Presentations:

“The 10x10 Project: Can we have both extreme energy efficiency and programmability?”

Andrew A. Chien, University of Chicago

Heterogeneous architectures promise as much as 100-fold energy benefits, but significant software, programmability, and algorithm challenges must be solved in co-design with these heterogeneous architectures. The 10x10 project is an aggressive approach to heterogeneity, exploiting "dark silicon". The 10x10 paradigm is a principled, systematic approach to heterogeneity in computer architecture. A 10x10 architecture exploits deep workload analysis to drive co-design of a federated heterogeneous architecture that exploits customization for energy efficiency, but federates a set of customized engines to achieve general-purpose coverage. The 10x10 team includes Prof. Wenmei Hwu (UIUC) and Dr. Laura Carrington (SDSC/PMaC Labs), and we gratefully acknowledge the support of the National Science Foundation and the DARPA PERFECT program.

“Collectively Achieve Fairest Distribution of Discrete Resources with Limited Sharing: Complexity and Protocols”

Wenjing Rao, University of Illinois at Chicago

We focus on the problem of decentralized resource allocation in a locally connected network. Bounded by the network constraint, each discrete resource can only be allocated to one of the local agents. However, a global "fairest distribution" has to be achieved, in a decentralized way, despite the connectivity constraint. Our research indicates that, not only does a fairest distribution exists (that is independent of the choice of a specific fairness index), such a global optimum can be approached via greedy steps of "local balancing" operations as well. We propose some decentralized protocols that can efficiently converge to a near-optimal solution by taking advantage of concurrent resource transfers via localized communication.

“Warehouse-scale Computing: Challenges and Opportunities”

Nicholas Kidd, Google, Inc.

Warehouse-scale computers power the services offered by companies such as Google, Amazon, Yahoo, and Microsoft’s online services division. They differ significantly from traditional datacenters: they belong to a single organization, use a relatively homogeneous hardware and system software platform, and share a common systems management layer. Most importantly, WSCs run a smaller number of very large applications (or Internet services), and the common resource management infrastructure allows significant deployment flexibility. The requirements of homogeneity, single-organization control, reliability, and enhanced focus on cost efficiency motivate designers to take new approaches in constructing and operating these systems. In this talk, I will discuss some of the challenges with WSCs, such as maintaining high availability and low "tail" latency for all the services on which your computation depends. WSCs also bring many opportunities, and I'll discuss a couple of ongoing projects at Google-Madison to bring those opportunities to bear. I'll also discuss why you should come to work for Google, and Madison, WI in particular!

Scaling Up Without Blowing Up

Douglas Thain, University of Notre Dame

A variety of distributed computer systems enable users to scale up scientific applications to hundreds to thousands of nodes. However, a universal problem across many systems is that, at some critical scale, the application or the system fails in some serious way that wastes computing resources and troubleshooting effort. I will give an overview of several related efforts at Notre Dame to design systems and techniques for distributed computing that are effective at scaling up real applications to productive levels, without creating a computational disaster.

“The Case for Limping-Hardware Tolerant Clouds”

Haryadi Gunawi, University of Chicago

With the advent of scalable parallel computing, thousands of devices are connected and managed collectively. This era is confronted with a new challenge: performance failure; systems often perform worse than expected due to large-scale management issues such as hardware failures, software bugs, and configuration mistakes. In this proposal, we target one overlooked cause of performance failure: limping hardware -- hardware whose performance degrades significantly compared to its specification. The growing complexity of technology scaling, manufacturing, design logic, usage, and operating environment increases the occurrence of limping hardware. We believe this trend will continue, and the concept of performance perfect hardware no longer holds. Therefore, we advocate that limping hardware should be considered as a new and important failure mode that future distributed systems should deal with. To corroborate this conjecture, in this talk, we raise and answer the following questions: What are the cases of limping hardware observed in production? What are the impacts on deployed systems? What are the design flaws? How should future generation systems manage limping hardware? Based on our initial findings, we conclude that although today's scale-out systems employ redundancies, they are not capable of making limping hardware "fail in place". Users, administrators, and developers often use laborious time-consuming ad-hoc methods to pinpoint and replace limping hardware. As a result, performance failures cascade, productivity is reduced, and energy is wasted. This leads us to introduce the concept of limping-hardware tolerant clouds in which scale-out systems can properly anticipate, detect, recover, and utilize various cases of limping hardware, isolating the negative implications from user applications.

“Toward Smart HPC via Active Learning and Intelligent Scheduling”

Zhiling Lan, Illinois Institute of Technology

As HPC systems and their applications are growing large, complex and dynamic, it becomes essential to incorporate much more intelligence into resource control and scheduling. In this talk, I will present our on-going research projects to achieve smart HPC in which information about resources and applications will be automatically gathered, distributed, and acted on for improving performance, reliability, and energy efficiency. Specifically, our research focuses on two main thrusts: (1) active learning to provide multilevel analysis of workload characteristics, power requirements, fault/error/failure patterns, and performance-reliability-power tradeoffs from disparate data; and (2) intelligent scheduling to provide automated and self-optimized management of system resources and workloads based on the learned knowledge.

“Pushing experiments to the Internet's edge”

Fabian E. Bustamante, Northwestern University

This talk will present Dasu, a measurement experimentation platform for the Internet’s edge. Dasu supports both controlled network experimentation and broadband characterization, building on public interest on the latter to gain the adoption necessary for the former. I discuss some of the challenges we faced building a platform for the Internet’s edge, describe our current design and implementation, and illustrate the unique perspective it brings to Internet measurement. Dasu has been publicly available since July 2010 and has been installed by over 90,000 users with a heterogeneous set of connections spreading across 1,802 networks and 147 countries.

“Increasing Network Resiliency by Optimally Assigning Diverse Variants to Routing Nodes”

Cristina Nita-Rotaru, Purdue University

Cloud computing made the global IT infrastructure become dependent on a relatively small number of very large distributed systems managed as clouds. To achieve adequate scale and availability, cloud computing systems need two distributed system capabilities: consistent global state replicated across the network; and a distributed messaging system that connects cloud components. However, both systems are vulnerable to intrusions and the algorithms and tools necessary to build them at cloud scale, guaranteeing their integrity and performance under intrusion attacks, do not exist in practice. Networks with homogeneous routing nodes are constantly at risk as any vulnerability found against a single routing node could be used to compromise all routing nodes. Introducing diversity among routing nodes can be used to address this problem. With only a limited number of manageable variants (e.g. several different operating systems), the choice of assignment of these variants to routing nodes is critical to the overall network resiliency. Intuitively, one might think a random assignment of these limited variants would do well, but our results show a random assignment can result in a system with poor resiliency. We present the Diversity Assignment Problem (DAP), the assignment of variants to nodes in a network, and we show how to compute the optimal solution in medium-size networks. We also present a greedy approximation to DAP that scales well to large graphs. Our solution shows that a high level of overall network resiliency can be obtained even from variants that are weak on their own. For real-world systems that grow incrementally over time, we provide an online version of our solution. Lastly, we provide a variation of our solution that is tunable for specific applications (e.g., BFT). This work will appear in DSN 2013.

“Parallel Networks and Storage for Predictable End-to-End Data Movement”

Eun-Sung Jung, Argonne National Laboratory

The need in scientific communities to rapidly and reliably share large datasets among several distributed institutions are growing. These data transfers move over parallel and diverse network paths, involve parallel storage systems, and vary widely with respect to their file size and number. Such requirements demand more sophisticated data transfer tools whose features are capable of dealing with the complexities of the end-to-end path in terms of parallelism and diverse configuration options; optimize for heterogeneous networks with different characteristics requiring different protocols and configurations; contain I/O mechanisms to efficiently read and write to parallel storage; and handle the varying demands of widely different data transfer workloads.

We have been developing algorithms and mechanisms to meet the aforementioned requirements in the following areas:

- Exploiting the parallelism and heterogeneity in the network infrastructure for data movement.
- Leveraging parallel I/O interfaces to improve the performance of reading/writing to storage.
- Reliable parallel end-to-end data transfer combining network data movement and I/O to storage.

First, multiple network paths between the sites may be involved in the data transfer and these paths may differ in their latency and bandwidth. Sophisticated algorithms that exploit this parallelism and diversity in the network paths are needed to achieve efficient data movement. Second, file systems are increasingly parallel. We need to exploit parallel I/O interfaces routinely used in high-performance computing (HPC) to efficiently read and write from storage systems. Finally, when moving large quantities of data across end-to-end storage system-to-storage system paths, it is essential to do end-to-end checksum verification. Due to the incurring overhead, parallel checksum would be a possible solution for the problem. Putting all together, high speed end-to-end data transfer would be attainable.

“System Design Considerations of Big Data Processing”

Xian-He Sun, Illinois Institute of Technology

Technology advances are unbalanced. CPU performance has been improving at a much faster pace than memory and storage technologies during the last three decades, which has led to the so-called memory-wall problem. In the meantime, newly emerged applications, such as online gaming, social networks, and Cloud computing, are all data intensive, which has led to the so-called big-data problem. The lasting memory-wall problem compounded with the newly emerged big-data problem has changed the landscape of computing. CPU speed is no longer the performance bottleneck of a computing system, the data access speed is, whereas the data access speed is limited by the performance of memory and file systems. However, historically computing systems are designed and developed to utilize CPU performance, not the sustained data accessing. A paradigm change is needed to support data-centric computing. In this talk we first review the history and concepts of the big-data and memory-wall problems. We then discuss the challenges of design advanced memory and I/O systems for data-centric computing. Finally, we present some our recent results in understanding and optimizing the performance of memory and I/O systems from the data-centric point-of-view.

“Towards a Trustworthy Android Ecosystem”

Yan Chen, Northwestern University

Mobile security has become a big threat. In this talk, we first present the mobile seucrity work in the Lab of Internet and Security Technology (LIST) at Northwestern University. Then we focus on one recent work on mobile malware detection. We evaluate the state-of-the-art commercial mobile anti-malware products for Android and test how resistant they are against various common obfuscation techniques (even with known malware). Such an evaluation is important for not only measuring the available defense against mobile malware threats but also proposing effective, next-generation solutions. We developed DroidChameleon, a systematic framework with various transformation techniques, and used it for our study. Our results on ten popular commercial anti-malware applications for Android are worrisome: none of these tools is resistant against common malware transformation techniques. Moreover, the transformations are simple in most cases and anti-malware tools make little effort to provide transformation-resilient detection. Finally, in the light of our results, we propose possible remedies for improving the current state of malware detection on mobile devices.

“Recent Research in the V3VEE Project”

Peter Dinda, Northwestern University

(VMM) framework for modern architectures. Our current effort, the Palacios VMM, is a publicly available open-source codebase that can be embedded into various host OSes, including Linux and the Kitten lightweight kernel. Palacios is on track for its 5th release this summer. It has been used for a range of research at Northwestern, including the design, implementation, and evaluation of very low overhead techniques for virtualizing supercomputers at scale, achieving very low overheads in overlay networking on high-end interconnects, achieving scalable tracking of memory content sharing across a large machine, and realizing power-, energy-, and performance-directed adaptive computing on NUMA hardware. This talk will present the V3VEE project, Palacios, and then touch on three current research efforts:

- Guarded execution of privileged code in the guest
- Adaptive memory mapping on NUMA machines to optimize for power, energy, or performance
- VMM-based emulation of Intel's hardware transactional memory specification

We anticipate that posters or other presentations will expand on these and other topics.

“Ethos: A Layered Approach to Secure Applications”

Jon A Solworth, University of Illinois at Chicago

Traditionally, operating systems have provied a low-level interface in which application programmers were free to implement any functionality. This has obvious benefits for performance, as described by the end-to-end principle. But it has extraordinarily negative security implications, where vulnerabilities in any application can lead to system-wide compromise. Ethos is an OS designed and implemented to provide high security to applications. It does this in part by providing strong, yet simple security services including authorization, authentication, and encryption (including for networking). Ethos is also designed to have more abstract interfaces with much cleaner failure semantics, eliminating many "gotchas" which have resulted in security holes. These interfaces guarantee many security properties to every applications while simplifying application programming. This talk gives an overview of Ethos semantics and implementation, and describes it security benefits.

“Coordinating Application and System Adaptation for Power, Performance, and Accuracy Management”

Hank Hoffmann, University of Chicago

Many applications support tradeoffs between the accuracy of their results and their resource requirements (e.g., time, power, energy). Similarly, many computing systems (i.e., hardware and system software) support tradeoffs in the performance delivered to applications and their resource usage. Adaptive, or autonomic techniques, can be used to automatically configure an application or system in its tradeoff space and maintain satisfactory behavior in the face of unpredictable events (e.g., component failure or fluctuations in application workload). As the number of such adaptive applications and systems increases, they are increasingly likely to interact, raising the question: Can coordinated management of both application and system produce better outcomes than allowing each piece to adapt without knowledge of the other? This talk describes a general framework for creating adaptive applications and system components and an extension to this framework that can simultaneously manage the interaction of adaptive applications and systems. We then show some results demonstrating that coordinated management can achieve greater accuracy for a given energy goal or reduced power consumption for a given accuracy target.

“Diderot: a Domain-Specific Language for Portably Parallel Scientific Visualization and Image Analysis”

Gordon Kindlmann, University of Chicago

Many algorithms for scientific visualization and image analysis are rooted in the world of continuous scalar, vector, and tensor fields, but are programmed in low-level languages and libraries that obscure their mathematical foundations. Diderot is a parallel domain-specific language that is designed to bridge this semantic gap by offering programmers a high-level, mathematical programming notation that allows direct expression of mathematical concepts in code. Relative to previous work [1], recent advances in Diderot support portable parallelism (compiling for both pthreads in CPUs and OpenCL for GPUs), inter-strand communication (enabling complex particle system dynamics), and better support for higher derivatives. These features make Diderot an effective tool for developing new algorithms by exploratory programming. The high-level notation allows a concise and natural expression of the algorithms, while the parallelism allows efficient testing on realistic data sets. Recent successes include accurate visualization of the shape of Canny edges in three-dimensions, texture-based visualization of tensor fields, and particle-based sampling of ridge and valley features.

“The (Potential) Perks of Integrating Provenance Support into Database Engines”

Boris Glavic, Illinois Institute of Technology

Data Provenance, information on the origin and creation process of data, has been studied extensively by the database, workflow, and distributed computing communities. The database community has largely focused on the theoretical aspects and all reasonably complete relational provenance systems take the approach of representing provenance relationally and rely on query rewrites to provide provenance support. Typically, these systems are implemented as middleware solutions on top of a standard database. The main drawback of this approach is that its performance and scalability are very limited. In this talk I will make the case for integrating provenance support deep in the query execution engine of a database system. Based on my own experience in implementing a relational provenance system, I will first highlight the major reasons for the performance limitations of the query rewrite approach: (1) The SQL language and its implementations have not been designed with provenance support in mind. Thus, some parts of the provenance generation have to be expressed in overly complex ways, because we are missing the "right" operators to do it efficiently. (2) The data-flow in provenance processing (mostly append-only) is fundamentally different from regular query processing. While in regular query processing, operators like aggregation do vastly reduce the size of the data, this does not hold for provenance in general. (3) The provenance of a query tends exhibit structure that can largely be determined upfront. Furthermore, provenance information often contains substructures that overlap to a large extend. These properties could be exploited for compression, but the benefits that can be gained in a middle-ware solution from this type of compression are limited. I will outline how integrating provenance support into the execution engine can be used to address these limitations and present preliminary results based on implementing a provenance-aware aggregation operator.

Posters:

“A Decoupled Execution Paradigm for Data-Intensive High-End Computing”

Yanlong Yin, Illinois Institute of Technology

High-end computing (HEC) applications in critical areas of science and technology tend to be more and more data intensive. I/O has become a vital performance bottleneck of modern HEC practice. Conventional HEC execution paradigms, however, are computing-centric for computation intensive applications. They are designed to utilize memory and CPU performance and have inherent limitations in addressing the critical I/O bottleneck issues of HEC. In this study, we propose an innovative decoupled execution paradigm (DEP) and the notion of the separation of computing nodes and data (processing) nodes. The novelty of the new DEP execution paradigm is that the data nodes, collectively, take care of the data-intensive operations of the application. The computing nodes, collectively, take care of the computation-intensive operations. The application is executed in a decoupled but fundamentally more efficient manner for data-intensive HEC with the collective support from data processing nodes and computing nodes. The DEP execution paradigm is an evolutionary, if not revolutionary, execution model where I/O intensive operation is as important as computation. DEP's data-centric architecture could have an impact in future HEC systems, programming models, and algorithms design and development. It can significantly reduce costly data movement and is better than the existing execution paradigms for data-intensive applications. The initial experimental tests have confirmed its promising potential. Currently, we are in progress of building the upper level user language (LLVM based), run-time system (an MPI extension), and a tool that helps users to identify the data-intensive phases of their applications.

"Accelerating Scientific Workflow Applications with GPUs"

Dustin Shahidehpour, Illinois Institute of Technology

GPUs are one of most effective ways to provide acceleration on HPC resources. However, a gap still exists between scientific application and accelerators. In an effort to bridge this gap, the DataSys Laboratory at the Illinois Institute of Technology has developed GeMTC (GPU enabled Many-Task Computing), a CUDA-based framework which provides efficient support for Many-Task Computing workloads on accelerators. The GeMTC framework has been integrated into Swift/T, a parallel programming framework from Argonne National Laboratory and UChicago, providing GPU functionality for the Swift language. This project works to analyze several scientific applications already written in Swift script, and then identify the computationally intensive portions of the applications. By porting these portions of code into GeMTC functions, we can provide transparent GPU acceleration for the user. This work highlights preliminary efficiency and performance results conducted from several scientific applications including SciColSim, a collaboration graph analysis tool, and a Molecular Dynamics application that calculates kinetic and potential energies. These applications were chosen because they include many small, sequentially executed tasks that can easily be ported to run in parallel on GPUs. These results highlight applications that provide a use case for both parallel and embarrassingly parallel applications accelerated by the GeMTC framework. Future work is aimed at providing high level abstractions for easy adoption of the Swift/T + GeMTC stack and build upon the growing number of pre-compiled micro-kernels that GeMTC provides to Swift/T end users.

"Automatic Memory Deduplication Support in Hypervisor"

Fangzhou Yao, University of Illinois at Urbana-Champaign

Virtualization techniques are widely used nowadays in datacenters and cloud computing environments. Such environments are installed with large number of similar virtual machines sharing the same physical infrastructure [1]. Modern operating systems like Linux and Windows are termed as memory grabbers [6]. Thus, exploiting opportunities for optimization to reduce the cost becomes inevitable. In this paper, we focus on the memory usage optimization across virtual machines by de-duplicating the memory on per-page basis. We maintain a single copy of the duplicated pages using a copy-on-write semantics. Unlike some existing strategies, which are intended for only applications and need user’s advice [2][4][5], this system provides a periodic automatic memory de-duplication support within the hypervisor to achieve benefits from dynamic changes to memory footprint. We also implemented a prototype of this system within Xen hypervisor [3] to support both para-virtualized and full-virtualized instances.

“Blockus: Efficient Out-of-Core Computation Using Blocks”

Erik Bodzsar, University of Chicago

It is well-known that the MapReduce programming model is not expressive enough for many applications [2]. However, more expressive big data computing systems are in-memory and therefore have limited scalability. We propose a scale up model for big data processing. We use SSDs to eliminate the memory limitation and work on data sets bigger than memory with high efficiency. However, SSDs are still significantly slower than DRAM; using them as operating system swap space leads to low system performance [1]. We present Blockus, a single-machine prototype system that provides transparent out-of-core computation with a flexible programming model. Our research explores the use of higher levels of application knowledge to increase I/O efficiency by prefetching, caching and computation reordering. Users express computation as a sequence of parallel operations on matrix and vector blocks. We exploit the information about parallelism to decrease I/O latency by prefetching, and to do I/O-minimizing task scheduling. We propose two task schedulers. One exploits knowledge about memory contents to greedily execute tasks that require the least amount of I/O. The other exploits iterative algorithm structure by reversing execution order in every iteration to maximize data reuse between iterations. Initial results show 20-100% speedups over naïve mmap for simple iterative streaming algorithms on data sets that are 2x-5x bigger than memory size.

“Characterizing Broadband Services with Dasu”

Zachary Bischof, Northwestern University

We present a crowdsourced approach to broadband service benchmarking from end-systems. We describe its prototype implementation in the context of Dasu [1], a measurement experimentation platform for the Internet's edge. Our approach is based on the observation that, by leveraging minimal monitoring information from the local host and home routers, one can achieve the scale of "one-time" end host approaches with the accuracy of hardware-based solutions. We demonstrate the broadband characterization functionality of our implementation in Dasu and showcase its user interface. We present results showing that a large scale and continuous view enabled by an end-host approach allow us to capture, among other features, the wide range of service performance experienced by users across geographic locations and over time. Dasu support for broadband benchmarking currently includes traditional low-level metrics (such as latencies to the first public IP hop, last private IP hop, primary and secondary DNS servers, and egress points; download and upload throughput measured by NDT; packet loss; and DNS lookup performance) as well as application-level metrics including BitTorrent throughput, average throughput rates measured by YouTube, web browsing performance (page-loading time) for popular websites (ranked by Alexa.com). Measurements are extensible in terms of both their type and target. Dasu also collects passive performance metrics when available. Dasu users are provided with summaries of and historical data about the collected measurements, including a comparison of the average performance seen by other Dasu users in the same region on alternative ISPs. Since many users have restrictions on their monthly bandwidth usage, the interface includes a history of bandwidth used by BitTorrent as well as other traffic from the localhost. When UPnP is enabled at the gateway, Dasu uses byte counters to accurately keep track of the user's total bandwidth consumption.

“Community sensing under (soft) control”

John Rula, Northwestern University

Community sensing leverages the pervasiveness and power of new mobile devices, such as smartphones and tablets, to enable ordinary citizens to collect, transport and verify data. For these systems, their effectiveness is a direct function of the coverage provided by their user base, which in turn depends on the scale and mobility patterns of the participants. The limited scale of most platforms and the strong spatial and temporal regularity of human mobility, however, means that the uncoordinated movement of participants will leave interesting areas uncovered or data that never reaches its destination. In this paper we present Crowd Soft Control (CSC), an approach to exert limited control over the actions of participant users by leveraging the built-in incentives of location-based gaming and social applications. By pairing community sensing with location-based applications, CSC allows researchers to reuse a hosting application’s incentives (e.g. game objectives) to soft control the actions of partici- pating users, increasing the coverage of community sensing campaigns. We have implemented a CSC framework and used it in three case studies. We use our prototype implemen- tation and case studies through experiments that demonstrate the low cost of integration and minimal overhead of CSC, and illustrate its potential benefits.

“Compiler optimization for massively parallel data flow”

Timothy Armstrong, University of Chicago

Distributed, dynamic data flow is an execution model well-suited for many large-scale parallel applications, particularly scientific simulations and analysis pipelines running on large, distributed-memory clusters. Swift is a high-level declarative language that allows flexible data flow composition of functions written in other programming languages such a C or Fortran. This presentation will describe the challenges involved in developing Swift/T: a high performance, scalable implementation of Swift for distributed memory clusters. A particular focus will be the application of compiler optimization techniques. We show that compiler optimization can reduce communication overhead by 70-93% on distributed memory systems at scales up to thousands of cores. With compiler optimization, the high-level Swift language becomes with hand-coded coordination logic for certain common styles of computationally intensive applications.

“Distributed Key-Value Store on Cloud”

Tonglin Li, Illinois Institute of Technology

ZHT is a zero-hop distributed hash table, which has been tuned for the requirements of high-end computing systems. ZHT aims to be a building block for future distributed systems, such as parallel and distributed file systems, distributed job management systems, and parallel programming systems. The goals of ZHT are delivering high availability, good fault tolerance, high throughput, and low latencies, at extreme scales of millions of nodes. ZHT has some important properties, such as being light-weight, dynamic allowing node joins and leaves, fault tolerant through replication, persistent, scalable, and supporting unconventional operations such as append (providing lock-free concurrent key/value modifications) in addition to insert/lookup/remove. We have evaluated ZHT's performance under a variety of systems, ranging from a Linux cluster with 512-cores, to an IBM BlueGene/P supercomputer with 160K-cores. Using micro-benchmarks, we scaled ZHT up to 32K-cores with latencies of only 1.1ms and 18M operations/sec throughput. This work provides three real systems that have integrated with ZHT, and evaluate them at modest scales. 1) ZHT was used in the FusionFS distributed file system to deliver distributed meta-data management at over 60K operations (e.g. file create) per second at 2K-core scales. 2) ZHT was used in the IDAStore, an information dispersal algorithm enabled distributed object storage system, to manage chunk locations delivering more than 500 chunks/sec at 32-nodes scales. 3) ZHT was also used as a building block to MATRIX, a distributed job scheduling system, delivering 5000 jobs/sec throughputs at 2K-core scales. We compared ZHT against other distributed hash tables and key/value stores and found it offers superior performance for the features and portability it supports.

“Elastic Fidelity: Trading-off Computational Accuracy for Energy Reduction”

Georgios Tziantzioulis, Northwestern University

As technology scales to smaller sizes the variability of transistor increases, which is followed by an increase in the probability of timing errors. In addition, the demand for better power efficiency in future computer systems pushes operational voltage near threshold, effectively exacerbating the error rate. An important factor on how much we can decrease operation voltage are the guard bands that need to be maintained to assure correctness, however, these are under the assumption that all computations needed to be error free. This assumption is not required for all applications. Music files, images, videos, even ranking algorithms (like the ones used by google) and in general anything that has to do with human perception, are inherently error tolerant. Our project, Elastic Fidelity, combines the above trend of computer systems’ reliability and the remark about inherently fault-tolerant applications and tries to capitalize on that by exploring the concept of exchanging computational accuracy for power efficiency. In order to explore the feasibility and limits of this concept we need to overcome several challenges. In our work we try to explore the effect of timing errors on the quality of applications' output. Other aspects that we try to research are the possible limitations, and thus changes that needed to be made, in the programming abstraction, system software and computer architecture. In this direction we have enhanced a computer architecture simulator with fault injection capabilities, elaborated on how different ISAs can implement such a scheme and also model the effect of sub nominal voltage operation of CPU functional units (ALU, FPU).

“Enabling and Optimizing Parallel File Write in HDFS”

Xi Yang, Illinois Institute of Technology

Today, two camps of parallel processing systems are HPC systems (like supercomputers) and Datacenters (MapReduce based clusters). They own the superiority and efficiency on solving different computing problems. For better use of resources in some scenarios, there is demand to allow MPI based HPC applications to access the data-intensive file systems equipped in Datacenters, such as HDFS. Due to the semantic gaps, this HPC-datacenter merging is facing many difficulties. One severe contradiction is that the N-1 parallel file writes are the common usage cases for the MPI based HPC applications but distributed file systems HDFS does not support them. N-1 file write refers to the case where N processes share one file and write to different parts of that file in parallel. N-N file write refers to the case where each of the N processes has one independent file to write. This contradiction motivates our work - to enable and optimize MPI based N-1 parallel file writes in HDFS. Typical I/O systems support two types of I/O: synchronous and asynchronous I/O operations. To satisfy HPC application’s usage, we also design and implement these two types. The design idea consists of two major considerations: avoiding the block-level parallel write and enabling the file-level parallelism. The implementation consists two parts - the implementation in MPI-IO and the one in HDFS. To avoid block-level parallel write, the MPI-IO-side implementation needs select aggregators properly so that each target block is in charge of one client process, similar to the Collective I/O scheme. To enable the file-level parallelism, the HDFS client needs the corresponding lock distribution scheme and metadata committing management. Our evaluation results show that we successfully achieved the following contributions. 1. Scalable aggregated bandwidth for parallel file write. 2. Reduced I/O response time. 3. Effective buffer for burst write workload.

“Enabling Dynamic Memory Management Support for MTC on NVIDIA GPUs”

Ben Grimmer, Illinois Institute of Technology

Many-Task Computing emphasizes utilizing many computational tasks over a short period of time. Tasks can be either dependent or independent and are arranged as directed acyclical graphs (DAGs) [1]. Until recently there had been no support for MTC workloads on accelerators, but the development of GeMTC [2] enables Many-Task Computing to run efficiently on NVIDIA GPUs. One major complication with enabling MTC on NVIDIA GPUs is due to the memory management system. Due to overheads, the standard had been for applications to perform all allocations once at the beginning, reducing the importance of efficient memory management. To support MTC applications on NVIDIA GPUs, we need efficient memory management throughout the lifetime of the application. This paper presents a dynamic memory management system, which allows for efficient dynamic memory operations. We compare our results to the default CUDA approach; preliminary results highlight the ability to perform memory operations 8x faster than the default CUDA memory management system.

“Experiments at the Internet’s Edge with Dasu”

Mario A. Sanchez, Northwestern University

Dasu [1] is an extensible measurement experimentation platform for the Internet’s edge. Dasu is composed of a distributed collection of clients, hosted by participating end hosts, and a core set of services for managing and coordinating experimentation. Dasu supports and builds on broadband characterization as an incentive for adoption to capture the network and service diversity of the commercial Internet. This poster presents Dasu in action, focusing on its experiment delegation mechanism and showing how it enables third-party experimentation and maintains security and accountability. Dasu’s management services include a Configuration Service, an Experiment Administration Service, a Coordination Service and a Data Service. Upon initialization, Dasu clients use the Configuration Service to announce themselves and obtain various configuration settings. Clients periodically request experiment tasks from the Experiment Administration (EA) Service, who assigns experiments based on client’s characteristics (such as its IP prefix, geographic location, or operating system) and experiments’ requirements. Clients also contact the Coordination Service to submit updates about completed tasks and retrieve resource usage and experiment constrains. Finally, clients use the Data Service to report results of completed experiments as they become available. Dasu enables sophisticated experiments by supporting the assignment of tasks to clients according to arbitrarily complex experiment logic based on clients’ characteristics. Dasu adopts a two-tiered architecture for the EA Service, with a Primary EA server responsible for resource allocation, and a number of Secondary EA servers in charge of particular experiments. This two-tier architecture enables third-party experimentation, with each participating group hosting their own Secondary EA server. The Primary EA server acts as a broker, allocating Dasu clients to experiments by assigning them to the responsible Secondary EA server, based on clients’ characteristics and resource availability.

“FusionFS: a distributed file system for large scale data-intensive computing”

Dongfang Zhao, Illinois Institute of Technology

State-of-the-art yet decades-old architecture of HPC systems has segregated compute and storage resources, bringing unprecedented inefficiencies and bottlenecks at petascale levels and beyond. This paper presents FusionFS, a new distributed filesystem designed for extreme scalability while achieving significantly higher throughput in both data I/O and metadata operations. It is a user-level filesystem that runs on the compute resource infrastructure, and enables every compute node to actively participate in the metadata and data management. FusionFS has been implemented and deployed on an IBM BlueGene/P supercomputer at 1K-node and scales with throughputs over 158GB/s and 130K metadata ops/sec, delivering up to 98X better performance than the GPFS parallel file system deployed. We evaluate a real bioinformatics application (the Basic Local Alignment Search Tool -- BLAST) at scales up to 1K-nodes, and show it offering up to 32X better performance on FusionFS compared to GPFS.

“Galaxy: Pushing the Power and Bandwidth Walls with Optically Connected Disintegrated Processors”

Yigit Demir, Northwestern University

The scalability trends of modern semiconductor technology lead to increasingly dense multicore chips. Unfortunately, physical limitations in area, power, off-chip bandwidth, and yield constrain single-chip designs to a relatively small number of cores, beyond which scaling becomes impractical. Multi-chip designs overcome these constraints, and can reach scales impossible to realize with conventional single-chip architectures. However, to deliver commensurate performance, multi-chip architectures require a cross-chip interconnect with bandwidth, latency, and energy consumption well beyond the reach of electrical signaling. We propose Galaxy, an architecture that enables the construction of a many-core “virtual chip” by connecting multiple smaller chiplets through optical fibers. The low optical loss of fibers allows the flexible placement of chiplets, and offers simpler packaging, power, and heat requirements. At the same time, the low latency and high bandwidth density of optical signaling maintain the tight coupling of cores, allowing the virtual chip to match the performance of a single chip that is not subject to area, power, and bandwidth limitations. Our results indicate that Galaxy attains speedup of 2.2x over the best realistic single-chip alternatives with electrical or photonic interconnects (3.4x maximum), and 2.8x smaller energy-delay product (6.8x maximum). We show that Galaxy scales beyond 4K cores and attains 2.5x speedup at 6x lower laser power compared to a similar-scale Macrochip with silicon waveguides.

“GeMTC: GPU enabled Many-Task Computing”

Scott J. Krieder, Illinois Institute of Technology

Current software and hardware limitations prevent Many-Task Computing (MTC) workloads from leveraging hardware accelerators (NVIDIA GPUs, Intel Xeon Phi) boasting Many-Core Computing architectures. Some broad application classes that fit the MTC paradigm are workflows, MapReduce, high-throughput computing, and a subset of high-performance computing. MTC emphasizes using many computing resources over short periods of time to accomplish many computational tasks (i.e. including both dependent and independent tasks), where the primary metrics are measured in seconds. MTC has already proven successful in Grid Computing and Supercomputing on MIMD architectures, but the SIMD architectures of today's accelerators pose many challenges in the efficient support of MTC workloads on accelerators. This work aims to address the programmability gap between MTC and accelerators, through an innovative CUDA middleware, namely GeMTC, that enables MIMD programmability of SIMD architectures. GeMTC consists of a daemon SuperKernel running on the device that executes MTC tasks mapped to precompiled device functions. The C API enables GeMTC to be utilized by parallel programming frameworks such as Swift. The SuperKernel receives tasks from Swift, executes some associated work with the given parameters, and returns a result back to Swift. Users include gemtc calls within a given swift script and all of the “heavy lifting” including data transfers, kernel launches, etc. are handled implicitly and transparently from the user. There are a number of motivating MTC applications from an array of domains that fit this model. In this work we present results from a Molecular Dynamics code which is calculating kinetic and potential energies. In addition we provide preliminary results for MTC workloads on the Intel Xeon Phi coprocessor. This work will enable a broader class of applications to leverage the growing number of accelerated high-end computing systems.

“HPC Analytics for Extreme Scale Computing”

Li Yu, Illinois Institute of Technology

Log data is an incredible asset for troubleshooting in large-scale systems. Nevertheless, due to the ever-growing system scale, the volume of such data becomes overwhelming, bringing enormous burdens on both data storage and data analysis. To address this problem, we present a 2-dimensional online filtering mechanism to remove redundant and noisy data via feature selection and instance selection. The objective of this work is two-fold: (i) to significantly reduce data volume without losing important information, and (ii) to effectively promote data analysis. We evaluate this new filtering mechanism by means of real environmental data from the production supercomputers at Oak Ridge National Laboratory and Sandia National Laboratory. Our preliminary results demonstrate that our method can reduce more than 85% disk space, thereby significantly reducing analysis time. Moreover, it also facilitates better failure prediction and diagnosis by more than 20%, as compared to the conventional predictive approach relying on RAS (Reliability, Availability, and Serviceability) events alone.

“Linking and Loading Scientific Workflows”

Casey Robinson, University of Notre Dame

My research is focused on finding the implicit dependencies in a scientific workflow. Most workflow systems only transfer the files explicitly specified as required, relying on the fact that the correct executable or configuration file is available at the execution site. Knowledge of all the dependencies in a scientific workflow provides many benefits; more accurate estimation of disk usage and successful execution of all programs in the workflow. Reducing the number of files required to be specified also improves the usability of the system. Previous attempts[1] at solving this problem monitored a test run of the workflow, which results in a historical record of the application but not a general executable workflow. My research approaches the linking problem from a static viewpoint to create a comprehensive view of the dependencies.

“MATRIX: MAny-Task computing execution fabRIc at eXascales”

Anupam Rajendran, Illinois Institute of Technology

Scheduling large amount of jobs/tasks over large-scale distributed systems plays a significant role in achieving high system utilization and throughput. Today’s state-of-the-art job management/scheduling systems have predominantly Master/Slaves architecture, which has inherent scalability issues at petascale and beyond, and is vulnerable to single point failure. In designing the next-generation distributed job management system, we must address new challenges such as load balancing. In this work, we propose an adaptive work stealing technique to achieve distributed load balancing at extreme scales, those found in today’s petascale systems toward tomorrow’s exascale systems. The parameter space of work stealing is investigated through SimMatrix, a light-weight SIMulator of MAny-Task execution fabRIc at eXascale, with up to millions of nodes, billions of cores, and tens of billions tasks. Based on the optimal parameter space we explored through SimMatrix, we implemented work stealing in a real system MATRIX, MAny-Task computing execution fabRIc at eXascale. MATRIX utilizes ZHT, a Zero hop distributed Hash Table, for distributing tasks and managing metadata of all tasks, and supports both high performance computing (HPC) and many-task computing (MTC) workloads. Via MATRIX, we study performance of the work stealing technique on a Blue Gene/P machine with up to 1K nodes (4K cores), and in the process validating the simulation results. Our results show that work stealing is a scalable and efficient load balancing approach (with more than 90% efficiency possible) for extreme-scale distributed systems.

“Measuring Power Consumption on IBM Blue Gene/Q”

Sean Wallace, Illinois Institute of Technology

In addition to pushing what is possible computationally, state-of-the-art supercomputers are also pushing what is acceptable in terms of power consumption. Despite hardware manufacturers researching and developing efficient system components (e.g., processor, memory, etc.), the power consumption of a complete system remains an understudied research area. Because of the complexity and unpredictable workloads of these systems, estimating the power consumption of a full system is a nontrivial task. We provide system-level power usage and temperature analysis of early access to Argonne's latest generation of IBM Blue Gene supercomputers, the Mira Blue Gene/Q system. The analysis is provided from the point of view of jobs running on the system. We describe the important implications these system level measurements have as well as the challenges they present. Using profiling code on benchmarks, we will also look at the new tools this latest generation of supercomputer provides and gauge their usefulness and how well they match up against the environmental data.

“MinimaLT: Minimal Latency Networking Through Better Security”

Xu Zhang, University of Illinois at Chicago

MinimaLT is a new network protocol that provides ubiquitous encryption for maximal confidentiality, including protecting packet headers. MinimaLT provides server and user authentication, extensive Denial-of-Service protections, and IP mobility while approaching perfect forward secrecy. We describe the protocol, demonstrate its performance relative to TLS and unencrypted TCP/IP, and analyze its protections, including its resilience against DoS attacks. By exploiting the properties of its cryptographic protections, MinimaLT is able to eliminate three-way handshakes and thus create connections faster than unencrypted TCP/IP.

“NoVoHT: a Lightweight Dynamic Persistent NoSQL Key/Value Store”

Kevin Brandstatter, Illinois Institute of Technology

With the increased scale of systems in use and the need to quickly store and retrieve information, key/value stores are becoming an important element in the design of large-scale storage systems. Key/value stores are well known for their simplistic interfaces, persistent nature, and excellent operational efficiency – they are also known as NoSQL databases. This paper presents the design and implementation of a non-volatile hash table (NoVoHT). NoVoHT was designed from the ground up to be lightweight, fast, and dependency-free. Our goal was to create a fast persistent key/value store that could be easily integrated and operated in lightweight Linux OS typically found on today’s

supercomputers. We also aimed to develop a system that performed as close as possible to an in-memory hash map, but with the added benefit of being persistent. We also extended the traditional key/value store interface (e.g. insert, lookup, remove) to include a novel operation (e.g. append) that has allowed NoVoHT to efficiently support lock-free concurrent write operations. NoVoHT is also dynamic, supporting live migration across node boundaries. We have run comparisons at significant scales against some of the more commonly used key value stores and have shown that NoVoHT can perform similarly or better than other systems such as Kyoto Cabinet, and BerkeleyDB. We observed up to 165K+ operations per second, up to 32X better performance than competing systems. We have evaluated NoVoHT with both traditional mechanical disks (HDD) as well as with solid state disks (SSD), and have deployed NoVoHT as the persistent back-end of a distributed hash table (ZHT) on an IBM BlueGene/P supercomputer at up to 32K-cores.

“Power-Aware Job Scheduling on Production HPC Systems”

Zhou Zhou, Illinois Institute of Technology

As the exascale computing come into being, the energy cost for the operation of HPC system is more daunting than ever before . Many research have been done about regulating the energy consumption, however, most of them can not avoid affecting the HPC system s utilization or the fairness of job scheduling. In this paper, we propose an job-power-aware scheduling mechanism for HPC system with the objective to reduce energy cost and not degrading system utilization. We have made the observation that the electricity price can be swaying in a wide range during a day and the job's power profile in HPC system can also be very diverse. The novelty of our scheduling mechanism is we take the variation of energy price and job's power profile into consideration and try to make a better decision regards timing for scheduling each job. We verified the effectiveness of our algorithm for Cluster-based system. And experiments, conducted with real job traces form production system, show that our energy cost aware algorithm can reduce the energy cost in HPC system as much as 10% while on bring neglectable impact to the system's utilization.

“Scalable Authentication Infrastructure”

Wenyuan Fei, University of Illinois at Chicago

At Internet-scale, it is long-standing open problem to provide all the properties needed by a user authentication infrastructure. Here a strong trust model is essential, so that those which rely on authentication can control their exposure to weak or adversarial third parties. Yet authentication systems with strong trust models have been inefficient, suffering from high latency, excessive bandwidth, and high CPU load. These inefficiencies significantly impede wide-scale deployment. We introduce Scalable Authentication Infrastructure (SAyI), a public-key based authentication infrastructure which scales to the Internet. It provides strong trust, protects privacy, and provides security. SAyI is very efficient: it is designed to minimize bandwidth and latency through a careful integration of authorization and authentication. In SAyI, irrelevant certificates do not negatively impact performance. An Internet user authentication is guaranteed to complete in a single Internet round trip, significantly faster than alternative authentication infrastructures.

“Scaling Work Queue for the Cloud with Hierarchy”

Michael Albrecht, University of Notre Dame

There is a large class of applications known as Master-Worker programs, where a computational workflow is divided into discrete sub-tasks by a central manager and are then distributed to remote worker nodes. These applications scale easily from tens to hundreds of compute cores and the simple structure of the framework allows for easy deployment across a multitude of resources but the centralized design poses a challenge for scaling up to the increased data sizes and thousands to tens of thousands of simultaneous workers necessary for modern scientific workflows. Further exacerbating these challenges is the ongoing shift from using shared, semi-local Grid resources to dedicated remote cloud providers. The reduced bandwidth between local data sources and compute resources in the cloud can constrain the capacity of the master, reducing the available parallelism. Furthermore both the links and the resources are often metered, introducing a new monetary consideration when allocating and controlling resources. To address these challenges we introduce an intermediate level of hierarchy within the master-worker paradigm, and demonstrate the benefits gained in terms of data transfer overhead, master capacity, and task throughput.

“Shifting GEARS to Enable Guest-context Virtual Services”

Kyle Hale, Northwestern University

We argue that the implementation of VMM-based virtual services for a guest should extend into the guest itself, even without its cooperation. Placing service components directly into the guest OS or application can reduce implementation complexity and increase performance. In this paper we show that the set of tools in a VMM required to enable a broad range of such guest-context services is fairly small. Further, we outline and evaluate these tools and describe their design and implementation in the context of Guest Examination and Revision Services (GEARS), a new framework within the Palacios VMM. We then describe two example GEARS-based services—an MPI communication accelerator and an overlay networking accelerator—that illustrate the benefits of allowing virtual service implementations to span across the VMM, guest, and application. Other VMMs could employ the ideas and tools in GEARS.

“Simple and Secure Networking”

Yaohua Li, University of Illinois at Chicago

Secure networking is difficult to get right: TLS/SSL, often deployed with Public Key Infrastructure (PKI), has proven vulnerable to various attacks[1]. Kerberos design is outdated since it is based on symmetric key cryptography. Browsers trust many CAs, leaving the users vulnerable to all of them. Furthermore, TLS implementation has numerous and complex APIs, resulting in many vulnerabilities even in security critical and widely deployed software written by experts[3].
We designed netStackGo to better support secure networking, a strong trust model[4], and low complexity. A user can authenticate cryptographically to different organizations, where each organization chooses its own authentication servers. For users, netStackGo provides encrypted, authenticated, and authorized networking. For programmers, a simple interface without complex and arcane rules, such as for certificate validation, simplifies their job and eliminates many vulnerabilities. For system administrators, their systems are better understood and more easily protected.

“Software Engineering Challenges and Tools in Distributed Computing Workspace for Civil Engineering Applications”

Peter Sempolinski, University of Notre Dame

Recently we designed a workspace for certain civil engineering simulation applications to be run in distributed computing environments. On this poster we very quickly overview the overall structure of this system and then highlight specific software engineering challenges. We discuss a toolkit which we developed for templating complex simulations. This file management tool is intended for distilling a user specified parameters into the complex inputs of our simulations, as well as manage the files of these simulations for input, output and deletion. Also, as these simulations require an odd assortment of dependency files, we have to insure that all these pieces are sent to whatever distributed computing back-end we select. We also briefly discuss the broader applications which motivated this work.

“The 10x10 Project: Can we have both extreme energy efficiency and programmability?”

Raihan ur Rasool, University of Chicago

Heterogeneous architectures promise as much as 100-fold energy benefits, but significant software, programmability, and algorithm challenges must be solved in co-design with these heterogeneous architectures. The 10x10 project is an aggressive approach to heterogeneity, exploiting "dark silicon". The 10x10 paradigm is a principled, systematic approach to heterogeneity in computer architecture. A 10x10 architecture exploits deep workload analysis to drive co-design of a federated heterogeneous architecture that exploits customization for energy efficiency, but federates a set of customized engines to achieve general-purpose coverage. The 10x10 team includes Prof. Wenmei Hwu (UIUC) and Dr. Laura Carrington (SDSC/PMaC Labs),

and we gratefully acknowledge the support of the National Science Foundation and the DARPA PERFECT program.

“Toward Petascale Cosmology Simulations”

Jingjin Wu, Illinois Institute of Technology

Numerical simulations are vital for cosmologists to model the universe and researchers are always making efforts to enhance cosmological codes, such as fully utilizing the ever-growing supercomputers for high-fidelity simulations and including new physical processes especially those that are ignored before. We target cell-based AMR(Adaptive mesh refinement) cosmology application-the Adaptive Refinement Tree(ART) code, and improve the performance of ART through I/O optimization and hierarchical task mapping techniques.

“Towards a Provenance-aware Distributed Filesystem”

Chen Shou, Illinois Institute of Technology

It has become increasingly important to capture and understand the origins and derivation of data (its provenance). A key issue in evaluating the feasibility of data provenance is its performance, overheads, and scalability. Distributed file systems have so far proposed a central system for provenance collection [1]. This is a performance bottleneck, especially for file systems meant for extreme-scales. In previous work [2], we already integrated two recent research projects, SPADE [3] and FusionFS [4] to present a prototype of a provenance-aware distributed filesystem. The system is able to collect provenance distributedly, which offers excellent scalability while retaining the provenance overhead negligible under certain conditions. However, due to the implementation of SPADE (in Java), the system is limited to small scale (up to 32-node). In order to provide a better scalability for data provenance collection in extreme-scale computing, we are now leveraging ZHT (Zero-hop Distributed Hash Table) [5] for provenance storage, and based on that, developing a more efficient provenance system (in C++) along with FusionFS. Compared with SPADE, the new system is more light-weight, and has a better query scalability and storage load balancing. The new system is expected to scale up to 1K-node.

“Understanding the cost of the cloud for scientific applications”

Iman Sadooghi, Illinois Institute of Technology

Commercial clouds bring a great opportunity to the scientific computing area. Scientific applications usually need huge resources to run on, however not all of the scientists have access to significant high-end computing systems, such as those found in the Top 500 list. Cloud has gained the attention of scientists as a competitive resource to run HPC applications at a lower cost. But as a different infrastructure, it is unclear whether clouds are capable of running scientific applications with a reasonable performance. Before we can start using existing public cloud platforms for scientific or in general, high I/O demanding applications, we have to study the raw performance of public clouds in terms of compute, memory, network and I/O. The goal of this work is to assess the ability and the cost of the Amazon EC2 cloud running scientific applications using customized instances against the local systems with no virtualization. compare the cost of the public cloud with a private cloud. Then we measure the performance and analyze the cost of memory, CPU, network, and I/O for each instance type of Amazon EC2

“Understanding the Costs of Many-Task Computing Workloads on Intel Xeon Phi Coprocessors”

Jeffrey Johnson, Illinois Institute of Technology

Many-Task Computing (MTC) aims to bridge the gap between HPC and HTC. MTC emphasizes running many computational tasks over a short period of time, where tasks can be either dependent or independent of one another. MTC has been well supported on Clouds, Grids, and Supercomputers on traditional computing architectures, but the abundance of hybrid large-scale systems using accelerators has motivated us to explore the support of MTC on the new Intel Xeon Phi accelerators. The Xeon Phi is a PCI-Express based expansion card comprised of 60 cores supporting 240 hardware threads to produce up to 1 teraflop of double-precision performance in a single accelerator. These cards are already being integrated into super-computing clusters such as Stampede which hosts over 6,400 Xeon Phi Accelerators totaling in over 7 petaflops of double-precision performance. This work provides an in depth understanding of MTC on the Intel Xeon Phi and presents our preliminary results of running several different workloads on pre-production Intel Xeon Phi hardware. By utilizing Intel’s provided SCIF protocol for communicating across the PCI-Express bus we have achieved over 90% efficiency near or outperforming OpenMP offloading tasks over 300 uS with our batch framework. This performance opens the opportunity for the development of a framework for executing heterogeneous tasks on the Xeon Phi alongside other potential accelerators including graphics cards for MTC applications. Our framework will provide fine granularity for executing MTC applications across large scale compute clusters. It will be integrated with our existing graphics card framework, GemTC, to provide transparent access to GPU’s, Xeon Phi’s, and future generations of accelerators to help bridge the gap into exascale computing

“Understanding Timing Errors Patterns”

Yuanbo Fan, Northwestern University

Circuit-level timing speculation has been proposed as a technique to improve overall system efficiency by eliminating overheads arising from worst-case design assumptions. Under this design paradigm, the processor runs at voltage, frequency, and thermal operating points which would not guarantee signal setup time constraints for all logic paths. The system is augmented with timing error-detection and correction techniques, so that a timing error no longer leads to catastrophic system failure. Instead, we can trade off error rate for energy savings. Furthermore, recent work has shown that in real programs timing errors are often correlated to specific static instructions – a concept known as timing error locality. This property allows the hardware to dynamically predict many timing errors and consequently lower recovery costs. While this phenomenon has been studied, the relationship between gate-level circuit structure and program values is not well understood. This work seeks to understand how these two interact to develop robust models for studying and predicting timing error patterns.

“VMM-based Emulation of Intel Hardware Transactional Memory”

Maciej Swiech, Northwestern University

We describe the design, implementation, and evaluation of emulated hardware transactional memory, specifically the Intel Haswell Restricted Transactional Memory (RTM) architectural extensions for x86/64, within a virtual machine monitor (VMM). Our system allows users to investigate RTM on hardware that does not provide it, debug their RTM-based transactional software, and stress test it on diverse emulated hardware configurations. We are able to accomplish this approximately 60 times faster than under emulation. A noteworthy aspectof our system is a novel page-flipping technique that allows us to completely avoid instruction emulation, and to limit instruction decoding to only that necessary to determine instruction length. This makes it possible to implement RTM emulation, and potentially other techniques, far more compactly than would otherwise be possible. We have implemented our system in the context of the Palacios VMM, and it will be publicly available when the is paper is published. Our techniques are not specific to Palacios, and could be implemented in other VMMs.

“When is Multi-version Checkpointing Needed?”

Guoming Lu, University of Chicago

The scaling of semiconductor technology and increasing power concerns combined with system scale make fault management a growing concern in future high performance computing systems. Greater variety of hardware and software errors, higher error rates, longer detection intervals, and so called “silent” errors are all expected. Traditional checkpointing models and systems mostly assume that error detection is nearly immediate and thus preserving a single checkpoint is sufficient. We define a richer model for future systems that captures the reality of latent errors, i.e. errors that go undetected for some time, and use it to derive optimal checkpoint intervals. We also develop a multi-version checkpoint scheme that enables recovery from latent errors. With the richer system model, we explore the potential importance of multi-version checkpoint systems, characterizing opportunities and costs of increasing resilience. Specific results include understanding the limits of single checkpoint systems for a range of error and detection rate scenarios two to as many as a dozen checkpoints may be needed to achieve acceptable error coverage, and to achieve reasonable system efficiency, multiple versions (three to sixteen) can be a significant benefit. Studies of several future exascale machine scenarios show that two checkpoints are always beneficial, but when checkpoints schemes are optimized with Flash memories, as many as seven checkpoints are beneficial.

“Towards Distributed Batch Scheduling”

Xiaobing Zhou, Illinois Institute of Technology

This paper is going to discuss a distributed system service, distributed job launch, at extreme scales. We will use ZHT (A Light-weight Reliable Persistent Dynamic Scalable Zero-hop Distributed Hash Table) as backend storage system to keep all the job metadata information, and leverage Slurm (basically slurmd) to launch jobs on the nodes. The system consists of many servers (up to tens of thousands), every one responsible of a group of compute nodes (AKA. partition). Each server runs a ZHT server for job metadata management, and each compute node runs a job launch Daemon (starting with the slurmd code base). Job data and metadata is distributed among all the servers, and the servers are fully connected with each other being able to address every other one directly. Job data and metadata is replicated by different consistency models, and the whole system is dynamic, with nodes (both the server and compute node) joining and departing. Users submit jobs to dedicated servers using some hash function (e.g. job_id modulo number of server) to ensure good load balancing. Each job many has attributes, such as job id, max running time, size (number of nodes required to be allocated), executable files, priority, etc. Each server has a job queue, which sorts all the submitted jobs according to the job priority. Jobs are launched based on combination of scheduling polices, such as FIFO, backfill, gang –scheduling, etc. We will do micro benchmark to see how this distributed job launch service outperforms Slurm with centralized scheduler.

“The Global View Resilience Model”

Zachary Rubenstein, University of Chicago

As the state of the art proceeds towards exascale computation, programmers will have to write applications designed to run on unreliable systems. Consequently, High-Performance Computing will need to adapt a new programming model which allows for application-driven error detection and recovery. We propose the Global View Resilience model, which treats computation as a series of transitions between globally-visible states. In our framework, this model manifests as an interface combining a globally-visible address space with multi-version, snapshot-based protection of globally addressable data. In GVR, the application has fine-grained control over the ways that errors are identified and handled. For a given Global Data Storage (GDS) object, the application can specify the priority of preserving the data within that object, the routines with which the object will be periodically examined for errors, and the routines which will be used for recovery in the event that an error is found. All GDS objects are globally accessible through one-sided communication paradigms. We have studied several aspects of the GVR model. One significant aspect of GVR is that it stresses the importance of preserving more than one backup version of state. It is important to exploit multi-version state preservation when we consider the prospect of latent errors, which are errors that only manifest after long periods of time. We have found that, given our model, a large number of checkpoints may be beneficial in order to contend with latent errors. Additionally, we have studied the prospect of augmenting real-world numerical methods with GVR in order to improve their reliability. We have experimented with different snapshot-taking and error-checking and recovery methods to observe the potential to preserve performance of iterative linear solvers when faced with soft errors.

“Mastering chaos with cost-effective sampling”

Mona Rahimi, DePaul University

Large datasets make it challenging to apply data analysis techniques such as classification, clustering and recommendation modeling. Therefore, there is an immense interest in selecting a small subset of a dataset in a way that preserves the information contained within the original dataset thus making it easier to perform data analysis. In addition to classification, a representative data sample is also beneficial for the purposes of visualizing large datasets; furthermore, same techniques can help reduce the cost of knowledge discovery techniques. In this project, our goal is to develop algorithms for selecting the most representative subset from a large dataset and use that subset to train the classifier while still maintaining a relatively high accuracy in classifying the data. We are planning to apply our methodology to Lung Cancer Database Consortium (LIDC) dataset, which includes Computed Tomography (CT) lung images, to select the most informative cases which are to be annotated by the radiologists manually. In long term, the proposed approach can reduce the number of cases to be interpreted by multiple radiologists, and therefore control the costs of medical diagnosis. The early results that we have collected so far were developed using several simpler datasets (such as IRIS). These preliminary results demonstrate that it is indeed possible to choose a small subset of data points that will result in a relatively high classification accuracy.

“Increasing Concurrency in Deterministic Runtimes with Conversion”

Timothy Merrifield, UIC

The current "state of the art" in deterministic runtime systems still come with a heavy performance penalty. While efficient hardware solutions (such as RCDC) have been proposed, we feel that an efficient software-only solution for determinism is within reach.

We present several techniques that we believe will significantly reduce the overhead in such systems. At the core of this proposal is Conversion, an implementation of version-controlled memory built in to the Linux kernel.

“Building Capable, Energy-Efficient, Flexible Visualization and Sensing Clusters from Commodity Tablets”

Xiao Yan, Loyola University Chicago

We explore the application of clusters of commodity tablet devices to problems spanning a “trilogy” of concerns: visualization, sensing, and computation. We conjecture that such clusters provide a low-cost, energy-efficient, flexible, and ultimately effective platform to tackle a wide range of problems within this trilogy. This is a work in progress, and we now elaborate our position and give a preliminary status report. A wide range of Android tablet devices are available in terms of price and capabilities. “You get what you pay for” w.r.t. display resolution, sensors, and chipset---corresponding to the trilogy. $200 gets one a 1280x800-pixel touch display, quad-core CPU with GPU, and various input sensors (camera, accelerometer, barometer, etc.) When arranged in a suitable geometry, such as a rectangle, such devices form an innovative whole that is more powerful than the sum of its parts, with CPUs and sensors in addition to a much higher combined display resolution than that of conventional HD monitors. Scholarly merit: The tablet cluster serves as a testbed for exploring a wide range of research questions in both technical and application domains. Numerous applications in education and research are imaginable, such as environmental and security monitoring, calculating and visualizing the energy footprint of an organization, facial recognition involving multiple cameras, exploring molecules in three dimensions, visualizing relationships among versions of a text, teaching the color space to art students, mapping audiovisual content from sequential to concurrent playback, etc. Project status: We have made preliminary progress along multiple fronts. Specifically, we have developed an Android/Java prototype app for clustering multiple devices and drawing to the resulting combined virtual display on top of the Skeenzone middleware for distributed mobile applications (http://code.google.com/p/skeenzone). We have also developed a an Android/Scala prototype app for navigating RESTful web services as data sources.

“Network Technologies used to Aggregate Environmental Data”

Paul Stasiuk, Loyola University Chicago

The goal of NOSWCEM or Loyola Weather Service (lws) project is to design and build a system of interconnected environmental monitoring devices that can intelligently and autonomously control the environment around them based on set thresholds and triggers. Ideally, the devices will also have the ability to aggregate their data and easily display this data in various ways:

through a user interface in the room where the device is located or via a web interface. The prototyping of the lws system gave us an opportunity to benchmark communication protocols that could be used in the transmitting of environmental data. Additionally, we were able to benchmark the use of a document based database's range queries based on compound indices. Our system can not only scale, but also provide interaction between users and the environment on a more intelligent and extensible level than other environmental monitoring solutions. The open-source nature of the project allows for others to actively contribute to developing the system, further molding it to their needs. The currently deployed system includes a framework for extending functionality beyond simply environmental data collection. For example, controlling the environment based on thresholds set by the user; lights, temperature and humidity. Our conclusions are twofold: (1) we have evidence that priority message queuing see's at least double the performance over HTTP based protocols for large amounts of queries from the nodes and (2) database range queries on compound time indices versus ISO times see only marginal performance increases; leaving us inconclusive the us of compound indices.

"A Polyglot Approach to Bioinformatics Data Integration: Phylogenetic Analysis of HIV-1"

Steven Reisman, Loyola University Chicago

RNA-interference has potential therapeutic use against HIV-1 by targeting highly-functional mRNA sequences that contribute to the virulence of the virus. Empirical work has shown that within cell lines, all of the HIV-1 genes are affected by RNAi-induced gene silencing. While promising, inherent in this treatment is the fact that RNAi sequences must be highly specific. HIV, however, mutates rapidly, leading to the evolution of viral escape mutants. In fact, such strains are under strong selection to include mutations within the targeted region, evading the RNAi therapy and thus increasing the virus' fitness in the host. Taking a phylogenetic approach, we have examined 3000+ HIV-1 strains obtained from NCBI'S database for each of the HIV genes, identifying conserved regions at each hypothetical and operational taxonomical unit within the tree. Integrating the wealth of information available from each genome's record, we are able to observe how conserved regions vary with respect to their distribution throughout the world. This was made possible through the development of a new software system, developed such that similar analyses can be conducted for any species or gene of interest, not just HIV-1. In addition to the phylogenetic signal which we can recognize from the HIV-1 genomes examined, we can also identify how selection varies across the genome. Taking this evolutionary approach, we have detected regions ideal for targeting by RNAi treatment. The software system mentioned above provides access to the National Center for Biotechnology Information's (NCBI) GenBank in multiple ways: It converts GenBank data to the FASTA format for for analysis using desktop tools, and it exposes the data in the form of a RESTful web service. We have implemented this system using polyglot approach involving multiple languages (Python and Scala), libraries (Flask and BioJavaX), and persistence mechanisms (text files and MongoDB NoSQL databases).