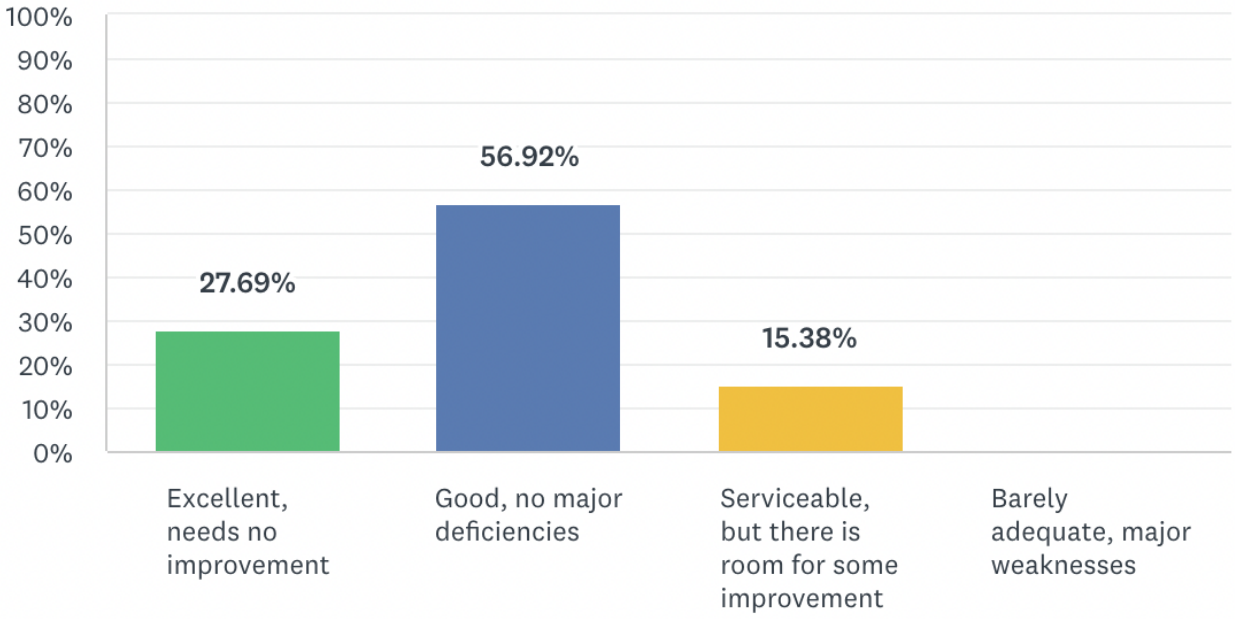
2021 Minerva User Survey Comments and Response
Scientific Computing and Data
March 23, 2022

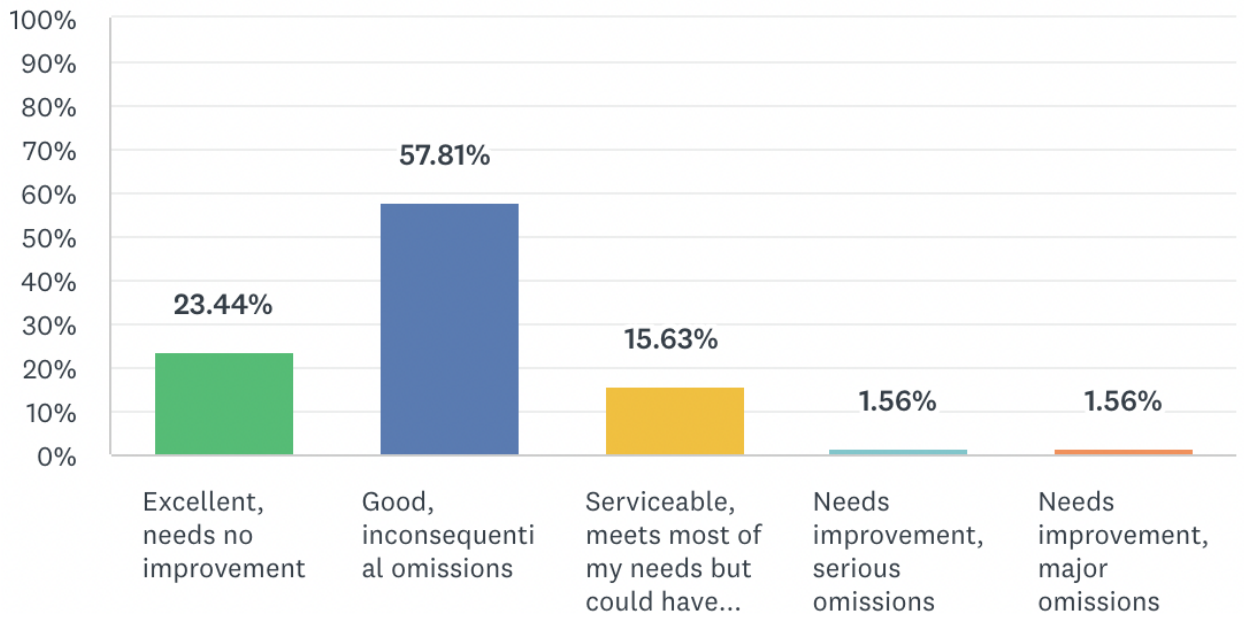
The 2021 Minerva user survey —distributed in January 2022—solicited feedback from 662 active Minerva users. Of these, 68 users responded (10.3% response rate)- 64 comments received

We asked five questions:

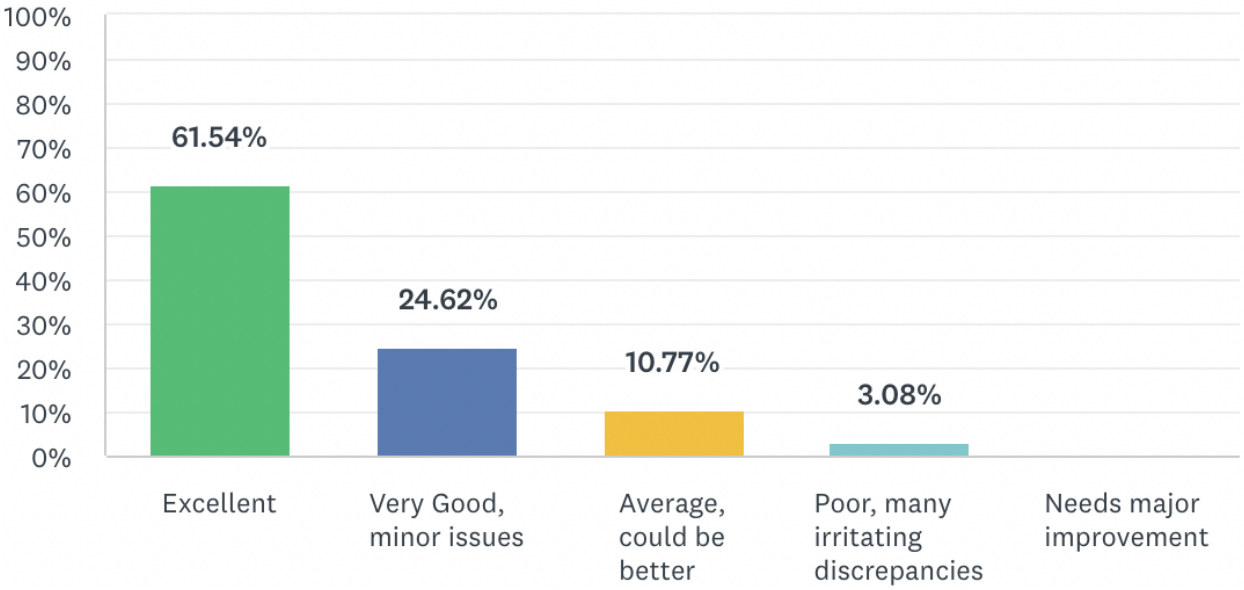
Q1: Overall, how satisfied are you with the LSF queue structure, compute and storage resources (GPUs, high-memory nodes, TSM, etc)?



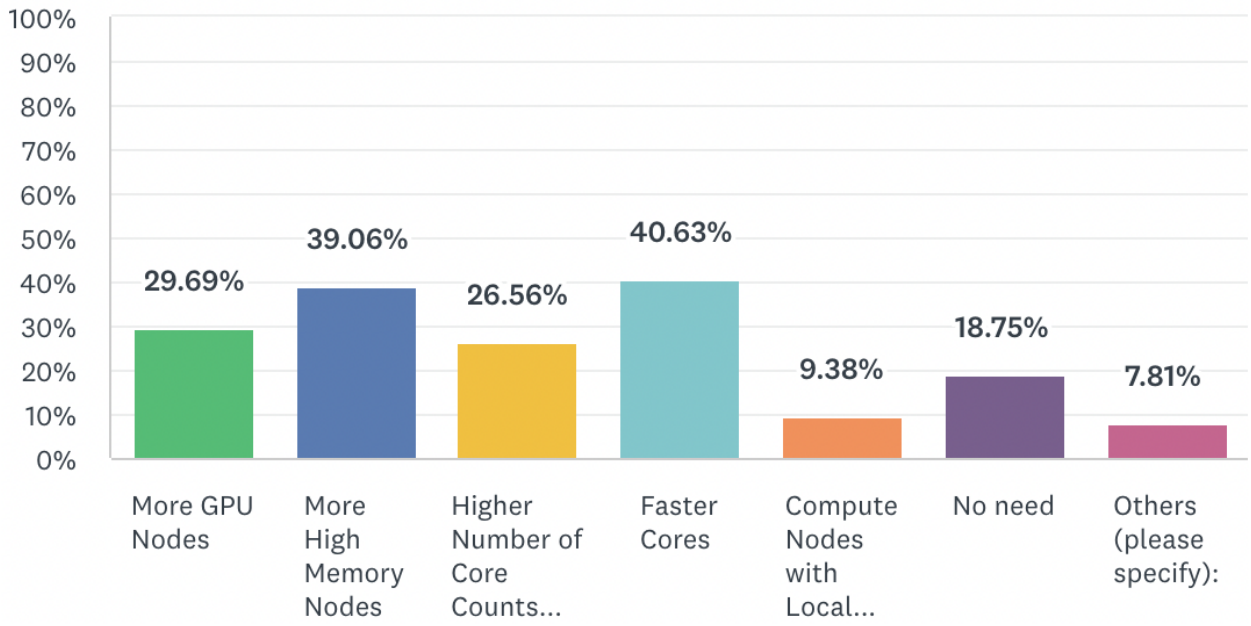
Q2: Please rate current software environment (packages and services such as database, data transfer, container etc).



Q3: Please rate your satisfaction with operations (ticket system, responsiveness of staff, documentation, user support etc).



Q4: Which of the following would you most prefer for future Minerva expansion?



Q5: What suggestions do you have for improving our service?

All comments and responses are as following:

System Issues

- Please schedule maintenance on evenings and/or weekends. It is extremely disruptive to do during the working day.
    - We have been always trying to schedule the PMs on weekends to minimize disruptions, sending out announcements 2 weeks ahead, with reminders followed. Please understand, however, that not all PMs can be scheduled on weekends (e.g., need to work with vendors) though we try hard to avoid weekdays. We will try our best to maximize the availability and usability of Minerva without interruptions.
    - Please note, we only had **one** PM without login access (jobs still running) in 2021 (i.e. Saturday May 29th).

- Need more GPU nodes.
    - We are adding 2 more GPU nodes (2TB RAM & 4 *80GB A100 on each) to serve more runs, especially heavy-memory jobs. We will send out an announcement with instructions once ready for production in early April.
    - We have set job limits on gpu queues for heavy users to prevent the resources solely occupied by a handful of users.
    - Our recent analysis based on the usage statistics of the whole GPU partition also indicates that we still have a satisfactory amount of resources to meet the needs of our community. We will keep monitoring the usage status of the GPU nodes to maximize their overall productivity.

    - If you need help on getting GPU resources, please always communicate with us at hpchelp@hpc.mssm.edu

- Need more high-memory compute nodes. ● The other improvement would make the process for using high-memory nodes, etc more accessible or make it a process where administrative approval must occur due to the demands of the resource.
    - We have deployed the new CATS (COVID and Translational Science) machine this Jan. It includes 55 high memory nodes with 1.5 TB of memory, 64 Intel Xeon Platinum 8358 2.9 GHz Processors per node, for a total of 3,520 cores. Please add "`#BSUB -R himem`" in your LSF job submission script to access them.
    - All NIH funded projects reported are eligible to run jobs on the new CATS machine. We have enabled CATS eligibility according to the records we have.
    - To check the eligibility of your project on CATS nodes, you can run "`mybalance`". If you have any questions on this, please send us a ticket at hpchelp@hpc.mssm.edu
    - In total, we have 92 himem nodes on Minerva.

- The storage of home quota is too small. 20GB space is not enough for software installation.
    - The home directory is to store frequently used items such as source code, binaries, and scripts. It's not designed for intensive I/O and has a storage limit of 20GB. You may want to use your work directory (/sc/arion/work/$USER) for your local software installation or project directory (/sc/arion/projects/xxx) if there is for group share software.

- Remove the recently implemented two-factor authentication requirement to SSH into Minerva from within the MSSM network: this is already required to join the MSSM network in the first place to obtain a login prompt, so there is NO real added security as a result of this policy, regardless of your interpretation of HIPAA requirements. On the other hand, the inability to use passwordless SSH is very inconvenient and causes substantial work inefficiencies.
    - For security reasons, we are required to use the 2-factor authentication for Minerva login.
    - You can set ControlMaster to reuse ssh connections for all hosts and set an alias for hostname to make it easier for multiple ssh connections. The setting details is at page 11 of our our training slides (edit %USERPROFILE%\.ssh\config on Windows or ~/.ssh/config on Linux/Mac): https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2021/09/MinervaIntro_Sep15-2021-compressed.pdf

- Having internet access for a few specialized compute jobs would be very useful.
    - The Minerva compute nodes do not have internet connection by default for security reasons except for those in the interactive queue. You need to be on an interactive node or one of the login nodes to talk to the outside world.

- But we do have proxy set up on compute nodes which allows you to connect to some sites. You can load it with `ml proxies`, and check the proxy settings by `ml show proxies`
- If you have a specific situation where an internet connection during runtime is a must for your workflow please contact us at hpchelp@hpc.mssm.edu to discuss further details about the workflow and possible options.

Job Scheduler & Queues

- The waiting time for submitting a job is sometimes long. But it is in general bearable.
  - We added 55 new CATS nodes (64 compute cores and 1.5 TB of memory per node) this year for NIH-funded research projects, which helps reduce the queue wait time.
  - We set the global job slot limit to 4,000 (the maximum number of running jobs for each user is 4,000 by default) on 03/18/2021 to avoid the whole cluster taken by the jobs of few users in response to users' feedback.
  - We have been also putting extra restrictions on certain users who use heavy resources such as memory in order to better balance the resources among users. You can check the slot limit by "blimits -u userID" while you have running jobs.

- Job queuing can often take a long time, despite a lack of pending jobs on a queue. Obviously the job queuing is still serviceable, but it is not always evident why pending jobs take a while to get a node.
  - There could be many reasons for a job to stay in the queue for longer than usual; The whole cluster may be all busy; Your job requests a large amount of compute resources such as amount of memory and compute cores so that no compute node can satisfy the resource requirement at the moment; Your job would overlap with a scheduled PM. To see the pending reason use the bjobs command with the "-l" flag:
    `$ bjobs -l <JobID>`
  - You can also use LSF commands such as bhosts, bqueues, and bjobs to check the status of the cluster and your job. Our latest slides on LSF is at https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2021/09/Minerva_LSF_2021-09-22.pdf.

- Overall, very satisfied. Primary exception is when a very small (1-3) users engage very large resources, essentially leaving very little available for everyone else (even for small jobs), which can lead to very long queue wait times. ● Tweaks to the priority of dispatches continue to be needed, but nothing too egregious. ● More queues for fast jobs with a limit on the number of jobs a user can request.
  - We set a global job slot limit to 4,000 (the maximum number of running jobs for each user is 4,000 by default) on 03/18/2021 to avoid the whole cluster taken by the jobs from a few users in response to users' feedback.

    - We have been also putting extra restrictions on certain users who use heavy resources such as memory in order to better balance the resources among users. You can check the slot limit by "blimits -u userID" while you have running jobs.
    - We also have the LSF fairshare scheduling policy implemented. This approach schedules jobs from each user with weights applied to their priority, based upon the user's historical usage information, but does not restrict the number of jobs a user can submit or run. We set the maximum number of *pending* jobs for each user to be 20,000 by default to regulate the overall load on the LSF scheduler.
    - We will keep adjusting our queue structure and dedicated nodes to reduce the turnaround time and increase the throughput along with the dynamic job needs.

- Most of the express queue is hogged, not allowing for other users to operate. Need another express queue with a smaller limit on the number of jobs.
    - You may want to use the LSF command "bqueues" to check the load status of the queues, which would be helpful for you to determine which queue is less crowded and where your jobs should be submitted to.

- The only thing missing is a language level scheduler like Dask for Python.
    - DASK is already available. You may simply load the python module to use it.

      $ ml python
      $ python
      …
      >>> from **dask_jobqueue** import **LSFCluster**

      In case you use DASK within your own conda environment you will need to install the dask-jobqueue package yourself to enable it.

- I've found the slurm scheduler to be more preferable to LSF, don't know if you guys agree with that but I would prefer slurm to LSF.
    - We received mixed opinions from users on switching our LSF job scheduler to SLURM. It may also cause a lot of interruptions to users with pipelines set up with LSF. So we will not switch to SLURM. We will keep in mind this option for the future.

- Need more interactive nodes.
    - Minerva has a limited number of compute nodes reserved exclusively for jobs running in interactive mode, accessible via the "interactive" queue. In addition to those dedicated nodes, regular compute nodes can also be used to open an interactive session by submitting a batch interactive job to the non-interactive queues such as "premium", "express" and "gpu" using the **bsub -I, -Is, and -Ip** options. For example,

      ```
      bsub -P acc_xxx -q express -n 2 -R "span[hosts=1]" -R
      rusage[mem=7000] -W 3:00 -Is /bin/bash
      ```

- One pain point for me is always the resource request guessing game. How many CPUs do I need? How much memory do I need? How long will it take to run? I would love easier ways (or maybe just better documentation of existing good ways) to get insight into the resources actually being used by jobs, so I know whether my resource specifications are accurate. ● Instructions on how to optimally request the correct amount of memory so that requested GPUs are not wasted.
    - Unfortunately, in most cases it's practically impossible to figure out in advance the amount of compute resources required for a production job at a glance over a job script or a quick check of the size of input data files. You will have to try a couple of short test runs with smaller data sets first before performing production runs to get a reasonable estimate. The LSF output log file of successfully finished test jobs will give you an idea about the amount of compute resources such as memory, walltime, and the number of compute cores/threads that would be required to successfully complete the production jobs.

- Another part of the guessing game is how to submit jobs of a size that will actually get run in a reasonable time instead of languishing in the queue. Should I submit 100 jobs requesting 4 CPUs each? Or maybe 200 jobs with 2 CPUs each? Or 400 jobs with 1 CPU each? Which option will finish most quickly? Which option will utilize minerva's resources most efficiently? As a mere user, I'm not familiar enough with Minerva's architecture to know. Some guidance from the people who are in the best position to know would be useful.
    - You can use LSF commands such as bhosts, bqueues, and bjobs to check the status of the cluster and your job. Our latest slides on LSF is at https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2021/09/Minerva_LSF_2021-09-22.pdf.
    - We will have our training session on Minerva April 1 and LSF April 8th. If you would like to schedule a 1:1 session, please email us at hpchelp@hpc.mssm.edu

- Better queue interface and better documentation.
    - We have installed a wrapper script authored by Harm van Bakel, which will make it easier to interact with the LSF job scheduler on Minerva, as suggested.
      To load them up, `$ ml LSFqueue`
      To get more info on the module, `$ ml help LSFqueue`
      And a detailed readme file at
      /hpc/packages/minerva-centos7/LSFqueue/1.0/README.txt
    - You can get quick commands listed by running “minerva_help” on login nodes.
    - Our latest slides on LSF is at https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2021/09/Minerva_LSF_2021-09-22.pdf.
    - For the complete guide to the LSF commands, see the on-line user manual(https://www.ibm.com/docs/en/spectrum-lsf/10.1.0?topic=reference-command)

- In larger groups, computing work might be assigned to one or very few bioinformaticians. They quickly exhaust their accounts by running calculations on behalf of many colleagues. That's because preprocessing steps for large omics data are usually the most computationally demanding part of lab studies. I wonder whether HPC considered some ways how to boost their accounts or give them a slower rate of priority decrease.
    - In certain special cases, the priority of jobs may be manually increased upon request. To request priority change you may contact MSSM Scientific Computing Support at hpchelp@hpc.mssm.edu. We will need the job IDs and reason to submit the request.

- I would love better tools for assessing GPU queue load. It would be great to more easily know how many of which type of GPU's are available before I submit the job and how many other users are ahead of me in a queue. I waste a lot of time reconfiguring my resource request when I am stuck in pending which I would not need to do if I had this information.
    - You can check the overall load status of the GPU queue by using the LSF commands “bqueues gpu”. Use the command “bhosts gpu” to see the occupation status of each GPU node. Note, the lg03a* is v100 gpus, while the lg07c is a100 gpus.
    - Here is the guide on how to request gpu resources at https://labs.icahn.mssm.edu/minervalab/documentation/gpu-etiquette/

Software/packages

- There have been some times when the system version of a package or program is broken or installed in a non-standard place or way, which makes it hard to do things efficiently ● Certain modules are broken, a reassessment of modules' functionality may be useful.
    - Sorry for the inconvenience. We will do more thorough testing and sanity checks before releasing modules. If you find a broken package/module, which you need to use, please let us know.

- More up to date versions of python and pip for virtual environment work. ● R versions and packages need to be maintained more up to date.
    - We support multiple versions of python 2, python 3 and R on Minerva, and other packages as well. You can see what versions are available and which one is the default using the module command “ml avail” or “ml spider”. The version number has to be specified explicitly to pick a specific version if not the default. E.g.

        $ *ml avail python*
        $ *ml python/3.7.3*

- Making often used pipelines (e.g. RNAseq et al.) available for all users.
    - Most of the often used packages/pipelines are available in our module system, and the instructions on how to build your environment with them using the

module commands are available at https://hpc.mssm.edu/about/modules. If there is an essential package that is not available on Minerva yet, we can make it available for all users upon request at hpchelp@hpc.mssm.edu

- My entire group uses snakemake and I can't understand why it is not supported as a module. Having to create custom conda environments for every little pipeline is cumbersome.
    - Snakemake is available via python packages. To access it, use

        $ *ml python*
        $ *which snakemake*
        /hpc/packages/minerva-centos7/py_packages/3.7/bin/snakemake

- I would like to see more IDE's like pycharm made available for editing code while in minerva.
    - We already have PyCharm on Minerva. To use it simply load the pycharm module:

        $ ml pycharm

- Like to have Docker support.
    - We can only support singularity on the shared HPC system due to security reasons. You can pull/run docker images with singularity. See our documents at Page 16 at https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2021/09/MinervaIntro_Sep15-2021-compressed.pdf and more detailed use on singularity training slide at https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2021/04/Singularity-04302021.pdf

- MATLAB parallel server products are missing. Right now MATLAB is operational on only one Node.
    - The MATLAB Distributed Compute Server (MDCS) product from Mathworks is available on Minerva. Please contact us at hpchelp@hpc.mssm.edu if you need help on configuring it for your workflow or need more information about it.

- It'll be great if RStudio server can be incorporated. The current RStudio-on-the-fly makes it a little difficult to run computations that require a long time, and sometimes multiple attempts are needed to open a session.
    - Rstudio server is not free. More importantly, the 2 factor authentication is not working with Minerva authentication env. Thus, for the reason of security and cost, we cannot support the Rstudio server this year.
    - Please note, even for rstudio server, you will need to wait for the resources to be allocated to your session.
    - We will continue to improve the in-house RStudio-on-the-fly code.

- Rstudio over the web is fantastic but it would be even better if it had the same packages as the current R/4.0.3 module. ● Overall great. We absolutely LOVE the containerized Rstudio implementation - it's outstanding. It would be great if 1) it were more straightforward to keep the version of R up-to-date and 2) we could launch Rstudio jobs without running into a 'locked' node, which necessitates submitting a new job and re-entering the queue. ● I would love to be able to run RStudio on minerva AND then be able to have my R session submit jobs to minerva from within RStudio. I use packages like batchtools to manage parallel cluster computations from within R. The current solution for running RStudio in a container doesn't support this, which means I'm stuck using R at the console when I want to submit jobs.
    - Our on-the-fly Rstudio over Web implementation is basically designed mostly for beginner users, which opens a RStudio session on the web and it comes with some basic packages only. They're not exhaustive at all and might be different from those available in the R modules on the system. You'll need to install and maintain your own R related packages if any. Though this would be helpful in the initial stages of development, testing and debugging, we would strongly recommend that you use the command line versions of fully loaded R, available via the system modules for production.
    - Lock files are created by RStudio program itself by design. If there already exists an active session owned by another user, your RStudio session cannot start on that node. You may want to slightly modify the LSF input parameters such as -n, -M, and -W so that yours can be sent to another node by the scheduler. You may need to try a couple of times to have yours dispatched to a node with no active RStudio sessions on.

- I keep having issues launching Jupyter notebooks on Minerva. I know of at least a couple of other people who have also encountered this problem.
    - Please contact us by opening a ticket at hpchelp@hpc.mssm.edu and provide a bit more details about the issues you encountered so that we can take a look.

Tickets

- Occasional missed tickets but very rare. ● I'm sure you get many annoying queries, so I sympathize, but usually I am contacting user support as a last resort and have already checked my job script many times.
    - We had some resource constraints last year, which was challenging. Lack of manpower has caused slower than usual response time to help researchers on some specific topics. If a ticket is missed by mistake please let us know. If you think our response is not satisfactory, you can always request 1-1 sessions with one of our staff members to communicate more effectively and discuss further details. We will try our best to respond in time and meet your expectations as always.

- It would be great if the HPC staff members name was automatically inserted into their response emails. When I have a long exchange, I like to know who I'm talking to. Clicking the ticket link for the website to see the name after doesn't work.
    - We will try not to forget to include our names in response.

Tutorial/Training

- It'd be nice if the minerva group can host more detailed sessions on how to submit complicated jobs (e.g., how to loop through parallel scripts and run those; etc.) ● More assistance in figuring out the best way to parallelize things would be nice. This could be in the form of documentation, or presentations, or "office hours", or something else. ● The community needs better documentation on job submission best practices. I think there is a lot of unnecessary resource usage because users do not know how to properly request what they need and are unaware of them taking too many resources. ●Templates and tools for everyday tasks. There's not really a tutorial script for different features of the Queue and compute structure/resources. When there is it, it is always buried deep in a PowerPoint where you have to check multiple and hope the PowerPoint where you found your answer is up to date. I rely on so many other universities and hospitals' websites to help with our Minerva and spend much time finding practical advice that works on our system. ● Provide more detailed guidance for new users / coders.
    - Our documentation on LSF Queues and policies is at https://labs.icahn.mssm.edu/minervalab/documentation/lsf-queues/
    - Examples for batch job submissions are found in our website: https://labs.icahn.mssm.edu/minervalab/documentation/job-execution/ https://labs.icahn.mssm.edu/minervalab/documentation/multiple-serial-jobs/ And in our latest training slides: https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2021/09/Minerva_LSF_2021-09-22.pdf
    - We have several training classes a year and announce these through the Minerva mailing list and post them on our hpc website (https://labs.icahn.mssm.edu/minervalab/)
    - You can also request 1-1 sessions in case you need further assistance by opening a ticket at hpchelp@hpc.mssm.edu
    - If your research group would like to schedule an additional tutorial or tutorials focused on special HPC topics, you can reach out to our computational scientists (hpchelp@hpc.mssm.edu).

- Another thing would be helping labs set up an organizational structure or at least some documentation on doing so. I got thrown into the deep end to handle our lab's Minerva when I started. I wasn't experienced enough to fix my lab's file structure and system or understand Linux best practices, especially within this system. Maybe I should've looked harder externally, but I feel many researchers lose time on the same things. Figuring out how to start up, what to do to begin or continue a project, and small things like setting up a temporary python or R environment in our file system or writing backup scripts. I

understand staffing issues and priorities make a lot of what I said tough. If there could be a common place for labs to share basic helper scripts, documentation on tasks, and/or templates, I would love to contribute, and I'm sure other labs would too.
    - You can always request 1-1 sessions or group training sessions tailored to meet the needs of your group by sending us an email at hpchelp@hpc.mssm.edu

Documentation on Website

- Support is excellent, documentation really lacks. ● The documents and helps are not described very clearly. ● Documentation needs to be better. The PDFs of training powerpoints are useful to an extent, but an actual documentation system is needed.
    - We are continuously working to improve our documentation. We have been updating the changes on the system and adding new services.

- Add documentation and links to often used (sequencing) databases (human genome et al.) to the "Documentation" webpage
    - https://labs.icahn.mssm.edu/minervalab/resources/data-ark/

TSM

- Overall, it's great. I wish there were regular backups, and don't like that archives expire but that's fine. ● The improvement would be providing backups of certain parts of the cluster. ● Again, a spot for backing up some of the data. Clusters can be dangerous if data is lost and not recoverable. ● There's not really a tutorial script for incrementally backing up via. DSMC backup.
    - We don’t recommend users to use TSM for Backup unless it is really really needed for your work. TSM archive can meet the needs of most users, to keep copies for the data that is not needed for a while. Please use TSM archive when you can. If you do have important files constantly changing and you really need to backup them, please complete the Minerva TSM Backup Request Form. We will contact you to discuss it.
    - Our detailed documentation on TSM is at https://labs.icahn.mssm.edu/minervalab/archiving-data

Others

- The VPN is so bad that I get kicked off multiple times a day, sometimes every 10 minutes. ● There were a few times that the connection was a bit laggy recently. But overall it is great.
    - If you have any VPN related issues/questions, please reach out to school IT by opening a ticket at ASCIT@mssm.edu.
    - There are multiple factors that affect the speed and quality of your network connection to Minerva. We try our best to notify users as quickly as possible when we notice any network/connection issues on our end. Please note that the

stability of your local internet can affect your connection speed as well especially if you're connecting to Minerva remotely.
    - If campus network is fine, if you still experience a frequent drop of your ssh connections, you may use the following ssh configurations to keep your ssh session alive:
      On your own computer (not Minerva), edit the ~/.ssh/config file:
      Host minerva
      Hostname minerva.hpc.mssm.edu
      ServerAliveInterval 240
      ServerAliveCountMax 2

      With this configuration, the SSH client sends a packet to the server every 240 seconds (4 minutes) to keep the connection alive. If the client does not receive a response after two tries (as specified by the ServerAliveCountMax setting), it closes the connection.

- dropped jobs.
    - Please let us know by opening a ticket at hpchelp@hpc.mssm.edu if you experience this kind of issue. Most likely it's due to an intermittent communication drop between the client compute node and the master node. In many cases jobs are still running on the node and the outputs might be retrievable.

- It also is tough to get any moderately computational researchers interested in using Minerva's excellent features. The job system is scary to them, the login is tedious, and there's no safe way to link to Sinai's google drive. Not all of these things can get fixed, but if we can in the house find a way to connect to google drive (which is the standard for storage in many labs), it would save a lot of time, money, and make Minerva much more approachable for new comp researchers who have their complete workflows in Drive.
    - We provide a command line utility "gdrive" for interfacing with Google Drive. Load the gdrive module to connect to your Google Drive account at ISMMS from Minerva:
      $ ml gdrive
      For instructions about how to use this tool:
      https://github.com/prasmussen/gdrive/#Usage

Thank you for your nice words!

- I started using Minerva 4 months ago and I have not encountered any problems.
- It is excellent - always a place for improvement, though.
- I'm sure improvements can always be made, but HPC is always super responsive when I ask for new software/help with current environments/software.
- This is really excellent and always prompt & helpful.
- I've been quite happy with the response time whenever I've had to contact Minerva's support.
- Staff is very responsive and I have no issues.

- The service we get, response time, attending to specific needs - all are exceptional!!!!!
- Staff is *the best*.
- Very helpful.
- No need for more resources at this time, availability has been great!
- I am very happy with the team and they have done a great job.
- I want to start by saying Minerva is an incredibly helpful and necessary system with so much to offer. I love using it,
- I want to reiterate that Minerva is excellent, and my research would be nowhere close to where it is without it. Your team is so responsive and works so hard, and this system is incredible.
- Consistent and smooth once you get the hang.

.