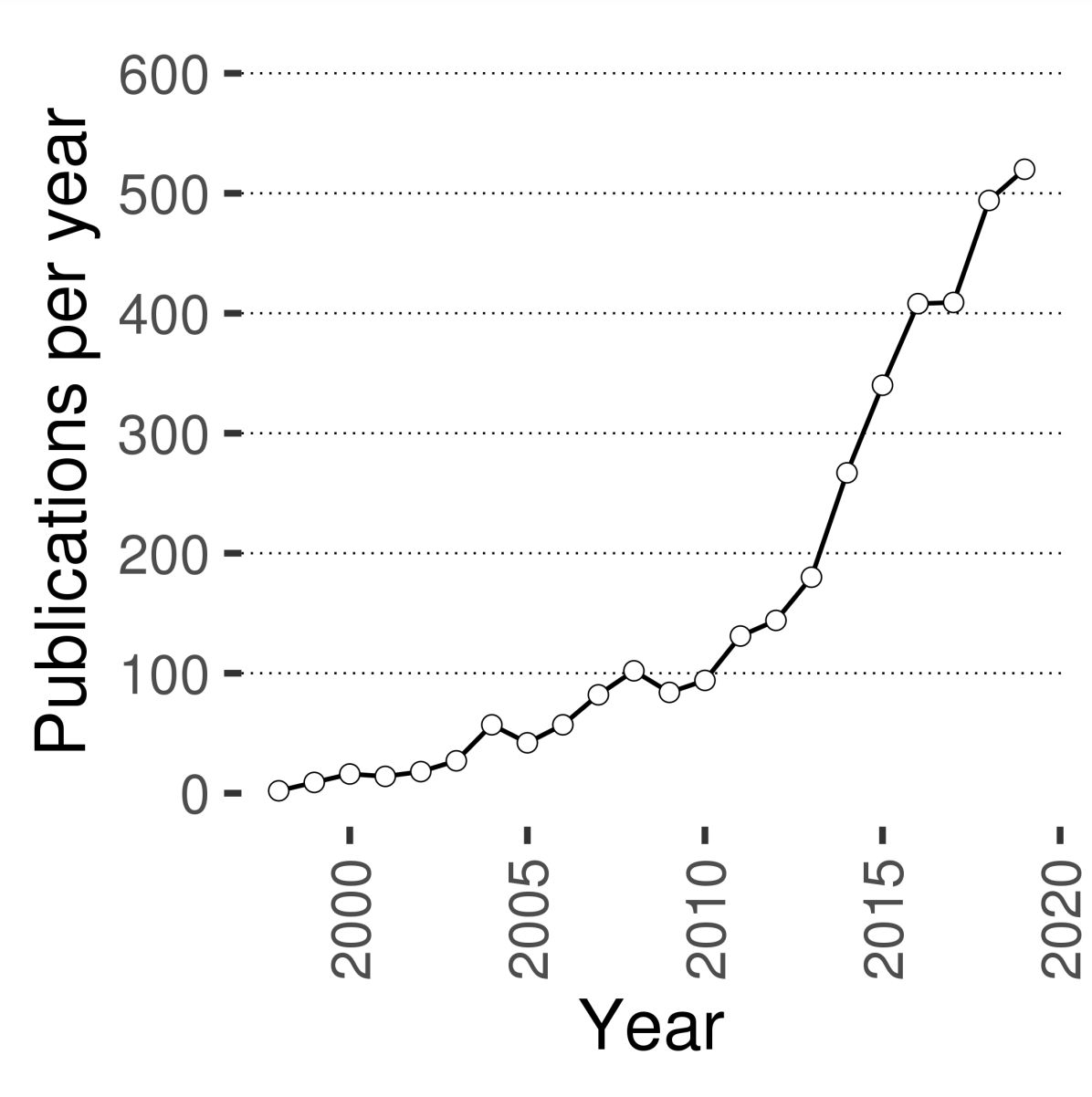
CPU utilization



Storage



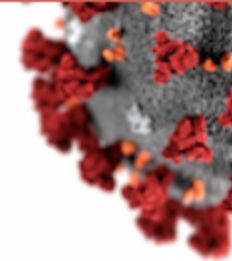
Publications



Covid Research

COVID-19 Research Support

52.4+ Million CPU hours used
1.2+ Million jobs run



Sample projects ([All projects](#)):

- Investigating the transcriptional regulators of hyper-inflammatory responses to SARs-CoV2 infection [[NIDDK](#)]
- Modeling unreported SARS-Cov2 infection from observed cases, computational modeling of Covid-19 biological systems. [[NIDDK](#)]
- Cryo-electron microscopy of virus proteins and complexes [[NCI_CCR](#)]
- Covid-19 transmission modeling and crossover trials [[NIAID](#)]
- Simulations of infection spread in populations [[NIDDK](#)]
- Genetic Determinants of Susceptibility to Severe COVID-19 Infection [[NIAID](#)]
- Structure of complexes of the SARS-CoV-2 ectodomain with a variety of ligands [[NIEHS](#)]

Biowulf users with COVID-related projects should [contact the HPC staff](#) to get increased priority for their jobs.

<https://hpc.nih.gov/cv19.php>

Biowulf - training and outreach

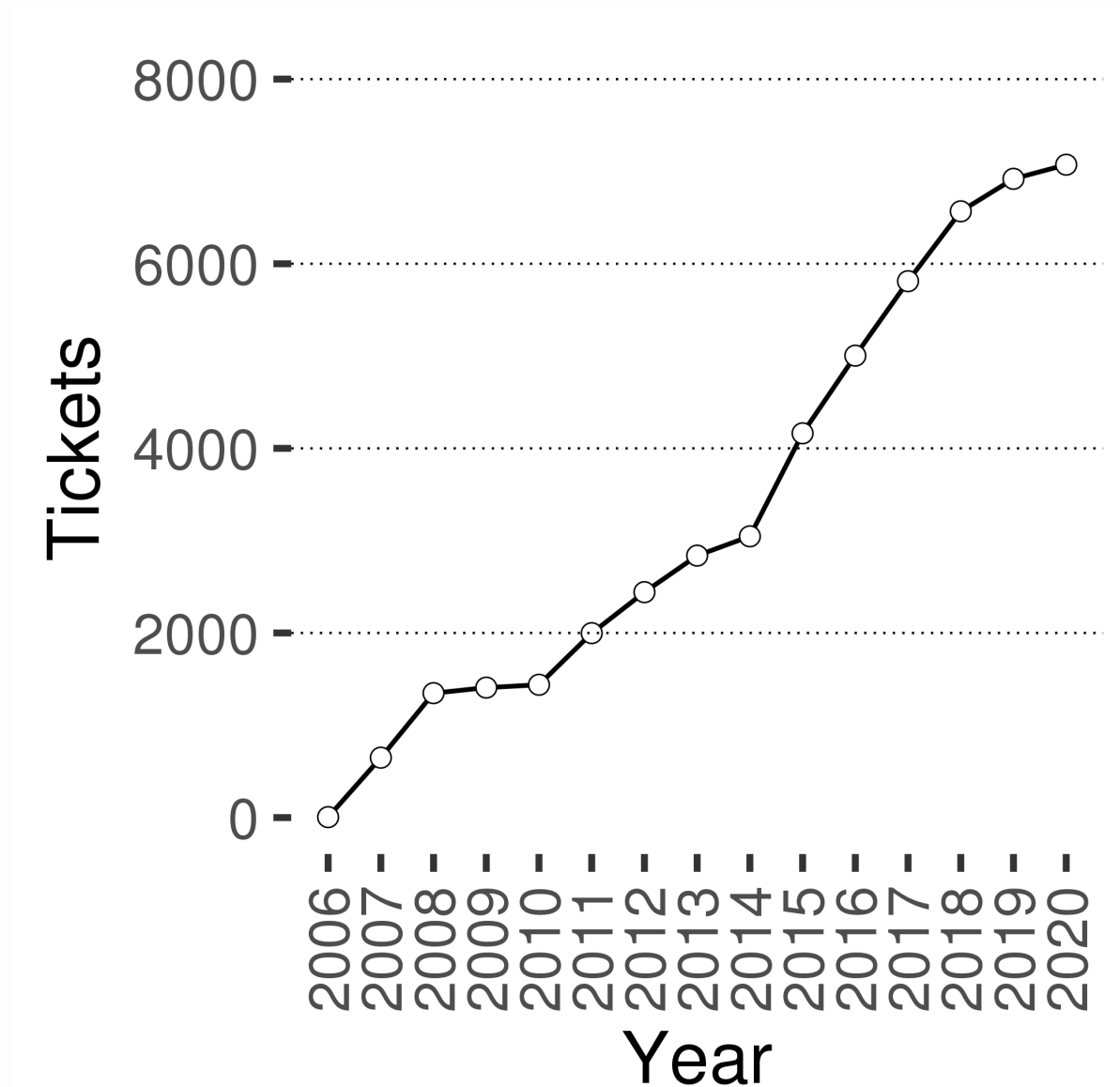
Training

- Online, self-paced classes (intro to `biowulf`, *bash*)
- Live action classes (e.g. Deep learning by example)
- Documentation (<https://hpc.nih.gov>) and tutorials (e.g. <https://github.com/NIH-HPC/snakemake-class>)

Outreach

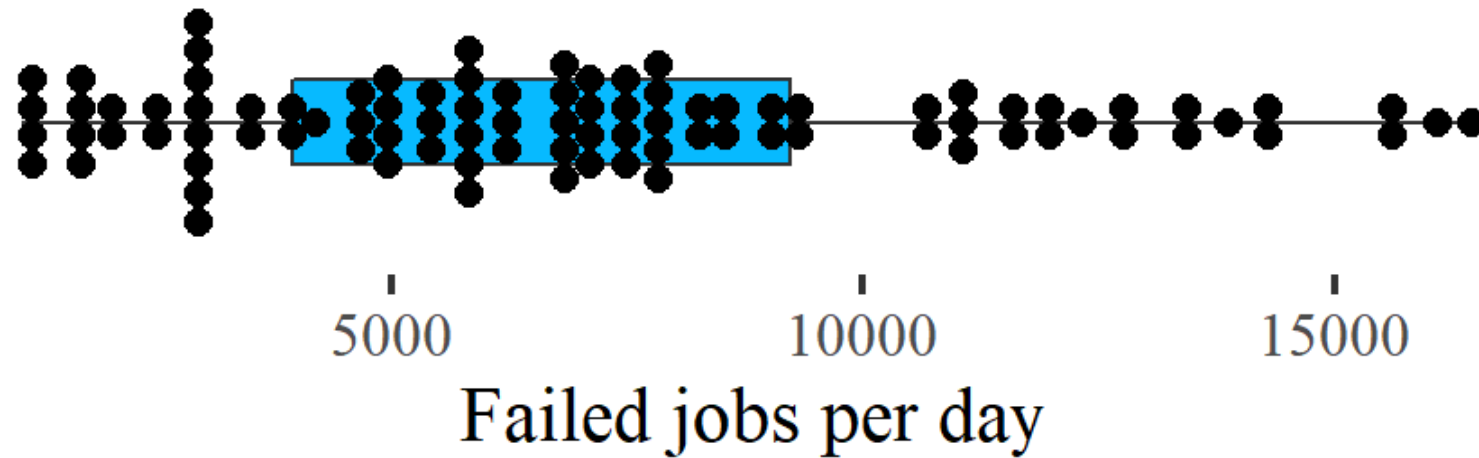
- Monthly *Zoom-in* consult
- Meeting with individual groups
- Cranky email from admin when you broke something

Support - `staff@hpc.nih.gov`



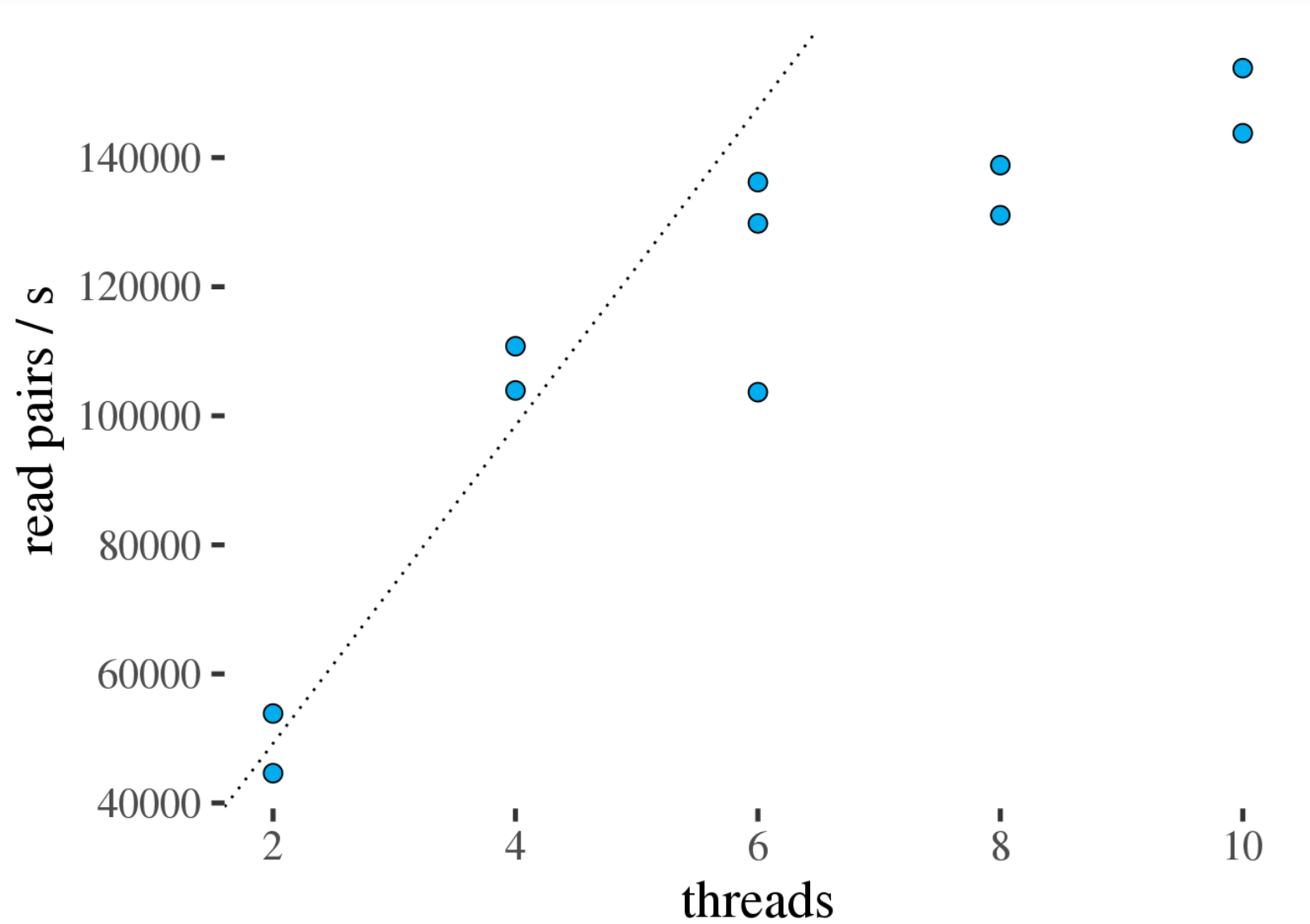
Best practices

Test before running large production jobs

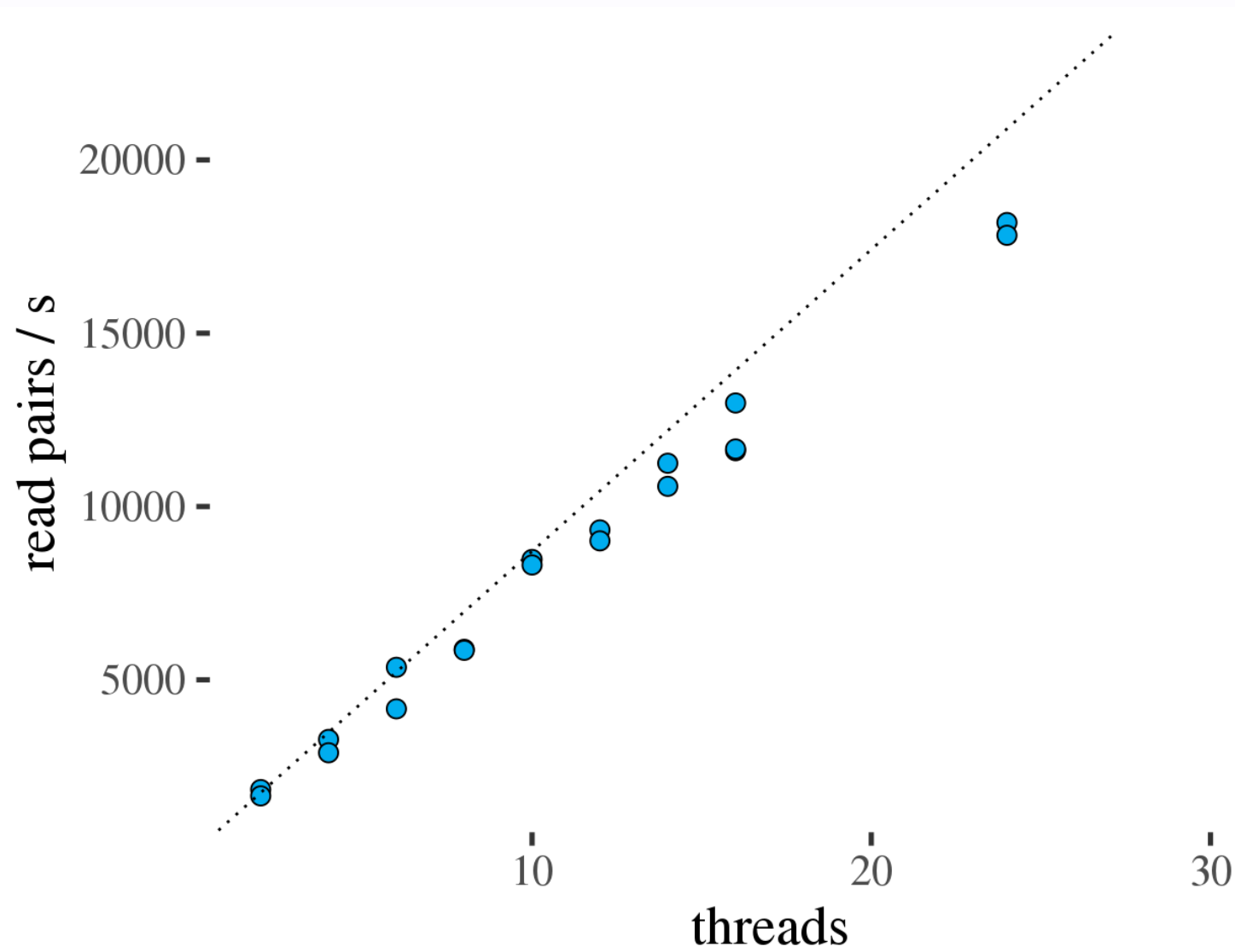


Scaling matters

fastp



bwa

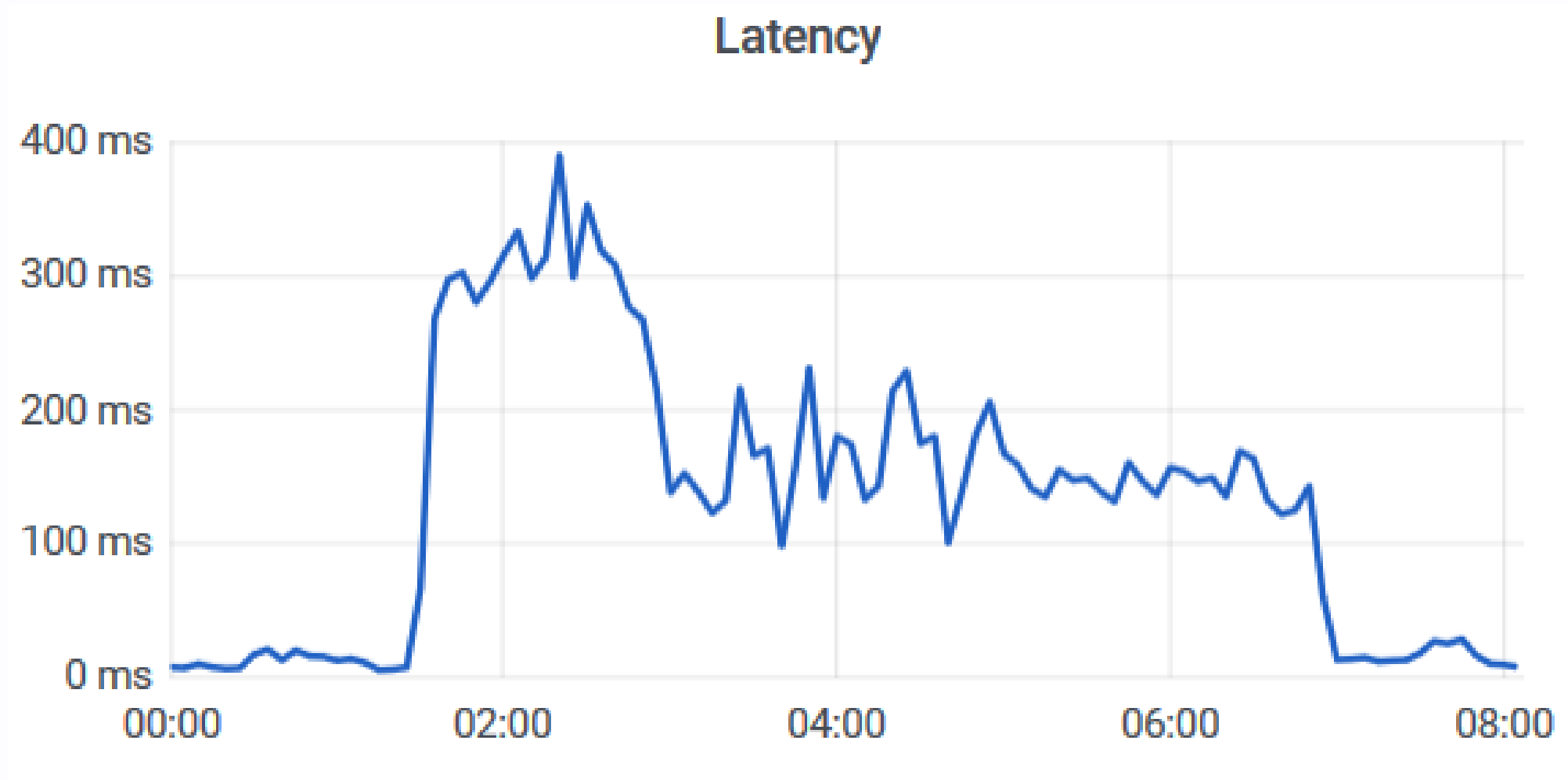


Be nice to the **file**
systems and use
`/lscratch`
and pipes

Do not swarm

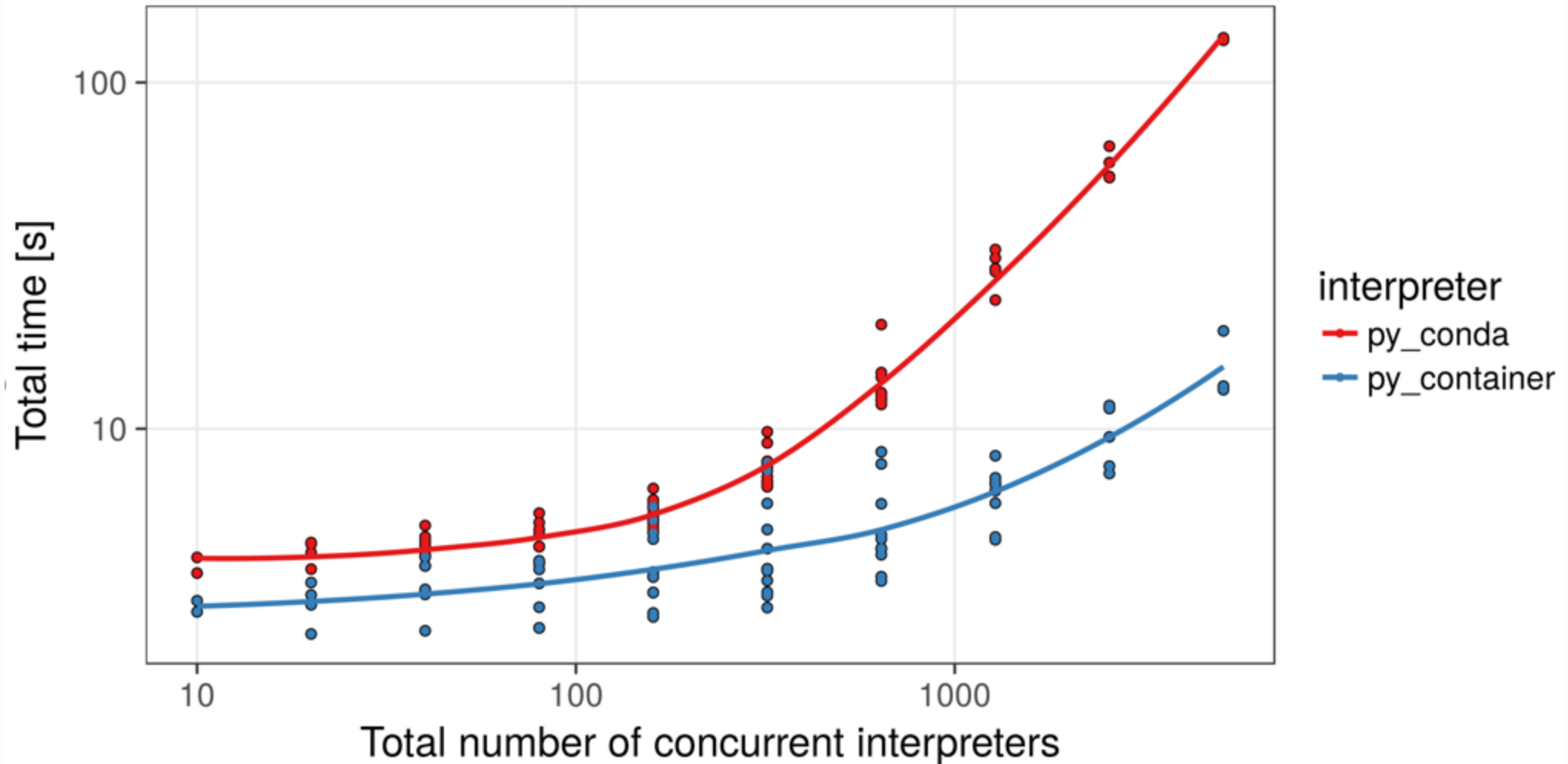
/scratch

otherwise this happens



Interpreter startup is
expensive

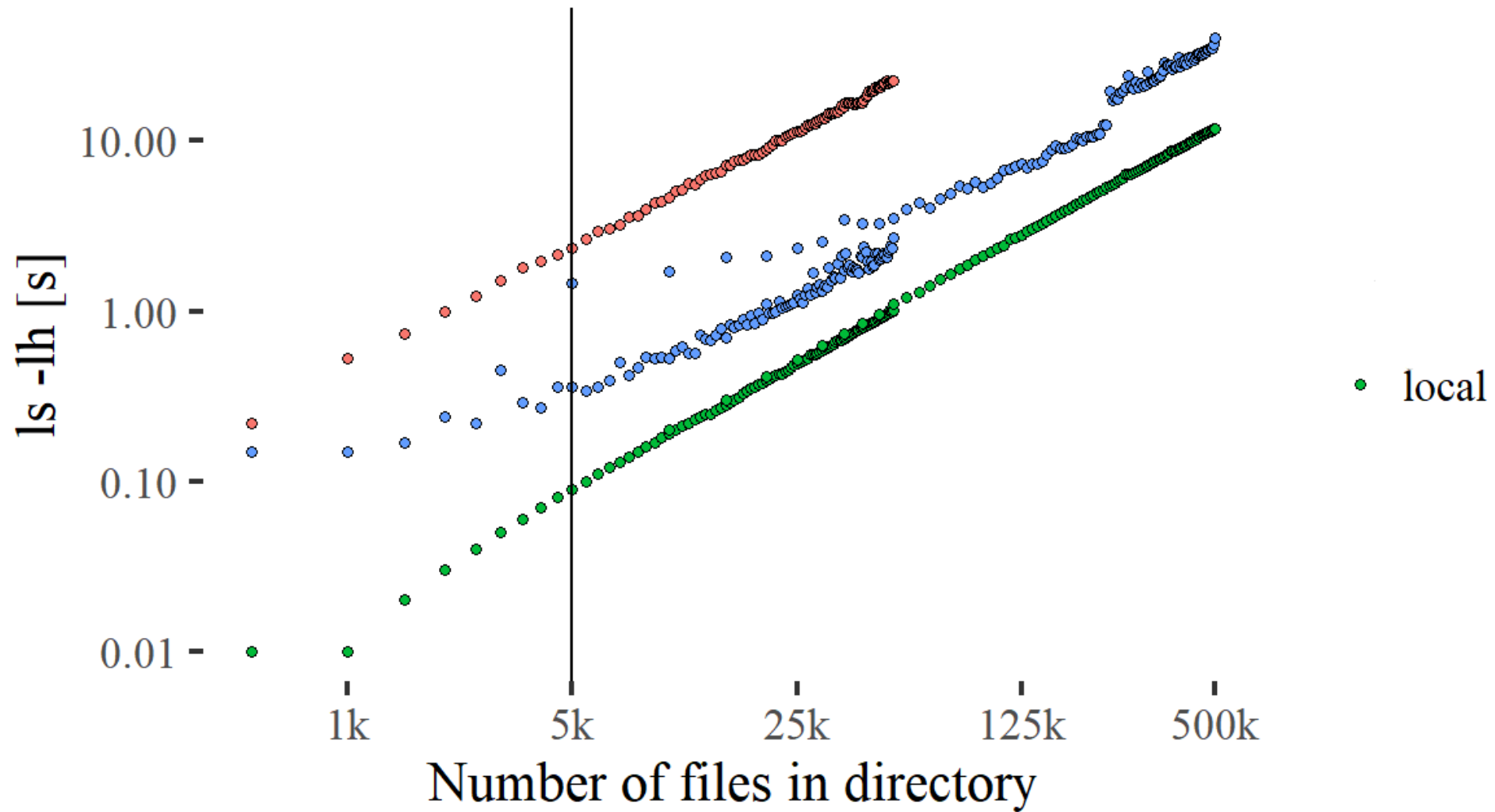
For example: concurrent python processes



Make sure you have
space in data for your
jobs

Keep **< 5000 files** in a
directory

listing files in large directories



and listing files is the least of the problems

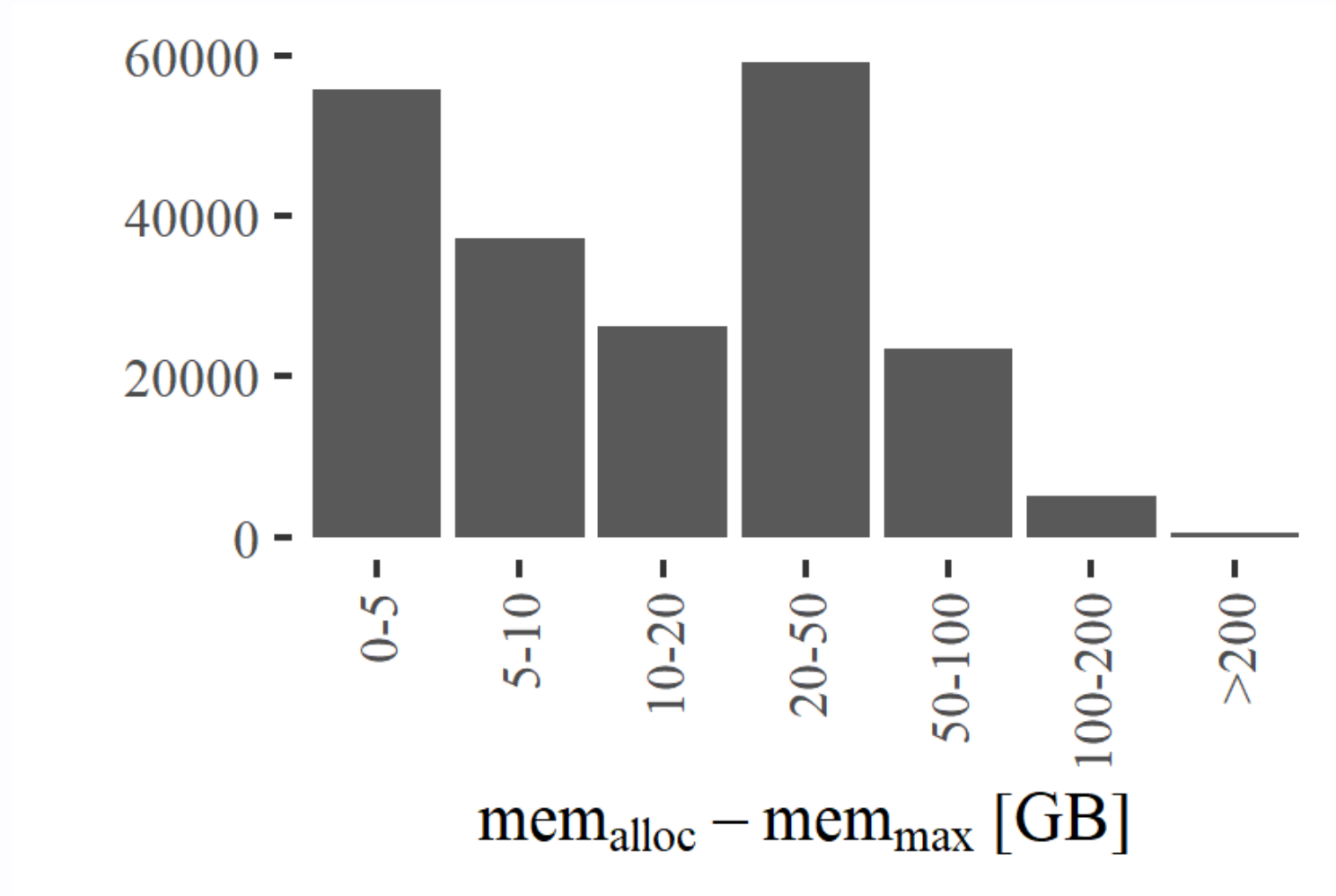
Biowulf storage is not for
archiving(*)

(* except the object store,
kind of

Backup your data

Have **good resource estimates** for your jobs

Completed jobs on norm over last 10 days



Don't **abuse** newwall

Use **job arrays** /
swarms when you can

Try to run jobs with

walltime > 15 min

Keep startup files **simple**
(no conda please)

Please **read your**
email

Be **detailed** and
specific when you file
an issue

Please **acknowledge**
biowulf in your
publications

Talk to us

`staff@hpc.nih.gov`

Questions?

Discussion?

Suggestions?

Cloud computing

Different ways to use cloud computing

Low level

spin up your own cloud resources and orchestrate your workflow

Mid level

- Google, AWS, and Azure provide life science APIs.
- snakemake/dsub/nextflow/gcloud pipelines using cloud resources
- some pre-defined pipelines

Example

```
(gcp) $ export BUCKET=gs://$PROJECT-life-science-test
(gcp) $ gsutil mb $BUCKET
(gcp) $ gsutil cp /fdb/app_testdata/bam/grch38/ENCFF374UMR.bam $BUCKET
(gcp) $ dsub \
  --provider google-v2 \
  --project $PROJECT \
  --regions us-east4 \
  --disk-size 200 \
  --name "samtools index" \
  --image quay.io/cancercollaboratory/dockstore-tool-samtools-index \
  --input INPUT_BAM="gs://$BUCKET/$BAM" \
  --output OUTPUT_BAI="gs://$BUCKET/samtools/submit_one/output/*.bai" \
  --command 'export BAI_NAME="$(basename "${INPUT_BAM}").bai"
             samtools index \
             "${INPUT_BAM}" \
             "$(dirname "${OUTPUT_BAI}")/${BAI_NAME}"' \
  --wait --summary
```


High level - specialized systems built on top of the low and mid level resources

- Firecloud
- Terra
- Seven Bridges CancerGenomicsCloud
- ISB cancer gateway