Project Shelob Summary

Keywords: GPU programming software and tools, GPU cluster, HPC and scientific computing, computer science education, workforce development

Shelob is a project for computer science research, education, training, and development built around a GPU cluster. The proposed cluster will deploy 72 next-generation NVIDIA "Kepler" GPUs. It will be used to develop a large set of open source GPU-enhanced software [**?**]: (i) the Cactus CaKernel – a general framework for the development of parallel scientific applications which can take advantage of heterogeneous architectures; (ii) Pluto compiler advancement – a compiler to generate effective code for heterogeneous systems; and (iii) ParalleX – a new systems paradigm being developed by the STE||AR group. It will also be used to develop a variety computational science applications, and in more than 20 distance learning courses and workshops to educate a new generation of CISE researchers.

The project will be headed by PI Honggao Liu and a select team of Co-PIs and Senior Investigators from the Louisiana State University Center for Computation and Technology. The PIs are investigators in several large multi-institutional projects, including Cactus, Pluto, STE||AR, and LA-SiGMA. The team members, researchers from other state institutions, and smaller CCT research efforts will use the Shelob cluster to advance GPU-enabled open source software.

Intellectual Merit: Heterogeneous computing environments pairing CPUs with GPUs have demonstrated their value for scientific computation, and are rapidly becoming the configuration of choice for new high performance computing systems. Although only 10 of the top 500 supercomputers employ GPUs, they include 3 of the list's top 5. NCSA, TACC and NCCS have announced their intent to build GPU enhanced supercomputers. These machines offer the potential for new discoveries in a wide range of fields. However, parallel programming is made difficult by the languages and methods needed to deal with high core count nodes, heterogeneous acceleration devices, and their interaction. The need for highly complex hybrid programming models presents an enormous software development crisis and requires a transformation in computational research and education to achieve efficient use.

The software developed during the Shelob project will be of significant benefit to the CISE communities. The Cactus framework represents a technique for building optimized, reusable components for incorporation into new programs. By extending existing modules (thorns) to utilize GPU methods, a user of the Cactus framework can immediately benefit without direct GPU programming knowledge. The Pluto compiler is a source-to-source transformation system that can generate code for general-purpose multicore systems as well as for GPUs, staring from C source with a few added directives. We will enhance Pluto to handle heterogeneous environments with multiple GPUs. The STE||AR group uses the ParalleX execution model and its implementation in the experimental runtime system HPX (High Performance ParalleX) to improve the reliability and programmability of data-intensive massively parallel computing platforms using GPUs in a user friendly way. The ParalleX notion of percolation provides for a perfect high level abstraction of massively data-parallel execution using GPUs. All applications built on top of HPX will immediately benefit from the results. LA-SiGMA is a statewide NSF funded computational materials science project involving nearly 50 faculty at 7 Louisiana campuses. LA-SiGMA has several computational teams developing a variety of computational materials science applications. The most active of these teams is the GPU team, which is developing GPU accelerated parallel materials science applications.

Broader Impact: The Shelob project will include training and education at all levels, from a Beowulf/GPU bootcamp for high school students to more than 20 annual CCT workshops and computational sciences distance learning courses for our students. The Shelob cluster will also be used in the CCT and LA-SiGMA REU and RET programs. The computational sciences community will benefit greatly from the distribution of the Cactus framework, the enhanced Pluto compiler, the ParalleX execution model, the LA-SIGMA materials science codes, and other scientific computing codes that will be developed on the cluster.

Project Shelob Description

1 Introduction

We propose to transform computer and computational science research and education throughout the state of Louisiana by graduating our users from a massively parallel paradigm to a heterogeneous parallel paradigm. The process involves open source GPU-enhanced software, a GPU enhanced run time environment, and a general framework for the development of GPU enhanced codes. Funds are requested to purchase a GPU-based heterogeneous cluster with 72 next-generation NVIDIA "Kepler" GPUs. Large multi-institutional projects including Cactus, Pluto, STE||AR, and LA-SiGMA, together with numerous smaller research efforts will use this machine to develop open source codes, the HPX GPU enabled runtime environment, and the CaKernel framework. Together, they will enable new discovery on the next generation of supercomputers. A comprehensive set of distance learning classes and workshops will use the machine to train a new generation of researchers and students across various disciplines.

Modern science and engineering disciplines are undergoing a profound transformation using cyberinfrastructure (CI). Louisiana has made great strides in the past ten years modernizing its educational and research facilities through comprehensive investments in CI. Ten years ago, during the administration of Governor Murphy J. Foster, Louisiana embarked on the long term goal of establishing itself as an international leader in computational sciences research and education. Today, LSU's Center for Computation and Technology (CCT), and the Louisiana Optical Network Initiative (LONI) boast of over 115 TF of parallel computing capacity including a TeraGrid node. The LONI Institute and its offspring the NSF funded LA-SiGMA project are products of this statewide investment.

High performance computing (HPC) has evolved dramatically since the launch of the LONI initiative. Multi-core processors have emerged as the computing engines in all commodity platforms. GPUs are becoming cost-effective and powerful processors for general-purpose computing. This has raised interest in GPU use for many scientific and engineering applications. In fact, the movement towards GPU-enhanced supercomputers has already begun. As of June 2011, three out of the top five computers on the Top500 list use GPUs to achieve their peek performance. This is significant as only 10 GPU machines are on the list, suggesting this number is likely to grow rapidly. In the United States, the National Center for Supercomputing Applications (NCSA) has recently deployed its 153 TF GPU cluster Forge, and Texas Advanced Computing Center (TACC) is planning to build a Peta-scale heterogeneous machine, Stampede, in 2012. The replacement for Jaguar at the Oak Ridge Leadership Computing Facility, Titan, is expected to run at over 20 PF by employing Kepler GPUs. In Europe, many supercomputing centers have recently provided access to heterogeneous supercomputers whose performance exceeds hundreds of TF.

Parallel programming used to be as "simple" as endeavoring to place one task per node. Once this task partitioning and the related data layout and communication scheduling issues were worked out, programmers could code for a node without cumbersome rules and restrictions. Current machine architectures are making things harder. The continued increase in single-thread performance between CPU generations has slowed due to insurmountable heat dissipation challenges. This has resulted in an increase in the processing core count per chip to maintain performance. More cores increase the burden on the programmers by adding another level of parallelization. Much more disruptive has been the maturing of GPUs for general purpose scientific computing. These devices provide several times the performance of contemporary CPU chips [**?**], but require extensive code modifications. The modifications are non-trivial: codes must be parallelized to a much greater degree and adapted for a baroque, spare, and uncompromising memory hierarchy. The CPU component must be adapted to manage execution and orchestrate data movement. Languages such as NVIDIA's proprietary CUDA (Compute Unified Device Architecture) [**?**] and Kronos' multi-vendor, multi-target (*e.g.*, GPU, many-core chip) OpenCL [**?**] have facilitated programming given these constraints, but they have not removed them. Despite the potential gains, hybrid programming involving CPUs and GPUs "simply" exacerbates software development challenges.

To meet these challenges, a new approach to heterogeneous programming and a concerted education effort is needed. The goal of this proposal is to obtain the funds for the purchase a Kepler GPU accelerated cluster which will allow the state to take the next step in supercomputing, and produce a workforce ready for the new generation of heterogeneous parallel computers like Forge, Stampede and Titan.

The researchers in LSU's CCT, HPC group, members of Cactus, Pluto, STE||AR, and LA-SiGMA projects, and the investigators in the proposal are well qualified to meet these goals. The CCT has administered a (now five years old) GPU cluster, Spider, with 32 NVIDIA Tesla GPUs since 2005. The LSU HPC group administers 16 CPU clusters totaling over 110 TF including the TeraGrid Queen Bee resource, in addition to a 2-node GPU cluster with 6 Fermi M2070 GPUs. The CCT events team includes 4 full time staff who host a variety of computational sciences workshops. Last summer, the CCT hosted 12 workshops each lasting from several days to a week. The Cactus project provides a general parallel framework for a variety of scientific applications with short ranged correlations. This request will enable Cactus to add elements to the framework which enable GPU acceleration, aiding all who use Cactus to quickly derive performance benefits. The Pluto compiler is a source-to-source transformation system that can generate code for general-purpose multicore systems as well as for GPUs, staring from C source with a few added directives. We will enhance Pluto to handle heterogeneous environments with multiple GPUs. The STE||AR group uses the ParalleX execution model and its implementation in the experimental runtime system HPX (High Performance ParalleX) to improve the reliability and programmability of data-intensive computing platforms using GPUs. The implementation of the ParalleX notion of percolation gives all applications written based on HPX the immediate and direct benefit of being seamlessly integrated with the special GPU accelerator hardware. LA-SiGMA is a statewide NSF funded computational materials science project involving nearly 50 faculty at 7 Louisiana campuses. LA-SiGMA has several computational teams developing a variety of computational materials science applications. The most active of these teams is the GPU team, which is developing GPU accelerated parallel materials science applications. LA-SiGMA is also responsible for a wide range of distance learning computational sciences courses in addition to an annual computational materials science workshop.

The machine we propose to purchase will include 72 Kepler GPUs on 24 nodes. Each node will have 2 sockets with 16 Sandy Bridge cores and a total of 1536 GB memory. It will be used both for code development by the Cactus, STE||AR, and LA-SiGMA teams as well as many other computational sciences projects, and for more than 20 computational sciences workshops and courses each year to train a new generation of computational sciences researchers to use heterogeneous computing to enable new discovery on the next generation of supercomputers.

The payoff from this project will be significant. Hundreds of researchers throughout the state of Louisiana will use the Shelob cluster to develop a new generation of codes including the Cactus general framework and the HPX runtime system. These codes will be ready when the next generation of GPU-accelerated supercomputers come on-line two to three years from now. In order to effectively use these supercomputers to enable new discovery, researchers will need to employ GPU-enhanced massively parallel codes. The Shelob cluster will be used to train a new generation of researchers to program using CUDA, OpenCL, PGI Accelerator and use the Pluto compiler in preparation for the new breed of supercomputer.

The remainder of this proposal is organized as follows. In the next section, Sec. 2 we describe the main projects that will contribute to this project. We describe the hardware to be purchased in Sec. 3, while in Sec. 4 we sketch how the Shelob cluster will be housed and maintained. In Sec. 5 we discuss the use of the Shelob cluster for research and education. Finally, we conclude in Sec. 6.

2 Contributing Projects

2.1 The LSU Center for Computation and Technology

The Center for Computation and Technology [**?**], or CCT, is an interdisciplinary research center located on the campus of Louisiana State University in Baton Rouge, Louisiana. CCT advances LSU's Flagship Agenda and promotes economic development for the state by using computational applications to aid research and develop solutions that benefit academia and industry.

CCT is an innovative research environment, advancing computational sciences, technologies and the disciplines they touch. CCT researchers include 48 faculty, 13 research staff, 9 postdocs, 21 support staff and nearly 60 students from many LSU departments and colleges. The CCT staff is composed of 70 experts in several areas including cyberinfrastructure and framework development, computer science, electrical and computing engineering, mathematics, astrophysics, numerical relativity, coastal modeling and simulation, computational fluid dynamics, computational biology, computational chemistry, computational materials science, and AVATAR (Arts, Visualization, Advanced Technologies and Research).

Researchers at the CCT use advanced cyberinfrastructure to enable research in many different fields. By uniting researchers from diverse disciplines, ideas and expertise are disseminated across LSU departments to foster knowledge and invention. Many of these research groups will use the Shelob cluster, both directly for their research or to help educate their students.

The CCT is also the host or the lead organization for large multi-institutional projects including the LONI Institute, the Cactus framework, the Pluto compiler, STE||AR, and LA-SiGMA, and for smaller single-institution projects including Coastal Modeling, Next-Generation GPU Design, and Geometric and Visual Computing. These projects are all described next.

2.2 LONI and LONI Institute

The Louisiana Optical Network Initiative [**?**], or LONI, brings 8 Louisiana research universities, 2 partner organizations, and 70 community colleges together for collaboration and resource sharing via a 40 Gbps optical network. Funded by the state of Louisiana, the LONI network and institute represents a unique resource for enabling computational science, research, and education. The LONI network's connections to Internet2, National LambdaRail (NLR), and the Arkansas Research and Education Optical Network (ARE-ON), enable effective access to national resources, such as NSF XSEDE. In addition, the LONI Institute provides computational science faculty and research staff at 6 member institutions, and offers coordinated training and education opportunities.

The LONI network is a 40 Gbps fiber optic network that runs throughout Louisiana and into Mississippi, connecting their research universities to one another enabling efficient and robust collaborations. It is funded by the State of Louisiana at $40 million over 10 years with an additional $10 million for computing resources totaling over 85 TF of total capacity. At the core of these resources is Queen Bee, a 50 TF Dell Linux cluster. The cluster contains 668 eight-core nodes, amounting to 5344 Intel Clovertown cores, with 5.4 TB (terabytes) of memory and 54.4 TB of local disk, and 425 TB of parallel filesystem space. Half of Queen Bee's computational cycles have been contributed to the TeraGrid community since LONI joined TeraGrid in 2007.

The LONI Institute [**?**] is a virtual organization composed of six Louisiana research universities that house HPC platforms connected by the LONI network. Member universities include LSU, Louisiana Tech, University of New Orleans, Southern, Tulane, and University of Louisiana at Lafayette. The Institute was created in 2007 with a grant of $7 million (matched by $8 million from the institutions) from the State's Board of Regents. The Institute's main research thrusts are in computational science, computational materials, and computational biology. Through the LONI Institute, twelve tenure-track faculty members and six computational scientists have been hired at the member institutions. These include two senior and highly respected computational materials researchers (Co-PI Mark Jarrell at LSU and Lawrence Pratt at Tulane). They are complemented by the world-renowned John Perdew at Tulane, whose many seminal contributions to Density Functional Theory are a major driving force for computational materials research.

The LONI Institute is a unique virtual organization in which hiring and other decisions are shared among the six institutions. The Institute drives research and education, making Louisiana much more competitive for industrial partnerships that depend on computing advances for their competitive edge. LONI Institute researchers initiate projects in cooperation with industry to advance economic development in Louisiana and work toward creating University-Industry Research Centers. An example of such statewide collaboration is the Louisiana Alliance for Simulation-Guided Materials Applications (LA-SiGMA) project. LA-SiGMA is funded by an NSF EPSCoR RII program, which is led by the LONI Institute and has broad-based participation from computational and experimental scientists and engineers across the state.

Over 100 researchers in the computing disciplines in the LONI Institute will access the Shelob cluster, both directly for their research or to help educate their students.

2.3 Cactus Framework on Heterogeneous Systems

Despite the growing trend to make GPUs available for supercomputing, there is a paucity of tools to fully exploit the hardware's potential. To address this issue, a team at LSU CCT, in collaboration with researchers from Poznan Supercomputing and Networking Center in Poland designed and implemented a programming framework, *CaKernel*. This framework supports development of large scale scientific applications which are primarily using stencil computations in their numerical kernels.

The Cactus computational framework is the foundation of CaKernel. Cactus [**? ? ?**], is an open-source, modular, highly portable, programming environment for collaborative research using high-performance computing. Cactus is distributed with a generic parallel computational toolkit providing parallelization, domain decomposition, coordinates, boundary conditions, interpolators, reduction operators, and efficient I/O in different data formats. There are at least 30 groups worldwide using Cactus for their research work in cosmology, astrophysics, computational fluid dynamics, coastal modeling, quantum gravity etc. The Cactus framework is a vital part of the Einstein Toolkit, an NSF-funded (CIGR) collaboration which enables large parts of the world-wide research in numerical relativity by providing the necessary core computational tools. Cactus is part of the software development effort for Blue Waters, and in particular the Cactus team is working with NCSA to produce development interfaces and paradigms for large scale simulation development. Through the LONI's HPCOPS (TeraGrid) award, the Cactus group is developing a Science Gateway which will include capabilities for simulation assembly, compilation and deployment across the NSF TeraGrid/XSEDE resources. Through the NSF EPSCOR CyberTools project, the Cactus team is involved in educational initiatives including an REU, K-12 education and graduate mentoring.

Automated Code Generation is an important technique to bridge the gap between a physics equation and its discretization on one hand, and an efficient implementation in a low-level language such as C, C++, CUDA or OpenCL on the other hand. We employ *Kranc* [**? ?**], which accepts a set of equations and a discretization specification in Mathematica syntax, and emits complete, self-contained Cactus modules. Kranc applies various transformations and optimizations (e.g. loop blocking, loop fission, multi-threading, loop unrolling, vectorization) as necessary for the target architecture. As Kranc's input describes the physics system at a very high level, it can perform optimizations that are inaccessible to a low-level language. In the Einstein Toolkit [**?**], Kranc is used to generate a highly efficient open-source implementation of the Einstein equations, a complex system of equations with more than 5,000 terms in its kernel. Current Kranc development includes using OpenCL as target language.

The CaKernel programming framework, although still in the development stage, addresses the following challenges facing the computational science community: portability, programmability, performance, and productivity. CaKernel has been designed and implemented in Cactus as a programming abstraction to develop highly efficient parallel scientific applications which can take advantage of heterogeneous architectures. Due to the flexibility and extensibility of the Cactus framework no changes to the Cactus flesh are necessary, guaranteeing that existing features and user-implemented modules are not adversely affected by this addition. In particular, the parallelization across GPUs is handled in Cactus via MPI without any modifications to the existing framework. Thanks to Cactus, the kernel code written in CaKernel are guaranteed to be portable on almost all supercomputers available for research. With the combination of a set of highly optimized templates and a kernel descriptor, CaKernel can automatically generate computational kernels in CUDA, OpenCL, C, and potentially other programming languages from developer's C-style numerical codes. The programmability of CaKernel is greatly improved via the automatic code generation enabled in CaKernel to reduce possible human errors in writing code, with the ever-growing complexity of scientific codes in mind. CaKernel not only makes it easy to write and execute computational kernels, but also makes it possible to optimize the kernel without changing the kernel code itself. The performance can be fine-tuned by modifying/swapping the templates or adjusting the kernel parameters. These highly optimized templates that are completely transparent to application developers hide the complexity of lower-level programming and optimization. This design enables software engineers or domain experts to contribute their expertise at the maximum scale, thus enhancing the overall programming productivity.

The Development of CaKernel has so far been carried out on a 8-node GPU cluster with 2 NVIDIA Tesla C1060 cards in each node. Due to complexity and wide scope of potential application codes, there is still a lot of research to be done before CaKernel reaches a level of maturity where good performance can be obtained on a regular basis with a minimum of effort. We are therefore in great need of a new quick turnaround GPU system that is big enough to enable routine performance and scalability tests. The new hardware architecture and development environment on the Shelob cluster will also enable us to prepare our researchers for the existing and planned national level GPU clusters, e.g., Forge and Titan.

2.4 Extending the Pluto compiler

As mentioned earlier, the emergence of GPUs as cost-effective and powerful processors for general-purpose computing has raised great interest in their use for many scientific and engineering applications but further exacerbate the software development challenges. Developing high-performance applications for GPUs and accelerators is even more complicated than programming general-purpose multicore processors. Further, there is currently no way of writing portable code that can execute on an NVIDIA GPU or an AMD/ATI GPU or the Cell processor – each has its own programming model. The need for a portable programming model to program GPUs/Accelerators prompted an industry-wide consortium to come together and create the OpenCL language as an extension to C. The Pluto compiler builds on this approach, seeking to mask the difficulties of multi-core and GPU programming [**? ? ? ? ? ? ?**].

We will enhance the Pluto compiler by adding the ability to automatically generate OpenCL code from annotated C programs provided by the user. The proposed work is motivated by recent advances in *polyhedral-based approaches* for powerful transformations of affine computations which have enabled the development of Pluto's automatic parallelization/optimization system [**?** **?**]. In addition, Pluto will be enhanced to explore automatic code generation for heterogeneous systems with multiple GPUs, where the GPUs can communicate directly (e.g., using the NVIDIA GPUDirect technology).

The proposed enhancements will be applicable over a broad range of scientific and engineering applications. For example, in doing these enhancements, we will work closely with application developers from the LA-SiGMA project. The Shelob cluster will be critical for compiler development, testing and training users to develop "performance programming" skills on next-generation heterogeneous systems. Such a system could result in significant productivity gains, and could improve the return-on-investment for existing and future hardware. As was done with the Pluto system, the OpenCL transformation system will be made publicly available.

2.5 The STE||AR Group

STE||AR stands for "Systems Technologies, Emergent Parallelism, and Algorithms Research" [**?**]. The centerpiece of this group's work is the ParalleX execution model and its implementation in the experimental runtime system HPX (High Performance ParalleX) [**?**]. The goal is to enable scientists and developers to write applications which expose more parallelism, scale better, and generally show improved performance compared to more conventional programming models such as MPI [**?**].

ParalleX is a new (and still experimental) parallel execution model. It's purpose is to overcome the limitations imposed by current hardware and software development practice. ParalleX synthesizes several ideas such as lightweight synchronization semantic constructs supporting message-driven computation in a global address space. The idea is to mitigate overheads, expose parallelism, and provide runtime adaptive scheduling and load balancing. It targets two types of applications. The first type includes those requiring excellent strong scaling, allowing for a dramatic reduction of execution time for fixed workloads. The second type covers those needing the highest levels of sustained performance through massive parallelism. Such applications are presently unable (through conventional practices) to effectively exploit more than a relatively small number of cores in a multi-core system. Such applications are unlikely to utilize high-end computing systems with the hundreds of millions of cores expected by the end of this decade.

Four factors are inhibiting these two forms of scalability: 1) starvation – the insufficiency of available useful work either globally or locally; 2) latency – the distance measured in time (e.g., cycles) for a remote access or service request; 3) overhead – the critical-time work required to manage parallel resources and concurrent tasks which would not be required for pure sequential execution; and 4) waiting – caused by contention or delays due to conflicts for shared physical or logical resources. It is recognized that each of these is a consequence of the fundamental strategy implicit in the execution model.

HPX (High Performance ParalleX) is an experimental, modular, and performance oriented representation of the ParalleX execution model targeted at conventional architectures (currently, Linux based systems, such as SMP nodes and conventional clusters) [**?**]. The most important design objective of HPX is to create a state-of-the-art parallel runtime system providing a solid foundation for highly scalable applications while remaining as efficient, as portable, and as modular as possible.

The overarching goal of using HPX in this project is to conduct fundamental research that will result in formal techniques and tools to enable the development of correct programs for data-intensive parallel computing. A major goal is to dramatically accelerate data intensive applications on future platforms of unprecedented scale, comprising multi/many core processor components, deep memory hierarchies and GPU based heterogeneous architectures. While we do not expect to provide all answers to the crisis looming in data-intensive/parallel computing, we plan to address the following questions in this project:

1. How can we reliably program data-intensive computing platforms based on GPUs to exploit massive parallelism and to serve best the varied tasks that may be executed on them?
2. How can we express high-level parallelism at this scale in a natural way for users?
3. What new programming abstractions (including models, languages and algorithms) can accentuate these fundamental capabilities?
4. How can programming systems for data-intensive computing platforms using GPUs be designed to support extremely high levels of concurrency, reliability, efficiency, and availability?

The ParalleX execution model exposes the interaction with and integration of accelerator hardware such as GPUs through one of its intrinsic models - percolation. Percolation utilizes the work queue model to minimize the number of system memory accesses which would have to be otherwise initiated by the GPUs to obtain new work with related operand data. This alleviates unnecessary processor stalls and enables the GPUs to utilize their internal execution resources optimally. Percolation allows to fully overlap and synchronize the computations in the GPU and the main cores by combining code and data into special ParalleX parcels (messages), which are being sent to the GPU for execution. This high level abstraction allows to integrate and fully parallelize the accelerated computation with the message driven, split-phase transaction execution on the main cores.

Our choice of ParalleX as the target platform and HPX as the implementation framework for our planned research is motivated by the key challenges of scalability, efficiency, power consumption, and reliability as well as programmability. We expect these challenges to dominate data-intensive systems towards the end of the decade, distinguishing them from the conventional platforms and enabling ultra-scale performance and capacity.

2.6 LA-SiGMA

The Louisiana Alliance for Simulation-Guided Materials Applications (LA-SiGMA) [**?**] is an NSF-sponsored virtual organization of seven state institutions of higher learning focusing on computational materials science. Members include nearly 50 faculty, 9 postdocs and nearly 70 students from Tulane, Xavier, University of new Orleans (UNO), LSU, Southern, Louisiana Tech, and Grambling. Three members (Xavier, Southern and Grambling) are historically black colleges or universities. This group is composed of three science drivers (SDs) using a set of common computational methods, combined with experimental studies, to study electronic and magnetic materials (SD 1), energy storage and generation materials (SD 2), and biomolecular materials for drug delivery (SD 3). The computer codes used range from those written by individual research groups to open source and commercial software packages. The goals of the electronic and magnetic materials SD are to develop and validate methods that enable the study of complex phenomena in correlated electronic and magnetic materials ranging from transition metal oxides to organic magnets. As a consequence of their exotic and diverse physical properties, transition metal oxides are considered to be the frontier of research on "emergent research device materials." The targets of the energy materials SD include electrode materials for supercapacitors, hydrogen storage materials, and catalytic process for the formation of biofuels. The goals of the biomolecular materials SD are to develop, apply, and validate experimentally multi-scale computational tools that will enable the design of novel drug delivery vehicles.

The glue that holds the three SDs together are the formalisms, algorithms and codes being developed in this project. The cybertools and cyberinfrastructure (CTCI) group of participants, led now by Co-PI Ramanujam, allows Alliance members to more efficiently utilize the next generation of 21st-century supercomputers. The SD teams work hand-in-hand with computer scientists, computer engineers, and applied mathematicians in CTCI to develop the formalisms, algorithms and codes needed to efficiently utilize the next generation of supercomputers. Members of the SDs together with CTCI are broken into several computational teams to address common codes and high-performance computing challenges.

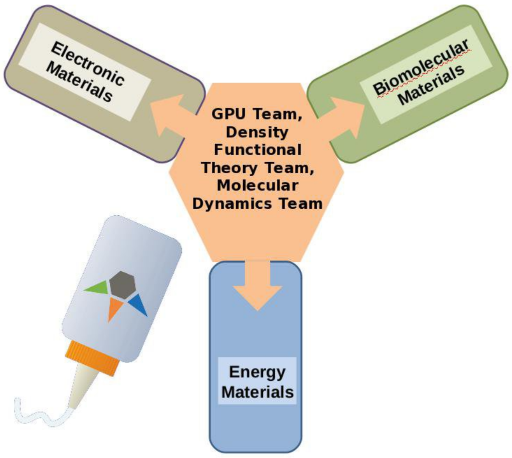

Figure 1: LA-SiGMA is composed of three Science Driver teams. The SD teams are glued together by the computational teams, the largest of which is the GPU team.

The largest of these computations teams is the GPU team, composed of over 30 LA-SiGMA students, post-docs, and faculty from 4 LA-SiGMA campuses working together to develop GPU accelerated codes. The GPU team is developing a series of GPU-enhanced codes that are heavily used in the community, including Hirsch-Fye Quantum and continuous time Monte Carlo, Density Matrix Renormalization Group, parallel tempering for glasses and molecular systems, Monte Carlo Simulations in Open Ensembles, and Variational Monte Carlo. Each code release will be available on our web portal and be accompanied by documentation and completely worked examples. In addition, the LA-SiGMA GPU team is coordinating merging of continuum simulation codes with the LAMMPS molecular dynamics code. Highly tuned code elements that have more general applicability, such as the thin rectangular matrix product and update needed by some of the Monte Carlo codes, will be released on their own as separate code modules. The development versions of all the codes will be continuously available through anonymous SVN access. Last summer many of the LA-SiGMA and CCT REU students and 5 faculty and students from the Louisiana School for Math, Sciences and the Arts (LSMSA) worked with the GPU team (c.f. Sec. 5).

Collaborations between LA-SiGMA students and post-docs are enhanced through an innovative Computational Science Collaboratorium which brings together CTCI, GPU team, and members of all the SD's and computational teams into the same space with resources for the development of the next generation of codes. The collaboratorium is located in the CCT on the LSU campus. This space contains cubicles for 15 LA-SiGMA students or post-docs in computer science, physics, chemistry, mechanical engineering, electrical and computer engineering, etc. There is a separate space for faculty and staff across the hall. The collaboratorium has a meeting area with three white boards for discussion as well as modern teleconferencing and videoconferencing equipment so that the meetings may be shared with other campuses. Each desk in the collaboratorium has a high-performance Intel Westmere desktop computer, each with a NVIDIA Fermi GPU card. In addition there is a 1TF multi-core workstation, with a Fermi M2070 GPU. These desktops and workstation are being used for the development of GPU accelerated codes. In addition to the Intel compilers, MPICH, and OpenMP, each of these machines also has PGI GPU accelerator compilers installed. The PGI compilers simplify the development of GPU codes, by allowing the programmer to add OpenMP-like compiler directives to existing Fortran or C programs that control the code execution on the GPU. This allows codes to be developed and tested quickly without the need CUDA, at least in the initial stages of development.

The impact of the Shelob project on the LA-SiGMA collaboration will be significant. One of the main goals of LA-SIGMA is to graduate its researchers from serial and parallel computing to the heterogeneous computing which will characterize the national leadership class capability computing in 2-3 years. The Shelob cluster will help us meet this goal, both as a machine we can use to design and test the codes being developed by the LA-SIGMA computational teams and as a machine which can be used in LA-SiGMA courses and workshops described in Sec. 5 Finally, and most significantly, the LA-SiGMA computational teams is developing a series of GPU-enhanced codes that are heavily used in the community. Each code release will be available on our web portal and be accompanied by documentation and completely worked examples.

2.7 Coastal Modeling

Coastal engineers and scientists have utilized GPU's in modeling the impacts of storm surges and water waves on civil infrastructures, such as levees, bridges, piers, and oil and gas drilling rigs. Applications include the 3D CFD model using the volume of fluid method for free surface flows and waves in the Cactus computational framework, and the Smooth Particle Hydrodynamic model for wave and structure interaction [**?**]. The LSU Department of Civil and Environmental Engineering has also developed Lattice Boltzmann models for shallow water flow and mass transport associated with dam break or levee breach using GPUs [**? ?**]. The Shelob cluster will enable coastal researchers at LSU to address rich and urgent scientific problems pertinent to Louisiana hurricane protection and coastal restoration.

2.8 Next-Generation GPU Project

Since the release of the NVIDIA G80 in 2006 [**?**] and with the rapid acceptance of CUDA with which the G80 could be programmed for non-graphical purposes, the needs of the scientific computing community have joined those of 3D graphics developers in driving the designs of GPUs. Subsequent GPU generations have increased their usefulness for scientific computing by refining useful features such as higher double-precision throughput, more flexible synchronization capabilities, and greater memory hierarchy flexibility [**?**]. These newer features focus on kernel performance itself, other new features of recent designs help improve GPU utilization by allowing multiple active kernels. Many proposed GPU enhancements are based on a system view in which the CPU plays only a supporting role [**? ?**]. Still, there are some problems which have some elements that perform well on GPUs, and other elements which execute more efficiently on CPUs; when these elements are fine-grained efficiency on hybrid GPU/CPU implementation of these problems suffer. Next generation GPUs will need to accommodate such problems.

The Next-Generation GPU Project led by SI Koppelman will investigate GPU and CPU designs which support very tight coupling between GPU and CPU to support hybrid execution. The starting point for this investigation will be a set of well tuned codes with deft GPU/CPU work divisions running on state-of-the-art production-scale hardware. This exactly describes the Shelob cluster and the codes to be developed for it. The Next-Generation GPU Project will leverage this experience by developing candidate next-generation GPU designs.

The project will look closely at which GPU and CPU features are useful for tightly coupled hybrid codes. Early programming techniques initially concentrated on shifting work to the GPU with little consideration given to CPU participation beyond shuttling data [**? ? ?**]. The emphasis is now shifting to problems that require, or at least can benefit from, a more careful division of work between the devices, see for example [**?**]. For some codes work division is complex as it must balance the benefit of execution efficiency on the preferred processor with the overhead of data migration. It is important to understand how the division is influenced by such things as data layout, synchronization, etc. Achieving this understanding will not just produce better tuned hybrid codes, but it will also provide hints for features to include in follow-on generations of GPUs.

It is easy to develop a list of candidate features for such a study. They include memory organization and bandwidth provisioning, interaction with low-level caches, and GPU scheduling flexibility. There are complementary issues for the CPU, including performance and capabilities of the vector (SIMD) unit and cache organizations. To go beyond these obvious features one needs extensive experience with tightly integrated GPU/CPU codes, such codes are currently being developed by the LA-SiGMA, CaKernel (Cactus), and STE||AR (HPX) projects, and development would move to the Shelob machine described in this proposal.

With the help of domain and computational experts the codes will be factored and tuned for execution on Shelob. In the process coding alternatives will be developed that exercise some of the system's features, and their performance will be carefully measured. This exploration process will provide us with well-tuned codes and the know-how for porting additional codes. It will also inevitably lead to ideas on how next-generation GPU/CPU compute clusters might be designed, as we find places where small additional or modified capabilities would lead the much better performance. These ideas will be pursued with simulation and other studies of next-generation GPUs.

Also, by exploring these issues in depth on the Shelob cluster, programming methodologies can be defined and contribute to the other elements of the project. Exposure to these issues will raise the experience levels of participating students and directly help support future systems.

2.9 Geometric and Visual Computing

The Shelob cluster will be very helpful to the geometric and visual computing group in LSU led by SI Li, especially on the digital archiving, modeling, and processing of massive geometric spatial and temporal data, such as urban scanning, architectural, archaeological, forensic scan data, and medical imaging data. The proposed heterogeneous computing environment can provide efficient data processing toolkits in facilitating and augmenting many important late-stage large scale geometric data processing tasks in information modeling and analysis.

Processing large-scale geometric data is usually computationally expensive and sometimes prohibitive. Instead of processing them directly, we are studying efficient domain decomposition techniques that partition the mapping and modeling problems into solvable sub-domains, and we are also developing processing algorithms and optimization solvers accordingly. The Shelob cluster will play an important role in speeding up several tasks whose realtime or interactive response are critical.

A few closely related projects include:

1. Massive Heterogeneous Volumetric Data Mapping for Urban Modeling and Environment Exploration: the scan raw data collected by vehicles/robots equipped with laser scanners need to be matched and mapped in a dynamical environment efficiently;
2. Digital Forensic Facial Reconstruction: we aim in automating assembly, completion, and modeling of excavated damaged/fragmented skeletons/skulls and other archaeological data in digital environment to benefit archaeological/forensic/medical tasks, where real-time semantic manipulations/feedback from domain experts and specialists is highly desirable;
3. MRI-guided Four-dimensional Lung Tumor Motion Modeling and Radiotherapy Management: the tumor movement prediction and temporal parametric motion model refinement need to be performed on real-time during radiotherapy delivery so that the radiation beam can correctly target the tumor without killing nearby normal tissues.

All these projects will require efficient parallel processing and can greatly benefit from the GPU-enhanced Shelob cluster.

3 Hardware Description

The Shelob cluster is a Hewlett-Packard SL230 cluster which include 24 compute nodes, one head/control node, and 100 TB scratch storage with a QDR InfiniBand interconnect. Shelob will deploy 72 next-generation NVIDIA Kepler GPUs. Kepler is a follow-on to Fermi and uses a 28 nm production process. The Kepler cards will triple the dual-precision floating point performance of Fermi and hit up to 6 double precision GFlops. It will have support for DDR3 and GDDR5 memory and a 128-bit memory interface.

Each of 24 compute nodes contains two 8-core 2.6 GHz Intel Xeon Sandy Bridge-EP E5-2670 processors, and 3 NVIDIA Kepler GPUs. There is one dedicated x16 PCIe bus for each of the three GPUs in a node. These GPUs will serve as companion processors to the multi-core CPUs. The Sandy Bridge-EP processors have a base clock speed of 2.6GHz with a 115 W TDP, 20MB of Level 3 cache memory, and each one of them could process up to 16 threads simultaneously using Intel's Hyper-Threading technology. The Sandy Bridge-EP processors will feature Turbo Boost technology, the AVX and AES instruction set extensions, as well as support for VT-x / VT-d / VT-c virtualization. The Sandy Bridge-EP chips also include up to 2 QPI (Quick Path Interconnect) links, 40 PCIe Gen3 lanes, 4 DMI 2.0 lanes, and an integrated quad-channel DDR3 memory controller supporting up to three DIMMs per channel.

One HP Proliant DL380 G7 server, containing identical processors, will provide 16 cores for head-node support. The system will run the Red Hat Enterprise Server Linux operating system. When operational, the system will deliver about 8 TF of CPU and over 140 TF of GPU peak performance.

Each compute node contains 64 GB (8×8GB) of memory. The memory subsystem uses Intel's QPI running at 8GTps over 4 memory channels. The memory consists of 8 8GB 2Rx8 PC3-10600R-9 RDIMM sticks. The RDIMMs provide a maximum speed of 1600MHz with an additional register that buffers the address and command signals thus reducing the electrical loading on the signals. By reducing the electrical load from each DIMM, more DIMMS can be populated on a given memory channel. The RDIMMs also have full support for ECC which provides commands and address data protection, and supports automatic retries when an error is detected for uninterrupted operation in the case of transient errors.

The Shelob cluster has three networks providing a high-speed InfiniBand interconnect, a console network and an administrative network. The QDR InfiniBand has a peak signal rate of 40Gbps and a peak data rate of 32Gbps. Each node has a Mellanox Connect-X2 InfiniBand adapter on the motherboard connecting to the enclosure. Each enclosure connects to a Voltaire QDR 36 Port InfiniBand switch resulting in a non-blocking fat-tree InfiniBand network. The console network is a 10/100Mbps Ethernet network that provides access to the iLO (integrated Lights Out) remote management port on each server. The console network with KVM provides an out-of-band management solution for the cluster. An HP ProCurve 2610-48 Ethernet switch is used to organize the management network in a simple star topology. The administrative network is a full 1Gbps Ethernet. It is used for tasks such as server provisioning, job scheduling, server monitoring and basic system administration. The complete system will occupy 1 42U 19in rack.

HP has significant experience and success in deploying supercomputer and high performance cluster solutions. HP offers a complete range of turnkey x86-64 solutions, some with GPUs, as well as extensive consulting and design services for HPC deployments. The Shelob cluster will include the HP Cluster Management Utility (CMU) for control, configuration and provisioning. The Shelob system will have 3-year hardware and software warranties which include on-site and remote support for installation with a response time of 12 hours, overnight part replacement, and next-business-day on-site hardware repair during University business hours (8:00am-6:00pm CST Monday through Friday).

4 Environment / Deployment

Implementation and Project Management The Project Management Board will consist of PI (Liu), four co-PIs (Jarrell, Kaiser, Brandt, and Ramanujam), and three Senior Investigators (Chen, Hall, and Koppelman). The project organization will be led by Liu and the remaining Co-PIs and investigators who represent senior leadership in different scientific disciplines across LSU: Honggao Liu, Deputy Director of CCT, Principle Investigator of HPCOPS and TeraGrid/XSEDE Site Lead for LONI; Mark Jarrell, Principle Investigator of the LONI Institute and LA-SiGMA, and Professor of Physics; Hartmut Kaiser, Team Lead of STE||AR group, and Research Professor of Computer Science; Steve Brandt; Team Lead of Cactus Computational Framework, and Research Professor of Computer Science; Jagannathan Ramanujam, Professor of Electrical and Computing Engineering; Qin Chen, Associate Professor of Civil and Environmental Engineering; Randall Hall, Professor of Chemistry; and David Koppelman, Associate Professor of Electrical and Computing Engineering.

PI Liu has been the Principal Investigator on the LSU/LONI's HPCOPS project funded by NSF in 2007 to bring LONI's HPC resources into the national TeraGrid, and on the TeraGrid Extension: Bridging to eXtreme Digital (XD) project to extend LONI's operations on TeraGrid/XSEDE through March 2012. Liu was the the Director of HPC at LSU and LONI in 2008-2011, and managed over 20 professional staff who maintained HPC hardware and software resources and provided supports to the research computing community. Liu has overseen all HPC activities and led the HPC development efforts at LSU and LONI, and has been instrumental in establishing HPC at LSU as a nationally recognized facility for providing HPC services and production cycles to researchers on campus, in the state, throughout the nation, and across the world.

SI James Lupo, HPC Enablement Manager at CCT, will serve as the primary project manager, reporting directly to PI Liu. Lupo will be responsible for coordinating all elements of the operation and implementation of the Shelob cluster. Lupo, the project manager for LONI's TeraGrid project, has supported some 2,000 active LSU, LONI, and TeraGrid users, with a small group of consultants.

From the experience of installing many clusters, we believe the full deployment of the Shelob cluster can be accomplished within 90 days from commitment of funds for the project. The leadership team is skilled and experienced in national and local computing research infrastructure, and will ensure that the acquisition, deployment and operations of the heterogeneous computing environment is smooth and successful.

Quality of the Physical Infrastructure The Shelob system will be housed in LSU's Frey Computing Services Center (CSC) machine room. The CSC has 14,000 square feet of raised floor space, and houses six HPC systems including Tezpur, a 360-node cluster. It currently has a 500KW UPS, two 120-ton chillers backed up by the campus chilled water loop, and a 1.25MW backup generator. There is sufficient capacity within the power backup system for the additional electrical load of the Shelob cluster. Two chillers are operated in redundant fail-over fashion, with one being adequate to carry the full heat load. There is sufficient automatic generator capacity to carry the CSC under full load for extended periods of time. LSU's Network Operation Center (NOC) monitors the climate in the machine room continuously. All critical electrical, HVAC, fire detection and fire protection equipment, leak detection, temperature, humidity and security systems are continuously monitored in the NOC on a 24 x 7 x 365 basis.

Technical Expertise Needed and Available LSU HPC and CCT staff have decades of experience in running medium to large cluster systems. Red Hat Enterprise Server is already installed on a number of our clusters, and our system administration staff is quite familiar with it. With funding requested in this proposal, an experienced LSU system administrator will dedicate their effort to administration of the Shelob cluster for the project's full three-year duration.

Plans for Instrument Operations and Maintenance The Shelob cluster's batch queues will, unless it unexpectedly proves problematic, share LDAP user information and allocation constraints with other LSU HPC compute clusters. In the project's first six months, the PI and staff will study usage policies from other interactive clusters toward determining a phase 1 policy for managing interactive time allocations and procedures. A oversight committee consisting of the PIs and co-PIs will meet twice yearly to review these cluster usage policies, and give feedback on possible improvements.

5 Outreach and Education Plan

Training and education at all levels, from primary school through graduate school and beyond, is an essential component of this project. Programs are offered both to train a core set of users, who will develop codes for the Shelob cluster and for the next generation of heterogeneous machines such as Forge and Titan, and to develop a pipeline to sustain the project in the future. The proposed Shelob cluster will be used for summer workshops to train students to use PGI GPU accelerator compilers and CUDA. GPUs will be introduced as components of the machines built and used by high school students in our long-running Beowulf bootcamp. To help establish a pipeline into our graduate programs, the Shelob cluster will be used by undergraduate students in the CCT and LA-SiGMA REU programs. To establish a pipeline into our undergraduate programs, K-12 local teachers participating in LA-SIGMA summer RET programs will also use the cluster.

Beowulf/GPU Bootcamp. For the past four years the CCT has offered a week-long camp to introduce high school students to the field of supercomputing. In this camp students are given hands-on experience with all aspects of supercomputing, from the hardware to the applications level. They interact with research professors and learn about new career paths.

During the week, students take apart and reassemble compute nodes, install the operating system and system software, and learn to program their new cluster with Python + MPI. For many students, this will be their first exposure to programming, for others it is a chance to learn about parallel programming and its application to science. Students learn about different forms of parallelism at the bootcamp, and in part this activity is carried out with an interactive calculation of vector addition or the heat equation. Students also hear lectures from domain scientists, describing research of special interest to the Louisiana area, from hurricane forecasting to gravitational wave signals (i.e. the LIGO observatory in Livingston Parish).

This year we will update the program by including a discussion of GPUs and their importance in modern computing. The students will not only take apart and reassemble GPU hardware similar to what will be in the Shelob compute nodes, but will explore simple data parallel programming of GPUs from within Python. Students will run their programs on Shelob, and gain experience about modern scientific research. Students also will be taken on a tour of the machine room facilities and see Shelob firsthand.

Research Experiences for Undergraduates (REU). Both LA-SiGMA and the CCT offer REU programs sponsored by the NSF. They are nine-week programs where students work collaboratively on a wide variety of computational science projects, and become familiar with interdisciplinary research. The LA-SiGMA REU program recently completed its first year. It involves 30 students every summer, 5 students at each of the following Louisiana institutions: Louisiana State University, Louisiana Tech University, Southern University, Tulane University, University of New Orleans, and Xavier University. The CCT REU is in its second year. During the summer of 2010 its NSF funding was complemented with additional funds from the Board of Regents to support 15 college students from all over the States. Another 10 students participated in the program this last summer. Three high school students also joined the REU programs last summer.

Students work with the faculty and research staff at CCT and LA-SiGMA, learning how to use the cutting-edge cyberinfrastructure on campus to examine various science phenomena such as gravitational waves that result from colliding black holes, explore new materials for energy storage or revolutionary electronic devices, or to design new kinds of physical interaction devices to extend computer visualizations. Since most research groups at the CCT collaborate with international researchers, REU students are exposed to how international collaborations work. Last year, nearly half of these students worked with the GPU team using the most current cyberinfrastructure tools with individually designed training sessions targeted to their specific degree of preparation. These students will be able to develop their codes on the Shelob GPU cluster, and participate in summer workshops (see below) which will employ the cluster to teach heterogeneous programming and parallel throughput. We hope that these students will return to LSU for graduate school so that they can continue to develop their codes and deploy them on future heterogeneous supercomputers.

The last two summers, the visit of Edward Seidel, Assistant Director for Mathematical and Physical Sciences at NSF and previous Director of the CCT, was a highlight of the program. Seidel participated in the final presentations of the REU students and addressed all REU students on campus in the evening both years.

Research Experiences for Teachers (RET) programs associated with the LA-SiGMA project are offered annually in New Orleans, Baton Rouge, and Ruston. For example, in the summer of 2011, three teachers were selected for the LSU's program, three for Southern University's, two for UNO's, and five for LA Tech/GSU's program. These teachers work at a mix of public and private schools. Three of the teachers are faculty members at the Louisiana School for Math, Science, and the Arts (LSMSA). LSMSA is the state's flagship high school, a boarding school for gifted and talented grades 10-12 Louisiana students. LSMSA sends the majority of its graduates to LA universities, mostly to LSU. Two of the LSMSA faculty were members and active participants in the GPU team last summer. They developed classroom demonstration modules, and they are using the present set of GPU workstations. These LSMSA faculty have committed to return next summer and help us to extend this program to other LA high school students.

Distance learning graduate level classes are shared via synchronous video and are available to students throughout the state. They are, for the most part, specialized graduate courses that emphasize computational methods that could not be taught at the individual institutions due to the lack of a critical mass. Last year, nine graduate courses were offered, in many cases involving faculty from different locations:

- *Computational Physics*: Computing for Petascale Systems by Karen Tomko (Ohio Supercomputer Center) and Juana Moreno (LSU).
- *Simulations of Quantum Many-Body Systems* by Matthias Troyer (ETH Zurich) and faculty at the Ludwig-Maximilians-Universität, Munich; École normale supérieure, Paris; Max Planck Institute for the Physics of Complex systems, Dresden; University of Massachusetts, Amherst; University of Wyoming; and Louisiana State University (Co-PI Jarrell).
- *Advanced Solid State Physics with Computation* by Co-PI Mark Jarrell (LSU).
- *Computational Solid State Physics* by Rongying Jin (LSU) and Cyrill Slezak (Hillsdale College, Michigan).
- *Introduction to Scientific Computing* by Gabrielle Allen (LSU, NSF), and Frank Löffler (LSU).
- *GPU Programming* by SI David Koppelman (LSU), Programming GPUs for graphics and non-graphical computations using OpenGL, OpenGL Shader Language, and CUDA,
- *GPU Microarchitecture* by SI David Koppelman (LSU), Low-level design and programming of GPUs, including organization, memory hierarchy, and native instruction sets, with examples.
- *Performance Optimization of Applications on GPUs* by Co-PI Ramanujam (LSU): Optimization of science applications, GPUs, heterogeneous systems, CUDA, OpenCL
- *Program Parallelization* by Co-PI Ramanujam (LSU): Compiler optimizations, source-to-source transformations. Programming paradigms and support for HPC architectures.

This fall we have added two more courses:

- *Condensed Matter Physics* by Adrienn Ruzsinszky (Tulane).
- *Statistical Mechanics* by SI Randall Hall (LSU) and Colin Wick (LATech).

Computational Sciences Workshops are run by the CCT continuously throughout the summer. Last summer, we held 12 workshops on a broad range of subjects. These included two workshops from the Virtual School of Computational Science and Engineering (one on GPU programming and the other in MPI/OpenMP), introductory workshops on ParalleX and HPX, a Shodor workshop on Computational Thinking from a Parallel Perspective, a week long hands-on workshop on DFT methods, two one-week workshop for middle school girls (Alice in Computerland), and several others. In addition, intensive training for the REU programs discussed above involves workshop in basic Unix/Linux, C and C++ programming, MPI, OpenMP, CUDA, an introduction to Cactus, etc. which are open to non-REU students.

HPC training is provided throughout the semester, open to anyone at LSU and across the state. It not only includes tutorials on HPC, MPI, OpenMP, Parallel programming concepts, debugging with TotalView and

DDT, grid computing, the Cactus framework, CUDA programming, Shell scripting, but also science-specific tutorials, e.g., on “Electronic Structure Calculations in Quantum Chemistry” and “Scientific Workflow and Visualization in VisTrails”. The Shelob cluster will have a huge impact on these tutorials, courses and workshops. All of the distance learning courses listed above involve computational projects, which will utilize the cluster. It will also be heavily utilized in the workshops. Furthermore, since the Shelob cluster will be under local control it can be “reserved” to meet the needs of these activities.

6 Conclusion

We propose to purchase a Kepler-based GPU cluster that will be the centerpiece of an effort to transform computational sciences education and research throughout the state of Louisiana. Large scale multi-institutional projects including Cactus, Pluto, STE||AR, and LA-SiGMA, together with numerous smaller research efforts will use this machine to develop open source codes, a GPU-enhanced run time environment, as well as a general framework which will enable new discovery on the next generation of supercomputers. A comprehensive set of distance learning classes and workshops will use the machine to train a new generation of researchers and students across various disciplines.

The movement towards GPU-enhanced multi-core supercomputers has already been made, creating an enormous software development crisis. As of June 2011, three out of the top five fastest computers in the world use GPUs to achieve their peek-performance. While there are only ten GPU machines on the current Top500 list, this number is likely to grow rapidly. In the United States, the NCSA has recently deployed its 153 TF GPU cluster, and TACC is planning to build a Peta-scale heterogeneous machine Stampede in 2012. The replacement for Jaguar at the Oak Ridge Leadership Computing Facility, Titan, which is expected to run at over 20 PF, will employ Kepler GPUs. There is a great need for open source, well documented codes to run on these machines as well as the Cactus framework which will accelerate the development of a wide variety of codes. This is matched by the need for a comprehensive set of computational sciences courses and workshops to train the next generation of researchers to use these codes and machines to enable new discovery on the next generation of supercomputers.

The researchers in the LSU’s CCT, HPC group, members of the Cactus, Pluto, STE||AR, and LA-SiGMA projects, and the investigators in the proposal are well qualified to address this need. The LSU HPC group and the CCT administers most of the capacity of the state’s LONI network, including a TeraGrid node. The CCT events team includes 4 full time staff who host a variety of computational sciences workshops. Last summer, the CCT hosted 12 workshops each lasting from several days to a week. The Cactus project provides a general parallel framework for a variety of physics applications with short ranged correlations. The Pluto compiler will be enhanced to handle heterogeneous environments with multiple GPUs. The STE||AR group uses the ParalleX execution model and its implementation in the experimental runtime system HPX to improve the reliability and programmability of data-intensive computing platforms using GPUs. The implementation of the ParalleX notion of percolation gives all applications written based on HPX the immediate and direct benefit of being seamlessly integrated with the special GPU accelerator hardware. LA-SiGMA is a statewide NSF funded computational materials science project involving nearly 50 faculty at 7 Louisiana campuses. LA-SiGMA has several computational teams developing a variety of computational materials science applications. The most active of these teams is the GPU team, which is developing GPU accelerated parallel materials science applications. LA-SiGMA is also responsible for a wide range of distance learning computational sciences courses in addition to an annual computational materials science workshop.

The payoff from this project will be significant. Hundreds of researchers throughout the state of Louisiana will use the Shelob cluster to develop a new generation of open source codes including the Cactus general framework. These codes will be ready when the next generation of GPU-accelerated supercomputers come on line two to three years from now. In order to effectively use these supercomputers to enable new discovery, researchers will need to employ GPU-enhanced massively parallel codes. The Shelob cluster will be used to train a new generation of researchers to program using CUDA, OpenCL, Pluto compiler, and PGI Accelerator in preparation for the new breed of supercomputer. They will uniquely enable new discoveries from effective use of the machines.

Project Shelob Data Management Plan

Access, Sharing, and Intellectual Property Issues With few exceptions data will be publicly available. This includes anonymous read-only access to the code depository. Any user will be able to obtain development versions as well as release versions of the code. The codes are to be publicly available under an open-source license (such as GPL) and we expect that contributors will not provide code that cannot be distributed under this license. All contributions will be reviewed and approved, providing a means to detect violations.

Privacy A privacy statement will be provided to users of the Shelob project wiki and repository. That statement will explain that posts are public, which should be obvious to users. Code contributors will be made aware of the license.

Preservation and Archiving The software packages themselves will be maintained in an SVN repository, so all released packages are preserved to maintain historical context. The servers are routinely backed up, protecting against system failure and vandalism. The scratch file system on Shelob uses RAID technology to minimize the chance of loss of data and code undergoing active development. The repository and wiki will be hosted on CCT systems, and are expected to be supported long after the life of the grant.

The Center of Computation and Technology is currently in the process of setting up a general solution for data management. This solution will be published at http://www.cct.lsu.edu/NSF_DataManagement_Plan.

Project Shelob Supplementary Documents

Personnel Involved in Project:

1. Honggao Liu; Louisiana State University; PI
2. Mark S. Jarrell; Louisiana State University; Co-PI
3. Hartmut Kaiser; Louisiana State University; Co-PI
4. Steven Brandt; Louisiana State University; Co-PI
5. Jagannathan Ramanujam; Louisiana State University; Co-PI
6. Qin Chen; Louisiana State University; SI
7. Randall W. Hall; Louisiana State University; SI
8. James A. Lupo; Louisiana State University; SI
9. David Koppelman; Louisiana State University; SI
10. Xin Li; Louisiana State University; SI
11. Juana Moreno; Louisiana State University; SI
12. Mayank Tyagi; Louisiana State University; SI
13. Jian Tao; Louisiana State University; SI
14. Peter Diener; Louisiana State University; SI
15. Frank Löffler; Louisiana State University; SI
16. Ravi Paruchuri; Louisiana State University; SI
17. Bhupender Thakur; Louisiana State University; SI

Project Shelob References