Results of Researcher Assessment: Configuration of Major HPC Expansion, 2022

This document contains four sections:

- Section 1: Introduction
- Section 2: Survey questions -- Questions as presented, verbatim, to participants
- Section 3: Results -- Participant responses
- Section 4: Appendix -- Complete fine-grained results for data summarized and/or simplified in Results section

Section 1: Introduction

A major expansion of HPC, expected to be online in Spring 2022, will be undertaken. In order to determine the specifics of the hardware configuration, NJIT researchers' hardware needs and preferences were ascertained via a survey, the results of which are presented in this document. The information gleaned will inform the configuration discussions with the hardware vendor.

The 43 survey respondents were nearly all tenured or tenure-track faculty. Fourteen departments across three NJIT colleges were represented; Computer Science, Mathematical Sciences, and Mechanical and Industrial Engineering predominated.

The survey addressed the main elements of researchers' use of NJIT's HPC clusters: applications and compilers, amount of dependence on CPUs vs. GPUs, resources needed for their work (number of cores, amount of RAM, high-speed node interconnect, amount of storage, etc.), to what extent current resources are adequate or inadequate, and in what ways the expansion is likely to affect their research.

Results indicate that:

- CPU users (all researchers) and GPU users (about half the researchers) reported that if more resources (cores, memory, storage) were available, they would use them
- 69% use parallel processing in their applications - hence the need for ubiquitous high-speed node interconnect
- 45% of researchers affirmed that their application(s) would benefit from the availability of a parallel file system (PFS); currently there is no PFS
- About half of applications used were researchers' own code; own-code and open-source/commercial applications covered an extremely wide range of engineering and
  scientific applications
- 40% of researchers reported that the time to completion for their most compute-intensive runs was not acceptable
- In open comments, researchers emphasized increased efficiency and productivity, quicker results and more publications, more competitive training for students, and ability to address research questions requiring large amounts of data analysis that they are currently unable to effectively pursue
    - One researcher emphasized the vital importance of the much higher default storage quota for research that the expansion would make possible; although the survey did not specifically cover this, default storage quota is a major concern among faculty researchers

Section 2: Survey questions

Below are the survey questions in their entirety. The survey began with a text display of definitions, followed by a body of mandatory questions (some with sub-questions, depending on previous answers), and ended with an invitation to write in further comments.

Definitions displayed at start of survey

The current IST-managed high performance computing (HPC) clusters referred to in this assessment are:

- Lochness
  - Public-access and privately-owned nodes, both CPU and GPU
- Stheno
  - CPU and GPU nodes, owned by the Dept. of Mathematical Sciences

The expansion will include a parallel file system (PFS); currently NJIT's HPC infrastructure does not have a PFS.

A PFS provides cluster nodes shared access to data in parallel. It enables concurrent access to storage by multiple tasks of a parallel application, to facilitate high-performance through simultaneous, coordinated input/output operations between compute nodes and storage.

Questions

Demographics

1. What is your NJIT position?
   Choices: Faculty, Postdoc, Academic research staff (please describe)

   1a. If *Faculty:*
   Choices: Tenured, Tenure-track, Non-tenure-track
2. What is your college/school?
   Choices: Newark College of Engineering; College of Science and Liberal Arts; Ying Wu College of Computing; Martin Tuchman School of Management; College of Architecture and Design
3. What is your Department?
   Choices: *College-specific lists*
4. For approximately how long have you and/or your research group been using IST-managed high performance computing (HPC) resources?
   Choices: Less than 6 months, 6 to 12 months, 1+ -to 2 years, 2+ to 5 years, more than 5 years, Don't know
5. What is the general classification of computations for which you and/or your research group use IST-managed HPC?
   Choices: *Extensive multiple-answer list*
6. Please provide a brief, specific description of the computational work in [*above response*] research for which you and/or your research group use IST-managed HPC.
   Answer: *write-in*

Resource usage

7. What applications, including your own code, do you run on the Lochness and/or Stheno clusters? You can specify up to five applications.

   7a. For each application: Is this application your own code or open-source/commercial?
   Choices: Own code; open-source/commercial

   7b. If open-source/commercial, what is the name of the application?
   Answer: *write-in*

   7c. What is the application's function in your research?
   Answer: *write-in*

   7d. How important is this application for your research?
   Choices: Minimally, Slightly, Moderately, Very, Extremely
8. How often do you submit jobs to be run on the Lochness and/or Stheno clusters?
   Choices: Several times a day; Once daily; Every few days; Weekly; Monthly
9. Do you compile, or re-compile, your applications prior to processing?
   Answer: yes/no

   9a. If yes, what compilers do you use?

Answer: *write-in*

10. What are the maximum resources you typically request for your CPU applications for: Number of cores; Memory, GB/core: Storage, GB?
    Answer: *write-ins*
11. Given sufficient resources, what is the maximum amount you would request for your CPU applications for: Number of cores; Memory, GB/core: Storage, GB?
    Answer: *write-ins*
12. Do your applications(s) make use of GPUs? If yes, how many?
    Answer: yes/no/don't know; *write-in*
    12a If yes, What are the maximum resources you typically request for your GPU applications for
    Number of cores; Number of GPUs; Memory, GB/core: Storage, GB?
    Answer: *write-ins*
    12b If yes, Given sufficient resources, what is the maximum amount you would request for your GPU applications for Number of cores; Memory, GB/core: Storage, GB?
    Answer: *write-ins*
13. Do any of your applications require more than 1TB/node of RAM?
    Answer: yes/no/don't know
14. What is the maximum number of jobs you simultaneously submit to a cluster?
    Answer: *write-in*
15. Given sufficient resources, what is the maximum number of jobs you would be likely to simultaneously submit to a cluster?
    Answer: *write-in*
16. Do you require multiple nodes to run your application(s)?
    Answer: yes/no/don't know
17. Do your application(s) mainly depend on:
    Choices: Number of cores available, amount of RAM available, Disk I/O, Mixed - depends on the application; Don't know
18. What is your preference in CPU processor type?
    Choices: Intel; AMD; No preference; Unsure
19. Do your applications(s) make use of parallel processing?
    Answer: yes/no/don't know
20. Would your application(s) significantly benefit by using a parallel file system (PFS)?
    Answer: yes/no/don't know
21. Do your application(s) require a high-speed, low-latency compute node interconnect (e.g., InfiniBand) for minimally adequate performance?
    Answer: yes/no/don't know
22. What is the typical maximum time for your most compute-intensive runs to complete?
    Choices: Several minutes; several hours; about a day; several days; about a week; several weeks; more than several weeks, OTHER - please approximate
23. Is the "chosen time, above" maximum amount of time to completion acceptable?
    Answer: yes/no
    23a If no, What is the maximum time to completion that would be acceptable?
    Answer: *write-in*
24. What is the maximum amount of data you need to store per run, or series of runs, for post-processing?
    Choices: Less than a few GBs; Between a few GBs and a TB; Between a TB and a PB; more than a PB; Don't know
25. What type(s) of data do you need to store? (choose any)
    Multiple choices: Numerical; Text; Images; Video; Other-specify
26. How frequently does the data that you store need to be accessed?

Choices: Several times a day; Once a day; Every few days; Once a week; Every few weeks; Every few months; About once a year; Other - specify

27. How long does this data need to be retained?
    Choices: A few days; A few weeks; A few months; A year; Several years; OTHER
28. Other than yourself, how many individuals require access to this data?
    Choices: None; 1 to 5; 6 to 20; OTHER

Open comments

29. (Optional) Please provide comments on how this major HPC expansion is likely to affect your research.
    Answer: *Extensive write-in*

Section 3: Results

Participant responses to all survey questions are presented below. Where responses are excerpted (i.e., participants' extensive written comments) or summarized for brevity and/or clarity, complete response details are presented in the **Appendix**.

Demographics

Position, College/school, department, research area

Survey questions 1 - 4: What is your NJIT position? --College/school? --Department? How long have you been using IST-managed HPC resources?

Forty-two invitees completed the survey; an additional invitee provided only basic demographic information and final comments. Hence, while most of the reported results reflect 42 participants, a few reflect 43.

Figure 1 shows participants' positions; over three quarters of participants were tenured or tenure-track faculty.

Figure 2 shows participants' college affiliations, which were nearly equally divided among Newark College of Engineering, Ying Wu College of Computing, and the College of Science and Liberal Arts.

Figure 3 shows participants' departments within their colleges; Mechanical and Industrial Engineering, Computer Science, and Mathematical Science predominated across the three colleges.

Figure 4 shows how long participants have been using IST-managed HPC resources; nearly three quarters have used the resources for over two years, with over half of those using them for over 5 years. (*Note:* No one reported using HPC resources for 6+ months to a year.)

Figure 1

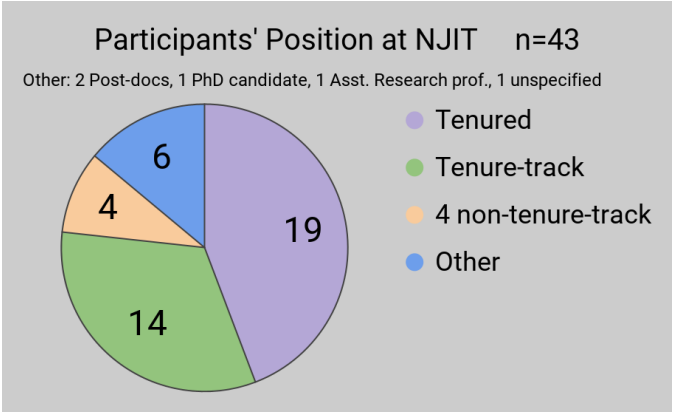


Figure 2

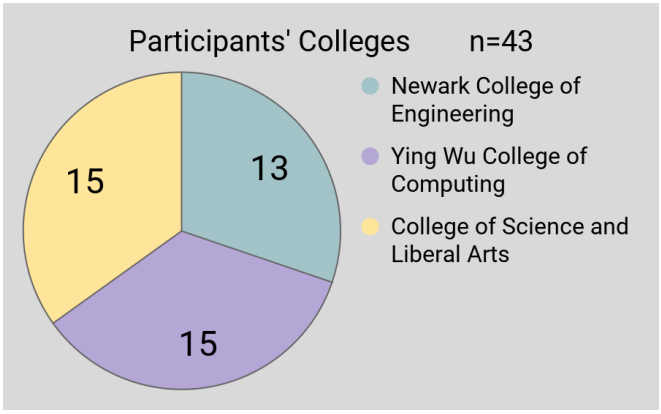


Figure 3

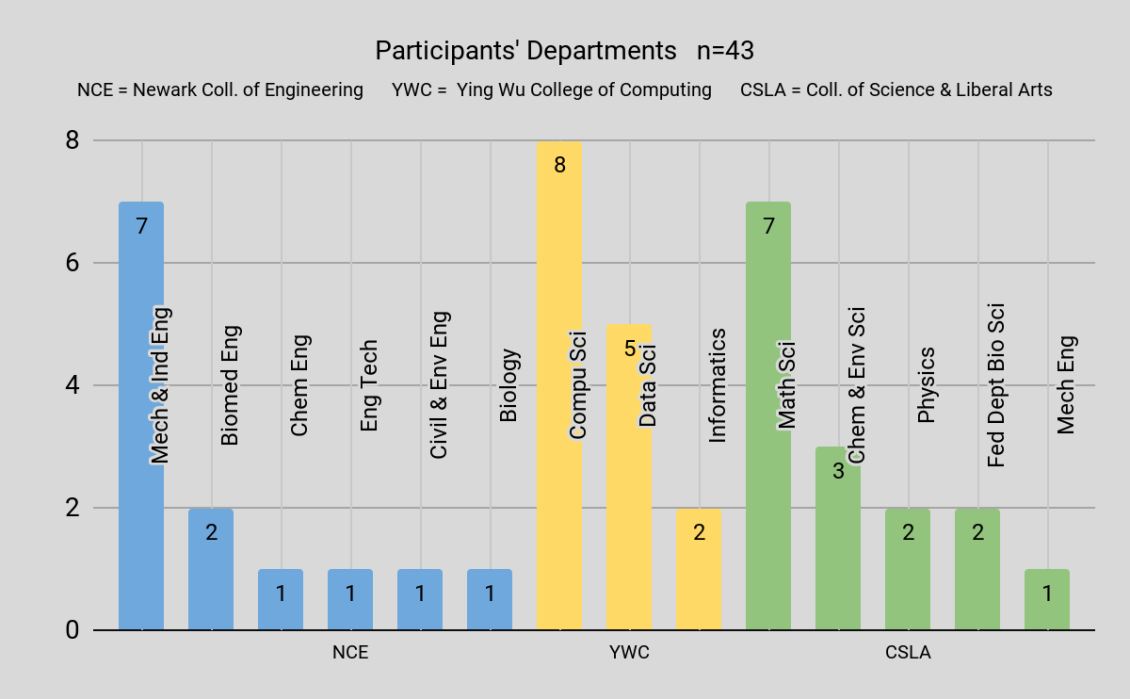
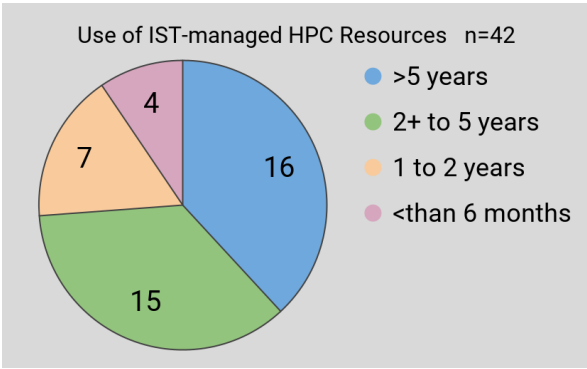


Figure 4



General and specific classifications of computations using IST-managed HPC

Survey questions 5 - 6: General classification of research area(s) using IST-managed HPC resources; Brief, specific description(s).

Participants could specify any number of *general* areas of computational use from an extensive list, followed by more *specific* write-in descriptions. These responses are listed in Table 1. Each row in the SPECIFIC column represents one entry of a specific description within a specified general area. Since many participants selected several general areas, each followed by a specific description, the number of entries (91) exceeds the number of participants (43).

(Note: A few participants repeated their specific description entries under different general classification choices. Thus, the occasional verbatim repetition of specific descriptions reflects participants' identical descriptions under different general classifications.)

Table 1

Areas of Computations using IST-managed HPC	
GENERAL	SPECIFIC
Neural networks	Our group uses extensive large amount data and images processing using neural network
	Robust machine learning with 01 loss

	We have trained deep neural networks on GPUs where the deep neural networks are used for performing classification and regression to solve problems in solar physics, space weather and climate monitoring.
	Computational neuroscience and neural networks. Simulations in CPU of coupled differential equations using Python, Brian2, Pytorch, and Matlab.
	We are developing deep neural networks specifically for graph-structured data.
	My group's use of the system has been light, but we have been expanding into neural networks that require a lot of resources. The research is on graph neural networks.
	AI, machine learning (deep learning model selection based on genetic algorithm for visual data classification, hybrid models based on deep learning and genetic algorithm, DNN hyper-parameter optimization based on genetic algorithm for object recognition and robot grasping)
	Deep Learning model training
	We have used HPC to implement many types of deep learning like LSTM, RL, and other AI methodologies like CMA-ME in order to allow game agents to learn the complexities of adversarial games.
	We solve inverse problems with genetic algorithms and perform classification with neural networks.
	Machine learning for human motion analysis
	Large natural language models
	Using generative adversarial networks (a type of deep learning) to infer the parameters of biophysical ODE models
Computational fluid dynamics	Cryogenic multiphase flow for space applications
	My research involves fluid simulations of blood flow, including cell interactions, deformation, and transport through microvascular networks. The code is one I developed, is written in Fortran, and currently uses openMP parallelism. It scales well up to 32 cores, but if nodes are available with higher core counts that would be great. With regard to typical job run-times, I have run many different types of simulations with this code with run times as small as a few hours and as large as 1-2 months (on the HPC cluster at Rutgers). This code currently runs on CPUs, but I will be porting it to run on GPUs over the next year.
	Use COMSOL Multiphysics software to perform numerical simulations of the Navier-Stokes equations.
	My research group uses computational fluid dynamics approaches, and complex systems analyses in conjunction with high-performance computing in massively parallel architectures, to develop applications in enhanced oil recovery and collective migration of bacterial cells.
	Blood flows
	Solving nonlinear PDE's mostly using finite difference methods, on both CPUs and GPUs
	Solution of PDEs by finite difference methods, spectral methods, and by integral equations. Some processing of data. Some use of Matlab and other commonly used software packages.
	We develop in-house large scale, massively parallel (MPI) fluid dynamics software
	My group develops fast and accurate algorithms to solve free and moving boundary problems in fluid dynamics other application areas
	Solving computational fluid dynamics of biofluids

	We study the flow of complex fluids. Our objective is to study their properties and develop models for their behavior. The simulations involve discretizing the governing conservation equations using a finite element technique, resulting in a system of equations. The system of equations is then solved using fast multigrid solvers.
	We perform simulations the so called GITM model which describes the dynamics of the Earth's ionosphere (via the NavierStokes equations). This allows us to study the ionospheric response to solar wind pressure changes, etc, providing a better understanding of space weather.
Statistical analysis	Using software such as PAML, we have run statistical analyses that compare patterns of molecular evolution across species. A paper using these methods, which cites the cluster and staff at NJIT is here: https://academic.oup.com/mbe/article/38/4/1372/5991404
	Machine learning with big data
	Comparing different machine learning models on different datasets
	Video analytics, AI, machine learning (Use of deep learning for statistical shape modeling, anomaly detection, Statistical process control, video analytics for visual surveillance, video Analytics on IoT devices, edge-based video analytics for real-time traffic monitoring in a smart city)
	We perform statistical analysis and uncertainty quantification from solving the inverse problem.
	Apply SA to software engineering research
	Applying new statistical and machine learning methods to large data sets.
Computational PDE	Run parameter sweeps of a PDE-solver written by the PI in Matlab. Also use COMSOL Multiphysics software to perform numerical simulations of different PDEs, typically in two space dimensions, with variable parameters.
	Solving nonlinear PDE's mostly using finite difference methods, on both CPUs and GPUs
	Solution of PDEs by finite difference methods, spectral methods, and by integral equations. Some processing of data. Some use of Matlab and other commonly used software packages.
	My group develops fast and accurate algorithms to solve free and moving boundary problems in fluid dynamics other application areas
	We solve PDEs using computational tools.
	Solving PDE of solid mechanics and fluid dynamics
Computational Biophysics	Aggregation of amyloid proteins and their interaction with lipid bilayers
	Develop biophysical models that requires parallel computation
	My research group uses computational approaches, ranging from molecular dynamics, Monte Carlo simulations, and complex systems analyses in conjunction with the theoretical formalisms of equilibrium and non-equilibrium statistical mechanics, and high-performance computing in massively parallel architectures, to develop applications in targeted drug delivery.
	Blood clot formation
	Solving biomechanics simulation problems
	Simulating systems of ODEs modeling the electrical activity of neurons and cardiac cells involved in generating circadian (~24-hour)

	rhythms
Computational physics and chemistry	Gas clathrate formation
	We have trained deep neural networks on GPUs where the deep neural networks are used for performing classification and regression to solve problems in solar physics and space weather.
	Discrete element based simulations (solving large number of coupled ordinary differential equations)
	Gas phase atmospheric and environmental chemistry, Combustion energy models.
	Non-adiabatic quantum dynamics methods to study light-induced charge transfer reaction
	We perform simulations of plasma dynamics in the Earth's radiation belts and solar wind using Particle-in-Cell code TRISTAN-MP. These simulations allow us to study different wave generation mechanisms in astrophysical and near-Earth plasmas, to study particle acceleration and scattering in a self-consistent way.
Materials research	Bio-inspired materials
	My research group uses computational approaches, ranging from molecular dynamics, Monte Carlo simulations, and complex systems analyses in conjunction with the theoretical formalisms of equilibrium and non-equilibrium statistical mechanics, and high-performance computing in massively parallel architectures, to design and optimize engineered nanomaterials for enhanced delivery of antibodies to the disease cells.
	Topological data analysis using GUDHI library
	Density Functional Theory (DFT) calculations of energy materials - capacity-voltage correlation; chemo-mechanical properties of nanomaterials and their heterostructure
	Mechanics of Materials
	First Principle
Monte Carlo	The phylogenetic software we use frequently involve Markov chain Monte Carlo searches.
	Modeling of segregative processes induced by vibrations and structural reorganization of discrete granular systems.
	My research group uses computational approaches, ranging from molecular dynamics, Monte Carlo simulations, and complex systems analyses in conjunction with the theoretical formalisms of equilibrium and non-equilibrium statistical mechanics, and high-performance computing in massively parallel architectures, to develop applications ranging from targeted drug delivery to enhanced oil recovery.
	Running multiple data simulations involving large data sizes
	Monte Carlo simulations
	We run many monte carlo simulations to obtain many results to be analyzed. We implement particle filters, simulated annealing, and other Monte Carlo methods.
Bioinfomatics	We have used the HPC to run phylogenetic searches using programs such as MrBayes and PAML. These searches work to optimize molecular (DNA) and morphological data for species to reconstruct their evolutionary relationships.
	Forecasting cancer survival time and disease risk from DNA sequence data

	Computational neuroscience and neural networks. Simulations in CPU of coupled differential equations using Python, Brian2, Pytorch, and Matlab.
	Develop deep learning methods for analysis of genomic data, which requires the use of GPU, and a large number of CPU. The genomic data is also typically large, it is common to need a few terabytes.
	Actually it is MEDICAL INFORMATICS. We are processing large medical terminologies.
Computational Chemistry	Quantum mechanical calculations using mainly Density Functional Theory, towards the study of catalytic systems and general organic chemistry mechanisms.
	Update molecules position and velocities based on force/interaction between each atom, for each time interval
	Combustion, Environment and Atmospheric Chemistry, Global Warming
Biophysics	Cell migration
	Computational neuroscience and neural networks. Simulations in CPU of coupled differential equations using Python, Brian2, Pytorch, and Matlab.
Condensed matter physics	Discrete element based simulations (solving large number of coupled ordinary differential equations)
	Density Functional Theory (DFT) calculations of density of states (DOS), band structures of various nanomaterials
Electro-magnetism, Wave propagation	The above PDEs are for modeling wave propagation.
	We perform simulations of the whistler wave propagation in the Earth's ionosphere (full wave model) and magnetosphere (ray tracing). Whistler waves play important role in the electron dynamics in the inner magnetosphere, so it's important to understand their propagation from the source region (in particular, VLF transmitters, lightnings)to the interaction region. Our simulations help us to better understand these processes, including wave energy transmission from the lower ionosphere to magnetosphere and in the opposite direction.
Granular science	CPU-intensive simulations (discrete granular dynamics and Monte Carlo) of important phenomena in granular science, such as the transmission of waves through granular system and density relaxation promoted by external impulses.
	Discrete element based simulations (solving large number of coupled ordinary differential equations)
Image forensics	Video analytics, AI, machine learning (Use of CNNs in source camera identification, recaptured image forensic, computer graphics image forensic, GAN-generated image detection, source social network identification)
Software verification, static analysis	Analyze billions lines of code for locating bugs/vulnerabilities
Steganalysis and image forensics	Video analytics, AI, machine learning (Use of CNN for feature learning for steganalysis, multi-domain feature learning for CNN-based image steganalysis, use of CNN for real-time spatial steganalysis)
Transportation	Video analytics, AI, machine learning (Use of deep learning for traffic safety solution, traffic incident and anomaly detection, real-time collision prediction system, deep learning based edge traffic flow prediction, Video-based ITS vehicle illegal activity detection, Vehicle

data analysis	detection and classification based on DCNN)

OTHER areas	
OTHER GENERAL	OTHER SPECIFIC
Computational Biomechanics	Running finite element analyses and probabilistic simulations
High Performance Data Analytics (HPDA)	Large-scale graph algorithms and combinatorial computing
Deep neural networks for medical image segmentation and medical diagnosis	Data pre-processing, training machine learning programs on data, data post processing
Computational neuroscience and neural networks. Simulations in CPU of coupled differential equations using Python, Brian2, Pytorch, and Matlab.	Computational neuroscience and neural networks. Simulations in CPU of coupled differential equations using Python, Brian2, Pytorch, and Matlab.
Graph data analytics	We use and expanse an open source framework Arkouda to handle large graphs.
Social Network Analysis.	We are processing large amounts of Twitter Data. We would like to expand to Reddit.
Thermochemical properties of molecules and element. Kinetic models of fuel, atmospheric, and industrial chemistry	Development of thermochemistry properties of molecules and molecular systems, Reaction kinetics as function of temperature for processes in the earths atmosphere and modeling of industrial chemical processes
Performance evaluation	Performance evaluation
Simulating ODE models of the dynamics of genes and proteins that constitute the molecular circadian clock	The general field I work in is computational or mathematical biology; the computational work involves simulating ODEs (sometimes PDEs) and inferring parameters for the models using optimization, data assimilation, and machine learning.

Resource Usage

Applications

Survey questions 7 - 9: Applications, including participant's own code, on Lochness and/or Stheno clusters; Frequency of job submissions to be run on Lochness and/or Stheno clusters; Compiling/re-compiling applications prior to processing.

Participants provided up to five applications, specified as either "own code" or "open-source/commercial", that they run on the Lochness and/or Stheno clusters. They described each application's function, and rated its importance, (*Extremely, Very, Moderately, Slightly, Minimally*), and provided the name of open-source/commercial applications.

Forty-three "own code" applications and 31 "open-source/commercial" applications were reported. As can be seen in Figures 5a and 5b, participants overwhelmingly rated applications, whether own-code or open-source/commercial, as *very* or *extremely important*, with *extremely important* accounting for nearly two thirds of all ratings. No applications were rated as *minimally important*.

Survey question 7, Table i in the **Appendix** lists **all reported applications,** along with their names (if open-source/commercial), functions, and importance ratings.

Figure 5a

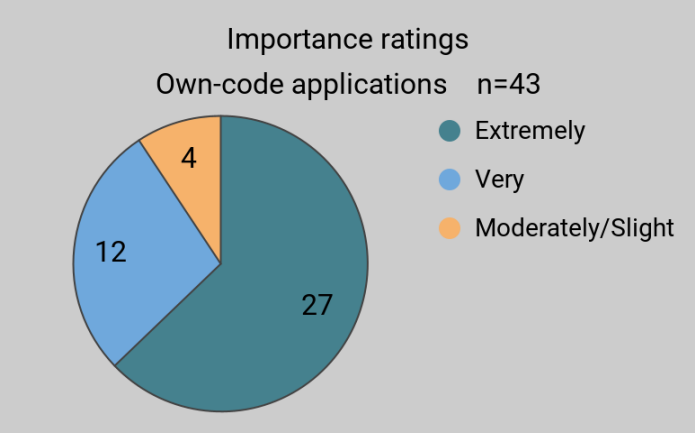
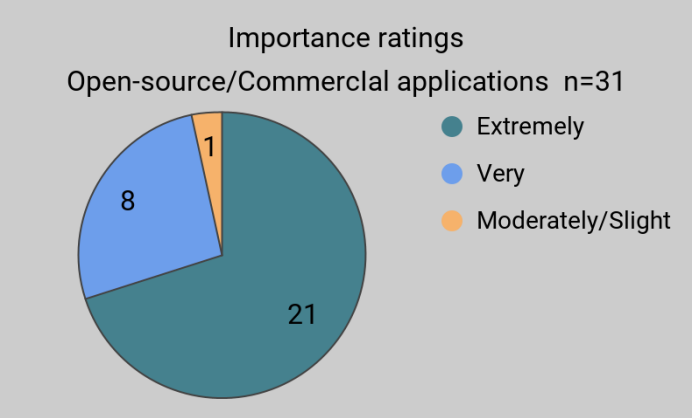


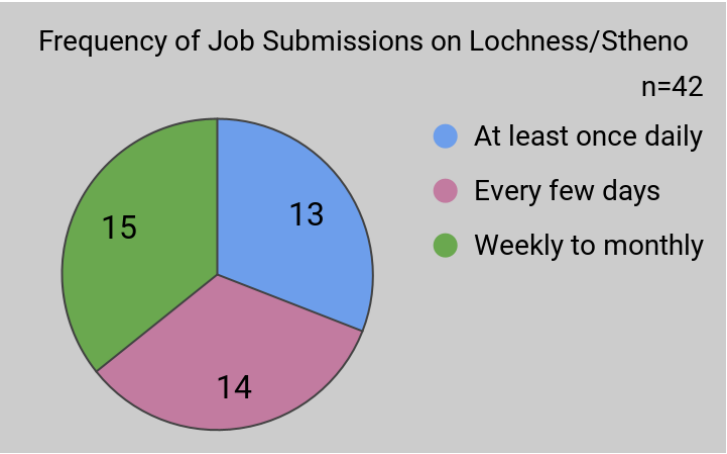
Figure 5b



Frequency of job submissions

Figure 6 shows how often participants submit jobs to be run on Lochness and/or Stheno clusters; about a third submit once daily (once to several times), a third submit every few days, about a third submit weekly to monthly. *Survey question 8, Figure i* in the **Appendix** provides a finer-grained breakdown of job submission frequency.

Figure 6



Compilers

Nineteen out of 42 participants indicated that they compiled or re-compiled their applications prior to processing. Table 2 lists the 13 compilers whose names were provided by those 19 participants. Several participants listed more than one compiler; hence, the total number of users exceeds 19.

Table 2

Compiler	Number of users	Compiler	Number of users	Compiler	Number of users
C	1	Gnu C	4	Intel suite	2
C++	3	Gnu C++	1	Intel-19.1.0.166 suite	1
Fortran	5	GNU compiler suite	2	Matlab	1
Fortran 90	1	Intel ifort	2	PGI suite	1
				TensorRT	1

Resource usage (current versus desired) for CPU applications

Survey question 10 - 11: Maximum resources typically requested for CPU applications for: Number of cores; Memory, GB/core; Storage, GB. Desired maximum amount given sufficient resources.

Participants provided the maximum resources that they typically request for their CPU applications, followed by what their requests would be given sufficient resources (‘desired” requests).

The paired figures-- 7a and 7b, 8a and 8b, 9a and 9b-- show, respectively, the amounts for the typically requested and desired requests for Number of cores; Memory, GB/core; and Storage, GB.

Note that while the y-axis scales are identical across pairs for ease of quantitative comparisons, the x-axis bar label quantity ranges are not. This reflects the differences in how the data fell into quantitative groups across actual typical requests and hypothetical desired requests. (Responses were *write-in*; participants did NOT choose among pre-set quantity ranges.)

As can be seen in the paired figures, there is a consistent trend of a greater number of participants *typically requesting* relatively small amounts of resources than *desiring* small amounts of resources, and a greater number of participants *desiring* relatively large amounts of resources than *typically requesting* relatively large amounts of resources. In addition, the largest desired request exceeds the largest typical request for each paired comparison: The largest *Number of cores typical request* is 640, while the largest *Number of cores desired request* is 1400; the largest *Memory, GB/core typical request* is 264, while the largest *Memory, GB/core desired request* is 1000; and the largest *Storage, GB desired request* is 20,000 while the largest *Storage, GB desired request* is 50,000.

Figure 7a

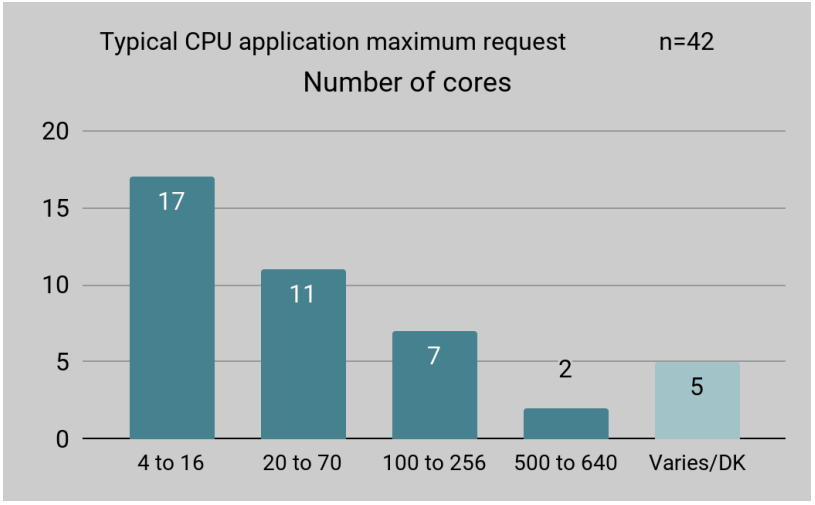


Figure 7b

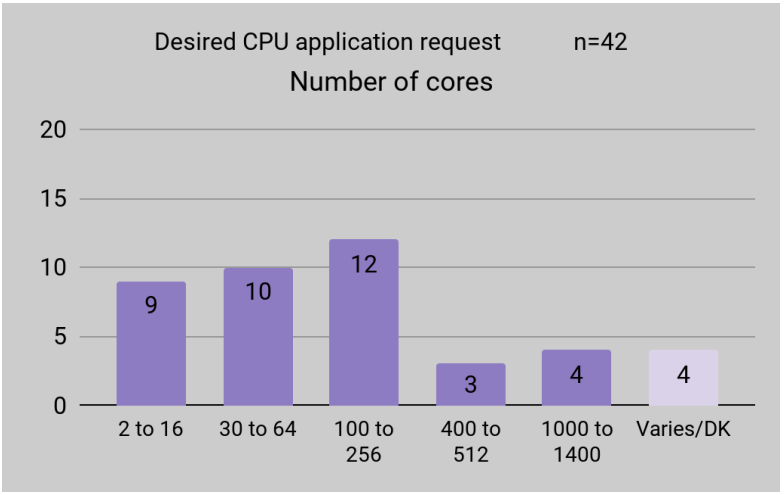


Figure 8a

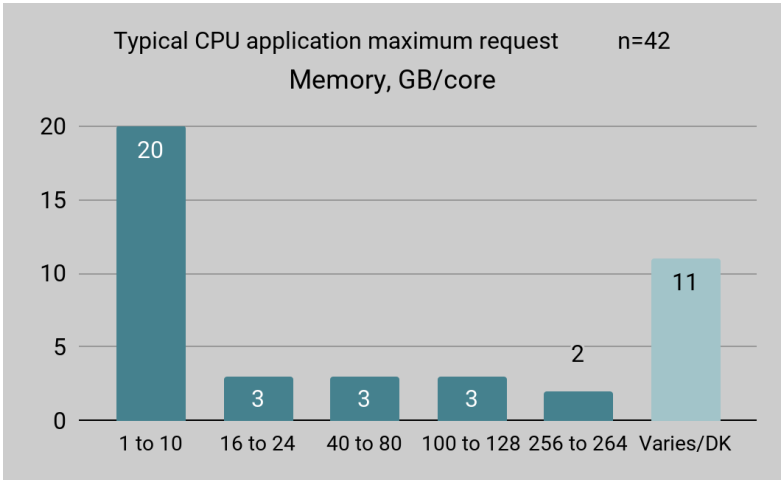


Figure 8b

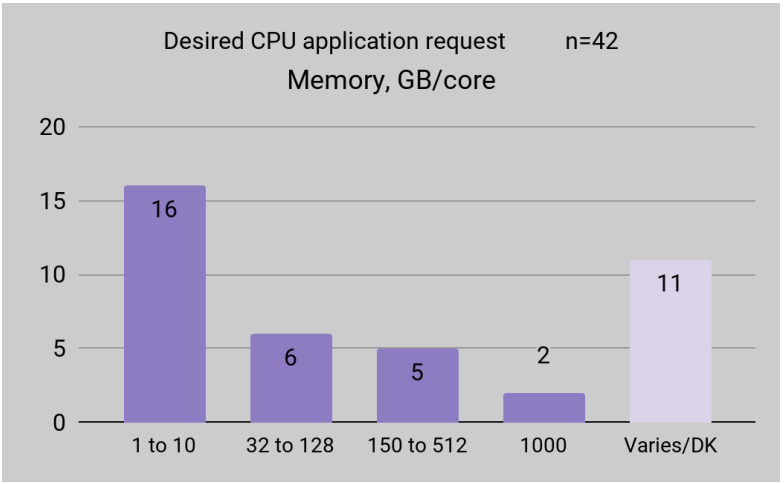


Figure 9a

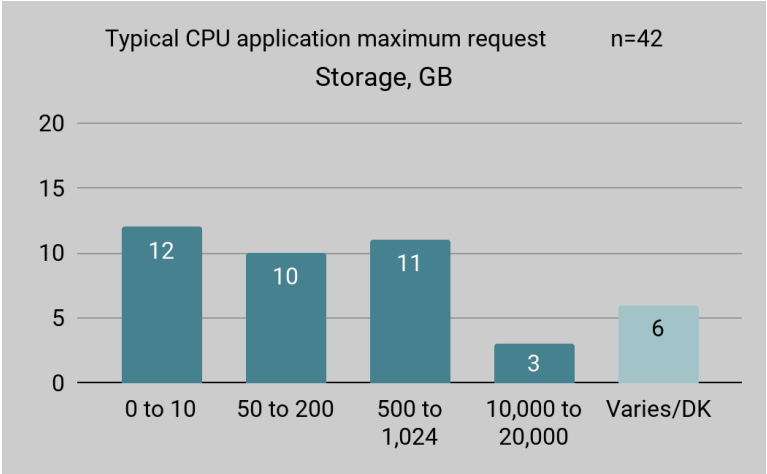
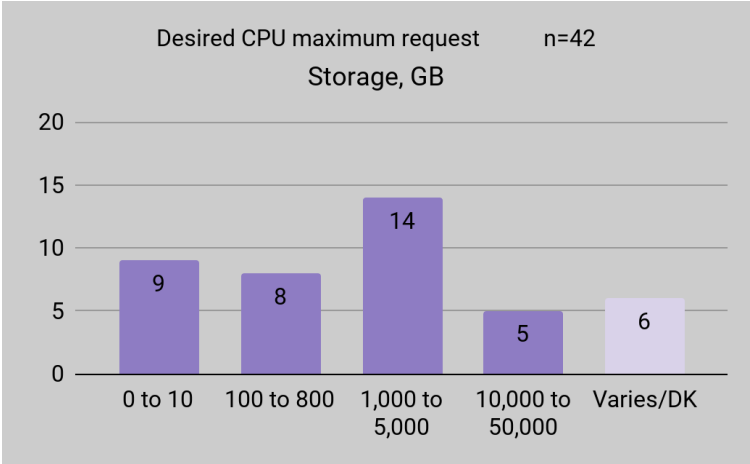


Figure 9b



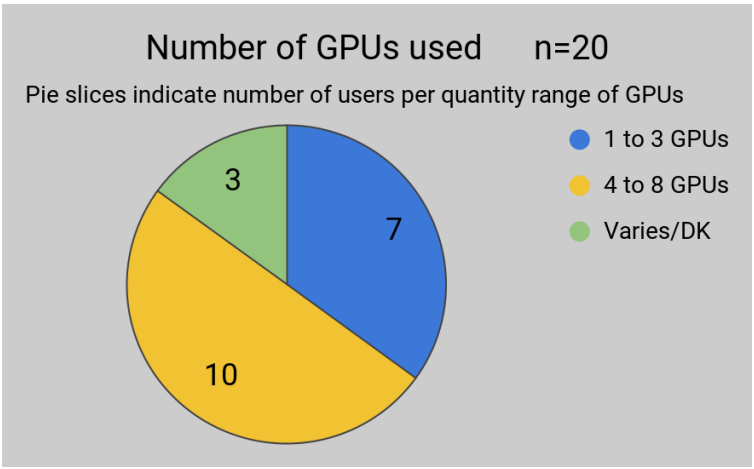
Resource usage (current versus desired) for GPU applications

Survey questions 12 - 14: Applications(s) make use of GPUs-- how many? Maximum resources typically requested for GPU applications for: Number of cores; Memory, GB/core; Storage, GB. Would-be maximum amount given sufficient resources.

Twenty out of 42 participants reported that their applications make use of GPUs; 17 reported not using GPUs, and 5 did not know.

Figure 10 shows reported GPU use from the 20 participant users; half (i.e., 10) reported using 4 through 8, and 7 used a maximum of 3, and the remaining 3 either used varying amounts or did not know. *Survey question 12 figure ii* in the **Appendix** shows a finer-grained breakdown of quantities of GPU use

Figure 10



As with the 42 participants reporting on CPU usage (above), the 20 participants who reported using GPUs provided the maximum resources that they *typically request* for their GPU applications, followed by what their requests would be given sufficient resources (*desired requests)*.

The paired figures-- 11a and 11b, 12a and 12b, 13a and 13b-- show, respectively, the amounts for the typically requested and desired requests for Number of cores; Memory, GB/core; and Storage, GB. Note that, as with the CPU use data reported above, while the y-axis scales are identical across pairs for ease of quantitative comparisons, the x-axis bar label quantity ranges are not. Again, this reflects the differences in how the data fell into quantitative groups across actual typical requests and hypothetical desired requests, reported as written-in quantities.

As can be seen in the paired figures, the largest desired request exceeds the largest typical request for *Number of cores* and for *Memory GB/core*: 10,000 versus 5000 and 1000 versus 160, respectively. For *Storage, GB, w*hile the maximum request (20,000) was identical for both typical and desired requests, a greater number of participants reported higher-quantity desired requests: 5 participants reported typical requests between 1000 and 20,000, while 9 participants reported desired requests in that range. In contrast with the CPU usage data, at least a quarter of the 20 GPU-users responded “Varies/Don't know” across all three paired queries.

Figure 11a

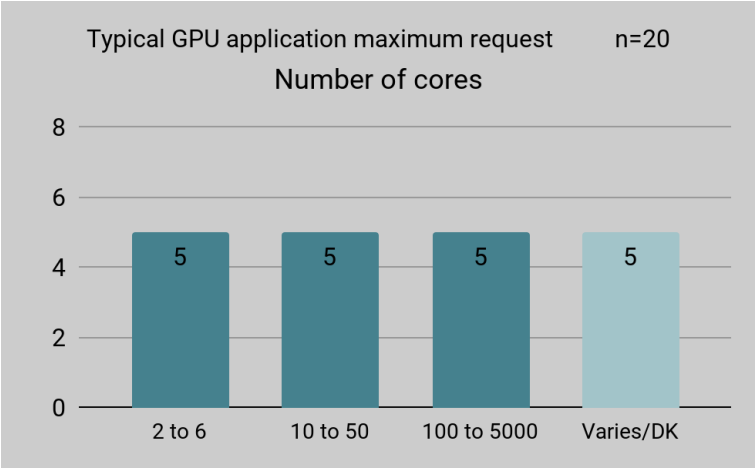


Figure 11b

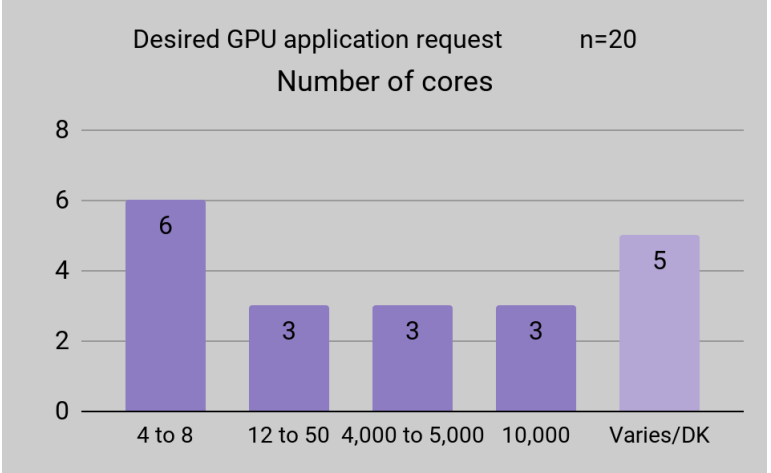


Figure 12a

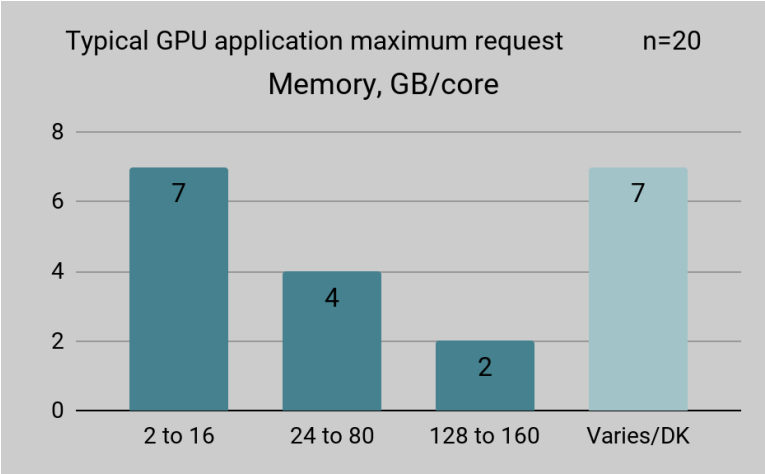


Figure 12b

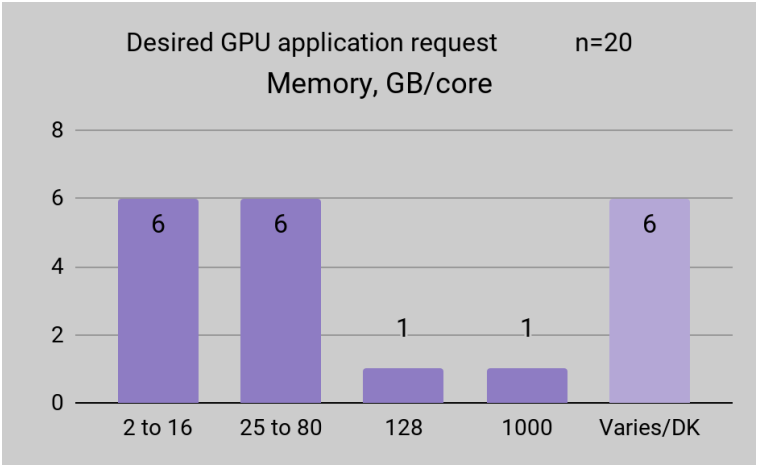


Figure 13a

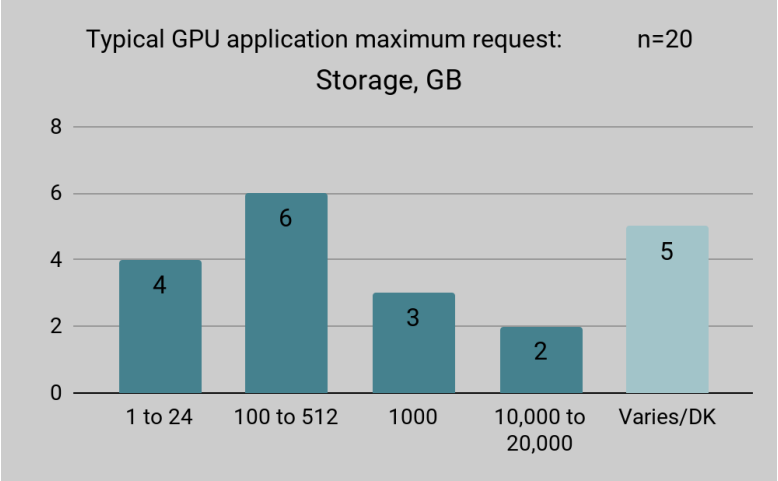
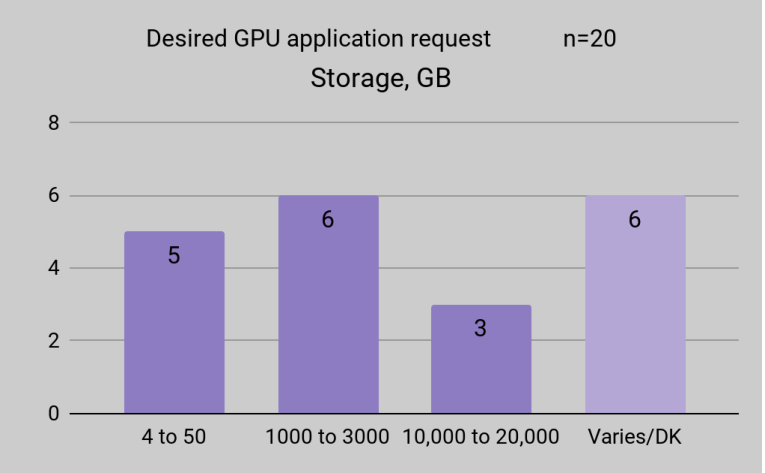


Figure 13b

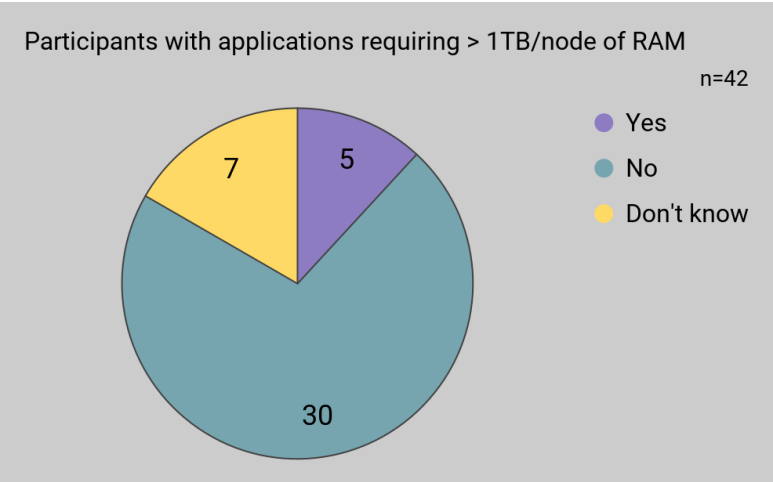


Various

Survey question 13 - Any applications require more than 1TB/node of RAM?

Large RAM usage: About 12% of the 42 participants reported applications requiring more than 1TB/node of RAM; see Figure 14.

Figure 14



Survey questions 14 and 15 - Maximum number of jobs simultaneously submitted to a cluster; Desired maximum number given sufficient resources.

Figure 15a shows the typical maximum number of jobs that participants reported simultaneously submitting to a cluster; Figure 15b shows the desired number of job submissions given sufficient resources. As with other paired bar graphs in these results (above), the two figures are identically scaled on the Y axis, while the numerical ranges on the x axis reflect the clustering in the written-in data. The figures show that nearly three times as many participants would like to submit at least 100 jobs (that is, 11 participants' *desired* report) than they actually submit (that is, 4 participants' *typical* report). In addition, the highest desired maximum number is 1024, in contrast to the highest typical maximum of 400.

Figure 15a

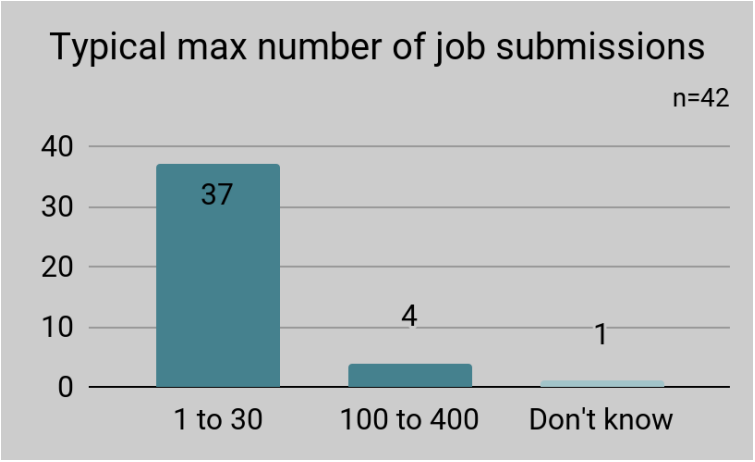
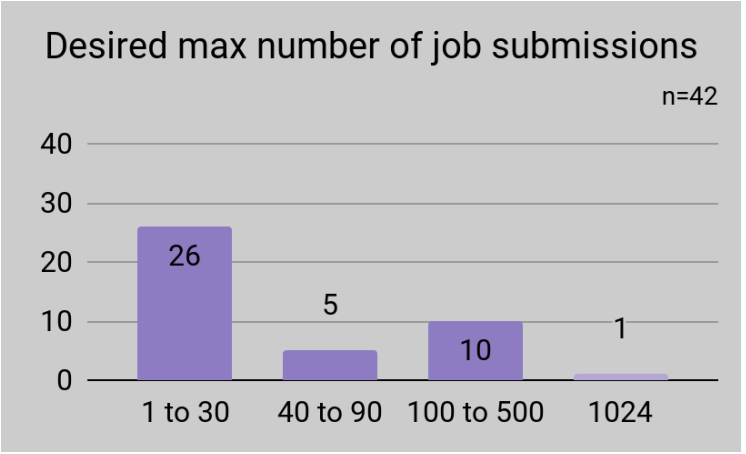


Figure 15b



Survey questions 16 and 17 - Require multiple nodes to run your application(s)? Mainly depend on available cores, RAM, mixed, Disk I/O?

Figure 16 shows the number of participants who reported requiring multiple nodes to run their application.
Figure 17 shows what resources participants' applications depended on. Note that in addition to Number of cores available, Amount of RAM available, and Mixed, participants could also choose Disk I/O-- but none chose this option.

Figure 16

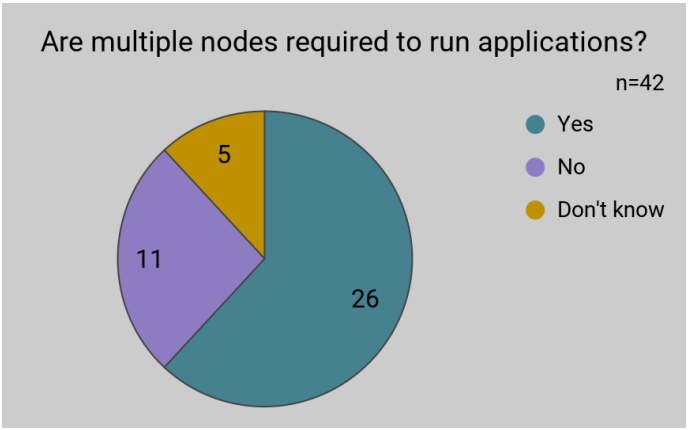
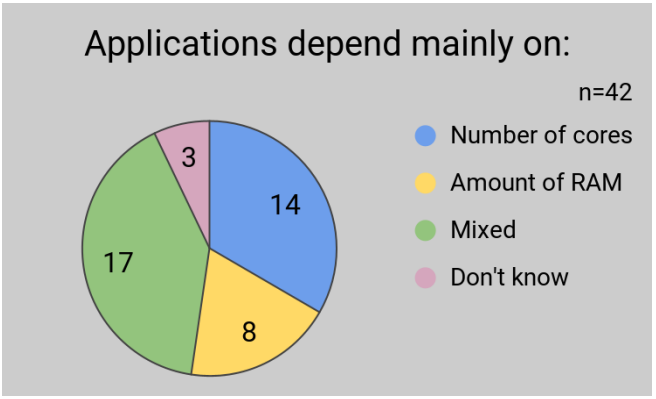


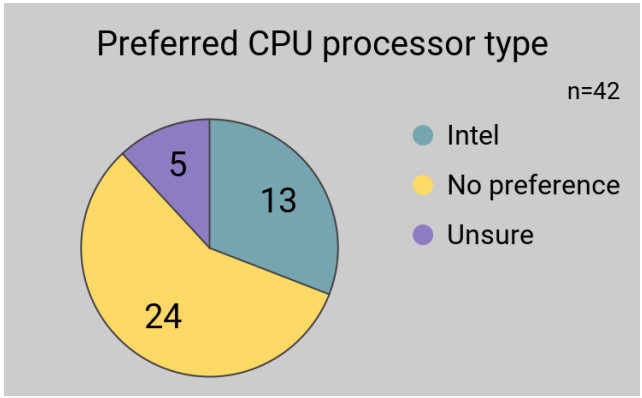
Figure 17



Survey question 18: Preference in CPU processor type

Figure 18 shows participants' preference in CPU processor type between Intel, AMD, No preference, and Unsure. Intel was the only selected processor type; none chose AMD, and over half had no preference.

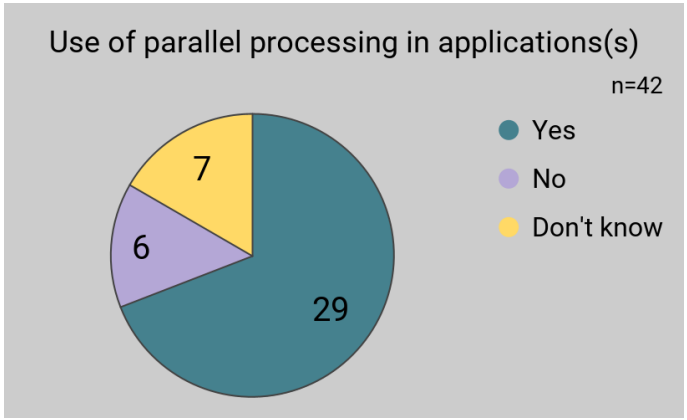
Figure 18



Survey question 19: Applications(s) make use of parallel processing?

Figure 19 shows that over two thirds of participants (29 out of 42) reported the use of parallel processing in their applications(s).

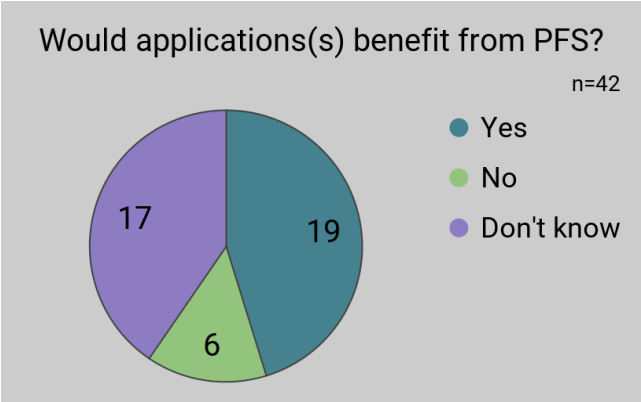
Figure 19



Survey question 20: Would your application(s) significantly benefit by using a parallel file system (PFS)?

Figure 20 shows that nearly half of the participants reported that their application(s) would benefit by using a parallel file system (PFS). Note that nearly as many did not know; only 6 out of 42 specified that their application(s) would **not** benefit.

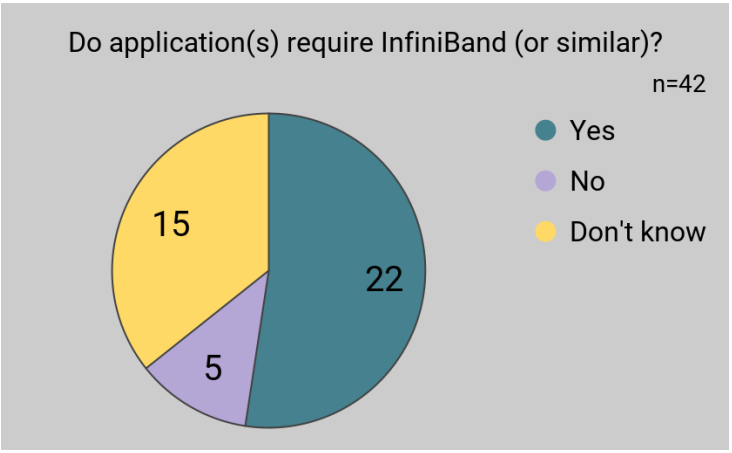
Figure 20



Survey question 21: Do your application(s) require a high-speed, low-latency compute node interconnect (e.g., InfiniBand) for minimally adequate performance?

Figure 21 shows that over half (22 out of 42) of the participants reported that their application(s) require a high-speed, low-latency compute node interconnect (e.g., InfiniBand) for minimally adequate performance. Note that 15 out of 42 did not know; only 5 out of 42 specified that their application(s) did **not** require a high-speed, low-latency compute node interconnect.

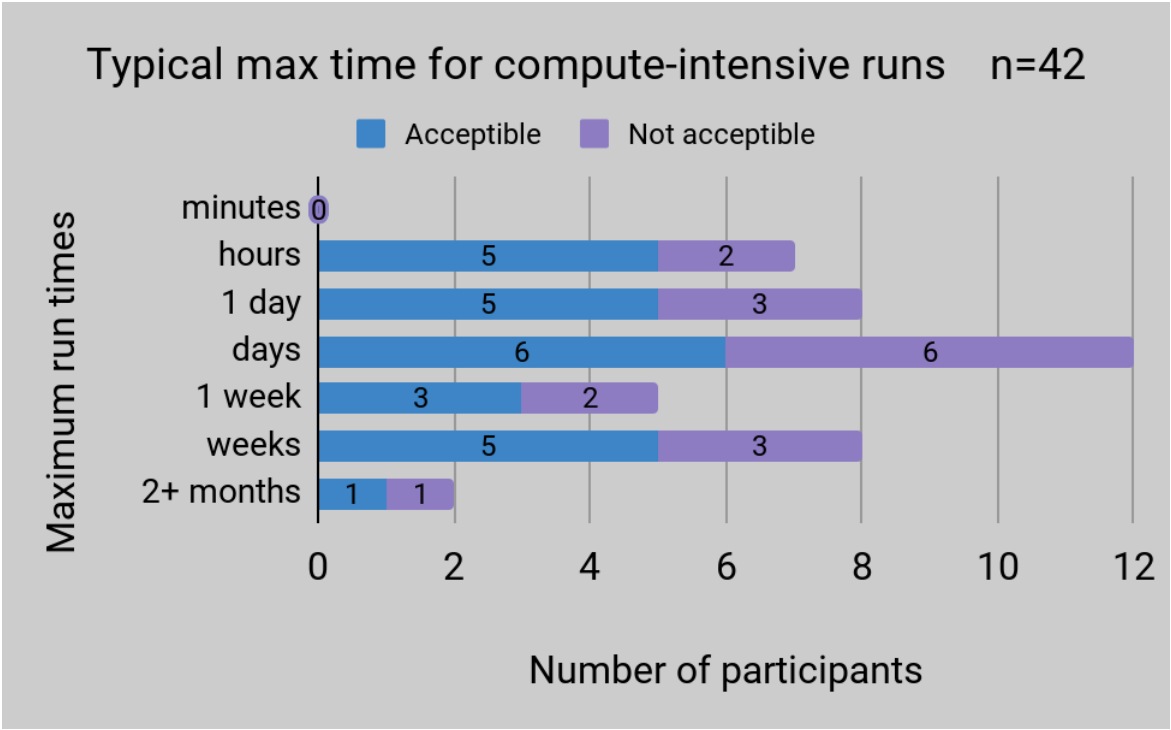
Figure 21



Survey question 22 - 23: Typical maximum time for most compute-intensive runs to complete; Is time to completion acceptable? If not, what would be acceptable?

Participants indicated the typical maximum time for their most compute-intensive runs to complete (selecting among *Several minutes; several hours; about a day; several days; about a week; several weeks; more than several weeks, Other - please approximate*) and then reported whether whether that time-to-completion was acceptable, and if not, provided (via write-in) what would be acceptable. Figure 22 shows the number of participants who indicated each time-to-completion, divided into those who were satisfied with that time (Acceptable) and those who were not (Not acceptable). The desired times provided by those participants who found their current time-to-completion not acceptable is listed in Survey question 23a, Table *ii*, in the **Appendix**.

Figure 22



Data storage; retention and access

Survey questions 24 and 25: What is the maximum amount of data you need to store per run, or series of runs, for post-processing? What type(s) of data do you need to store?

Figure 23 shows the maximum data that participants need to store; two thirds reported needed between a few GBs and 1 TB. Figure 24 shows the type of data to be stored; note that participants could indicate multiple types of data, hence the number of responses (75) exceeds the number of participants (42). Numerical was the most prevalent.

FIgure 23

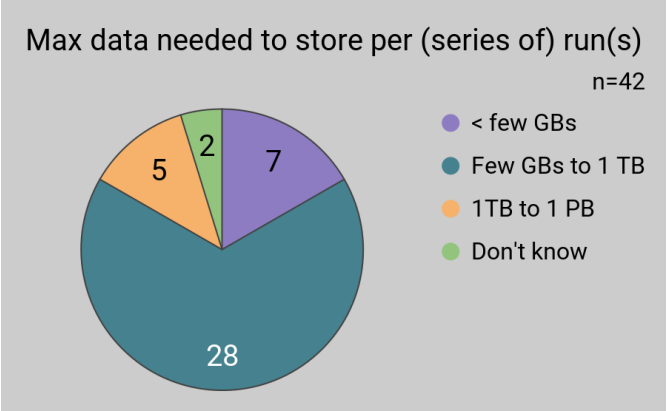
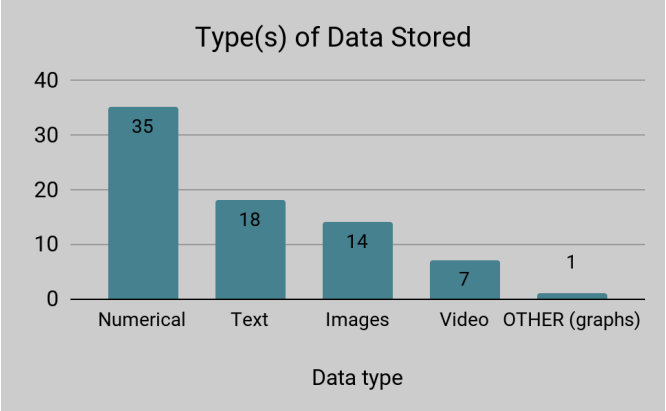


Figure 24



Survey questions 26 and 27: How frequently is stored data accessed? How long retained?

Figure 25 shows how frequently participants need to access stored data; over three quarters accessed data at least as often as once a week.

The five OTHER responses were: (1) Depends on application - could be daily, or weekly; (2) Down load data - need to store 1 to several weeks; (3) Post-processing when case studies have been completed - typically several months; (4) When the job is running; (5) Once, during analysis. Survey question 26 Figure *iii* in the Appendix shows a finer-grained break-down.

Figure 25 shows how long data need to be retained. The three OTHER responses were: (1) At least a year with the option to archive it for possible re-processing; (2) Various; (3) Depends on the job and result. Survey question number 27 Figure *iv* in the Appendix shows a finer-grained break-down.

Figure 25

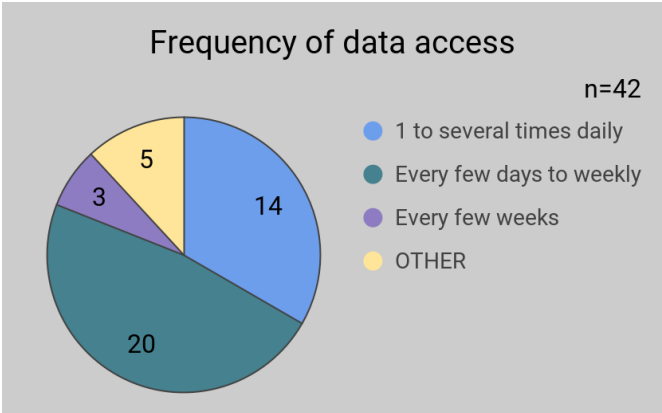
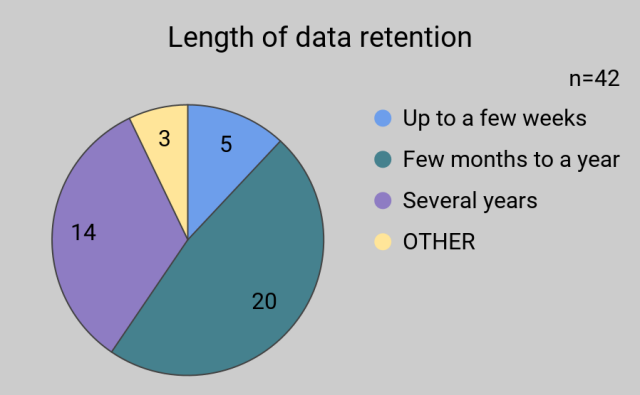


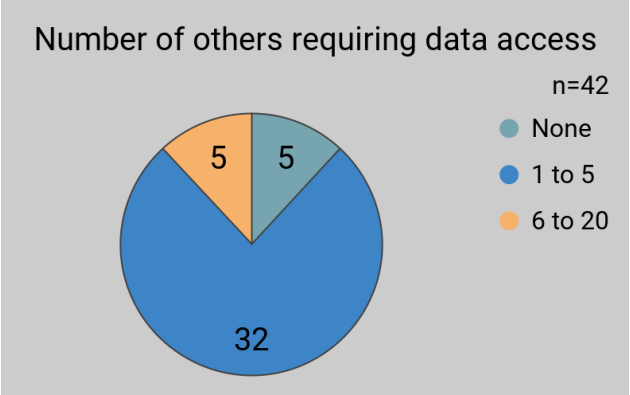
Figure 27



Survey questions 28: Other than yourself, how many individuals require access to this data?

Figure 26 indicates that the majority of participants report between 1 and 5 other individuals, in addition to themselves, requiring access to their stored data. None chose OTHER.

Figure 27



Open Comments

Survey question 29: (Optional) Please provide comments on how this major HPC expansion is likely to affect your research.

Twenty-six participants wrote in comments. Table 3 shows a **representative sample** selection, including comments from each of the three colleges represented. The complete comments are listed in Survey questions 29, Table *iii*, in the **Appendix**.

Table 3

Sample COMMENTS	
Participant's college	
Participant's Department	*Participant's comment*
College of Science and Liberal Arts	
Department of Mathematical Sciences	An HPC expansion will allow me to conduct numerical simulations far more effectively and efficiently. If jobs are completed with a higher speed, I can update my models faster and more effectively, resulting in better papers being published with greater frequency. This will also result in more successful grant proposals. I am currently often limited by the rather slow speed and scarce resources of the cluster.

Department of Physics	This expansion will have a tremendous impact on my research program. A significant increase in the number of CPU and GPU nodes will allow us to study phenomena that were previously inaccessible to us. These phenomena take place within living cells and they involve tracking the position of ~1,000,000 atoms over several 10-100 microseconds.
Ying Wu College of Computing	
Computer Science	It will affect our research significantly. In general, the more computing resources become available, the more advanced deep learning models we can try, test and use.
Computer Science	My current research has not been using Lochness so much. Students have been outsourcing their work to Google Colab (Pro). However, demands are rising and having the possibility to train simultaneously multiple neural networks of moderate size would be great to have.
Newark College of Engineering	
Mechanical and Industrial Engineering	Will allow us to actually get data needed to investigate the problems of interest. (The inability to do this has a ripple effect, i.e., few papers produced, low visibility in the community, difficulty in attracting external funding and good students, and the loop continues). Research computing resources have greatly improved over the time I have been here (+ 30 years), but they are still far below what they should be for a university that is marginally a "Research I" institution. NJIT is well overdue in allocating funding to improve its research computing infrastructure.

Section 4: Appendix

Survey question 7: (a) What applications, including your own code, do you run on the Lochness and/or Stheno clusters?

Table *i* lists all reported applications and their importance ratings in response to survey questions 7a-d.

Own-code applications and *open-source/commercial* applications are listed separately. Each table row represents one entry by an individual participant, with each participant having up to five separate entries. *Own code* and *open-source/commercial* application lists are each organized by importance rating (left-most column), such that *Extremely* important applications are listed first and *Slightly* important applications listed last. *Open-source/commercial* applications are alphabetized by name within each importance rating group. Note that this occasionally results in examples of a multiply-listed application (e.g., LAMMPS) being separately listed under different importance rating groups.

Table *i*

Own Code	
Importance	**Description of use**
Extremely	Streaming graph analytics
	Fluid flow simulations of blood cells and cancer cells through complex microvascular networks in 3D.

	This is Matlab code to solve both ordinary differential equations and partial differential equations. These equations describe processes that arise in fluid dynamics and biophysics. The output of this code is used to uncover new physics, and is typically relayed to experimental collaborators and other computational scientists. The results are then written up for publication and used in grant proposals.
	biophysical model
	Granular dynamics (i.e., discrete element) code
	Image processing code using RNN
	Study robust machine learning models based on 01 loss
	TSInet: Reconstruction of Total Solar Irradiance by Deep Learning
	cfd
	Most of this is code developed by doctoral students for their thesis work, plus some (occasionally) of my own.
	Climate modeling
	Boundary integral
	For materials simulation
	Lochness and stheno are suitable tools for addressing my needs.
	we develop large models for analyzing billions lines of code
	Simulating ODE models of biological systems
	Non-adiabatic quantum dynamics methods to study light-induced charge transfer reaction
	The code solves the Hamiltonian equations for the particle dynamics in a given field. We use this code for test-particle simulations of electron acceleration and scattering by whistler waves in the Earth's radiation belts.
	Lymphatic system transport of cancer cells and/or tumor-derived chemical species
	Segmentation and classification of medical images
	I use Fortran, C++, and python to create a multiscale computational framework to address problems related to health, environment, and energy.
	SolarUnet: Identifying and Tracking Solar Magnetic Flux Elements with Deep Learning
	LSTM-flare-prediction: Predicting Solar Flares Using a Long Short-term Memory Network
	Data assimilation and other optimization techniques for inferring parameters of biophysical ODE models from data
	CNNStokesInversion: Inferring Vector Magnetic Fields from Stokes Profiles of GST/NIRIS Using a Convolutional Neural Network
	RNN-CME-prediction: Predicting Coronal Mass Ejections Using SDO/HMI Vector Magnetic Data Products and Recurrent Neural Network
Very	Finding mechanisms that explain biophysics behind neurons and networks in neuroscience. Applications could include understanding of certain pathologies such as epilepsy but also cerebral functions.
	fortran based finite difference code for solving nonlinear PDEs
	Deep Learning on Graphs
	Deep autoencoders
	Video analytics, AI, machine learning (Core components of real-time video analytics)

	Research code for biomechanics
	Performance evaluation
	We conduct simulations which are then tested against experimental data as well as previous results.
	This is code written in COMSOL Multiphysics, which solves the partial differential equations of fluid dynamics. The results are then relayed to experimental collaborators. The results are then written up for publication and used in grant proposals.
	Discrete element simulations
	Numerical method development for flows involving complex interfaces
	The code solves Vlasov-Maxwell equations for the evolution of the particle distribution function in gyro-kinetic approximation. We use it for studying electron hole dynamics in space plasmas.
Moderately	The code solves Maxwell equations for the monochromatic wave propagation in the Earth's ionosphere.
	Monte Carlo simulation code developed by myself in the 1980's and modified over the course of many years.
	GPU-accelerated computing for immersed boundary method simulations
Slightly	C programs to be called from R

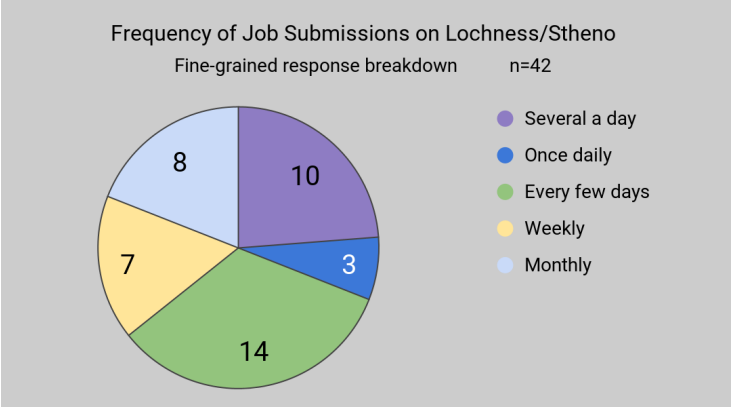
Open-Source or Commercial

Importance	Name	Description of use
Extremely	Arkouda	a framework to support high performance computing using Python
	Arkouda	Large scale data analytics
	Cryogenics	Study the physics of multiphase cryogenic flows in low gravity
	DeepHM (provided by IBM Research)	Use deep learning to infer parameters of mechanistic ODE models
	Gaussian - Computation Chemistry Calculations	Molecular Structure, Thermochemical Properties: Enthalpy of Formation, Entropy, Heat Capacities; Reaction Kinetics in the Earths Atmosphere and In Combustion, Reaction Chemistry in Industrial Processes
	Gaussian 16	Perform DFT calculations (optimizations of geometry, calculations of energy, frequencies, and other properties.
	GITM	The code solves a complex set of equations describing global dynamics of the ionosphere.
	GROMACS	Molecular dynamics simulations
	Kinetic Calculations	Rate constants of chemical reactions
	LAMMPS	The dynamic behavior of chemical molecules/atoms
	LAMMPS	Molecular dynamics simulations
	LAMMPS	Molecular Dynamics
	Matlab, GMRES, other commonly available	To supplement other user-written code

	nuts and bolts software packages - ODE solvers, matrix manipulation etc.	
	MrBayes	Phylogenetic Reconstruction
	optimization of neural networks for inference	Edge computing
	ParaView	It is an open-source, multi-platform application designed to visualize data sets.
	Python & ML Packages	Machine Learning
	R	Using the built-in functions plus programming features
	Sabberstone	Simulate a video game as well as AI agents
	TRISTAN-MP	The code solves Vlasov-Maxwell equations with Particle-in-Cell approach.
	VASP	Density Functional Theory
Very	Abaqus	Finite element software
	High Energy Finishing	Studying the flow of water and ceramic particles in a highly rotating capsule
	IMPETUS - AFEA	Discrete finite element code obtained from CertaSim. Code runs on a stand-along GPU in the Granular Science Laboratory. The software is state-of-the art with enormous capabilities to study rapid dynamic processes.
	LAMMPS	It is a classical molecular dynamics code with a focus on materials modeling
	Machine learning of graph structure of Medical Terminology	Predict placement of new concepts in network.
	PaML	Phylogenetic Reconstruction and Comparative Methodology
	PyTorch	Using PyTorch to run custom-made neural network models.
	Social Network Analysis	Learn the structure of Tweets of trolls and bad actors.
Slightly	ORCA	Perform other DFT calculations that are not yet implemented in Gaussian 16
	Python	Data analysis

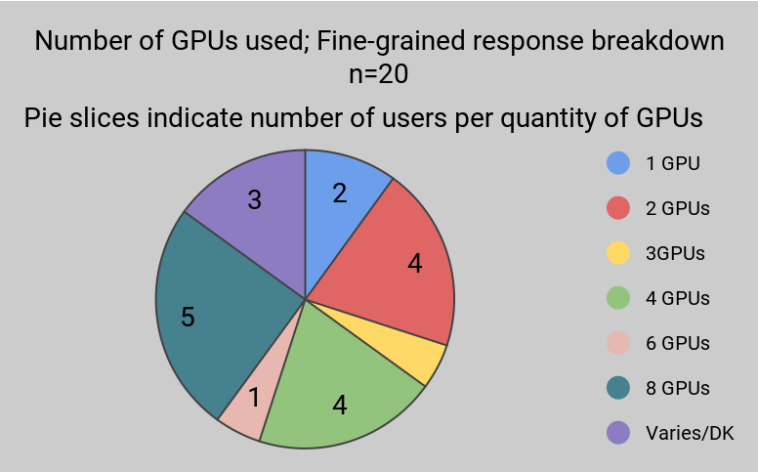
Survey question 8: How often do you submit jobs to be run on the Lochness and/or Stheno clusters? Figure *i* shows fine-grained results.

Figure *i*



Survey question 12: If your applications(s) make use of GPUs, how many do you use? Figure *ii* shows fine-grained results.

Figure *ii*



Survey question 23: I*f the typical maximum time for your most compute-intensive runs to complete is not acceptable, what amount of time would be?* Results are listed in Table ii. Note that a few of the 17 participants who considered their maximum run times unacceptable proposed run times that were longer rather than shorter; for instance, one participant whose maximum run time was “one day” responded with “1 week”. It is possible that some participants took the question to mean “how much time would you like to be allotted” rather than “how short would you like your turn-around time to be”.

Table *ii*

Compute-intensive Run Times			
Maximum run time	Number of participants deeming run time ACCEPTABLE n=25	Number of participants deeming run time NOT ACCEPTABLE n=17	Acceptable run times -- write-in responses from participants deeming run time NOT ACCEPTABLE
Several hours	5	2	3 days
			You live with what you have got.
One day	5	3	20 minutes
			few hours
			1 week
Several days	6	6	A few hours
			One day
			One day
			One day
			1 - 2 days
			1 month
One week	3	2	1 day
			30 days
Several weeks	5	3	within one week
			a week, probably. Well, acceptable is a strong word. Several weeks is acceptable, in the end, if there is no other option. But a week would be much, much better, obviously.
			several months, i.e., 2-3.
More -- write-in response	(write-in run time: 2 months) 1	(write-in run time; months) 1	Would like to have results within a couple of weeks instead of several months.

Survey question 26 :How frequently does the data that you store need to be accessed? Figure *iii* shows fine-grained results.

Survey question number 27: How long does this (Q26, above) data need to be retained? Figure *iv* shows fine-grained results.

Figure *iii*

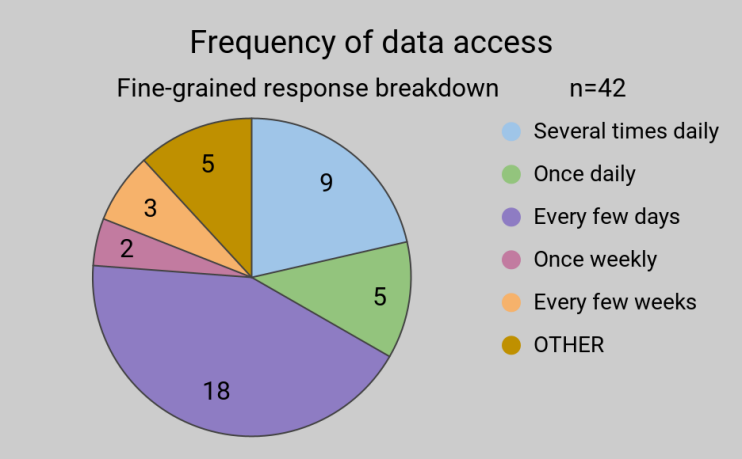
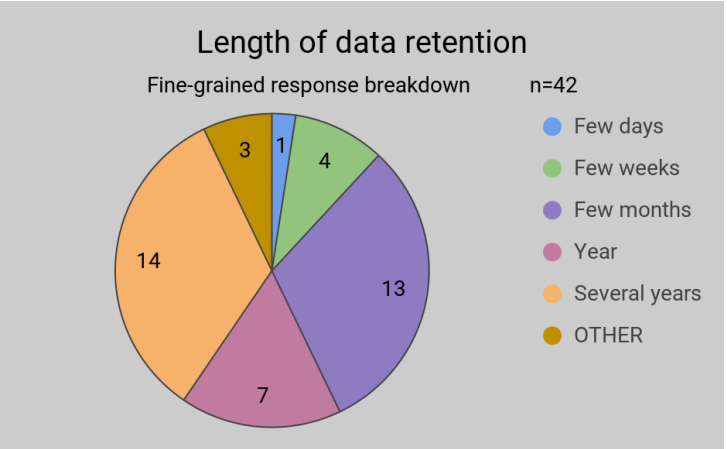


Figure i*v*



Survey question 29: Open comments

Table *iii* lists **all comments** by the 26 participants who submitted them.

Table *iii*

COMMENTS		
Participant's college		
ID #	*Participant's Department*	*Participant's comment*
College of Science and Liberal Arts		
1	Department of Mathematical Sciences	An HPC expansion will allow me to conduct numerical simulations far more effectively and efficiently. If jobs are completed with a higher speed, I can update my models faster and more effectively, resulting in better papers being published with greater frequency. This will also result in more successful grant proposals. I am currently often limited by the rather slow speed and scarce resources of the cluster.

2		Significantly: we are very often limited by how many simulations we can run, and, for some projects, by GPU availability.
3		It broadens and expands the range of projects that I (and other people) can realistically consider doing.
4		This expansion will significantly affect the research in my department (Math. Sci.) and will increase our ability to attract computational science faculty candidates
5		It will be very helpful
6	Department of Physics	This expansion will have a tremendous impact on my research program. A significant increase in the number of CPU and GPU nodes will allow us to study phenomena that were previously inaccessible to us. These phenomena take place within living cells and they involve tracking the position of ~1,000,000 atoms over several 10-100 microseconds.
7		Major HPC expansion would significantly help me with my research routine. I would be able not only to perform my main research simulations faster and with better accuracy, but, more importantly, it would substantially reduce my preliminary work for such simulations. The codes I use, especially PIC code, require a lot of testing before one submits the final simulation run. Such testing takes a lot of time now, in particular because one needs to test the code over a long simulation periods to see if the code generates some numerical instabilities, which are not seen in the beginning. This work takes a lot of time and it's also one the least interesting parts of my research routine.
8	Department of Chemistry and Environmental Science	This expansion will greatly accelerate our research. Our applications require running hundreds of jobs with less than 16 cores for up to 48 hours. Currently, the bottleneck is the amount of cores available to run the jobs. We do not need stronger nodes, but simply more of them available at any given time.
9		Significantly benefit research
10	Federated Department of Biological Sciences	I would really appreciate it if the expansion would provide higher storage quota for research. Currently it is 500GB but with remote sensing data and spatial-temporal modeling work, this space is far from enough when attempting to simulate over large spatial areas. My research program would also benefit a ton if the storage space purchased from HPC will be PERMANENT so that I don't have to pay for it once every year to keep the storage. Please let me know if you need additional information, but these two things have been REALLY important to run my research program. Many many thanks!
Ying Wu College of Computing		
11	Computer Science	Speed the research results and finding. It has a significant effect.
12		It will affect our research significantly. In general, the more computing resources become available, the more advanced deep learning models we can try, test and use.
13		My current research has not been using Lochness so much. Students have been outsourcing their work to Google Colab (Pro). However, demands are rising and having the possibility to train simultaneously multiple neural networks of moderate size would be great to have.
14		Really look forward to this expansion!

15		With sufficient resources, we can expand our research to more computationally intensive applications and make use of deep neural networks for real-time image processing and video analytics.
16	Data Science	This HPC Expansion is excellent and will strongly support my research.
17		The HPC expansion will make more students work on our projects simultaneously and our jobs can be executed quickly.
18		I have no idea how many GPUs one needs to do something...my (former) student did all the work on HPCs/Lochness. My current student is using BERT, but he is running it on Google Collab...I would guess that Google Collab never asked him what GPUs he needs. My (former) student had to break up data in his experiment into many batches and then the programs took maybe 24 hours to run on each batch. So obviously we don't have enough resources. We (currently) take whatever Google Collab is giving us and live with it and give up on NJIT.
19		It is very positive that an expansion takes place but the expansion must be designed such as these clusters are compatible with a cloud provider (AWS/Azure/GCP) - clusters must be connected to a core cloud provider. For example EKS-A clusters at NJIT federated to AWS.
20	Informatics	By all means, I strongly support this expansion, as I need more computing power that would be more easily available for my research. More GPUs and more resources in general that are available for the NJIT community would be great.
Newark College of Engineering		
21	Mechanical and Industrial Engineering	The significant increase in the number of public nodes as well as the addition of the new PFS will enable me (and my future group of grad students) to run more jobs simultaneously without needing as many privately owned nodes. This will greatly help our productivity and code development time. I will still purchase new nodes, but I will not have to acquire as many.
22		Will allow us to actually get data needed to investigate the problems of interest. (The inability to do this has a ripple effect, i.e., few papers produced, low visibility in the community, difficulty in attracting external funding and good students, and the loop continues). Research computing resources have greatly improved over the time I have been here (+ 30 years), but they are still far below what they should be for a university that is marginally a "Research I" institution. NJIT is well overdue in allocating funding to improve its research computing infrastructure.
23		it will majorly improve my group's ability to be proficient and obtain results
24		NJIT HPC should get maximum support from NJIT higher authority. NJIT currently has many computational faculties. Without HPC, research activities of many faculties will be significantly affected.
25	Department of Civil and Environmental Engineering	The data presented in the survey is based on the Kong clusters that our group used extensively, since the transition to Lochness we have been using an NSF computer allocation to a cluster in Texas so we can get our simulations to run faster and process the data more rapidly. With this major performance expansion, we would definitely return to use the NJIT-based cluster as the previously mentioned NSF cluster has a long wait queue.
26	Engineering Technology	It will certainly be a good plus, looking forward to it.