Report on CSU HPC (High Performance Computing) Study

Ricky Kwok
Co-Chair, Research Advisory Committee
ISTeC

May 5, 2008

1 Introduction

High performance computing (HPC) involves the innovative use of parallel hardware (i.e., computing machines with more than one processor) and parallel software to enable very fast problem solving. The CSU Information Science and Technology Center (ISTeC) is working with the Office of the Vice President for Research (VPR) to assess current activities and future opportunities that would benefit from high performance computing. Our findings from this study (as for the online survey form, the reader is referred to [1]; the questions of the survey form are reproduced in Table 3, Section 6) could potentially help CSU to better position itself in its presence of HPC in the region and future HPC-related resource acquisition/allocation.

2 My Interpretations of Findings in the HPC Study

There were 37 responses obtained (see Section 3 for a set of data summary charts and Section 7 for eight tables of raw data), representing different research groups on campus. The following are my personal interpretations. All errors are mine.

1. Out of the 37 responses, there are 31 different research areas. It reveals that using HPC is a popular computing paradigm for fast problem solving of research problems in various disciplines at CSU.

2. Apart from CoGrid node, Bioinformatics cluster, and HP Half Dome system, there are eight additional HPC resources reported (see Table 2). Combining this observation and the result from Q1d, it means that about one-third of the users do not have immediate access to HPC resources. These users may have to rely on HPC resources provided from other departments within CSU or other organizations outside CSU. These users might face strict restrictions imposed on them when using HPC. Among these users, where there are totally 11 such users, 7 users indicated that their HPC needs are not currently met while only 3 users indicated that their HPC needs are currently met.

3. The HPC resources at CSU are quite scattered geographically, implying possibly that most users will only access HPC resource that is closest to him/her (i.e., within his/her own department). Specifically, HPC resources from different departments are not linked together. As a suggested next step in this HPC Study, we can try to survey the frequency of using HPC for problem solving (e.g., on average, how much time a user spends in a week running simulations). With such information available, we can find out whether it is beneficial to combine the HPC resources at CSU to form a larger computational site, such that users can solve a larger problem with a shorter response time. Of course, this may raise a resource management problem across different departments. One possible approach would be incrementally adding other HPC nodes to the CoGrid system. Indeed, most responses indicated that a campus-wide HPC platform built by linking up all the individual clusters would be great so that the otherwise idle HPC resources could be time-shared.

4. In addition to the above two points, one observation worth noting is that, as evident in Q1e, a majority of the users are pessimistic about their HPC needs being met in the future. This further motivates our suggestion on exploring the possibility of combining CSU's HPC resources in order meet users' future needs collectively. On

the other hand, it is encouraging that over 70% of the respondents are willing to be involved in future HPC activities.

5. Concerning the needs for advanced visualization, out of the 26 users who answered whether their current needs are being met (among these users, 5 did not mention about their future needs), 69% replied yes; out of the 21 users who answered whether their future needs will be met, 67% replied yes. However, when we further look into the data, we found that out of the 5 users whose current needs are not being met, only one is optimistic that his future needs will be met. In addition, out of the 16 users whose current needs are being met, 3 users felt that their future needs will not be met. Hence, there is a slight trend that more users will be unsatisfied about the HPC support on advanced visualizations. Therefore, while the majority of the users are satisfied that their advanced visualization needs are being met currently and will be met in the future, it is indeed necessary (or beneficial in the near future) to further enhance our HPC systems to support advanced visualization.

6. More than half of the respondents were not aware of the course GRAD 510/511. This suggests that we should "advertise" more about the offering of the course. On the other hand, we should organize more tutorials/workshops on HPC, given the popularity of using HPC to solve research problems at CSU.

7. Space limitation (for accommodating HPC hardware) is a common concern among HPC resources owners.

8. Another major worry among HPC users (not technical oriented in HPC, e.g., biochemistry researchers) is the lack of adequate "consultancy" about software and hardware to use for their problems.

3 Summary of Findings

There were totally 37 responses obtained. Table 1 shows a summary of research areas of the respondents. Apart from CSU's CoGrid node, Bioinformatics cluster and HP Half Dome system, Table 2 lists HPC systems owned by the corresponding departments/organizations of the respondents.

Figures 1 to 11 present a summary of results about HPC usage, nature of usage, algorithms used, HPC resource used, whether HPC needs and visualization needs are met, compiler and language used, willing to be involved in future HPC activities, awareness of the course GRAD 510/511, and whether follow-up is required.

Table 1: A summary of research areas applying HPC.

Applied Economics	Discrete Mathematics	Oceanography
Applied Mathematics	Distributed Systems	Optimization
Assessment of Learning	Financial Economics	Quant. Methods in Social Sciences
Atmospheric Chemistry, Global Modeling, Pollution and Climate	Geodynamics	Resource Allocation
Bioinformatics	Greenhouse Gas Cycling	Satellite Data Assimilation
Compilers	Groundwater Modeling	Satellite Meteorology
Computational Fluid Dynamics	Hydrology, Geomorphology	Soil Carbon, Ecosystem Cycling
Computational Biology	Lex System Dynamics, Robotic Systems, Ecomplectric Power Systems	Solid State
Computational Mathematics	Microwave Remote Sensing and RF Systems	Spatial Statistics/Sampling
Computer Algebra	Networking	Wind Engineering and Fluid Mechanic
Computer Vision	North American Animal Disease Spread Model	

Table 2: List of HPC systems besides CoGrid node, Bioinformatics cluster and HP Half Dome system.

User	Department	Research Area	HPC System
Andrew S. Jones	CIRA	Satellite Data Assimilation	A 32-processor cluster.
Colette Heald	Atmospheric Science	Atmospheric Chemistry, Global Modeling, Pollution and Climate	Sun System nodes.
Dennis Harry	Geosciences	Geodynamics	Under development. Working with other units in WCNR to grow a Linix-based HPC grid that was initially created by NREL.
David Dandy	Chemical and Biological Engineering	Computational Fluid Dynamics	A 32-node Linux cluster (Red Hat) running openMosix.
Jeff Niemann	Civil and Environmental Engineering	Hydrology, Geomorphology	A few multiprocessor Sun workstations.
Mark Easter	NREL	Greenhouse Gas Cycling	Parallel processor (named “rubel”) at NREL, plus a database server.
Michael Kirby	Mathematics	Applied Mathematics	A 42-node AMD 64-bit Opteron cluster with 156GB total memory and over 5TB disk storage.
Taka Ito	Atmospheric Science	Oceanography	Dual core Opteron Linux cluster 16 nodes (2 CPUs per node, 4 cores per node).

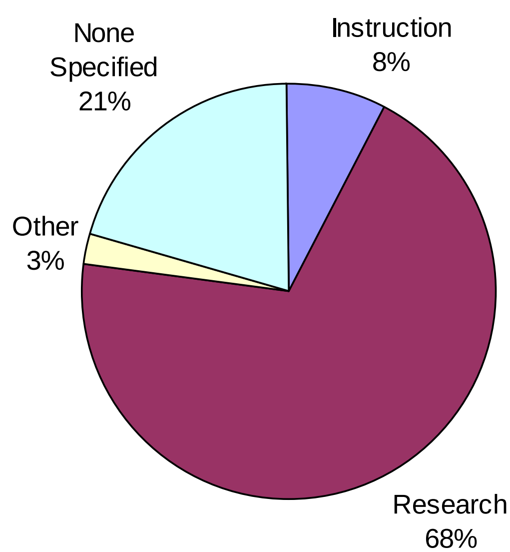

Figure 1: Q1 (Usage of HPC).

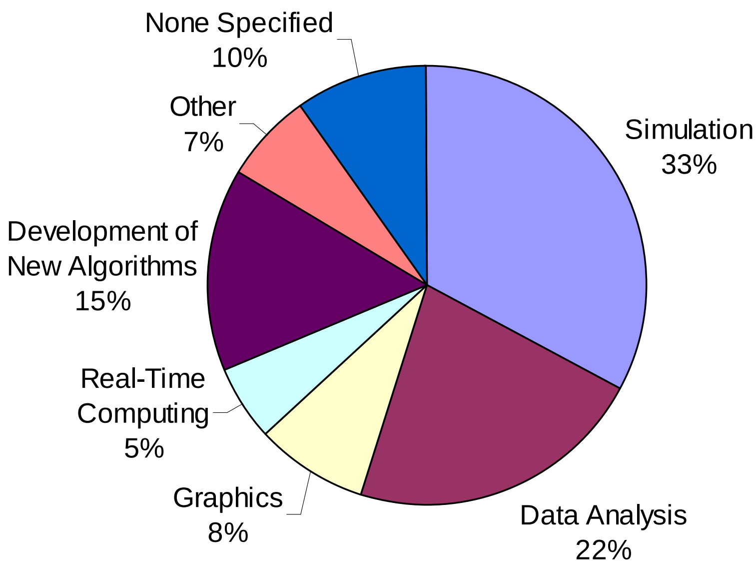

Figure 2: Q1a (Nature of Usage).

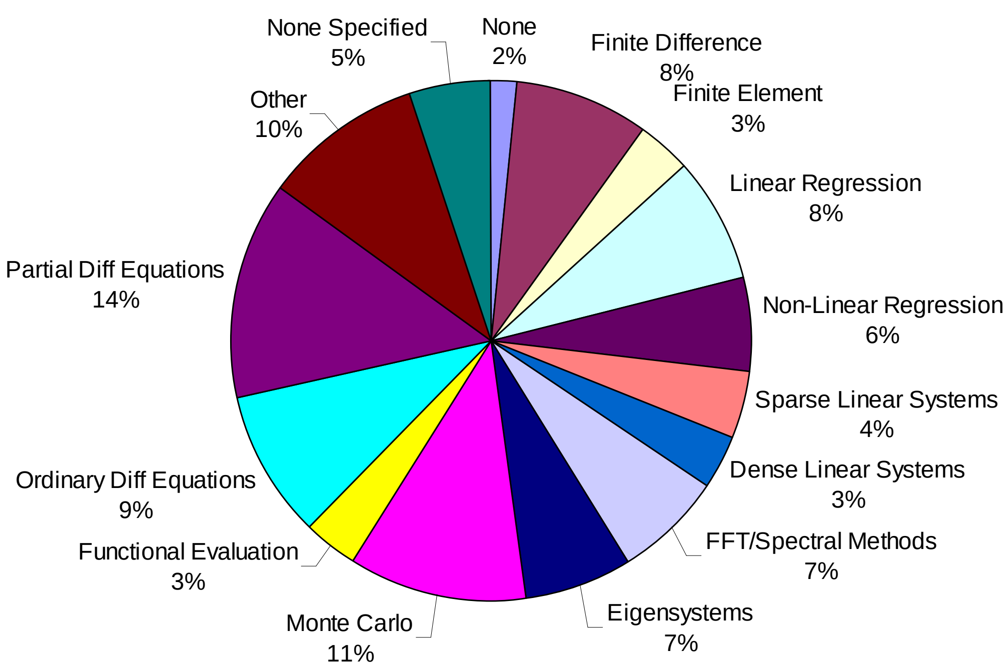

Figure 3: Q1c (Algorithm Used).

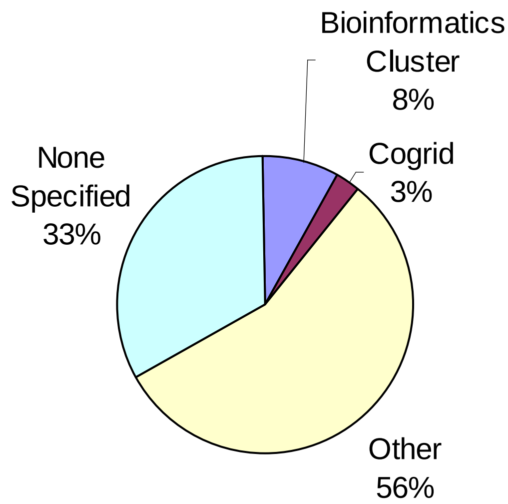

Figure 4: Q1d (HPC Resource Used).

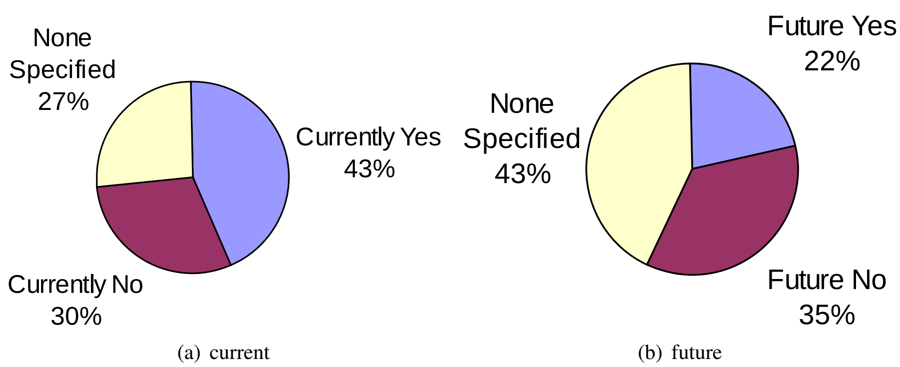

Figure 5: Q1e (HPC Needs Met?).

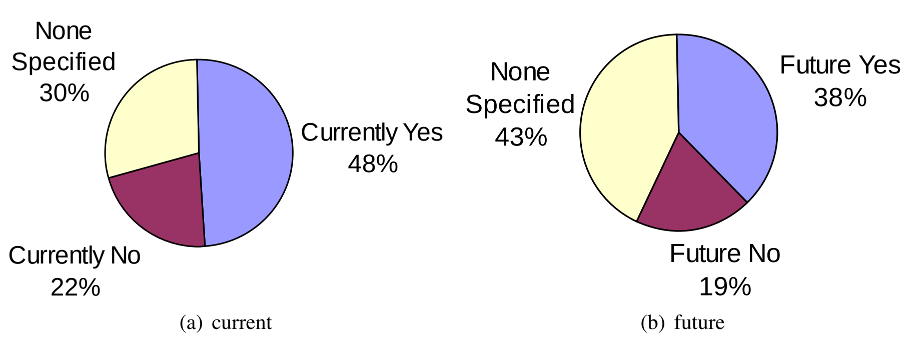

Figure 6: Q1f (Visualization Needs Met?).

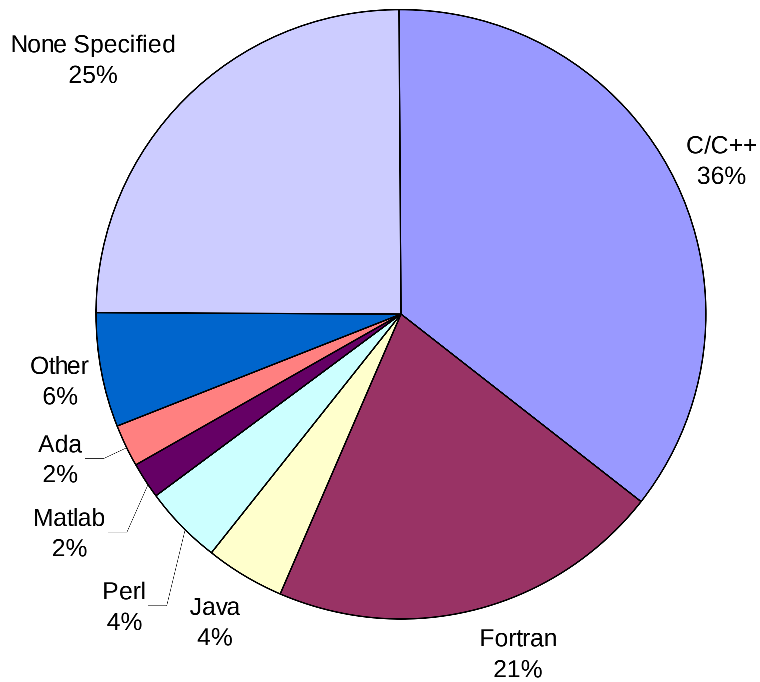

Figure 7: Q2 (Compiler Used).

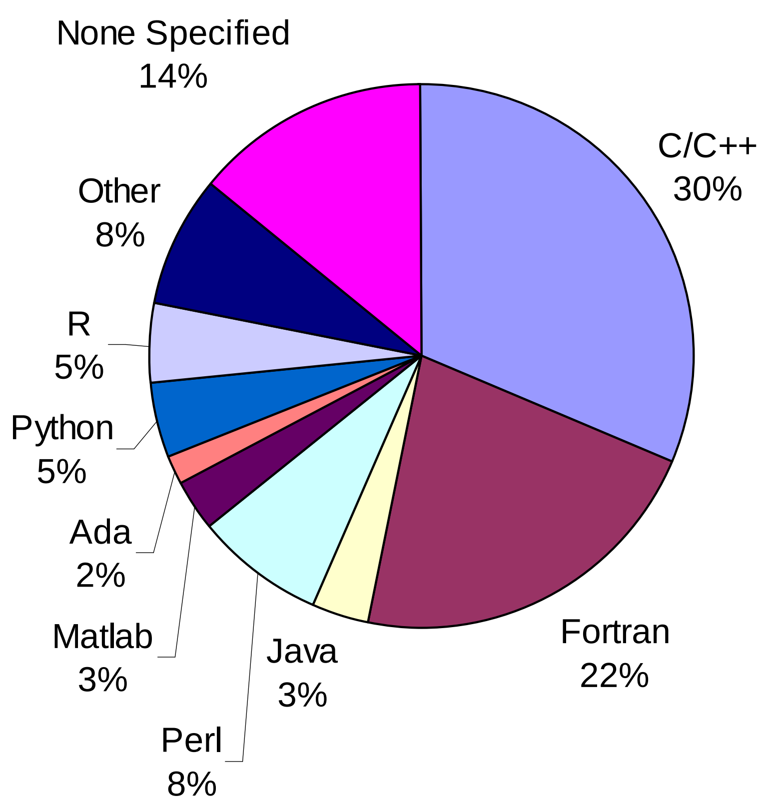

Figure 8: Q4 (Language Used).

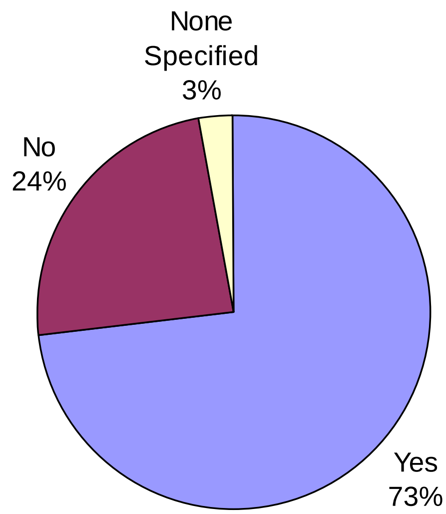

Figure 9: Q6 (Be Involved in Future?).

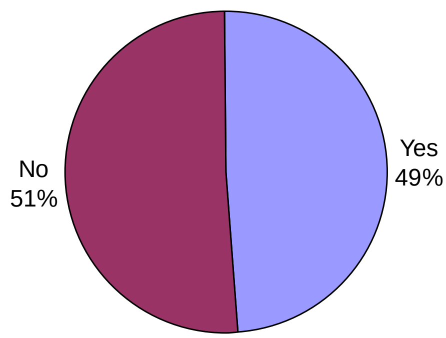

Figure 10: Q7 (Aware of GRAD 510/511?).

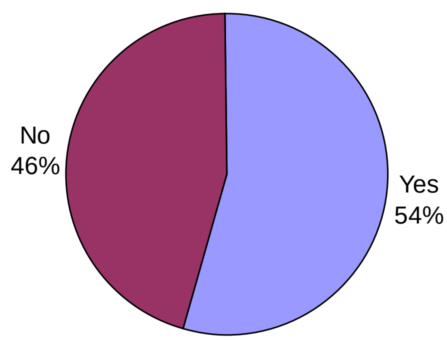

Figure 11: Q8 (Required Follow-Up?).

4 Summary of Representative HPC Activities

4.1 Small-angle X-ray Scattering

Professor Karolyn Luger, teamed up with colleagues from various other departments, most notably the Computer Science department (with Professors Jim Bieman and Michelle Strout), is leading a large scale effort in molecular structure research. Knowledge of molecular structure is essential for our understanding of biological systems. X-ray crystallography is widely used to determine the 3-dimensional structure of proteins, nucleic acids, and large nucleoprotein complexes. It exploits the propensity of X-rays to be scattered by the electron clouds around atoms. The "electron density" can be reconstructed from diffraction patterns obtained from single crystals of the target molecule. An atomic model is then built into the electron density, which in turn is refined against the data, eventually resulting in an accurate molecular structure in which the coordinates of every atom with respect to every other atom is known precisely. For more information about X-ray crystallography facility in the Department of Biochemistry and Molecular Biology, the reader is referred to [2].

While X-ray crystallography can provide details on structural information for mechanistic analyses, it is, however, restricted to describing low energy conformations of macromolecules within crystal lattices. Small-angle X-ray scattering (SAXS) offers complementary information about macromolecular folding, unfolding, aggregation, extended conformations, flexibly linked domains, shape, conformation, and assembly state in solution, at a lower resolution range. Combining X-ray crystallography with SAXS, it can allow multi-scale modeling to create complete and accurate atomic resolution structural images of macromolecules for modeling allosteric mechanisms, supramolecular complexes, and dynamic molecular machines. Figure 12 shows a typical SAXS flow.

Monte Carlo simulations are proving quite useful for modeling of SAXS data, and are expected to allow continued advances in SAXS data interpretation tools. Various groups have succeeded in developing computational method that can provide theoretical scattering patterns based on a model, which is adjusted until the predicated scattering pattern matches the measured pattern. This process, as with Monte Carlo simulations, is computationally intensive and therefore requires high-performance computing support.

4.2 Genomic Analysis

Professor P. Shing Ho (Chair of the Biochemistry and Molecular Biology deparment) is leading an effort to perform large scale genomic analysis of various phyla to study the evolution of functional DNA sequences. A recent endeavor by Professor Ho is a comparative phylogenomic analysis to study the evolutionary relationships between GC content, CpG-dinuleotide content (CpGs), potential nuclear factor I (NFI) binding sites, and potential Z-DNA forming regions (ZDRs) as representative structural and functional GC-rich genomic elements. Professor Ho's study is highly computationally intensive and has been done on a cluster of Linux machines using eight to 32 nodes of computing power. Sequences and annotations of prokaryotic genomes were accessed from the National Center for Biotechnology Information (NCBI) [3] and eukaryotic genomes from the Ensembl database [4].

4.3 Proteomics

Professor Jessica Prenni, Director of the CSU Proteomics and Metabolomics Facility [5], is leading an effort in proteomics. Proteomics is the large-scale study of proteins, in particular the structures and functions of proteins. Professor Prenni makes use of two software packages—Mascot [6] and SEQUEST [7]—in her research. Mascot is a software search engine that uses mass spectrometry data to identify proteins from primary sequence databases. Mass spectrometry, on the other hand, is an analytical technique that measures the mass-to-charge ratio of charged particles. Mascot implements a wide variety of proven searching methods to analyze mass spectrometry data. Using Mascot, the general approach for searching is to take a small sample of the protein of interest and digest it with a proteolytic enzyme such as trypsin. The resulting digest mixture is then analyzed by a mass spectrometer. Afterwards, the experimental mass values are compared with entries in a comprehensive primary sequence database (e.g., MSDB, NCBInr, SwissProt, dbEST) to find out the closest match(es).

SEQUEST is a tandem mass spectrometry data analysis program used for protein identification. In contrast to mass spectrometry, tandem mass spectrometry involves multiple steps of mass spectrometry selection. SEQUEST identifies collections of tandem mass spectra to peptide sequences that have been generated from databases of protein sequences. It identifies each tandem mass spectrum individually, and evaluates protein sequences from a database to compute the list of peptides that could result from each. Since the peptide's intact mass is known from the mass spectrum, using this

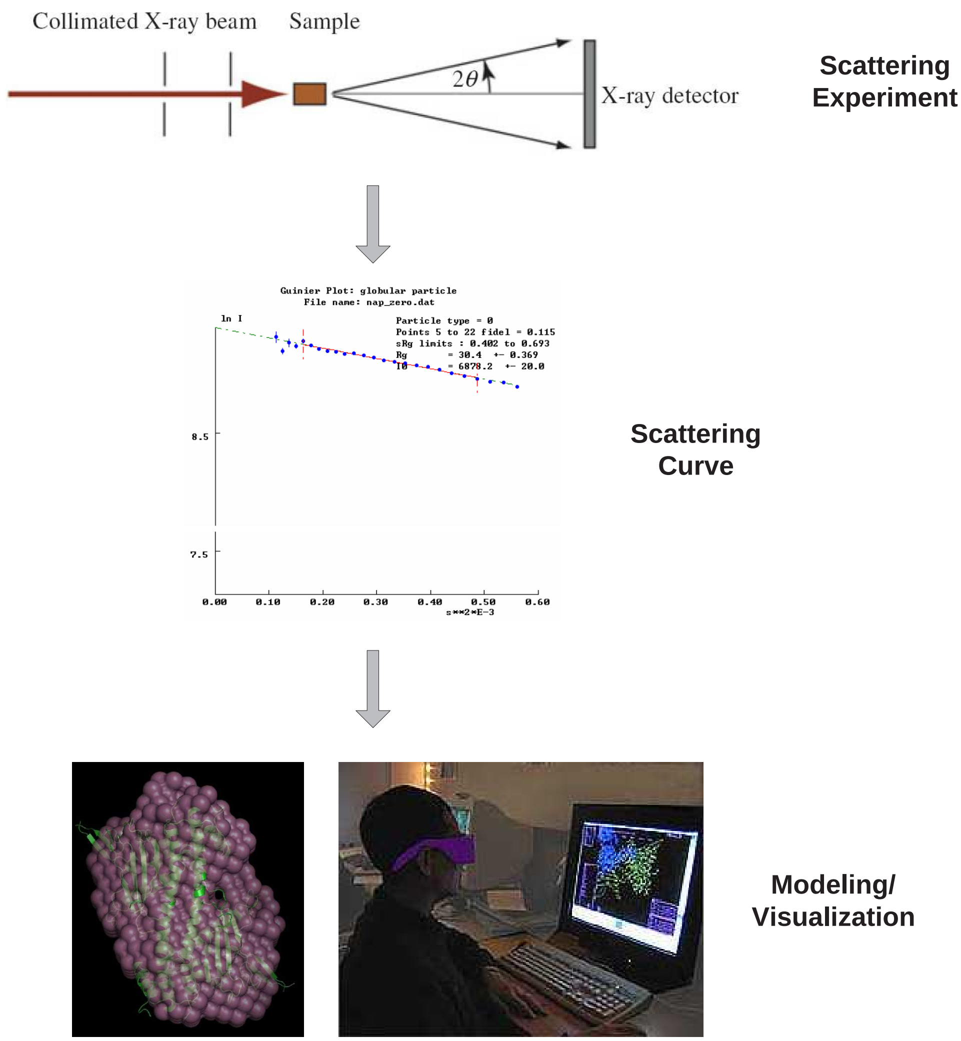

Figure 12: A typical SAXS flow.

information SEQUEST can determine the set of candidate peptides sequences that could meaningfully be compared to the spectrum. Specifically, for each candidate peptide, SEQUEST projects a theoretical tandem mass spectrum and compares these theoretical spectra to the observed tandem mass spectrum using a cross correlation approach. The candidate sequence with the best matching theoretical tandem mass spectrum is then reported as the best identification for this spectrum.

4.4 Pattern Analysis

Professor Michael Kirby (Mathematics department) is leading an effort in real-time pattern analysis with many interesting applications such as face recognition. A traditional approach to the investigation of unexplained phenomena is to infer laws from the patterns (information) in collected data. Presumably, the more information available with regard to a phenomenon, the better for the analysis, synthesis and modeling of patterns in data. However, with today's data acquisition technology, our ability to acquire data grows faster than our ability to process and analyze it. Specifically, high-dimensional and massive data sets are potentially a significant barrier to the investigation.

The understanding of patterns in high-dimensional and massive data sets is proving to be a key ingredient to knowledge discovery. For example, snapshots of clouds and sky give rise to a family of images with a common characteristic while at the same time exhibiting significant variations across the images. Likewise, a set of digital images of faces provides another example of an image family. While members of a family clearly possess distinct features, it is sensible to pose the question of whether an image, or set of unlabelled images, belongs to a certain family of images.

The analysis of patterns in data has typically been a subject in statistics and engineering. Recently, however, fundamental mathematical theory in areas such as differential geometry and topology has provided a new mathematical framework and insights for understanding large data sets residing in spaces of large ambient dimensions. The research at the Pattern Analysis Laboratory (PAL) [8], managed and directed by Professor Michael Kirby, emphasizes the transition of mathematical theory to efficient algorithms for exploring, understanding and modeling massive data sets.

For example, one of the approaches developed is geometric in nature and the main tool is the dimensionality reducing mapping. Specifically, each member of a family of patterns shares features common to the other members in the family. This correlation between images is a footprint of low dimensionality, or that a simpler representation exists. The question now becomes how does the data sit geometrically in its ambient space and how can this coherence be exploited for data reduction. The main techniques involved include optimal orthogonal expansions, Fourier analysis, radial basis functions, neural networks and wavelets.

The analysis of massive data sets involves a myriad of theoretical, algorithmic and computational challenges from which has emerged a new field of applied mathematics. It also promotes interdisciplinary collaboration, i.e., high-performance computing, for fast manipulation of high-dimensional and massive data sets.

4.5 Other HPC Activities

- Dennis Harry, Geosciences, research area: Geodynamics—his research is focused on understanding the dynamic processes associated with deformation of the Earth's crust and lithosphere, and the associated magmatic consequences. Key aspects are the multi-physics nature of the problem and spatial and temporal bandwidth (processes operating on microstructural scales impact results that Harry's team try to simulate on scales of 1000's of km, and processes operating on scales of days impact results that Harry's team tries to simulate on scales of 100's of millions of years. Impacts of the research relate to global energy resources and geological hazards.

  Finite element and finite difference codes are used. Some are home-grown and will continue to be developed. A growing tendency in my field is to take advantage of software libraries that are sustained by international collaborative efforts and some more specific codes that are sustained by international collaborative efforts. This is a work in progress. Dennis Harry is a Co-Chair of a working group that is part of the NSF-funded Computational Infrastructure in Geodynamics (CIG) to pursue these initiatives.

- Jeff Niemann, Civil and Environmental Engineering, research area: Hydrology, Geomorphology—his research is focused on hydrologic modeling, uncertainty analyses, and geomorphic modeling. His team develops programs to carry out Monte Carlo simulations of geomorphic models that extend for 10,000 or more years.

- Shaun Case, Clinical Sciences, research area: North American Animal Disease Spread Model—1) Case's team currently uses HPC systems at the University of Guelph Canada for Animal Disease/Epidemiology Simulation

Modeling, however, Case's team would really like to be able to use the resources here at CSU, CoGrid for instance, as well for the NAADSM project, which is part of The Animal Population Health Intitute here at CSU, and a joint project with USDA, University of Guelph and the Candian USDA equivalent. 2) Case's team uses software for equation fitting, on single processor systems, to help analyze the resulting simulation data, however, to do this on an HPC system would be a huge benefit to us. 3) Case's team would really like to discuss the options of using CoGrid for his research, as it is located here at our own University.

Case's team develops its own simulation software, which currently uses a form of Monte Carlo simulation, which the team will be building on in the future. Furture applications may contain neural network implementations, as well as, other forms of modeling.

- There are also numerous other HPC activities such as weather data assimilation and weather modeling, sequence analysis, genomics, functional genomics, analysis of network traces, network simulation, atmospheric composition modeling related to pollution and climate, computational fluid dynamics involving chemical vapor deposition, pyrolysis and microfluidics, modeling the ecosystem to simulate greenhouse gas cycling between soils and the atmosphere, ocean modeling, modeling of plant-soil ecosystems, etc.

5 CoGrid Job Submission System

Computational Grids enable the sharing, selection, and aggregation of a wide variety of geographically distributed computational resources (such as supercomputers, compute clusters, storage systems, data sources, instruments, people) and presents them as a single, unified resource for solving large-scale compute and data intensive computing applications (e.g., molecular modeling for drug design, brain activity analysis, and high energy physics).

CoGrid [9] was built with the goal of providing computational services for Colorado resident. At CSU, it is used by various departments, such as Chemical Engineering, Civil Engineering and Computer Science, to facilitate teaching and research activities.

CoGrid supports batch processing of jobs (e.g., a Fortran program, a MatLab program, a parallel program written using MPICH, etc.). To submit a job to the CoGrid system, we start by creating a script file (e.g., a simple csh shell script, Perl script, etc.) describing the "nature" of the job. Then, we need to enter the job requirements (e.g., required operating systems, applications, compilers, etc.) so that the system can determine the most appropriate grid/grids. After that, the job can be submitted for execution, and the system provides various functions to view the execution status of the job. Figure 13 presents a screen capture of the CoGrid job submission system.

The College of Engineering is currently exploring the possibility of establishing a centralized cluster computing facility based on a job management infrastructure similar to that of CoGrid.

6 Online Survey Form

The questions of the online survey form are reproduced in Table 3.

7 Raw Data

Tables 4 to 11 present the raw data results of the 37 responses from the survey.

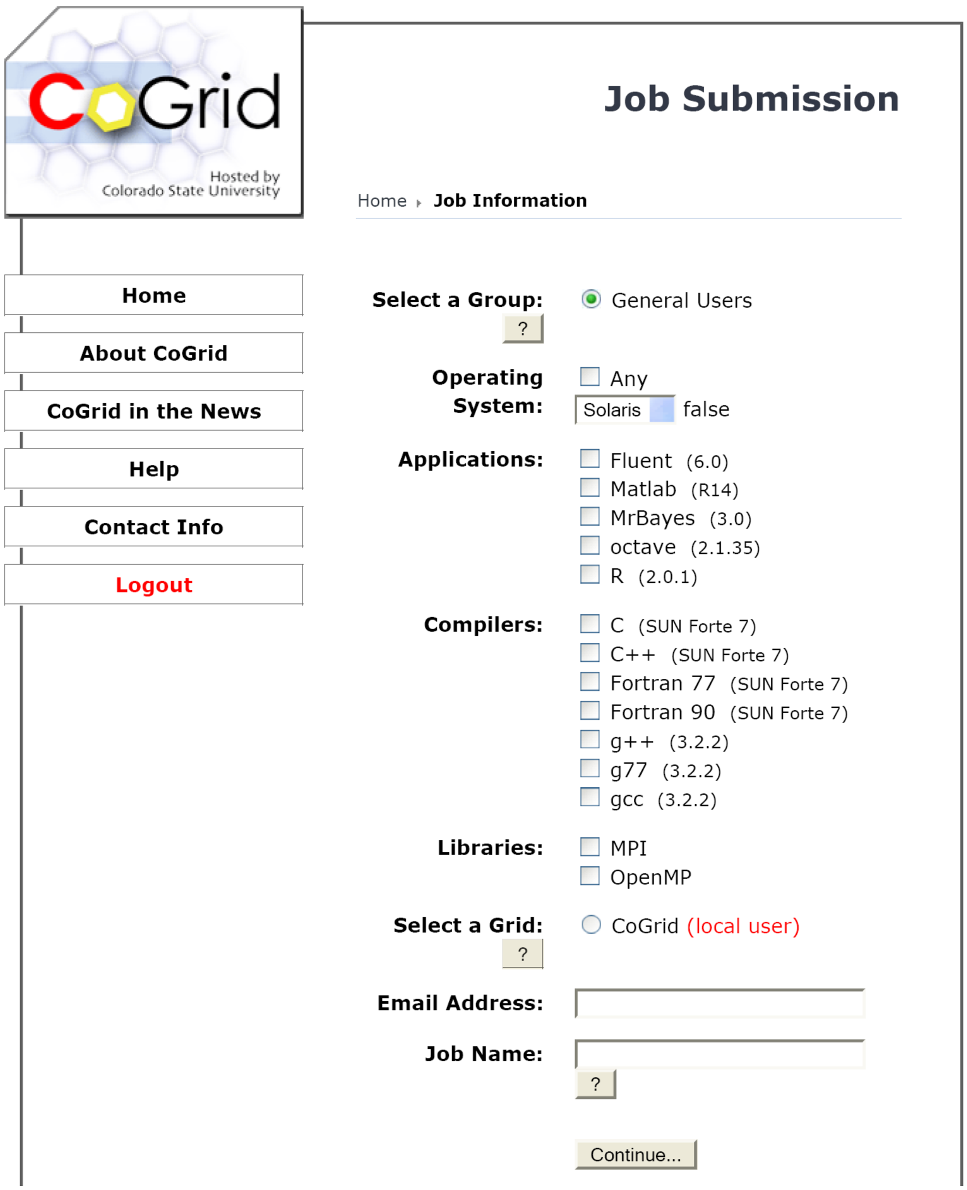

Figure 13: Screen capture of the CoGrid job submission system.

References

[1] CSU HPC Survey,
http://istec.colostate.edu/groups_research/hpc_survey/hpc_survey.cfm, 2008.

[2] Macromolecular X-ray Crystallography Facility,
http://www.bmb.colostate.edu/macro_xray.cfm, 2008.

[3] National Center for Biotechnology Information (NCBI),
http://www.ncbi.nlm.nih.gov/genomes/lproks.cgi, 2008.

[4] Ensembl Database, http://www.ensembl.org/, 2008.

[5] Proteomics and Metabolomics Facility, http://www.pmf.colostate.edu/, 2008.

[6] Mascot, http://www.matrixscience.com/, 2008.

[7] SEQUEST, http://fields.scripps.edu/sequest/, 2008.

[8] Pattern Analysis Laboratory,
http://www.math.colostate.edu/~kirby/PALindex.html, 2008.

[9] CoGrid, https://cogrid.colostate.edu/, 2008.

Table 3: The survey form.

Name: ____________________________ Email: ____________________________ Department: _______________________ Area of Research: ___________________	
1. Please indicate whether you currently use or have used HPC in (check all that apply):	☐ research ☐ instruction ☐ other (describe): ________________
If you checked 1 above, then please complete questions a-e below:	
1a) Please indicate the nature of your use of HPC (check all that apply):	☐ simulation ☐ data analysis ☐ graphics ☐ arts ☐ real-time computing ☐ development of new algorithms ☐ other (describe): ________________
1b) Please describe the fields in which you use or have used HPC (e.g., high energy physics, climate modeling, molecular simulation, fluid mechanics) :	____ ____ ____
1c) Please identify the mathematical algorithms used (check all that apply):	☐ none ☐ finite difference ☐ finite element ☐ boundary element ☐ linear regression ☐ non-linear regression ☐ sparse linear systems ☐ dense linear systems ☐ FFT/spectral methods ☐ eigensystems ☐ Monte Carlo ☐ functional evaluation ☐ ordinary diff equations ☐ partial diff equations ☐ other (describe): ________________ ☐ other (describe): ________________ ☐ not sure
1d) Please identify the resources you currently use for HPC (check all that apply):	☐ CSU's CoGrid node (http://cogrid.colostate.edu) ☐ Bioinformatics cluster ☐ HP Half Dome system ☐ other CSU (describe):________________ ☐ other external (describe): ________________
1e) Are your HPC needs being met by the combination of these resources?	Currently Yes ◯ No ◯ Future Yes ◯ No ◯
1f) Are your needs for advanced visualization being met?	Currently Yes ◯ No ◯ Future Yes ◯ No ◯
2. What compilers are needed?	____
3. Briefly describe programs you are currently running.	____
4. What programming languages are you using?	____
5. Do you have your own HPC system available? If yes, please describe or give a URL.	____
6. If you did not check any of 1 above, would you be interested in finding out more about how you might team with others at CSU to become involved in HPC?	Yes ◯ No ◯
7. Are you aware that CSU offers two courses, GRAD 510 and GRAD 511, specifically for graduate students and seniors in the area of HPC?	Yes ◯ No ◯
8. Would you like Ricky Kwok from ISTeC to follow up with you via a telephone call so that you can provide additional information and/or detail to your answers?	Yes ◯ No ◯

Table 4: Raw data results of the survey.

	Name	Department	Research Area	Q1 (Usage)	Q1a (Nature)	Q1d (Resource)	Q1e (Needs Met?)	Q1f (Visual. Met?)	Q2 (Compiler)	Q4 (Language)	Q5 (Own HPC?)	Q6 (Be Involved?)	Q7 (510/511)	Q8 (Follow-up?)
1	Ahmed Abdel ALim	Economics	Financial Economics	Instruction	Other	Other	Currently no; Future yes;	Currently no; Future yes;				Yes	No	Yes
2	Alexander Hulpke	Mathematics	Computer Algebra	Research	Development of new algorithms, other		Currently yes; Future no;	Currently yes; Future yes;	A C compiler (gcc is fine)	C/GAPs own language		Yes	Yes	Yes
3	Andrew S. Jones	CIRA	Satellite Data Assimilation	Research	Simulation, data analysis	Other	Currently yes; Future no;	Currently no; Future no;	Portland group Fortran compiler Perl 5 imsl libraries	Fortran 77-90-95, C, C++, Perl	Yes, a small 32-processor cluster at CIRA.	Yes	No	Yes
4	Andrey Ptitsyn	MIP	Bioinformatics	Research	Simulation, data analysis, development of new algorithms	Bio-informatics cluster	Currently no; Future no;	Currently yes; Future yes;		C/C++		Yes	Yes	No
5	Anton Betten	Mathematics	Discrete Mathematics	Research	Development of new algorithms	Other	Currently no; Future no;		C++	C++	No	No	Yes	Yes
6	Bogusz J Bienkiewicz	Civil and Environmental Engineering	Wind Engineering and Fluid Mechanic			Other	Currently no;	Currently no;				Yes	Yes	Yes
7	Brad Reisfeld	Department of Chemical and Biological Engineering	Computational Biology	Research	Simulation, data analysis	Bio-informatics cluster	Currently no; Future no;		Fortran, C	Fortran, C, Python	No	Yes	No	Yes
8	Bradly Thomas Johnson	Construction Management										Yes	Yes	Yes
9	Bruce Draper	Computer Science	Computer Vision	Research	Real-time computing, development of new algorithms	Other	Currently no;	Currently yes; Future yes;	gcc/g++ (GNU C/C++)	C++, R		No	Yes	No
10	Christos Papadopoulos	Computer Science	Networking	Research	Simulation, data analysis	Other			gcc	C/C++	No	Yes	No	No
11	Colette Heald	Atmospheric Science	Atmospheric Chemistry, Global Modeling, Pollution and Climate	Research	Simulation, data analysis, development of new algorithms	Other	Currently yes; Future yes;	Currently yes; Future yes;	Sun Studio compiler or comparable for Fortran90	Fortran 90, IDL	Yes, recently purchased Sun System nodes for computing.	Yes	No	No
12	Craig A. Bond	DARE	Applied Economics	Other: Have not used...but potentially could	Simulation, data analysis				No idea	Try to use pre-programmed languages as listed above.	No	Yes	Yes	Yes

Table 5: Raw data results of the survey.

	Name	Department	Research Area	Q1 (Usage)	Q1a (Nature)	Q1d (Resource)	Q1e (Needs Met?)	Q1f (Visual. Met?)	Q2 (Compiler)	Q4 (Language)	Q5 (Own HPC?)	Q6 (Be Involved?)	Q7 (510/511)	Q8 (Follow-up?)
13	Dennis Harry	Geosciences	Geodynamics	Research	Simulation, development of new algorithms	Other	Currently yes	Currently yes	GNU Fortran, C, and C++	Fortran, C, C++	Under development. Working with other units in WCNR to grow a Linix-based HPC grid that was initially created by NREL in our college.	Yes	No	Yes
14	David Dandy	Chemical and Biological Engineering	CFD	Research	Simulation, development of new algorithms	Other	Currently no; Future no;	Currently no; Future no;	C++, Fortran 77/90/95	Fortran, C++	Yes, it's a 32 node Linux cluster (Red Hat) running openMosix.	Yes	No	No
15	Domenico Bau	Civil and Environmental Engineering	Groundwater Modeling	Research	Simulation		Currently yes; Future yes;	Currently yes; Future yes;	Fortran, C++	Fortran, C++	Not yet	Yes	No	Yes
16	Donald Estep	Mathematics and Statistics	Computational Mathematics	Research	Simulation, development of new algorithms	Other	Currently yes;	Currently yes;		C++, C, Fortran	No	Yes	Yes	No
17	Edwin Chong	Electrical and Computer Engineering	Optimization	Research	Simulation, data analysis, development of new algorithms	Other	Currently no; Future no;	Currently yes; Future yes;	C, C++	Matlab	No	No	Yes	No
18	Hochheimer	Physics	Solid State				Currently yes; Future yes;	Currently yes; Future yes;	N/A	N/A	N/A	No	No	No
19	James Smith	Development	Distributed Systems	Research	Simulation, graphics	Other	Currently no; Future yes;	Currently yes; Future yes;		C++, Java	No	Yes	Yes	Yes
20	Jeff Niemann	Civil and Environmental Engineering	Hydrology, Geomorphology	Research	Simulation, data analysis	Other	Currently yes; Future no;			Fortran	A few multiprocessor sun workstations is the closest we have for our own HPC system.	Yes	No	No
21	Karen Kaminski	School of Education	Assessment of Learning									No	No	No
22	Kirk Sarell	Executive MBA Program										No	No	No
23	Luis Briceno	Electrical and Computer Engineering	Resource Allocation	Research	Simulation	Other	Currently yes; Future yes;	Currently yes; Future yes;	g++, Java, Perl	C++, Java, Perl		Yes	No	Yes

Table 6: Raw data results of the survey.

	Name	Department	Research Area	Q1 (Usage)	Q1a (Nature)	Q1d (Resource)	Q1e (Needs Met?)	Q1f (Visual. Met?)	Q2 (Compiler)	Q4 (Language)	Q5 (Own HPC?)	Q6 (Be Involved?)	Q7 (510/ 511)	Q8 (Follow-up?)
24	Mark Easter	NREL	Greenhouse Gas Cycling	Research	Simulation, data analysis	Other	Currently yes;	Currently no;	F77, F90, gcc	Fortran, C, C++, Perl, R	Parallel processor (named ”rubel”) at NREL, plus a database server.	Yes	Yes	Yes
25	Michael Kirby	Mathematics	Applied Mathematics	Research, instruction	Simulation, data analysis, graphics, real-time computing, development of new algorithms	Other	Currently yes; Future no;	Currently yes; Future yes;	C++, Fortran, Java, Matlab	Matlab, C++	Yes, www.math.colostate.edu/ kirby/FACILITIES.html	Yes	Yes	Yes
26	Michelle Strout	Computer Science	Compilers	Research	Automatic generation of parallel programs	CSU’s CoGrid node,TerraGrid and NCAR		Currently yes; Future no;	The vendor supplied compilers. It would also be helpful to have compilers such as the Pathscale compiler.	C, C++, Fortran 77, 9X, 20XX. I would like to start using some Python and newer languages like Chapel and X10.	No		Yes	Yes
27	Mike Lacy	Sociology	Quant. Methods in Social Sciences		Simulation, data analysis			Currently no; Future no;	Stata.	Stata.	No	No	No	No
28	Richard Casey	Bioinformatics Center	Support Bioinformatics at CSU	Research, instruction	Data analysis, graphics, other	Bio-informatics cluster	Currently yes; Future no;	Currently yes; Future no;	We have several Fortran/C compilers for serial and parallel computations on the Bioinformatics cluster. We don’t have an immediate need for additional compilers but would install other compilers if requested by the CSU community.	We support Fortran, C, Perl, Python, Unix shell scripts, PVM, MPI, LINDA. We will support other languages if requested by the CSU community.	We manage the CSU Bioinformatics cluster (www.bioinformatics.colostate.edu). This includes a 64-CPU Apple G5 cluster, HP Proliant server, high-performance graphics workstations, Oracle bioinformatics database, 14TB RAID array for data storage.	Yes	Yes	Yes
29	Robin Reich	Forest, Rangeland and Watershed Stewardship	Spatial Statistics/Sampling									Yes	No	Yes

Table 7: Raw data results of the survey.

	Name	Department	Research Area	Q1 (Usage)	Q1a (Nature)	Q1d (Resource)	Q1e (Needs Met?)	Q1f (Visual. Met?)	Q2 (Compiler)	Q4 (Language)	Q5 (Own HPC?)	Q6 (Be Involved?)	Q7 (510/ 511)	Q8 (Follow-up?)
30	Samee U. Khan	Electrical and Computer Engineering	Distributed Systems	Research	Simulation	Other	Currently yes; Future yes;	Currently yes; Future yes;	gcc, MPI, Ada	C++, Ada	http://hpc.uta.edu/	Yes	Yes	Yes
31	Shaun Case	Clinical Sciences	North American Animal Disease Spread Model	Research	Simulation, data analysis	Other	Currently no; Future no;	Currently no; Future no;	GNU C (gcc) (And maybe Fortran for some external GIS and mathemetical libraries, which we use.)	GNU C (gcc) Future implementations may make use of C++, however for reasons of portability to other parallel systems in use by our user base, we currently only use "C".	No	Yes	Yes	Yes
32	Stephen Hayne	Computer Information Systems	Collaboration	Research	Simulation, data analysis, graphics		Currently no;	Currently no;	SAS		I'm going to try and configure the COB's Half Dome for SAS.	No	Yes	No
33	Steven Miller	Atmospheric Science (CIRA)	Satellite Meteorology	Research	Simulation, data analysis, real-time computing, development of new algorithms		Currently yes; Future yes;	Currently yes; Future yes;	Portland group	Fortran 90	No	Yes	No	No
34	Steven Reising	Electrical and Computer Engineering	Microwave Remote Sensing and RF Systems									No	No	No
35	Taka Ito	Atmospheric Science	Oceanography	Research	Simulation	Other	Currently yes; Future no;	Currently yes; Future no;	Fortran, C, C++	Fortran	Dual core Opteron linux cluster 16 nodes / 2 CPUs per node / 4 cores per node.	Yes	No	Yes

Table 8: Raw data results of the survey.

	Name	Department	Research Area	Q1 (Usage)	Q1a (Nature)	Q1d (Resource)	Q1e (Needs Met?)	Q1f (Visual. Met?)	Q2 (Compiler)	Q4 (Language)	Q5 (Own HPC?)	Q6 (Be Involved?)	Q7 (510/ 511)	Q8 (Follow-up?)
36	Tom Hilinski	NREL	Soil Carbon, Ecosystem Cycling	Research	Simulation, data analysis, graphics, other	Other	Currently yes; Future no;	Currently yes; Future yes;	Standards-Compliant C++ compilers. (use GNU's, would prefer Intel). ANSI C Fortran77 occasionally F90 Weaknesses currently are in available of quality Linux/*nix debuggers. (Want TotalView, can't afford it!)	Mostly C++, C, F77, Perl; Other essential software: R, netcdf, GIS	Yes, NREL cluster "rubel".	Yes	No	No
37	Wade Troxell	Engineering	Lex System Dynamics, Robotic Systems, Ecomplectric Power Systems	Research	Simulation, graphics, real-time computing							Yes	Yes	No

Table 9: Raw data results of the survey.

	Name	Email	Q1b (Field)	Q1c (Algorithm)	Q3 (Program)
1	Ahmed Abdel ALim	aliabdel@lamar.colostate.edu		None	
2	Alexander Hulpke	hulpke@math.colostate.edu	Group theory	Other	GAP Standalone Coset Enumerator Standalone Knuth Bendix.
3	Andrew S. Jones	jones@cira.colostate.edu	Weather data assimilation weather modeling weather data mass storage via NCAR	Finite difference, linear regression, non-linear regression, sparse linear systems, dense linear systems, eigensystems, ordinary diff equations, partial diff equations	WRF community model WRF-Var data assimilation system ftp data access to NCAR mass store variety of source configuration and wiki interfaces at NCAR.
4	Andrey Ptitsyn	andrey.ptitsyn@colostate.edu	Sequence analysis, genomics, functional genomics	Other	
5	Anton Betten	betten@math.colostate.edu	Discrete mathematics: classification problems for designs, codes, incidence structures, finite geometries.	Other	Own library of C++ classes, parallelizes well, runs best on clusters of many nodes.
6	Bogusz J Bienkiewicz	bogusz@engr.colostate.edu			
7	Brad Reisfeld	brad.reisfeld@colostate.edu	Molecular modeling and simulation	Other	Computational chemistry: AMPAC, gaussian, Jaguar
8	Bradly Thomas Johnson	bradlyt@colostate.edu	Building information modeling		
9	Bruce Draper	draper@cs.oclostate.edu	Computer Vision, and models of biological vision.	Eigensystems, other	Qt (www.trolltech.com), GSL (GNU Scientific Library)
10	Christos Papadopoulos	christos@cs.colostate.edu	Analysis of network traces, network simulation.	Linear regression, FFT/spectral methods	Matlab, custom scripts
11	Colette Heald	heald@atmos.colostate.edu	Atmospheric composition modeling related to pollution and climate	Finite difference, linear regression, non-linear regression, FFT/spectral methods, eigensystems, Monte Carlo, ordinary diff equations, partial diff equations	Earth System Model codes developed within the scientific community (non-commercial) and IDL
12	Craig A. Bond	craig.bond@colostate.edu	Could see applications in environmental and resource economics....complex, non-linear environmental modeling with human systems	Linear regression, non-linear regression, sparse linear systems, dense linear systems, eigensystems, Monte Carlo, partial diff equations, other	GAMS, Matlab, GAUSS, other statistical programs like NLOGIT, Eviews, STATA
13	Dennis Harry	dharry@warnercnr.colostate.edu	Research is focused on understanding the dynamic processes associated with deformation of the Earth's crust and lithosphere, and the associated magmatic consequences. Key aspects are the multi-physics nature of the problem and spatial and temporal bandwith (processes operating on microstructural scales impact results that we try to simulate on scales of 1000's of km, and processes operating on scales of days impact results that we try to simulate on scales of 100's of millions of years. Impacts of the research relate to global energy resources and geologicaal hazards.	Finite difference, finite element, FFT/spectral methods, ordinary diff equations, partial diff equations	Finite element and finite difference codes. Some are home-grown and will continue to be developed. A growing tendency in my field is to take advantage of software libraries that are sustained by international collaborative efforts and some more specific codes that are sustained by international collaborative efforts. This is a work in progress. I am Co-Chair of a working group that is part of the NSF-funded Computational Infrastructure in Geodynamics (CIG) to pursue these initiatives.
14	David Dandy	dandy@colostate.edu	Computational fluid dynamics, involving chemical vapor deposition, pyrolysis, and microfluidics	Finite difference, boundry, sparse linear systems, ordinary diff equations, partial diff equations, other	Fluent, and codes written by graduate students in the group
15	Domenico Bau	domenico.bau@colostate.edu	Combined optimization algorithms and groundwater simulation models	Finite element, sparse linear systems, Monte Carlo, partial diff equations	Finite Element Models for simulation of groundwater flow and transport in subsurface aquifers Multi-objective Genetic Algorithm

Table 10: Raw data results of the survey.

	Name	Email	Q1b (Field)	Q1c (Algorithm)	Q3 (Program)
16	Donald Estep	estep@math.colostate.edu	Engineering mechanics, e.g. fluid flow problems, thermal problems, MEMS scale devices, chemical reactions modeling of fusion and fission reactors climate effects on ecology atomistic and molecular simulation	Finite difference, finite element, sparse linear systems, Monte Carlo, functional evaluation, ordinary diff equations, partial diff equations	Our own
17	Edwin Chong	Edwin.Chong@ColoState.edu	Simulation of networks, simulation of sensing systems, Monte Carlo sampling, optimization and search algorithms.	Eigensystems, Monte Carlo, functional evaluation, partial diff equations, other	Matlab
18	Hochheimer	dieter@lamar.colostate.edu			N/A
19	James Smith	jtsmith@digitalglobe.com	Image processing	FFT/spectral methods, Monte Carlo	Image processing. Specifically, GIS algorithms for projection of satellite imagery.
20	Jeff Niemann	jniemann@engr.colostate.edu	Hydrologic modeling, uncertainty analyses, geomorphic modeling	Finite difference, partial diff equations	Monte Carlo simulations, simulations of geomorphic models that extend for 10,000 or more years.
21	Karen Kaminski	karen.kaminski@colostate.edu		None	
22	Kirk Sarell	kirks@business.colostate.edu			
23	Luis Briceno	ldbricen@colostate.edu	Resource allocation in heterogeneous computing.	FFT/spectral methods	Mostly simulations created using c++ or java. I usually use a perl script to enable multiple simulations.
24	Mark Easter	marke@nrel.colostate.edu	We use the ecosystem models to simulate greenhouse gas cycling between soils and the atmosphere.	Linear regression, non-linear regression, Monte Carlo, ordinary diff equations, partial diff equations	Perl, CENTURY ecosystem model, SATURN carbon cycling model, R (statistics)
25	Michael Kirby	kirby@math.colostate.edu	Applied mathematics, pattern analysis, modeling and simulation.	Finite difference, linear regression, non-linear regression, sparse linear systems, dense linear systems, FFT/spectral methods, eigensystems, Monte Carlo, functional evaluation, ordinary diff equations, partial diff equations, other	Programs for analyzing patterns in data as well as modeling and simulation.
26	Michelle Strout		I am working with the SAXS group on campus, where our goal is to simulate protein structure and movement so as to match the raw data gathered through small angle x-ray scattering. I also have been looking at ways to automate the generation of parallel code for Dave Randall's group global cloud resolution model. My research also includes developing and automating techniques for parallelizing irregular computations such as molecular dynamics simulations.	Sparse linear systems, partial diff equations	Moldyn as a molecular dynamics benchmark. Irregular Gauss-Seidel and Jacobi as iterative solver benchmarks. The Shallow Water Model implementation as it is related to the global cloud resolution model. We have just started development of a scattering model for the SAXS project.
27	Mike Lacy	michael.lacy@colostate.edu	Statistics	Monte Carlo	Stata. (This is the preferred statistical program for many social scientists today.) It has routines for mixed models that require Monte Carlo integration to evaluate likelihood functions, and some fairly problems can take weeks on an ordinary fast PC. Having a fast UNIX server to do this would be quite helpful. The algorithms are NOT something for which any research worker would try to write his own code in FORTRAN, C, etc.

Table 11: Raw data results of the survey.

	Name	Email	Q1b (Field)	Q1c (Algorithm)	Q3 (Program)
28	Richard Casey	richard.casey@colostate.edu	We support numerous research groups on campus. This includes Computer Aided Drug Design, molecular modeling, genomics, proteomics, metabolomics, phylogeny, molecular electronic structure theory, and similar research areas. There are several collaborative agreements developing or in-place where Bioinformatics Center resources could be, or are, in use including the Harvard Screening Facility, the Colorado Sequencing Consortium, the Rocky Mountain Regional Center of Excellence in Biodefense.	Finite difference, finite element, sparse linear systems, dense linear systems, FFT/spectral methods, eigensystems, Monte Carlo, partial diff equations, other	We have about 200 applications currently installed on the Bioinformatics cluster. Our staff and I assist faculty, graduate students, post-docs, and others in running these apps. A few key apps are: Complete OpenEye suite (www.eyesopen.com) Gaussian 98, 03, 03 Parallel Array-Track DCA Mr. Bayes Blast/Fast (numerous versions) R / Bioconductor Delta2D
29	Robin Reich	robin@warnercnr.colostate.edu			
30	Samee U. Khan	samee.khan@colostate.edu	Distributed systems	Linear regression, ordinary diff equations	Data replication methods for a large-scale network.
31	Shaun Case	shaun.case@colostate.edu	1) We currently use HPC systems at the University of Guelph Canada for Animal Disease/Epidemiology Simulation Modeling, however, we'd really like to be able to use the resources here at CSU, CoGrid for instance, as well for the NAADSM project, which is part of The Animal Population Health Intitute here at CSU, and a joint project with USDA, University of Guelph and the Candian USDA equivalent. 2) We use software for equation fitting, on single processor systems, to help analyze the resulting simulation data, however, to do this on an HPC system would be a huge benefit to us. 3) We'd really like to discuss the options of using CoGrid for our research, as it is located here at our own University.	Linear regression, non-linear regression, FFT/spectral methods, Monte Carlo, functional evaluation, other, not sure	We write our own simulation software, which currently uses a form of Monte Carlo simulation, which we will be building on in the future. Furture applications may contain neural network implementations, as well as, other forms of modeling.
32	Stephen Hayne	stephen.hayne@colostate.edu	Empirical analysis of experiment data, regression (etc.) modeling of ebay data	Linear regression, non-linear regression, Monte Carlo, ordinary diff equations, partial diff equations	
33	Steven Miller	miller@cira.colostate.edu	Nothing formal, but have worked on clusters for near real-time computing processing of satellite data, and have run radiative transfer models using MPI.	Monte Carlo	IDL
34	Steven Reising	Steven.Reising@ColoState.edu			
35	Taka Ito	ito@atmos.colostate.edu	Ocean modeling	Finite difference, partial diff equations	Ocean general circulation model
36	Tom Hilinski	tom.hilinski@colostate.edu	Currently: Modeling of plant-soil ecosystems, both single site and spatial grids. Grids can be small in cell count, or continental or global, at various resolutions. Grid cells are typically either 1/2 deg, 10 km, or 1 km on a side. A lower-48 US grid can be over 8 millions cells. Our grid framework allows for with-cell heterogeneity, which can greatly increase computational requirements.	Ordinary diff equations, other	Our ecosystem models. R, ArcInfo, and a large number of small utilities.
37	Wade Troxell	wade.troxell@colostate.edu		Finite difference, eigensystems, ordinary diff equations, partial diff equations	Vensim, Powersim