ICL

2016/2017 REPORT

DESTINATION EXASCALE

THE UNIVERSITY OF TENNESSEE
KNOXVILLE

ICL 2016/2017 REPORT

DESIGNED BY **David Rogers** EDITED BY **Scott Gibson**

ICL

2016/2017 REPORT

FROM THE DIRECTOR

For many of us at ICL, the highlight of 2016 will likely be the cluster of seven awards we won from the US Department of Energy's Exascale Computing Project (ECP) this fall. These awards were the result of the tremendous effort our team put forth to write—or, in some cases, contribute to—seven compelling proposals, six of which were due on the same day in early August. In the broader picture, ICL played a significant role in the community movement of more than six years that culminated in the ECP. The seeds for the ECP were planted by the International Exascale Software Project (IESP), which, as the timeline on page 6 shows, we helped launch in 2009; and the work we did with the IESP and its successor, the Big Data Extreme-scale Computing (BDEC) project, was important in encouraging the development of the ECP by helping to foster the president's National Strategic Computing Initiative (NSCI) in 2014, which led to the creation of the ECP in 2016.

Our success with the ECP was a function of at least two very noteworthy achievements. First, because of our incredibly strong reputation, forged over more than a quarter of a century, we were one of a select group of universities invited to submit proposals. ICL is well known for translating leading-edge research into software that is widely deployed and heavily used at HPC facilities, public and private, large and small, all over the world. Many ICL alumni (see pages 34–35) toiled to build that reputation, and to them we express our deep gratitude. The other achievement is attributed to the team we have now. As our response to the ECP opportunity has so outstandingly demonstrated, the leadership of our research directors the last several years has dramatically raised our game with respect to developing competitive and high-impact funding proposals.

So, for the next three (or, potentially, seven) years, ICL will be part of the elite vanguard of researchers from DOE laboratories and designated universities that will create the software infrastructure needed to exploit the power of the nation's first exascale machines. Each of our research groups will have high-profile software projects of significant impact to the ECP. The numerical libraries group will develop Software for Linear Algebra Targeting at Exascale (SLATE); work collaboratively with SNL on Production-ready, Exascale-Enabled, Krylov Solvers (PEEKS); and work with a consortium of DOE labs and universities on the Extreme-scale Scientific Development Kit for the Exascale Computing Project (xSDK4ECP). The distributed computing group will use its PaRSEC framework to build Distributed Tasking for Exascale (DTE), and collaborate with ORNL, SNL, and LANL on an enhanced version of Open MPI for Exascale (OMPI-X). In the Exa-PAPI project, the performance group will enhance the PAPI environment with capabilities necessary to monitor the performance of exascale applications in a multitude of dimensions, including both hardware and software. And our MAGMA team will collaborate with LLNL's co-design Center for Efficient Exascale Discretizations (CEED). You can learn more about these projects on pages 20–21.

Our successes with the National Science Foundation in 2016 also guarantee that that we will continue to push forward with our traditional research agendas in all three ICL focus areas. The BEAST OpeN Software Attuning Infrastructure (BONSAI) project will enable application communities to optimize their key computational kernels at unprecedented speed using huge concurrent autotuning sweeps on HPC systems with GPU accelerators and manycore coprocessors. (I was reminded in November of how far back ICL's autotuning lineage goes when Clint Whaley and I won the SC16 "Test of Time Award" for the ATLAS paper we wrote in 1998.) In the Simulation and Modeling for Understanding Resilience and Faults at Scale (SMURFS) project, the distributed computing group will work with ICL alum Dorian Arnold at the University of New Mexico to use computer simulation to help research scientists build predictive fault tolerance profiles of their applications as they execute on platforms of unprecedented scale. Finally, the PAPI Unifying Layer for Software-defined Events (PULSE) project will break new ground by enabling developers to expose performance metrics of key software components found in the HPC software stack, and combine them with a wide variety of hardware performance metrics in the standard PAPI interface.

With all that's been conveyed here, the stage is set for plenty of inspiration and enthusiasm at ICL as we continue the research in our respective areas and participate in the nation's major push to achieve exascale computing early in the next decade. In the midst of reflecting on what promises to be a very bright future, I express my appreciation to our federal and industrial sponsors for their continued support of our efforts. And, as always, I extend my special thanks and congratulations to ICL researchers, staff, and students for their creativity, skill, dedication, and tireless efforts to keep ICL on the leading edge of scientific computing research throughout the world.

Jack Dongarra
DIRECTOR, ICL

INTRODUCTION

Situated in the heart of the University of Tennessee campus and at the nexus of academia, government, and industry, the Innovative Computing Laboratory impacts the world as a leader in advanced scientific computing and high performance computing (HPC) through research, education, and collaboration.

The unique challenges of today's computational research are characterized by large datasets and the need for greater efficiency, performance, and power. ICL's cutting-edge efforts, which now span more than 25 years, have evolved and expanded with the agility and focus required to address those challenges. ICL's work encompasses a solid understanding of the algorithms and libraries for multicore, manycore, and heterogeneous computing, as well as performance evaluation and benchmarking for high-end computing. In addition, ICL's portfolio of expertise includes high-performance parallel and distributed computing, with keen attention to message passing and fault tolerance.

The tools and technologies that ICL designs, develops, and implements play a key role in the use of supercomputing for discoveries in areas such as the life sciences and drug design, climate science, earthquake prediction, energy exploration, combustion and turbulence, advanced materials science, and more.

Claxton Building
HOME OF ICL

FROM LEADING-EDGE RESEARCH TO HIGH-IMPACT SOFTWARE

NUMERICAL LINEAR ALGEBRA

Numerical Linear Algebra algorithms and software form the backbone of many scientific applications in use today. With the ever-changing landscape of computer architectures, such as the massive increase in parallelism and the introduction of hybrid platforms utilizing both traditional CPUs as well as accelerators, these libraries must be revolutionized in order to achieve high performance and efficiency on these new hardware platforms. ICL has a long history of developing and standardizing these libraries in order to meet this demand, and we have multiple projects under development in this arena.

PERFORMANCE EVALUATION AND BENCHMARKING

Performance Evaluation and Benchmarking are vital to developing science and engineering applications that run efficiently in an HPC environment. ICL's Performance Evaluation tools allow programmers to see the correlation between the structure of source/object code and the efficiency of the mapping of that code to the underlying architecture. This relationship is important for performance tuning, compiler optimization, debugging, and finding and correcting performance bottlenecks. ICL's benchmark software is widely used to determine the performance profile of modern HPC machines, and has come to play an essential role in the purchasing and management of major computing infrastructure by government and industry around the world.

DISTRIBUTED COMPUTING

Distributed Computing is an integral part of the high performance computing landscape. As the number of cores, nodes, and other components in an HPC system continue to grow explosively, applications need runtime systems that can exploit all this parallelism. Moreover, the drastically lower meantime to failure of these components must be addressed with fault tolerant software and hardware, and the escalating communication traffic that they generate must be addressed with smarter and more efficient message passing standards and practices. Distributed Computing research at ICL has been a priority for nearly two decades, and the lab has several projects in that arena under active development.

HISTORY

ICL's founder, Dr. Jack Dongarra, established the lab in 1989 when he received a dual appointment as a Distinguished Professor at the University of Tennessee, Knoxville and as a Distinguished Scientist at Oak Ridge National Laboratory. Since then, ICL has grown into an internationally recognized research laboratory, specializing in numerical linear algebra, distributed computing, and performance evaluation and benchmarking.

1989

Level 3 Basic Linear Algebra Subprograms (BLAS) specification is developed to perform assorted matrix-multiplication and triangular-system-solving computations.

Parallel Virtual Machine (PVM), a software tool for parallel networking of computers, is designed to allow a network of heterogeneous Unix and/or Windows machines to be used as a single distributed parallel processor.

1992

Basic Linear Algebra Communication Subprograms (BLACS) are created to make linear algebra applications easier to program and more portable.

Linear Algebra Package (LAPACK), a standard software library for numerical linear algebra, is released.

1995

Version 1.0 of the **Scalable LAPACK (ScaLAPACK)** library, which includes a subset of LAPACK routines redesigned for distributed memory MIMD parallel computers, is released.

1997

Automatically Tuned Linear Algebra Software (ATLAS) represents a new paradigm in high-performance library production and maintenance and is developed to allow software to keep pace with the incredible rate of hardware advancement inherent in Moore's Law.

NetSolve, a client-server system that enables users to solve complex scientific problems remotely, is released.

1999

The **Heterogeneous Adaptable Reconfigurable Networked SystemS (HARNESS)** project designs and develops a pluggable lightweight heterogeneous distributed virtual machine (DVM) environment.

2002

The MPI plug-in for HARNESS, called **Fault Tolerant MPI (FT-MPI)**, provides support for fault-tolerant applications that is crucial for large long-running simulations.

2003

A new benchmark, known as the **HPC Challenge Benchmark**, is developed for DARPA consisting of four benchmarks: HPL, Streams, RandomAccess, and PTRANS

The **LAPACK for Clusters** project is developed to deliver the convenience and use of existing sequential environments bundled with the power and versatility of highly tuned parallel codes that execute on clusters.

2006

Research on **Fault-Tolerant Linear Algebra (FT-LA)** is established to develop algorithm-based fault tolerance into major dense linear algebra kernels.

2008

A dense linear algebra package at the forefront of multicore computing, the **Parallel Linear Algebra Software for Multicore Architectures (PLASMA)**, is designed to deliver the highest possible performance from a system of multiple sockets of multicore processors.

2009

The **International Exascale Software Project (IESP) project** brings together representatives from the major segments of the global high-performance community to plan and create a new software infrastructure for the extreme-scale systems of the future.

2010

Distributed Parallel Linear Algebra Software for Multicore Architectures (DPLASMA) allows sustained performance for distributed systems where each node features multiple sockets of multicore processors and, if applicable, accelerators.

2011

The **Parallel Ultra Light Systolic Array Runtime (PULSAR)** project offers a simple programming model for large-scale distributed-memory machines with multicore processors and hardware accelerators.

2013

The workshop on **Big Data and Extreme-scale Computing (BDEC)** is initiated to systematically map out and account for the ways in which the major issues associated with Big Data intersect with plans that are being laid out for achieving exascale computing.

The **Bench-testing Environment for Automated Software Tuning (BEAST)** project is created to embrace the nature of accelerators, which offer an order of magnitude more computing power and memory bandwith than standard processors.

Designed to measure performance that is representative of modern HPC capability and to serve as a complement to the TOP500 list, the **High Performance Conjugate Gradients (HPCG) benchmark** list is released for the first time.

2014

The **ARGO** project begins developing a new exascale Operating System and Runtime (OS/R) designed to support extreme-scale scientific computation.

Throughout its history, ICL has attracted numerous postdoctoral researchers and professors from multiple disciplines. Many of these experts came to UT specifically to work with Dr. Dongarra. To the right is a list of select ICL alumni who played key roles in the establishment and growth of ICL and went on to build distinguished careers at other organizations and institutions.

1993
The **TOP500** project is launched to improve and renew the Mannheim supercomputer statistics, which had been in use for seven years.

1994
Version 1.0 of a standardized and portable message-passing system, called **Message Passing Interface (MPI)**, is released.

1999
The **Performance Application Programming Interface (PAPI)** project defines a standardized, easy-to-use interface to provide access to hardware performance counters on most major processor platforms.

2000
The **High-Performance Linpack (HPL) Benchmark** for Distributed-Memory Computers, a software package that solves a (random) dense linear system in double-precision (64 bits) arithmetic on distributed-memory computers, is released.

2006
Four institutions merge technologies to initiate the **Open Source Message Passing Interface (Open MPI)**: FT-MPI from UTK/ICL, LA-MPI from Los Alamos National Laboratory, and LAM/MPI from Indiana University, with contributions from the PACX-MPI team at the University of Stuttgart.

2008
Matrix Algebra on GPU and Multicore Architectures (MAGMA) allows applications to fully exploit the power of heterogeneous systems of multi/manycore CPUs and GPUs/coprocessors to deliver the fastest time-to-accurate-solution within given energy constraints.

2012
The **Parallel Runtime Scheduling and Execution Controller (PaRSEC)** is designed to provide a generic framework for architecture-aware scheduling and management of micro-tasks on distributed manycore heterogeneous architectures.

User Level Failure Mitigation (ULFM), a set of new interfaces for MPI that enables message-passing programs to restore MPI functionality affected by process failures, is developed.

2014
The **Rapid Python Deep Learning Infrastructure (RaPyDLI)** project begins delivering productivity and performance to the deep learning community by combining high-level Python, C/C++, and Java environments with libraries supporting GPU accelerators and MIC coprocessors (Intel Xeon Phi).

2015
Work begins on **Performance API (PAPI) EX**, or PAPI-EX, a project to extend PAPI with measurement tools for changing hardware and software paradigms.

SELECT ICL ALUMNI

Emmanuel Agullo	INRIA, France
Ed Anderson	EPA
Thara Angskun	Suranaree University of Technology
Dorian Arnold	University of New Mexico
Zhaojun Bai	University of California, Davis
Richard Barrett	Sandia National Laboratories
Adam Beguelin	Sensr.net
Susan Blackford	CSP, INC.
Alfredo Buttari	CNRS
Henri Casanova	University of Hawaii, Manoa
Zizhong (Jeffrey) Chen	University of California, Riverside
Jaeyoung Choi	Soongsil University, Korea
Andy Cleary	Amazon
David Cronk	Leidos
Frederic Desprez	ENS-Lyon, France
Victor Eijkhout	University of Texas, Austin
Graham Fagg	Microsoft
Mathieu Faverge	University of Bordeaux
Karl Fuerlinger	Ludwig-Maximilians-University Munich
Edgar Gabriel	University of Houston
Sven Hammarling	Numerical Algorithms Group
Greg Henry	Intel
Julien Langou	University of Colorado, Denver
Kevin London	Microsoft
Hatem Ltaief	KAUST, Saudi Arabia
Bob Manchek	Stratus Technologies
Keith Moore	Independent Consultant
Shirley Moore	Oak Ridge National Laboratory
Antoine Petitet	ESI Group, France
Roldan Pozo	NIST
Yves Robert	ENS-Lyon, France
Ken Roche	University of Washington
Erich Strohmaier	Lawrence Berkeley National Laboratory
Martin Swany	Indiana University
Dan Terpstra	Living Waters for the World
Françoise Tisseur	University of Manchester
Bernard Tourancheau	Université Joseph Fourier de Grenoble
Sathish Vadhiyar	Indian Institute of Science, India
Robert van de Geijn	University of Texas, Austin
Reed Wade	Artella, New Zealand
Vince Weaver	University of Maine
Clint Whaley	Louisiana State University
Felix Wolf	TU Darmstadt
Haihang You	Chinese Academy of Sciences

2016 HIGHLIGHTS

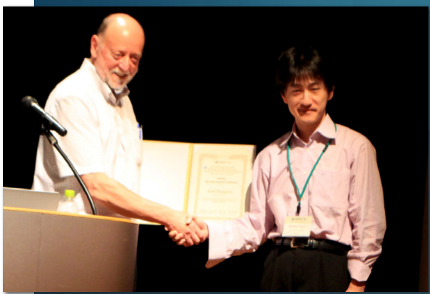
DESTINATION EXASCALE

ICL to Participate in a Project Essential to the Nation’s Security and Competitiveness

The Exascale Computing Project (ECP) was established in July 2015 as part of the National Strategic Computing Initiative. The aim of the ECP is to develop a capable exascale ecosystem that will encompass applications, system software, hardware technologies and architectures, and workforce development to meet the scientific and national security needs of the US Department of Energy in the early-2020s time frame. In 2016, ICL won proposals to participate in six of 35 software development awards and one of four co-design center awards for the ECP. First year funding for all the ECP software development awards totals $34 million. The four co-design centers were selected as part of a four year, $48 million funding award.

For its role in the ECP software projects and the co-design center, ICL will receive about $3.3 million in funding the first year and more than $3.4 million in each of the next two years, for an overall total of approximately $10.2 million. DOE’s vote of confidence in ICL as evidenced by these awards is a product of ICL’s more than 25 years of hard work in delivering solid open-source software to the research community, industry, and society in general.

ACCOLADES



Russian Academy of Sciences Headquarters
MOSCOW

Jack Dongarra Wins the HPDC Achievement Award

For his groundbreaking contributions to high performance computing, Jack Dongarra was honored with the 2016 High Performance Parallel and Distributed Computing (HPDC) Achievement Award, which he accepted at the 25th annual Association for Computing Machinery's HPDC conference in Kyoto, Japan.

The HPDC conference established the achievement award to recognize leading computer scientists who have made long-lasting and influential contributions to the foundation or practice of high-performance parallel and distributed computing and for seminal contributions and a sustained record of high impact in the field.

Paper on ATLAS Earns SC16 IEEE Test of Time Award

Clint Whaley and Jack Dongarra's "Automatically Tuned Linear Algebra Software," a paper written and published nearly 20 years ago for the SC98 supercomputing conference, received an IEEE Test of Time Award at SC16 in Salt Lake City. This paper was a major development in what would become the ATLAS software project.

Regarding the selection of this paper, the SC16 conference website stated: "This paper has received hundreds of citations, and new citations still appear. In addition to the portable performance that ATLAS provides to the CSE community, the autotuning strategies used in ATLAS have been an inspiration to other research teams who are doing similar work."

> Whaley, C., and J. Dongarra, "**Automatically Tuned Linear Algebra Software**", *1998 ACM/IEEE conference on Supercomputing (SC '98)*, Orlando, FL, IEEE Computer Society, November 1998.

Dongarra Becomes a Foreign Member of the Russian Academy of Sciences

On October 31, Vladimir Fortov, president of the Russian Academy of Sciences (RAS), informed Jack Dongarra that the organization had elected him as a foreign member. In making this esteemed list of well-known scientists from various countries and disciplines, Dongarra joins a number of other notable prior-elected computer scientists—such as Americans Don Knuth, professor emeritus at Stanford, and Mike Stonebraker, adjunct professor at the Massachusetts Institute of Technology. The RAS roster of foreign members also includes seven Nobel laureates, including famous American statesman Henry Kissinger.

2016 RESEARCH

What originally began over 25 years ago as in-depth investigations of the numerical libraries that encode the use of linear algebra in software, has grown into an extensive research portfolio. We have evolved and expanded our research agenda to accommodate the aforementioned evolution in HPC, which includes a focus on algorithms and libraries for multicore and hybrid computing. As we have gained a solid understanding of the challenges presented in these domains, we have further expanded our scope to include work in performance evaluation and benchmarking for high-end computers, as well as work in high-performance parallel and distributed computing, with efforts focused on message passing and fault tolerance. On the following pages, we provide brief summaries of some of our efforts in these areas.

ADAPT

Next-Gen Open MPI

The ADAPT project proposes to enhance, harden, and modernize the Open MPI library in the context of the ongoing revolution in processor architecture and system design. On the large systems expected before the end of this decade, the degree of parallelism (intra and inter node) will presumably increase by several orders of magnitude (based on the exascale roadmap predictions). To efficiently handle such systems, MPI implementations will have to adopt more asynchronous and thread-friendly behaviors to extract the best performance from more complex architectures.

The project team seeks to create a viable foundation for a new generation of Open MPI components that enables a rapid exploration of new physical capabilities, provides greatly improved performance portability, and works toward full interoperability between classes of components. ADAPT explores process placement, distributed topologies, file accesses, point-to-point and collective communications, and different approaches to fault tolerance.

ARGO

A new architecture for Exascale computing

The ARGO project is developing a new exascale Operating System and Runtime (OS/R) designed to support extreme-scale scientific computation. Disruptive new computing technologies, such as 3D memory, ultra-low-power cores, and embedded network controllers, are changing the scientific computing landscape. As it becomes clear that incremental approaches to operating systems and runtimes (OS/R) cannot grow into an exascale solution, we propose a novel radical approach.

ARGO is designed on a new, agile modular architecture that supports both global optimization and local control. It aims to efficiently leverage new chip and interconnect technologies while addressing the new modalities, programming environments, and workflows expected at Exascale. It is designed from the ground up to run future HPC applications at extreme scales.

CoDAASH

Co-design Approach for Advances in Software and Hardware

The Co-design Approach for Advances in Software and Hardware (CoDAASH) project focuses on understanding the relationship between algorithms and hardware platforms and how to jointly optimize the software and hardware to achieve efficient implementations for applications in materials science, chemistry, and physics. CoDAASH is a joint effort between the University of Tennessee, Knoxville; Iowa State University; University of Texas, El Paso; and the University of California, San Diego; and is funded by the United States Air Force Office of Scientific Research (AFOSR).

ICL's contribution focuses on expressing certain computational chemistry algorithms in the form of a data flow graph (DAG) and subsequently mapping the DAG representation of the kernels to the hardware platforms. This representation allows for capturing the essential properties of the algorithms (e.g., data dependencies), and enables computation at extreme scale in the era of manycore and highly heterogeneous platforms, by utilizing the hardware components (e.g., CPUs or GPUs) that perform best for the type of computational task under consideration. The dataflow-based form of these algorithms makes them compatible with next-generation task scheduling systems, such as PaRSEC.

DARE

Data-driven Autotuning for Runtime Execution

The objective of the Data-driven Autotuning for Runtime Execution (DARE) project is to provide application-level performance tuning capabilities to the end user. DARE's development motivation stems from the never-ending hurdles of performance tuning of the PLASMA and MAGMA linear algebra libraries. These hurdles motivated the development of a software architecture that combines three components: hardware analysis, kernel modeling, and workload simulation.

With DARE, the hardware analysis block builds a detailed model of the hardware, its computational resources (CPU cores, GPU accelerators, Xeon Phi coprocessors), and its memory system (host memories, device memories, multiple levels of cache). The kernel modeling block builds accurate performance models for the computational kernels involved in the workload, depending on granularity, place of execution, induced memory traffic, etc.; and the workload simulation block rapidly simulates a large number of runs to find the best execution conditions, while relying on the information provided by the other two blocks. The ultimate objective of DARE is to arrange the blocks in a continuous refinement loop that can serve as a framework for optimizing applications beyond the field of dense linear algebra.

DPLASMA

Distributed Parallel Linear Algebra Software for Multicore Architectures

FIND OUT MORE AT
http://icl.utk.edu/dplasma/

The Distributed Parallel Linear Algebra Software for Multicore Architectures (DPLASMA) package is the leading implementation of a dense linear algebra package for distributed heterogeneous systems. It is designed to deliver sustained performance for distributed systems where each node features multiple sockets of multicore processors, and if available, accelerators like GPUs or Intel Xeon Phi. DPLASMA achieves this objective by deploying the Parallel Linear Algebra Software for Multicore Architectures (PLASMA) algorithms on distributed memory systems by leveraging the state-of-the-art PaRSEC runtime.

In addition to traditional ScaLAPACK data distribution, DPLASMA provides interfaces for users to expose arbitrary data distributions. The algorithms transparently operate on local data, or introduce implicit communications to resolve dependencies, removing the burden of initial data reshuffle, and providing to the user a novel approach to address load balance.

EASIR

Extreme-scale Algorithms and Solver Resilience

The mission of the Extreme-scale Algorithms and Solver Resilience (EASIR) project is to close the performance gap between the peak capabilities of HPC hardware and the performance realized by high performance computing applications. To carry out this mission, the EASIR project team develops architecture-aware algorithms and libraries, and the supporting runtime capabilities, to achieve scalable performance and resilience on heterogeneous architectures.

The project team includes personnel from ORNL, Sandia National Laboratories, the University of Illinois, the University of California Berkeley, and the University of Tennessee (ICL). ICL's efforts focus on providing components and services in a vertically integrated software stack, from low-level runtime process and thread scheduling to multicore aware library interfaces, multicore dense linear algebra, scalable iterative methods, and advanced parallel algorithms that break traditional parallelism bottlenecks.

EMBRACE

Evolvable Methods for Benchmarking Realism and Community Engagement

FIND OUT MORE AT
https://sites.google.com/site/wsembrace/

The primary goal of the Evolvable Methods for Benchmarking Realism and Community Engagement (EMBRACE) project is to find the most rational way to design and develop forward-looking benchmarks followed by correctly interpreting their results. This concept has become difficult over time because the performance engineering side of the scientific computing community is broadening, its applications are diverse, and the hardware platforms are quickly evolving. The aim of the EMBRACE project then is to facilitate a technical and reasoned debate about benchmarking, and to keep this debate alive over time.

EMBRACE is a collaboration between ICL/UTK and Georgia Tech. ICL's contribution includes expertise in benchmark design and implementation, and community engagement. In November 2015, an EMBRACE Birds-of-a-Feather session was held at the SC15 conference in Austin, TX. The reception was very positive and the need for reimagining the benchmarking process, creation, and interpretation was clearly voiced by the attendees. A workshop for IPDS17 is upcoming and was met with enthusiastic response as a forum for discussion about the community's need for performance benchmarking.

FT-LA

Fault Tolerant Linear Algebra

FIND OUT MORE AT
http://icl.utk.edu/ft-la/

The Fault Tolerant Linear Algebra (FT-LA) research effort is aimed at understanding and developing Algorithm Based Fault Tolerance (ABFT) into major dense linear algebra kernels. With parallel machines currently reaching up to 300,000 cores, fault tolerance has never been so paramount. The scientific community has to tackle process failures from two directions: first, efficient middleware needs to be designed to detect failures, and second, the numerical applications have to be flexible enough to permit the recovery of the lost data structures.

At ICL, we have successfully developed fault-tolerant MPI middleware and, more recently, an FT-LA library that will efficiently handle several process failures. The project now supports soft (bit-flips) and hard (process failure) failures on a significant range of dense linear algorithms, from one-sided factorizations (Cholesky, LU and QR) to two-sided factorizations (Hessenberg, tri-diagonalization, and bi-diagonalization). Future work in this area involves the development of scalable fault-tolerant singular value decompositions and eigendecomposition, following the ABFT principles.

HPCC

The HPC Challenge Benchmark Suite

FIND OUT MORE AT
http://www.hpcchallenge.org/

The HPC Challenge (HPCC) benchmark suite is designed to establish, through rigorous testing and measurement, the bounds of performance on many real-world applications for computational science at extreme scale. To this end, the benchmark includes a suite of tests for sustained floating-point operations, memory bandwidth, rate of random memory updates, interconnect latency, and interconnect bandwidth. The main factors that differentiate the various components of the suite are the memory access patterns that, in a meaningful way, span the space of memory access characteristics, which is spanned by temporal and spatial locality. The components of the suite are brought together inside HPCC, which allows information to pass between the components and provide a comprehensive testing and measurement framework that goes beyond the sum of its parts.

Each year, the HPCC Awards competition features contestants who submit performance numbers from the world's largest supercomputer installations, as well as alternative implementations that use a vast array of parallel programming environments. The results are announced at the annual gathering of the International Conference for High Performance Computing, Networking, Storage and Analysis; and are available to the public to help track the progress of both the high-end computing arena and the commodity hardware segment.

HPCG

The High Performance Conjugate Gradients Benchmark

FIND OUT MORE AT
http://hpcg-benchmark.org/

The High Performance Conjugate Gradients (HPCG) benchmark is designed to measure performance that is representative of modern scientific applications. It does so by exercising the computational and communication patterns that are commonly found in real science and engineering codes, which are often based on sparse iterative solvers. HPCG exhibits the same irregular accesses to memory and fine-grain recursive computations that dominate large-scale scientific workloads used to simulate complex physical phenomena.

HPCG 3.0 Reference Code was released on November 11, 2015 for the SC15 conference in Austin, TX. In addition to bug fixes, this release positions HPCG to even better represent modern PDE solvers, which reflect the behavior of explicit methods that involve unassembled matrices, and aids in running HPCG on production supercomputing installations. The reference version is accompanied by binary releases from Intel and NVIDIA that are carefully optimized for the vendors' respective hardware platforms. Since its inception in 2013, the community's reception of the benchmark has been overwhelmingly positive, and the constant feedback leads to the continuous improvement of the code and its scope. The current HPCG Performance List was also released at SC16 and now features over 100 supercomputing sites. HPCG is a collaboration between ICL and Sandia National Laboratories.

HPL

The High Performance LINPACK Benchmark

FIND OUT MORE AT
http://icl.utk.edu/hpl/

The High Performance LINPACK (HPL) benchmark is a software package that solves a (randomly generated) dense linear system in double precision (64-bit) arithmetic on distributed-memory computers. Written in a portable ANSI C and requiring an MPI implementation, as well as either the BLAS or VSIPL library, HPL is often one of the first programs to run on large computer installations, producing a result that can be submitted to the biannual TOP500 list of the world's fastest supercomputers.

HPL 2.2, released in 2016, includes several bug fixes and accuracy enhancements based on user feedback. The major focus of HPL 2.2 is to improve the accuracy of reported benchmark results, and ensure scalability of the code on large supercomputer installations with hundreds of thousands of computational cores. The last version also featured a detailed time-of-run accounting to help with assessing power requirements at the time of execution, a metric which has been reported with TOP500 results since 2007 and is also highlighted on the Green500 list. In 2011, the LINPACK benchmark app for iOS achieved performance of over 1 Gflop/s on an Apple iPad 2, with per-watt performance easily beating supercomputing solutions, including the most power-efficient systems based on hardware accelerators. The App now achieves over 4 Gflop/s on the iPad Air and Pro.

MAGMA

Matrix Algebra on GPU and Multicore Architectures

FIND OUT MORE AT
http://icl.utk.edu/magma/

Matrix Algebra on GPU and Multicore Architectures (MAGMA) is a collection of next-generation linear algebra (LA) libraries for heterogeneous architectures. The MAGMA package supports interfaces for current LA packages and standards (e.g., LAPACK and BLAS) to allow computational scientists to easily port any LA-reliant software components to heterogeneous architectures. MAGMA allows applications to fully exploit the power of current heterogeneous systems of multi/manycore CPUs and multi-GPUs/coprocessors to deliver the fastest possible time-to-accurate-solution within given energy constraints.

New for 2016, MAGMA 2.2 features LAPACK-compliant routines for multicore CPUs enhanced with NVIDIA GPUs, and includes more than 400 routines, covering one-sided dense matrix factorizations and solvers, and two-sided factorizations and eigen/singular-value problem solvers, as well as a subset of highly optimized BLAS for GPUs. Tuning and support are added for the newest NVIDIA Pascal P100 GPU. The MAGMA sparse and MAGMA batched packages were added with the MAGMA 1.6 release and continuously extended and improved since then. MAGMA provides multiple-precision arithmetic support (S/D/C/Z, including mixed precision). Most of the algorithms are hybrid, using both multicore CPUs and GPUs, but starting with the 1.6 release, GPU-specific algorithms were added.

Open MPI

Open source MPI

FIND OUT MORE AT
http://icl.cs.utk.edu/open-mpi/

The Open MPI Project is an open-source Message Passing Interface (MPI) implementation that is developed and maintained by a consortium of academic, research, and industry partners. MPI primarily addresses the message-passing parallel programming model, in which data is moved from the address space of one process to that of another process through cooperative operations on each process. Open MPI integrates technologies and resources from several other projects (HARNESS/FT-MPI, LA-MPI, LAM/MPI, and PACX-MPI) to build the best MPI library available. A completely new MPI-3.1 compliant implementation, Open MPI offers advantages for system and software vendors, application developers, and computer science researchers.

ICL's efforts in the context of Open MPI have significantly improved its scalability, performance on manycore environments, and architecture-aware capabilities, such as adaptive shared-memory behaviors and dynamic collective selection, making it ready for the next-generation Exascale challenges.

OpenSHMEM

Communication Library, Specification, and Tools Ecosystem

FIND OUT MORE AT
http://www.openshmem.org/

OpenSHMEM is a Partitioned Global Address Space (PGAS) library interface specification that aims to provide a standard Application Programming Interface (API) for SHMEM libraries to aid portability across multiple vendors—including SGI, Cray, IBM, HP, Mellanox, and Intel. OpenSHMEM supports one-sided communication and is a perfect fit for applications with irregular communication patterns with small/medium-sized data transfers, since it is optimized for low-latency data transfers.

The OpenSHMEM Library API provides calls for data communication, group synchronization, data collection, data reduction, distributed locking of critical regions, and data and process accessibility to OpenSHMEM PEs. PEs put/get data to/from remotely accessible symmetric data objects on other PEs.

PAPI

The Performance API

FIND OUT MORE AT
http://icl.utk.edu/papi/

The Performance Application Programming Interface (PAPI) is the primary vehicle through which ICL gave performance measurement support to the larger HPC community. PAPI is an ongoing project that supplies a consistent interface and methodology for collecting performance counter information from various hardware and software components, including most major CPUs, GPUs and accelerators, interconnects, I/O systems, and power interfaces, as well as virtual cloud environments. Provided as a linkable library or shared object, PAPI can be called directly in a user program or used transparently through a variety of third-party tools, making it a de facto standard for performance counter analysis. Industry liaisons with Bull, Cray, Intel, IBM, NVIDIA, and others ensure seamless integration of PAPI with new architectures at or near their release.

In 2015, ICL began work on PAPI-EX with the University of Maine. The goal of PAPI-EX is to build support for performance counters available in recent generations of CPUs and GPUs, develop support for system-wide hardware performance counter monitoring, and create a sampling interface and user-facing command-line tool for PAPI. PAPI-EX will also incorporate a counter inspection toolkit to improve understanding of low-level hardware events. In 2016, ICL/UTK was awarded funding for the ECP EXA-PAPI project, which will augment PAPI-EX.

PaRSEC

Parallel Runtime Scheduling and Execution Controller

FIND OUT MORE AT
http://icl.utk.edu/parsec/

The Parallel Runtime Scheduling and Execution Controller (PaRSEC) is a generic framework for architecture-aware scheduling and management of micro-tasks on distributed manycore heterogeneous architectures. Applications we consider are expressed as a Direct Acyclic Graph (DAG) of tasks, with edges designating data dependencies. DAGs are represented in a compact problem-size independent format that can be queried to discover data dependencies in a totally distributed fashion, a drastic shift from today's programming models, which are based on sequential flow of execution.

PaRSEC orchestrates the execution of an algorithm on a particular set of resources, assigns computation threads to the cores, overlaps communications and computations, and uses a dynamic, fully distributed scheduler based on architectural features such as NUMA nodes and algorithmic features such as data reuse. PaRSEC includes a set of tools to generate the DAGs and integrate them in legacy codes, a runtime library to schedule the micro-tasks on heterogeneous resources, and tools to evaluate and visualize the efficiency of the scheduling. Many dense and sparse linear algebra extensions have been implemented, as well as chemistry, and seismology applications, which produced significant speedup in production codes.

PLASMA

Parallel Linear Algebra Software for Multicore Architectures

FIND OUT MORE AT
http://icl.utk.edu/plasma/

The Parallel Linear Algebra Software for Multicore Architectures (PLASMA) package is a dense linear algebra package at the forefront of multicore computing, designed to deliver the highest possible performance from a system with multiple sockets of multicore processors. PLASMA achieves this objective by combining state-of-the-art solutions in parallel algorithms, scheduling, and software engineering. Currently, PLASMA offers a collection of routines for solving linear systems of equations, least-square problems, eigenvalue problems, and singular-value problems.

PLASMA relies on runtime scheduling of parallel tasks, which is based on the idea of assigning work to cores based on the availability of data for processing at any given point in time. The concept, which is sometimes called data-driven scheduling, is closely related to the idea of expressing computation through a task graph, often referred to as the DAG (Directed Acyclic Graph), and the flexibility of exploring the DAG at runtime.

POMPEI

Rapid Python Deep Learning Infrastructure

The objective of the Programming with OpenMP4 for Exascale Investigations (POMPEI) Project is to investigate data structure programming paradigms and their interoperability with OpenMP 4.5 tasks. POMPEI will evaluate existing APIs (like SharP from ORNL) and co-design new APIs targeting numerical linear algebra routines and applications to harness the potential of extreme-scale heterogeneous systems.

The POMPEI project is a collaboration between ICL/UTK and ORNL. It is based on the MAGMA library and involves the development of a new unified programming approach for dense linear algebra on large-scale heterogeneous systems. POMPEI will demonstrate how OpenMP/OpenACC constructs can enhance the MAGMA library to interoperate with new data abstractions in a unified programming approach supporting various computing architectures.

PULSAR

Parallel Ultra-Light Systolic Array Runtime

FIND OUT MORE AT
https://bitbucket.org/icl/pulsar

The Parallel Ultra-Light Systolic Array Runtime (PULSAR) project provides a dataflow programming model inspired by systolic arrays, which were popularized by Hsiang-Tsung Kung and Charles E. Leiserson. PULSAR Runtime (PRT) offers a complete Application Programming Interface (API) for building and executing a Virtual Systolic Array (VSA)—a collection of Virtual Data Processors (VDPs) connected with channels and communicating using packets.

The runtime supports distributed memory systems with multicore processors and relies on POSIX Threads (aka Pthreads) for intra-node multithreading, and on the Message Passing Interface (MPI) for inter-node communication. The runtime also supports multiple NVIDIA GPU accelerators, in each distributed memory node, using the Compute Unified Device Architecture (CUDA) platform.

RaPyDLI

Rapid Python Deep Learning Infrastructure

The Rapid Python Deep Learning Infrastructure (RaPyDLI) project delivers productivity and performance to the deep learning (DL) community by combining high-level Python, C/C++, and Java environments with carefully designed libraries supporting GPU accelerators and Xeon Phi coprocessors. DL has made major impacts in areas such as speech recognition, drug discovery, and computer vision. This success relies on training large neural nets—currently, up to 10 billion connections trained on 10 million images—using either large-scale commodity clusters or smaller HPC systems where accelerators perform with high efficiency. This approach is of prime importance as the hardware accelerators enable much more sophisticated neural networks by increasing the available computational power by more than an order of magnitude.

RaPyDLI is a collaboration between ICL/UTK, Indiana University, and Stanford University, with each institution contributing their long-standing expertise in the field. Currently, ICL's focus for the RaPyDLI project is on efficient GPU kernel execution and optimization of scheduling strategies to reduce inefficiencies in the current code base in terms of performance and idle time.

LAPACK

The Linear Algebra PACKage

FIND OUT MORE AT
http://www.netlib.org/lapack/

ScaLAPACK

Scalable LAPACK

FIND OUT MORE AT
http://www.netlib.org/scalapack/

The Linear Algebra PACKage (LAPACK) and Scalable LAPACK (ScaLAPACK) are widely used libraries for efficiently solving dense linear algebra problems. ICL has been a major contributor to the development and maintenance of these two packages since their inception. LAPACK is sequential, relies on the BLAS library, and benefits from the multicore BLAS library. ScaLAPACK is parallel distributed and relies on the BLAS, LAPACK, MPI, and BLACS libraries.

LAPACK 3.7.0 was released in December 2016. LAPACK 3.7.0 includes TSQR for Least Square, Communication-avoiding Symmetric-indefinite Factorization with Aasen's triangular tridiagonalization, Level 3 BLAS use by Rook Pivoting form of LDL, the reduction to tridiagonal routine using the two stage algorithm. and an improved Complex Jacobi SVD. Since 2011, LAPACK has included LAPACKE, a native C interface for LAPACK developed in collaboration with INTEL, which provides NAN check and automatic workspace allocation. ScaLAPACK 2.0.0, which includes the MRRR algorithm and new Nonsymmetric Eigenvalue Problem routines, was released in November 2011. Two additional ScaLAPACK versions (2.0.1 and 2.0.2) were released in 2012 for minor bug fixes.

SILAS

Sustained Innovation for Linear Algebra Software

The main goal of the NSF-funded project on Sustained Innovation for Linear Algebra Software (SILAS) is to update two of the most widely used numerical libraries in the history of Computational Science and Engineering—LAPACK and ScaLAPACK—for the ongoing revolution in processor architecture and system design. Working with partners at the University of California, Berkeley and the University of Colorado, Denver, ICL used SILAS to enhance and harden these essential libraries in order to prepare them for the kind of extreme scale systems and applications that are now coming online.

SILAS is organized around three complementary objectives: 1) wherever possible, SILAS delivers seamless access to the most up-to-date algorithms, numerical implementations, and performance, by way of the familiar Sca/LAPACK programming interface; 2) wherever necessary, SILAS makes advanced algorithms, numerical implementations and performance capabilities available through new interface extensions; and 3) SILAS provides a well-engineered conduit through which new discoveries at the frontiers of research in these areas can be channeled as quickly as possible to all the application communities.

SparseKaffe

The Sparse direct methods via Run-time Scheduling and Execution of Kernels with Auto-tunable and Frequency-scaling Features for Energy-aware computing on heterogeneous architectures (SparseKafe) project will create fast and efficient sparse direct methods for platforms with multicore processors with one or more accelerators (e.g., GPUs or Xeon Phi coprocessors). SparseKaffe spans the platform pyramid, from desktop machines to extreme-scale systems consisting of multiple heterogeneous nodes connected through a high-speed network, with the goal of achieving orders of magnitude gains in computational performance, while also paying careful attention to the energy requirements.

The SparseKaffe project is a collaboration between ICL/UTK, the University of Florida, and Texas A&M University. ICL's work on the project will concentrate on dynamic runtime scheduling using the dataflow model, which will leverage, and be a natural extension of, ICL's work on runtimes as part of the MAGMA, PLASMA, and PaRSEC projects. The autotuning of the algorithm-specific computational kernels will apply the principles behind ICL's BEAST project.

SUPER

The Institute for Sustained Performance, Energy, and Resilience

FIND OUT MORE AT
http://super-scidac.org/

The Institute for Sustained Performance, Energy, and Resilience (SUPER), led by the University of Southern California, has organized a broad-based project involving several universities and DOE laboratories with expertise in compilers, system tools, performance engineering, energy management, and resilience to ensure that DOE's computational scientists can successfully exploit the emerging generation of high performance computing systems.

SUPER is extending performance modeling and autotuning technology to heterogeneous and petascale computing systems, investigating application-level energy-efficiency techniques, exploring resilience strategies for petascale applications, and developing strategies that collectively optimize performance, energy efficiency, and resilience. UTK/ICL work focuses on performance measurement, power and energy measurements, and resilience techniques for hard and soft errors.

TACOMA

Tensor Algebra Computations Over Manycore Architectures

The goal of the Tensor Algebra Computations Over Manycore Architectures (TACOMA) project is to design a high-performance tensor contractions package for heterogeneous systems with multicore CPUs and GPU accelerators. TACOMA will target the acceleration of a number of important applications, including high-order finite element method (FEM) simulations, machine learning, big data analytics, multiphysics, and more.

The TACOMA project is a collaboration between UTK/ICL, Lawrence Livermore National Laboratory (LLNL), the University of Paris-Sud, and Inria to develop software infrastructure for tensor contractions to, in part, expedite the porting of packages like BLAST to next-generation 150+ petaflop/s DOE machines (e.g., Sierra, LLNL's IBM Power9 machine). The TACOMA effort is multidisciplinary, incorporating linear algebra, languages, code generation and optimizations, domain science, and application-specific numerical algorithms. TACOMA is funded until March 2017, at which time it will be superseded by the CEED project.

TESSE

Task-based Environment for Scientific Simulation at Extreme Scale

The goal of the Task-based Environment for Scientific Simulation at Extreme Scale (TESSE) project is to use an application-driven design to create a general-purpose, production-quality software framework that attacks the twin challenges of programmer productivity and portable performance for advanced scientific applications on the massively parallel, hybrid, manycore systems of today and tomorrow.

The TESSE team is composed of researchers from Stonybrook, Virginia Tech, and ICL/UTK, who have designed a system that uses Directed Acyclic Graph (DAG)-based data flow as the basis of the software. This capability, with the extensions being explored by the TESSE project, will provide significant potential advantages in ease of composition, performance, and ease of migration to future architectures for irregular parallel applications. The TESSE team's next major goal is the ubiquitous existence of a powerful DAG-based data flow tool that complements, and is completely interoperable with, mainstream standard parallel programming models such as OpenMP and MPI.

TOP500

Ranking the 500 fastest computers in the world

FIND OUT MORE AT
http://www.top500.org/

With more than two decades of tracking supercomputing progress, the TOP500 list continues to provide a reliable historical record of large computer installations around the world. Seeing critical HPC metrics across all 500 machines and drawing a rich picture of the state of the art in the supercomputing field with respect to performance, energy consumption, and power efficiency are now easier than ever because of the list. Recent editions of the TOP500 have, for example, revealed a meteoric rise of accelerated and specialized architectures, as well as the integration of power and energy measurements from the Green500 project.

In November 2016, the number one spot in the TOP500 belonged squarely to the Chinese Sunway TaihuLight system, which nearly tripled the performance of the second-place finisher, Tianhe-2, also from China. Moreover, the presence of the Chinese permeated the list, as for the first time they featured the same number of systems as the US. Making this edition of the rankings especially diverse were new processing and networking technologies from the US and international companies. Of special note were Knights Landing processors from Intel, Pascal GPUs from NVIDIA, and new Infiniband interconnects from Mellanox and Intel.

UCX

Universal Communication X

FIND OUT MORE AT
http://www.openucx.org/

The Universal Communication X (UCX) is a collaborative effort between industry, laboratories, and academia to create an open-source, production-grade communication framework for data-centric and high-performance applications. UCX makes common network code available through a universal, portable, and performance-driven communication API that exposes low-level capabilities of high-performance networks, henceforth providing critical services for implementing features needed by high-level parallel programming constructs.

These services include lightweight remote memory access operations, lightweight synchronizations, active messages, atomic operations, etc. Open UCX has been successfully deployed to support communication middlewares, including OpenSHMEM, Open MPI, and PaRSEC.

ULFM

User Level Failure Mitigation

FIND OUT MORE AT
http://fault-tolerance.org/

User-Level Failure Mitigation (ULFM) is a set of new interfaces for the Message Passing Interface (MPI) that enables message-passing applications to restore MPI functionality affected by process failures. The MPI implementation is spared the expense of internally taking protective and corrective automatic actions against failures. Instead, it can prevent any fault-related deadlock situation by reporting operations whose completions were rendered impossible by failures.

Using the constructs defined by ULFM, applications and libraries drive the recovery of the MPI state. Consistency issues resulting from failures are addressed according to an application's needs, and the recovery actions are limited to the necessary MPI communication objects. Therefore, the recovery scheme is more efficient than a generic, automatic recovery technique, and can achieve the goals of enabling applications to resume communication after failure and maintaining extreme communication performance outside of recovery periods. A wide range of application types and middlewares are already building on top of ULFM to deliver scalable and user-friendly fault tolerance.

NEW FOR 2017

In winning the array of seven awards from the US Department of Energy's Exascale Computing Project (ECP) during the fall of 2016, ICL earned a place among the elite set of researchers from DOE laboratories who will create the software infrastructure for the nation's first exascale machines. In addition, several other projects will begin in the next year under new awards from DOE and the National Science Foundation. Following are brief summaries of the activities in which we will be involved when the work funded by these new awards gets underway in 2017.

AsyncIS

Asynchronous Iterative Solvers for Extreme-scale Computing

The Asynchronous Iterative Solvers for Extreme-scale Computing (AsyncIS) project focuses on improving the efficiency of iterative solvers by removing their synchronization points. Especially on architectures with massive amounts of parallelism, synchronization points are harmful to the performance of numerical algorithms. Removal will provide significant acceleration and prepare the linear algebra software stack for the parallelism levels of extreme-scale supercomputers. The AsyncIS project is a collaboration between Georgia Tech, ICL/UTK, Temple University, and Sandia National Laboratories, and is funded by the Department of Energy.

BONSAI

BEAST on OpeN Software Attuning Infrastructure

The BEAST on OpeN Software Attuning Infrastructure (BONSAI) project will develop a software infrastructure for using parallel hybrid systems at any scale to execute large, concurrent autotuning sweeps to make the optimization of computational kernels for GPU accelerators and manycore coprocessors faster. BONSAI builds on the BEAST project, which prototyped and validated an autotuning workflow consisting of generation and pruning of the parameter search space; compilation, benchmarking, and profiling of the kernels that pass the pruning; and collection, analysis, and visualization of the performance data.

ECP CEED

Center for Efficient Exascale Discretizations

The LLNL-led Center for Efficient Exascale Discretizations (CEED) co-design center will develop the next-generation discretization software and algorithms for a wide range of DOE and NNSA applications to run efficiently on future hardware. UTK/ICL will be instrumental in identifying, developing, and optimizing tensor contractions that are essential building blocks for these applications. We will be integrally involved in co-designing APIs with the LLNL scientists, external partners on the team, and vendors, and we will deliver a high-performance tensor contractions package through our MAGMA library.

ECP DTE

Distributed Tasking for Exascale

The Parallel Runtime and Execution Controller (PaRSEC) environment implements a dataflow paradigm that attacks the two sides of the exascale challenge: managing extreme-scale parallelism, while maintaining the performance portability of code. The Distributed Tasking for Exascale (DTE) project will transition the PaRSEC framework to exascale supercomputers, while extending it to further facilitate programmability and increase scientific productivity. DTE plans to develop PaRSEC to align with the critical needs of the ECP application communities in terms of scalability, interoperability, and productivity.

ECP EXA-PAPI

Extending PAPI-EX for Exascale

The Exascale Performance Application Programming Interface (Exa-PAPI) project extends PAPI-EX with performance counter monitoring capabilities for new and advanced ECP hardware and software technologies, fine-grained power management support, and integration capabilities for exascale paradigms such as task-based runtime systems. Exa-PAPI also adds events that originate from the ECP software stack, extending the notion of performance events to include not only hardware but also software-based information. Enabling the uniform monitoring of both types of events through one PAPI interface will transform exascale application development.

OMPI-X

Open MPI for Exascale

Despite their widespread use and popularity, neither Open Message Passing Interface (MPI) nor the MPI standard itself is fully ready for the changes in hardware and software the movement toward exascale will bring. OMPI-X will address a broad spectrum of issues in both the standard and the implementation: runtime interoperability for MPI+X and beyond, extending the MPI standard to better support coming exascale architectures, improving Open MPI scalability and performance, supporting more-dynamic execution environments, enhancing resilience in MPI and Open MPI, evaluating MPI tools interfaces, and maintaining quality assurance.

PEEKS

Production-ready, Exascale-enabled Krylov Solvers for Exascale Computing

The Production-ready, Exascale-enabled Krylov Solvers for Exascale Computing (PEEKS) project will explore the redesign of solvers and advance DOE's Extreme-scale Algorithms and Solver Resilience (EASIR) project for compatibility with the Trilinos software ecosystem. Products created from EASIR prototypes will be applied in developing interfaces and fundamental sparse kernels to make future GPU solvers and latency-tolerant solvers viable configure-time plugins in Trilinos. These innovations will greatly increase the ability to support new algorithms in the future.

PULSE

PAPI Unifying Layer for Software-defined Events

The PAPI Unifying Layer for Software-defined Events (PULSE) project focuses on enabling cross-layer and integrated modeling and analysis of the entire hardware system by extending Performance API (PAPI) with the capability to expose performance metrics for key software components found in the HPC software stack.

PULSE will enhance the impact of the abstraction and unification layer that PAPI provides to hardware events to also encompass MPI, OpenMP, LAPACK, MAGMA, and task-based runtimes software events.

SLATE

Software for Linear Algebra Targeting at Exascale

For a decade ICL has applied algorithmic and technological innovations in pioneering, implementing and disseminating dense linear algebra (DLA) software. The Software for Linear Algebra Targeting at Exascale (SLATE) project will converge and consolidate that software into a DLA library that will integrate seamlessly into the ECP ecosystem. This software will enable a wide range of applications to fully exploit the power of near-exascale and exascale systems as soon as they come online. SLATE is expected to substantially impact the ECP software ecosystem and applications.

SMURFS

Simulation and Modeling for Understanding Resilience and Faults at Scale

The Simulation and Modeling for Understanding Resilience and Faults at Scale (SMURFS) project will predict the complex interactions among a given application, a real or hypothetical hardware and software environment, and a fault tolerance strategy at extreme scale. UTK/ICL plans to design, develop, and validate new analytical and system component models that use semi-detailed software and hardware specifications to predict application performance in terms of time-to-solution and energy consumption. We will also gather valuable insights about application behavior at scale.

xSDK4ECP

Extreme-scale Scientific Software Development Kit for ECP

The Extreme-scale Scientific Software Development Kit for the ECP (xSDK4ECP) project will enable the seamless combined use of diverse, independently developed software packages for ECP applications. Project efforts will be prioritized according to the specific needs of the ECP and the National Nuclear Security Administration's Advanced Technology Development and Mitigation applications. Partners in this project are UTK/ICL, Argonne National Laboratory, Lawrence Berkeley National Laboratory, Lawrence Livermore National Laboratory (LLNL), Sanida National Laboratories, and the University of California, Berkeley.

2016 PUBLICATIONS

Abdelfattah, A., H. Ltaeif, D. Keyes, and J. Dongarra, "**Performance optimization of Sparse Matrix-Vector Multiplication for multi-component PDE-based applications using GPUs**", *Concurrency and Computation: Practice and Experience*, vol. 28, issue 12, pp. 3447 - 3465, May 2016.

Abdelfattah, A., A. Haidar, S. Tomov, and J. Dongarra, "**Performance, Design, and Autotuning of Batched GEMM for GPUs**", *ISC High Performance Computing*, Volume 9697 of the series *Lecture Notes in Computer Science* pp 21-38, 2016.

Abdelfattah, A., A. Haidar, S. Tomov, and J. Dongarra, "**Performance Tuning and Optimization Techniques of Fixed and Variable Size Batched Cholesky Factorization on GPUs**", *International Conference on Computational Science (ICCS'16)*, San Diego, CA, June 2016.

Abdelfattah, A., M. Baboulin, V. Dobrev, J. Dongarra, C. Earl, J. Falcou, A. Haidar, I. Karlin, T. Kolev, I. Masliah, et al., "**High-Performance Tensor Contractions for GPUs**", *International Conference on Computational Science (ICCS'16)*, San Diego, CA, June 2016.

Abdelfattah, A., A. Haidar, S. Tomov, and J. Dongarra, "**On the Development of Variable Size Batched Computation for Heterogeneous Parallel Architectures**", *The 17th IEEE International Workshop on Parallel and Distributed Scientific and Engineering Computing (PDSEC 2016)*, IPDPS 2016, Chicago, IL, IEEE, May 2016.

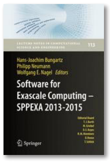
Abdelfattah, A., M. Baboulin, V. Dobrev, J. Dongarra, C. Earl, J. Falcou, A. Haidar, I. Karlin, T. Kolev, I. Masliah, et al., "**High-Performance Tensor Contractions for GPUs**", *University of Tennessee Computer Science Technical Report*, no. UT-EECS-16-738: University of Tennessee, January 2016.

Anzt, H., M. Baboulin, J. Dongarra, Y. Fournier, F. Hulsemann, A. Khabou, and Y. Wang, "**Accelerating the Conjugate Gradient Algorithm with GPU in CFD Simulations**", *VECPAR*, 2016.

Anzt, H., E. Chow, and J. Dongarra, "**On block-asynchronous execution on GPUs**", no. 291: *LAPACK Working Note*, November 2016.

Anzt, H., E. Chow, T. Huckle, and J. Dongarra, "**Batched Generation of Incomplete Sparse Approximate Inverses on GPUs**", *Proceedings of the 7th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems*, pp. 49–56, November 2016.

Anzt, H., E. Chow, J. Saak, and J. Dongarra, "**Updating Incomplete Factorization Preconditioners for Model Order Reduction**", *Numerical Algorithms*, 2016.



Anzt, H., E. Chow, D. Szyld, and J. Dongarra, "**Domain Overlap for Iterative Sparse Triangular Solves on GPUs**", In *Software for Exascale Computing - SPPEXA*, vol. 113 of *Lecture Notes in Computer Science and Engineering*: Springer International Publishing, pp. 527–545, September 2016.

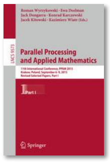
Anzt, H., J. Dongarra, M. Kreutzer, G. Wellein, and M. Kohler, "**Efficiency of General Krylov Methods on GPUs – An Experimental Study**", *The Sixth International Workshop on Accelerators and Hybrid Exascale Systems (AsHES)*, Chicago, IL, IEEE, May 2016.

Anzt, H., M. Kreutzer, E. Ponce, G. D. Peterson, G. Wellein, and J. Dongarra, "**Optimization and performance evaluation of the IDR iterative Krylov solver on GPUs**", *International Journal of High Performance Computing Applications*, May 2016.

Anzt, H. and E. S. Quintana-Ortí, "**Fine-grained Bit-Flip Protection for Relaxation Methods**", *Journal of Computational Science*, November 2016.

Anzt, H., S. Tomov, and J. Dongarra, "**On the performance and energy efficiency of sparse linear algebra on GPUs**", *International Journal of High Performance Computing Applications*, October 2016.

Aupy, G., A. Benoit, H. Casanova, and Y. Robert, "**Scheduling Computational Workflows on Failure-prone Platforms**", *International Journal of Networking and Computing*, vol. 6, no. 1, pp. 2-26, 2016.



Baboulin, M., J. Dongarra, A. Remy, S. Tomov, and I. Yamazaki, "**Dense Symmetric Indefinite Factorization on GPU Accelerated Architectures**", *Lecture Notes in Computer Science*, vol. 9573: Springer International Publishing, pp. 86-95, September 2015, 2016.

Benoit, A., A. Cavelan, Y. Robert, and H. Sun, "**Assessing General-purpose Algorithms to Cope with Fail-stop and Silent Errors**", *ACM Transactions on Parallel Computing*, August 2016.

Benoit, A., A. Cavelan, Y. Robert, and H. Sun, "**Optimal Resilience Patterns to Cope with Fail-stop and Silent Errors**", *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Chicago, IL, IEEE, May 2016.

Bosilca, G., A. Bouteiller, A. Guermouche, T. Herault, Y. Robert, P. Sens, and J. Dongarra, "**Failure Detection and Propagation in HPC Systems**", *Proceedings of the The International Conference for High Performance Computing, Networking, Storage and Analysis (SC'16)*, Salt Lake City, Utah, IEEE Press, pp. 27:1-27:11, November 2016.

Bosilca, G., T. Herault, and J. Dongarra, "**Context Identifier Allocation in Open MPI**", *University of Tennessee Computer Science Technical Report*, no. ICL-UT-16-01: University of Tennessee, January 2016.

Dong, T., A. Haidar, P. Luszczek, S. Tomov, A. Abdelfattah, and J. Dongarra, "**MAGMA Batched: A Batched BLAS Approach for Small Matrix Factorizations and Applications on GPUs**", *ICL Tech Report*, 08/2016.

Dongarra, J., "**Report on the Sunway TaihuLight System**", *University of Tennessee Computer Science Technical Report*, no. UT-EECS-16-742: University of Tennessee, June 2016.

Dongarra, J., M. A. Heroux, and P. Luszczek, "**High Performance Conjugate Gradient Benchmark: A new Metric for Ranking High Performance Computing Systems**", *International Journal of High Performance Computing Applications*, vol. 30, issue 1, pp. 3 - 10, February 2016.

Dongarra, J., "**The HPL Benchmark: Past, Present & Future**", *ISC High Performance*, Frankfurt, Germany, July 2016.

Haidar, A., B. Brock, S. Tomov, M. Guidry, J. Jay Billings, D. Shyles, and J. Dongarra, "**Performance Analysis and Acceleration of Explicit Integration for Large Kinetic Networks using Batched GPU Computations**", *2016 IEEE High Performance Extreme Computing Conference (HPEC '16)*, Waltham, MA, IEEE, September 2016.

Haidar, A., S. Tomov, K. Arturov, M. Guney, S. Story, and J. Dongarra, "**LU, QR, and Cholesky Factorizations: Programming Model, Performance Analysis and Optimization Techniques for the Intel Knights Landing Xeon Phi**", *IEEE High Performance Extreme Computing Conference (HPEC'16)*, Waltham, MA, IEEE, September 2016.

Haugen, B., "**Performance Analysis and Modeling of Task-Based Runtimes**", PhD diss., Department of Electrical Engineering and Computer Science, Knoxville, University of Tennessee, May 2016.

Herrmann, J., G. Bosilca, T. Herault, L. Marchal, Y. Robert, and J. Dongarra, "**Assessing the Cost of Redistribution followed by a Computational Kernel: Complexity and Performance Results**", *Parallel Computing*, vol. 52, pp. 22-41, February 2016.

Jagode, H., A. YarKhan, A. Danalis, and J. Dongarra, "**Power Management and Event Verification in PAPI**", *Tools for High Performance Computing 2015: Proceedings of the 9th International Workshop on Parallel Tools for High Performance Computing*, September 2015, Dresden, Germany, Dresden, Germany, Springer International Publishing, pp. pp. 41-51, 2016.

Jia, Y., P. Luszczek, and J. Dongarra, "**Hessenberg Reduction with Transient Error Resilience on GPU-Based Hybrid Architectures**", *30th IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, Chicago, IL, IEEE, May 2016.

Lopez, M., V. Larrea, W. Joubert, O. Hernandez, A. Haidar, S. Tomov, and J.. Dongarra, "**Towards Achieving Performance Portability Using Directives for Accelerators**", *The International Conference for High Performance Computing, Networking, Storage and Analysis (SC'16), Third Workshop on Accelerator Programming Using Directives (WACCPD)*, Salt Lake City, Utah, November 2016.

Luszczek, P., M. Gates, J. Kurzak, A. Danalis, and J. Dongarra, "**Search Space Generation and Pruning System for Autotuners**", *30th IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, Chicago, IL, IEEE, May 2016.

Masliah, I., A. Ahmad, A. Haidar, S. Tomov, J. Falcou, and J. Dongarra, "**High-performance Matrix-matrix Multiplications of Very Small Matrices**", *22nd International European Conference on Parallel and Distributed Computing (Euro-Par'16)*, Grenoble, France, Springer International Publishing, August 2016.

Newburn, C. J., G. Bansal, M. Wood, L. Crivelli, J. Planas, A. Duran, P. Souza, L. Borges, P. Luszczek, S. Tomov, et al., "**Heterogeneous Streaming**", *The Sixth International Workshop on Accelerators and Hybrid Exascale Systems (AsHES), IPDPS 2016*, Chicago, IL, IEEE, May 2016.

Wu, W., G. Bosilca, R. vandeVaart, S. Jeaugey, and J. Dongarra, "**GPU-Aware Non-contiguous Data Movement In Open MPI**", *25th International Symposium on High-Performance Parallel and Distributed Computing (HPDC'16)*, Kyoto, Japan, ACM, June 2016.

Yamazaki, I., S. Nooshabadi, S. Tomov, and J. Dongarra, "**High Performance Realtime Convex Solver for Embedded Systems**", *University of Tennessee Computer Science Technical Report*, no. UT-EECS-16-745, October 2016.

Yamazaki, I., S. Tomov, and J. Dongarra, "**Non-GPU-resident Dense Symmetric Indefinite Factorization**", *Concurrency and Computation: Practice and Experience*, November 2016.

Yamazaki, I., S. Tomov, and J. Dongarra, "**Stability and Performance of Various Singular Value QR Implementations on Multicore CPU with a GPU**", *ACM Transactions on Mathematical Software (TOMS)*, vol. 43, issue 2, 10/2016.

YarKhan, A., J. Kurzak, P. Luszczek, and J. Dongarra, "**Porting the PLASMA Numerical Library to the OpenMP Standard**", *International Journal of Parallel Programming*, June 2016.

2016 EVENTS

FEBRUARY 8-11
Network for Sustainable Ultrascale Computing (NESUS) Winter School & PhD Symposium
Timisoara, Romania

FEBRUARY 23-25
Open MPI Developer's Meeting
Dallas, TX

FEBRUARY 29-MARCH 3
MPI Forum
Chicago, IL

MARCH 12-16
Principles and Practice of Parallel Programming (PPoPP16)
Barcelona, Spain

MARCH 20-25
14th Copper Mountain Conference on Iterative Methods
Denver, CO

APRIL 4-7
GPU Technology Conference (GTC) 2016
San Jose, CA

APRIL 12-15
SIAM Conference on Parallel Processing for Scientific Computing
Paris, France

APRIL 18-22
Asia Supercomputing Community's 2016 Student Supercomputing Challenge (ASC16)
Wuhan, China

MAY 2-6
GPU Hackathon University of Delaware
Newark, DE

MAY 18-21
11th Scheduling for Large Scale Systems Workshop
Nashville, TN

MAY 18-19
Workshop on Batched, Reproducible, and Reduced Precision BLAS
Knoxville, TN

MAY 23-27
IEEE International Parallel & Distributed Processing Symposium (IPDPS)
Chicago, IL

ICL's Jack Dongarra served as co-chair of the Expert Committee and judge of the Asia Supercomputing Community's 2016 Student Supercomputing Challenge (ASC16). Hosted by the Huazhong University of Science and Technology, the challenge is billed as the world's largest supercomputing hackathon.

MAY 27
Parallel and Distributed Scientific and Engineering Computing (PDSEC 2016)
Chicago, IL

JUNE 6-8
International Conference on Computational Science (ICCS)
San Diego, CA

In May, ICL hosted the Workshop on Batched, Reproducible, and Reduced Precision BLAS (BBLAS). This workshop focused on extending the Basic Linear Algebra Software Library (BLAS) to provide greater parallelism for small-size operations, reproducibility, and reduced precision support.

JUNE 6-9
MPI Forum
Bellevue, WA

JUNE 12-14
Accelerated Data and Computing (ADAC) Workshop
Luguano, Switzerland

JUNE 15-17
Workshop on Big Data and Extreme Scale Computing (BDEC)
Frankfurt, Germany

JUNE 19-23
International Supercomputing Conference
Frankfurt, Germany

JULY 6-8
9th International Workshop on Parallel Matrix Algorithms and Applications (PMAA16)
Bordeaux, France

AUGUST 2-4
OpenSHMEM 2016: Third workshop on OpenSHMEM and Related Technologies
Baltimore, MD

AUGUST 3-6
9th Scalable Tools Workshop
Lake Tahoe, CA

AUGUST 16-19
Open MPI Developer's Meeting
Dallas, TX

AUGUST 24-25
Intel Science and Technology Center for Big Data Annual Research Retreat
Portland, OR

AUGUST 30-SEPTEMBER 1
Smoky Mountains Computational Sciences and Engineering Conference
Gatlinburg, TN

SEPTEMBER 13-15
IEEE High Performance Extreme Computing Conference (HPEC16)
Waltham, MA

SEPTEMBER 25-28
EuroMPI
Edinburgh, UK

SEPTEMBER 27-29
Exascale Requirements Review for Advanced Scientific Computing Research (ASCR) Workshop
Rockville, MD

OCTOBER 3-6
Workshop on Clusters, Clouds, and Data for Scientific Computing (CCDSC)
Lyon, France

NOVEMBER 1-4
Linux Plumbers Conference
Santa Fe, NM

NOVEMBER 13-18
SC16
Salt Lake City, UT

NOVEMBER 29
Exascale Computing Project PI Meeting
Lemont, IL

DECEMBER 5-6
PEEKS Kick-off Meeting
Albuquerque, NM

DECEMBER 5-8
MPI Forum
Dallas, TX

DECEMBER 12
TESSE Workgroup Meeting
New York, NY

The International Conference for High Performance Computing Networking, Storage, and Analysis (SC), established in 1988, is a staple of ICL's November itinerary. SC is vital to the growth and evolution of high performance computing in the United States because it is the only US event that elicits substantial participation from all segments of the HPC community, including hundreds of users, developers, vendors, research institutions, and representatives of government funding agencies. Such a talent-rich gathering enables participants to discuss challenges, share innovations, and coordinate relationships and collaborations with some of the best minds in scientific and high performance computing.

SC16 was held in Salt Lake City, Utah, on November 13–18. As usual, ICL had a significant presence at SC, with faculty, research staff, and students giving talks, presenting a poster and papers, and leading "Birds of a Feather" sessions. Since the University of Tennessee did not have a booth at the conference this time around, ICL provided its schedule of SC16 activities, a list of its attendees, and project information via a virtual booth online.

PARTNERSHIPS AND COLLABORATIONS

Since its inception in 1989, ICL has fostered relationships with many academic institutions and research centers, and proactively built enduring collaborative partnerships with HPC vendors and industry leaders, in the US and abroad. In this section, we recognize many of those partners and collaborators, most of whom we continue to work with today.

GOVERNMENT AND ACADEMIC

The exchange of ideas, expertise, and personnel between ICL and academic and government research institutions is an integral part of our success. Moreover, our relationships with these institutions routinely have a multidisciplinary dimension in that many of our collaborators are primarily focused on scientific areas different from our own, such as biology, chemistry, or physics.

Argonne National Laboratory	Georgia Tech	Indiana University	JICS Joint Institute for Computational Sciences
Berkeley Lab	Lawrence Livermore National Laboratory	Los Alamos National Laboratory EST. 1943	LSU
Michigan Technological University 1885	Massachusetts Institute of Technology	NCSA	NICS
Northwestern University	Oak Ridge National Laboratory	Pacific Northwest National Laboratory	Rice
The State University of New Jersey Rutgers	SDSC San Diego Supercomputer Center	Sandia National Laboratories	Stony Brook University
Stanford University	Texas A&M University	Berkeley University of California	UC Riverside University of California
University of Colorado Denver	University of Florida	University of Houston	Illinois University of Illinois at Urbana-Champaign
The University of Iowa	USC University of Southern California	University of Washington	Virginia Tech

INDUSTRY

The vendors and industry research leaders with whom we partner are very impprtant to us. They contribute significantly to our efforts to be a world leader in computational science research. Many have used our work, including our linear algebra libraries and performance analysis tools. As a result of these exchanges, we maintain close working relationships with many industry leaders.

INTEL SCIENCE AND TECHNOLOGY CENTER FOR BIG DATA

ICL joined the SciDB project of the Intel Science and Technology Center (ISTC) for Big Data, one of a series of research collaborations that Intel is establishing with universities in the US to identify and prototype revolutionary technology opportunities, and exchange expertise in various fields of high performance computing.

In the case of ICL, the lab will help improve the efficiency of large-scale data analytics by providing efficient codes for linear algebra on the Intel Xeon Phi coprocessor. The lab will also provide expertise on fault tolerance to help make the compute intensive portion of data management more resilient, which is essential given the large databases used in Big Data applications. Finally, the distributed nature of large data processing calls for optimal data distribution and redistribution operations, which has long been one of ICL's core strengths.

INTERNATIONAL COLLABORATIONS

Barcelona Supercomputing Center	Barcelona, Spain
Central Institute for Applied Mathematics	Jülich, Germany
Doshisha University	Kyoto, Japan
Dresden University of Technology	Dresden, Germany
École Normale Supérieure de Lyon	Lyon, France
École Polytechnique Federale de Lausanne	Lausanne, Switzerland
European Centre for Research and Advanced Training in Scientific Computing	Toulouse, France
European Exascale Software Initiative	European Union
Forschungszentrum Jülich	Jülich, Germany
Hokkaido University	Sapporo, Japan
High Performance Computing Center Stuttgart	Stuttgart, Germany
INRIA	France
ETH Zurich	Zurich, Switzerland
Kasetsart University	Bangkok, Thailand
King Abdullah University of Science and Technology	Saudi Arabia
Laboratoire d'Informatique de Paris 6 (LIP6)	Paris, France
Moscow State University	Moscow, Russia
National Institute of Advanced Industrial Science and Technology (AIST)	Tsukuba, Japan
Parallel and HPC Application Software Exchange	Tsukuba, Japan
Prometeus	Mannheim, Germany
Regionales RechenZentrum Erlangen	Erlangen, Germany
RIKEN	Wako, Japan
Rutherford Appleton Laboratory	Oxford, England
Soongsil University	Seoul, South Korea
Technische Universitaet Wien	Vienna, Austria
Tokyo Institute of Technology	Tokyo, Japan
Université Claude Bernard de Lyon	Lyon, France
University of Bordeaux	Bordeaux, France
University of Capetown	Capetown, South Africa
University of Manchester	Manchester, England
University of Paris-Sud	Paris, France
University of Umeå	Umeå, Sweden

LEADERSHIP

While we illustrate in this report that dedication to translating leading-edge research into high-impact software is a hallmark of the ICL brand, yet another is leadership. For us, this means bringing to bear the knowledge and social influence of our people to teach, motivate, and inspire others to maximize their efforts toward the goal of advancing the quality of high performance computing. We express our strong commitment to leadership through our involvement in conferences, workshops, and computational science education programs.

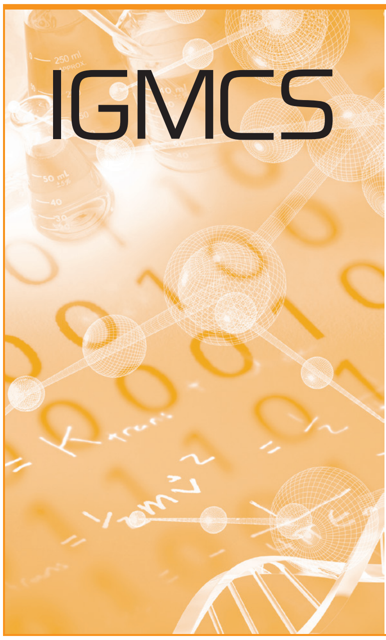
THE UNIVERSITY OF TENNESSEE KNOXVILLE

CENTER FOR INFORMATION TECHNOLOGY RESEARCH

The Center for Information Technology Research (CITR) was established in 2001 to drive the growth and development of leading-edge information technology research at the University of Tennessee. CITR's first objective is to develop a thriving, well-funded community in basic and applied information technology research at UT to help the university capitalize on the rich supply of research opportunities that now exist in this area. As part of this goal, CITR staff members currently provide primary administrative and technical support for ICL, helping maintain the lab's status as a world leader in high-performance and scientific computing research. CITR has also provided secondary support for other UT research centers.

CITR's second objective is to grow an interdisciplinary computational science program as part of the university curriculum. To this end, CITR helped establish the Interdisciplinary Graduate Minor in Computational Science (IGMCS) to offer UT graduate students an opportunity to acquire the balanced package of knowledge and skills required for today's computationally intensive research methods. CITR is also the sole provider of administrative support for the IGMCS.

FIND OUT MORE AT http://citr.cs.utk.edu/



INTERDISCIPLINARY GRADUATE MINOR IN COMPUTATIONAL SCIENCE

Addressing the need for a new educational strategy in computational science, CITR worked with faculty and administrators from several departments and colleges in 2007 to help establish a new university-wide program that supports advanced-degree concentrations in this critical new field across the curricula. Under the Interdisciplinary Graduate Minor in Computational Science (IGMCS), students pursuing advanced degrees in a variety of fields of science and engineering are able to extend their education with special courses of study that teach them both the fundamentals and the latest ideas and techniques from this new era of information-intensive research. Through the IGMCS, graduate students can augment their graduate work in their chosen field with courses from other disciplines that are specifically tailored to round out their education in computational science. The IGMCS curriculum, requirements, and policies are governed by a program committee composed of faculty members from participating IGMCS academic units and departments.

FIND OUT MORE AT http://igmcs.utk.edu/

BDEC
BIG DATA AND EXTREME-SCALE COMPUTING

BDEC Frankfurt
JUNE 2016

For several years, the United States, the European Union, Japan and China have each moved aggressively to develop their own plans for achieving exascale computing in the next decade. Such concerted planning by the traditional leaders of HPC speaks eloquently about both the substantial rewards that await the success of such efforts, and the unprecedented technical obstacles that apparently block the path to get there. But while these exascale initiatives, including the International Exascale Software Project (IESP), have understandably focused on the big challenges of exascale for hardware and software architecture, the relatively recent emergence of the phenomena of "Big Data" in a wide variety of scientific fields represents a tectonic shift that is transforming the entire research landscape on which all plans for exascale computing must play out.

The workshop series on Big Data and Extreme-scale Computing (BDEC) marks a distinctly new phase for the work of the IESP community, and is premised on the idea that we must begin to systematically map out and account for the ways in which the major issues associated with Big Data intersect with, impinge upon, and potentially change the national (and international) plans that are now being laid for achieving exascale computing.

In 2016, ICL was instrumental in organizing and staging the fourth BDEC workshop in Frankfurt, Germany. Along with Jack Dongarra, and following through with work they began with the IESP and the first three BDEC meetings in Charleston, Fukuoka, and Barcelona, several members of ICL's CITR staff, including Terry Moore, Tracy Rafferty, Sam Crawford, and David Rogers, played essential roles in making the fourth BDEC workshop a major success. A follow-up BDEC mini-workshop was held in Frankfurt, Germany during the International Supercomputing Conference (ISC 16). This one-day "reporting meeting" focused on current progress in the US, Europe, and Asia, and on organizing the writing effort to create a "Pathways to Convergence" report to the scientific computing community. The initial draft of that report was presented to the community in a successful "Birds of a Feather" session at SC16.

The next BDEC workshop, which will concentrate on finalizing the BDEC "Pathways" document, will be held in Wuxi, China in early March 2017. Given the ever-increasing emphasis that science, government, and industry continue to place on both Big Data and extreme-scale computing, this example of ICL's community leadership seems likely to become more and more prominent.

FIND OUT MORE AT http://exascale.org/

PEOPLE

ICL Group
FALL 2016

As the landscape in high performance computing continues to rapidly evolve, remaining at the forefront of discovery requires great vision and skill. To address this evolution and to remain a leader in innovation, we have assembled a staff of top researchers from all around the world who apply a variety of novel and unique approaches to the challenges and problems inherent in world-class scientific computing.

As part of an engineering college at a top 50 public research university, we have a responsibility to combine exemplary teaching with cutting-edge research. As such, we regularly employ bright and motivated graduate and undergraduate students. We have been, and will continue to be, very proactive in securing internships and assistantships for highly motivated and hardworking students.

CURRENT GROUP


Ahmad Abdelfattah
POST DOCTORAL RESEARCH ASSOCIATE


Hartwig Anzt
RESEARCH SCIENTIST I


George Bosilca
RESEARCH DIRECTOR


Aurelien Bouteiller
RESEARCH SCIENTIST II


Chongxiao Cao
GRADUATE RESEARCH ASSISTANT


Anthony Danalis
RESEARCH SCIENTIST II


Jack Dongarra
UNIVERSITY DISTINGUISHED PROFESSOR
DIRECTOR OF ICL


David Eberius
GRADUATE RESEARCH ASSISTANT


Teresa Finchum
ADMINISTRATIVE SPECIALIST II


Mark Gates
RESEARCH SCIENTIST I


Damien Genet
POST DOCTORAL RESEARCH ASSOCIATE

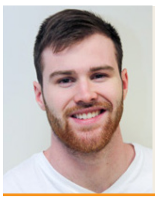

Scott Gibson
INFORMATION SPECIALIST I


Azzam Haidar
RESEARCH SCIENTIST II


Hanumantharayappa
GRADUATE RESEARCH ASSISTANT

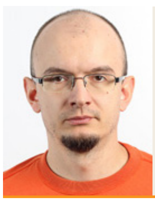

Thomas Herault
RESEARCH SCIENTIST II


Reazul Hoque
GRADUATE RESEARCH ASSISTANT


Harry Hughes
GRADUATE RESEARCH ASSISTANT


Heike Jagode
RESEARCH DIRECTOR


Cindy Knisley
FINANCIAL SPECIALIST


Jakub Kurzak
RESEARCH DIRECTOR


Julie Langou
RESEARCH LEADER


Nuria Losada
VISITING STUDENT

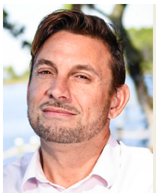
CURRENT GROUP


Xi Luo
GRADUATE RESEARCH ASSISTANT


Piotr Luszczek
RESEARCH DIRECTOR


Tyler McDaniel
GRADUATE RESEARCH ASSISTANT


Terry Moore
ASSOCIATE DIRECTOR


Phil Mucci
RESEARCH CONSULTANT


Thananon Patinyasakdikul
GRADUATE RESEARCH ASSISTANT


Yu Pei
GRADUATE RESEARCH ASSISTANT


Tracy Rafferty
PROGRAM MANAGER


Sangamesh Ragate
GRADUATE RESEARCH ASSISTANT


Stephen Richmond
GRADUATE RESEARCH ASSISTANT


Yves Robert
VISITING SCHOLAR


David Rogers
IT SPECIALIST III


Leighanne Sisk
ADMINISTRATIVE SPECIALIST I


Stanimire Tomov
RESEARCH DIRECTOR


Yaohung Tsai
GRADUATE RESEARCH ASSISTANT


Phil Vaccaro
GRADUATE RESEARCH ASSISTANT


Stephen Wood
VISITING POST DOCTORAL RESEARCH ASSOCIATE


Wei Wu
GRADUATE RESEARCH ASSISTANT

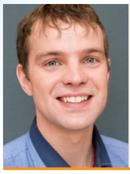

Ichitaro Yamazaki
RESEARCH SCIENTIST I


Asim YarKhan
RESEARCH SCIENTIST II


Dong Zhong
GRADUATE RESEARCH ASSISTANT

2016 VISITORS

Since its founding, ICL has always had a tradition of hosting many visitors from around the world. Some stay only briefly to give insightful seminars or presentations; others remain with us for as long as a year to collaborate, teach, or learn. Our connection to these researchers enables us to leverage an immense array of intellectual resources and work with the best and brightest people in the global HPC community.



Maksims Abalenkovs
UNIVERSITY OF MANCHESTER, UK

Emmanuel Agullo
INRIA
FRANCE

Jaeyoung Choi
SOONGSIL UNIVERSITY
SOUTH KOREA

Camille Coti
UNIVERSITE PARIS 13
FRANCE

Tim Davis
TEXAS A&M

Iain Duff
SCIENCE AND TECHNOLOGY FACILITIES COUNCIL, UK

Mathieu Faverge
INRIA
FRANCE

Karl Fuerlinger
LUDWIG-MAXIMILIANS UNIVERSITY, GERMANY

Sven Hammarling
NUMERICAL ALGORITHMS GROUP, UK

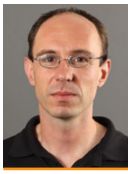
Torsten Hoefler
ETH ZURICH
SWITZERLAND

Emmanuel Jeannot
INRIA, FRANCE

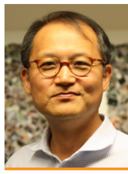
Khairul Kabir
NVIDIA

David Keffer
UTK DEPARTMENT OF MATERIALS SCIENCE & ENGINEERING

Raehyun Kim
SOONGSIL UNIVERSITY
SOUTH KOREA



Julien Langou
UNIVERSITY OF COLORADO



Myungho Lee
SOONGSIL UNIVERSITY
SOUTH KOREA

Yeongha Lee
SOONGSIL UNIVERSITY
SOUTH KOREA

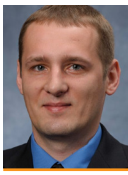
Peter Liaw
UTK DEPARTMENT OF MATERIALS SCIENCE & ENGINEERING

Robert Lucas
INFORMATION SCIENCES INSTITUTE

Dmitry Lyakh
ORNL

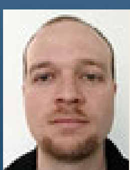
C.J. Newburn
NVIDIA

Sanjay Ranka
UNIVERSITY OF FLORIDA

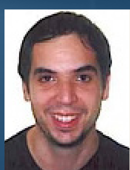


Oleg Shylo
UTK DEPARTMENT OF INDUSTRIAL & SYSTEMS ENGINEERING

Jakub Sistek
UNIVERSITY OF MANCHESTER, UK



Miro Stoyanov
ORNL



Pedro Valero-Lara
BARCELONA SUPERCOMPUTER CENTER

Frank Winkler
ORNL

Mawussi Zounon
UNIVERSITY OF MANCHESTER, UK

ICL ALUMNI

Carolyn Aebischer 1990-1993
Bivek Agrawal 2004-2006
Sudesh Agrawal 2004-2006
Emmanuel Agullo 2009
Jennifer Allgeyer 1993
Wes Alvaro 2007-2011
Ed Anderson 1989-1991
Daniel Andrzejewski 2007
Thara Angskun 2003-2007
Papa Arkhurst 2003
Dorian Arnold 1996-2001
Cedric Augonnet 2010
Marc Baboulin 2008
Zhaojun Bai 1990-1992
Ashwin Balakrishnan 2001-2002
Richard Barrett 1992-1994
Alex Bassi 2000-2001
David Battle 1990-1992
Micah Beck 2000-2001
Daniel Becker 2007
Dulceneia Becker 2010-2012
Adam Beguelin 1991
Annamaria Benzoni 1991
Tom Berry 1991
Vincent Berthoux 2010
Scott Betts 1997-1998
Nikhil Bhatia 2003-2005
Noel Black 2002-2003
Laura Black 1996
Susan Blackford 1989-2001
Wesley Bland 2008-2013
Kartheek Bodanki 2009
David Bolt 1991
Fernando Bond 1999-2000
Carolyn Bowers 1992
Barry Britt 2007-2009
Randy Brown 1997-1999
Bonnie Browne 2011-2012
Cynthia Browne 2005
Murray Browne 1998-1999
Antonin Bukovsky 1998-2003
Greg Bunch 1995

Alfredo Buttari 2004-2007
Giuseppe Bruno 2001
Anthony Canino 2012
Domingo Gimenez Canovas 2001
Henri Casanova 1995-1998
Cedric Castagnede 2012
Ramkrishna Chakrabarty 2005
Sharon Chambers 1998-2000
Zizhong Chen 2001-2006
Jaeyoung Choi 1994-1995
Wahid Chrabakh 1999
Eric Clarkson 1998
Andy Cleary 1995-1997
Michelle Clinard 1989-1991
Vincent Cohen-Addad 2012
Matthias Colin 2004
Charles Collins 2012
Stephanie Cooper 2011-2013
Tom Cortese 2002-2009
Camille Coti 2007
Jason Cox 1993-1997
Sam Crawford 2011-2016
David Cronk 1999 - 2010
Javier Cuenca 2003
Manoel Cunha 2006
Yuanshun Dai 2007-2013
Cricket Deane 1998-1999
Remi Delmas 2006
Frederic Desprez 1994-1995
Jin Ding 2003
Jun Ding 2001-2003
Ying Ding 2000-2001
Martin Do 1993-1994
Simplice Donfack 2014
Leon Dong 2000-2001
Tingxing Dong 2010-2015
Nick Dongarra 2000
David Doolin 1997
Joe Dorris 2015-2016
Andrew Downey 1998-2003
Mary Drake 1989-1992
Julio Driggs 2002-2004

Brian Drum 2001-2004
Peng Du 2005-2012
Eduardo Echavarria 2005
Victor Eijkhout 1992-2005
Brett Ellis 1995-2005
Shawn Ericson 2004
Zachary Eyler-Walker 1997-1998
Lisa Ezzell 2003-2004
Christoph Fabianek 2003
Graham Fagg 1996-2006
Mathieu Faverge 2010-2012
Shengzhog Feng 2005-2006
Don Fike 2000-2016
Salvatore Filippone 2004
Anna Finchum 2010
Mike Finger 1997
Markus Fischer 1997-1998
Len Freeman 2009
Xiaoquan Fu 2003-2004
Erika Fuentes 2003-2007
Karl Fuerlinger 2006-2008
Megan Fuller 2006
Edgar Gabriel 2003-2004
Lynn Gangwer 2000-2001
Tracy Gangwer 1992-1993
Nathan Garner 2001-2006
Kelley Garner 1998
Tina Garrison 1991
Adriana Garties 2011
Peter Gaultney 2011-2014
Christoph Geile 2008
Jean Patrick Gelas 2001
Boris Gelfend 1993
Jonathan Gettler 1996
Eric Greaser 1993
Stan Green 1992-1996
Alice Gregory 2004-2006
Jason Gurley 1997-1998
Bilel Hadri 2008-2009
Hunter Hagewood 2000-2001
Christian Halloy 1996-1997
Sven Hammarling 1996-1997

J. Mike Hammond 1994-1995
Hidehiko Hasegawa 1995-1996
Satomi Hasegawa 1995-1996
Chris Hastings 1996
Blake Haugen 2010-2016
David Henderson 1999-2001
Greg Henry 1996
Julien Herrmann 2011-2012
Holly Hicks 1993-1994
Alexandra Hicks-Hardiman 2009
Sid Hill 1996-1998
Tomoyuki Hiroyasu 2002-2003
George Ho 1998-2000
Josh Hoffman 2008-2010
Jeff Horner 1995-1999
Mitch Horton 2010-2012
Yan Huang 2000-2001
Aurelie Hurault 2009
Chris Hurt 2002
Paul Jacobs 1992-1995
Emmanuel Jeannot 2001-2006
Weizhong Ji 1999-2000
Yulu Jia 2011-2015
Weicheng Jiang 1992-1995
Song Jin 1997-1998
Patrick Johansson 2001
Aral Johnson 2009
Matt Johnson 2011-2013
Sean Jolly 1997-1998
Jan Jones 1992-2008
Kim Jones 1996-1997
Vijay Joshi 2011-2013
Khairul Kabir 2011-2016
Venkata Kakani 2007
Ajay Kalhan 1995
Balajee Kannan 2001
Madhuri Kasam 2007-2008
Kiran Kasichayanula 2010-2012
Ajay Katta 2010
David Katz 2002
Joshua Kelly 2000-2001
Supriya Kilambi 2008

Myung Ho Kim 2005-2006
Youngbae Kim 1992-1996
Jenya Kirshtein 2008
Michael Kolatis 1993-1996
Chandra Krintz 1999-2001
Tilman Kuestner 2010
Krerkchai Kusolchu 2010
Coire Kyle 2005
Amanda Laake 2003-2004
Xavier Lacoste 2012
Julien Langou 2003-2006
Jeff Larkin 2003-2005
Brian LaRose 1990-1992
Frank Lauer 2010
DongWoo Lee 2000-2002
Tracy Lee 1996-2012
Pierre Lemarinier 2008-2010
Todd Letsche 1993-1994
Sharon Lewis 1992-1995
Xiang Li 2001
Yinan Li 2006-2008
Weiran Li 2002
Chaoyang Liu 2000
Kevin London 1996-2005
Matt Longley 1999
Florent Lopez 2014
Hatem Ltaief 2008-2011
Daniel Lucio 2008
Richard Luczak 2000-2001
Teng Ma 2006-2012
Robert Manchek 1990-1996
Gabriel Marin 2013-2014
Tushti Marwah 2004
Theo Mary 2014
Donald McCasland 1994
Paul McMahan 1994-2000
Eric Meek 2003-2006
James Meyering 1991-1992
Jeremy Millar 1998-2002
Michelle Miller 1999-2003
Cindy Mitchell 2001-2002
Stuart Monty 1993
Eric Moore 2000
Keith Moore 1987-2007
Shirley Moore 1993-2012
Robert Morgan 1990-1991
Kishan Motheramgari 1997
Steven Moulton 1991-1993
Daichi Mukunoki 2012
Matthew Nabity 2008
Shankar Narasimhaswami 2004-2005
Rajib Nath 2008-2010
Fernando Navarro 2009
John Nelson 2011-2013
Donnie Newell 2010
Peter Newton 1994-1995
Jonas Nilsson 2001
Jakob Oestergaard 2000
Caroline Papadopoulos 1997-1998
Leelinda Parker 2002
Thibault Parpaite 2014
Dilip Patlolla 2007-2008
Andy Pearson 1989-1991
Paul Peltz 2003-2013
Theresa Pepin 1994
Antoine Petitet 1993-2001
Peter Pham 2012
Gregoire Pichon 2014
Vlado Pjesivac 2008
Jelena Pjesivac-Grbovic 2003-2007
James S. Plank 1991-1992
Ciara Proctor 2008
Tim Poore 2009
Roldan Pozo 1992-1994
Farzona Pulatova 2005-2006
Martin Quinson 2001
Tammy Race 1999-2001
James Ralph 2006-2014
Ganapathy Raman 1998-2000
Kamesh Ramani 2003
Mei Ran 1999-2004
Arun Rattan 1997
Sheri Reagan 1995-1996
Mike Reynolds 1994
George Rhinehart 2012
Jon Richardson 1990-1991
Ken Roche 1999-2004
Andrew Rogers 1997-1999
Tom Rothrock 1997-1998
Tom Rowan 1993-1997
Narapat (Ohm) Saengpatsa 2011
Kiran Sagi 2001-2005
Evelyn Sams 1998-1999
Ken Schwartz 1992-1993
Keith Seymour 1994-2009
Farial Shahnaz 2001
Brian Sheely 2009-2010
Zhiao Shi 2001-2007
Sergei Shinkarev 2005-2007
Majed Sidani 1991-1992
Shilpa Singhal 1996-1998
Matt Skinner 2008
Peter Soendergaard 2000
Raffaele Solca 2012
Gwang Son 2007-2009
Fengguang Song 2003-2012
Thomas Spencer 1999-2001
Erich Strohmaier 1995-2001
Xiaobai Sun 1995
Martin Swany 1996-1999
Daisuke Takahashi 2002
Judi Talley 1993-1999
Ronald Tam 2009
Chunyan Tang 2014-2015
Yuan Tang 2005-2006
Yusuke Tanimura 2003
Keita Teranishi 1998
Dan Terpstra 2001-2014
Joe Thomas 2002-2009
John Thurman 1998-1999
Francoise Tisseur 1997
Jude Toth 1993-1994
Bernard Tourancheau 1993-1994
Lauren Vaca 2004
Sathish Vadhiyar 1999-2003
Robert van de Geijn 1990-1991
Chad Vawter 1995
Eugene Vecharynski 2008
Scott Venckus 1993-1995
Antoine Vernois 2004
Reed Wade 1990-1996
Michael Walters 2001-2005
Mike Waltz 1999
Robert Waltz 1990-1991
Jerzy Wasniewski 2000
Vince Weaver 2010-2012
Scott Wells 1997-2010
David West 1990-1992
R. Clint Whaley 1991-2001
Jody Whisnant 1997-1998
James White 1999
Scotti Whitmire 1995-1996
Susan Wo 2000-2001
Felix Wolf 2003-2005
Jiayi Wu 2004-2007
Qiu Xia 2004-2005
Tinghua Xu 1998-2000
Tao Yang 1999
Erlin Yao 2012-2013
Kevin Ye 2015
Jin Yi 2009-2010
Haihang You 2002-2009
Lamia Youseff 2007
Brian Zachary 2009-2010
Omar Zenati 2012
Yuanlei Zhang 2001-2005
Junlong Zhao 2002
Yong Zheng 2001
Luke Zhou 2000-2001
Min Zhou 2002-2004

ACKNOWLEDGMENTS

For 25 years, our knowledge and hard work have earned the trust and support of many agencies and organizations that have funded, and continue to fund, our efforts. Without them we simply would not be able to conduct leading-edge research.

The main source of support has been federal agencies that are charged with allocating public research funding. Therefore, we acknowledge the following agencies for supporting our efforts, both past and present:

- AFOSR – Air Force Office of Scientific Research
- U.S. Department of Energy – Office of Science
- NNSA – National Nuclear Security Administration
- NSF – National Science Foundation

In addition to the support of the federal government, we have solicited strong support from private industry, which has also played a significant role in our success and growth. We gratefully acknowledge the following vendors for their generosity and support:

- AMD
- intel
- MathWorks™
- NVIDIA®

INTEL PARALLEL COMPUTING CENTER

The Intel Parallel Computing Center (IPCC) program, renewed with ICL in 2015, is composed of universities, institutions, and labs that are leaders in their field, focusing on modernizing applications to increase parallelism and scalability through optimizations that leverage cores, caches, threads, and vector capabilities of microprocessors and coprocessors.

The objective of the Innovative Computing Laboratory's IPCC is the development and optimization of numerical linear algebra libraries and technologies for applications, while tackling current challenges in heterogeneous Intel® Xeon Phi™ coprocessor-based high performance computing. In collaboration with Intel's MKL team, the IPCC at ICL will modernize the popular LAPACK and ScaLAPACK libraries to run efficiently on current and future manycore architectures, and will disseminate the developments through the open source MAGMA MIC library.

NVIDIA GPU CENTER OF EXCELLENCE

The Innovative Computing Laboratory joins a very small and select group of labs given a GPU Center of Excellence designation. ICL/UTK's GPU COE focuses on the development of numerical linear algebra libraries for CUDA-based hybrid architectures. ICL's work on the Matrix Algebra on GPU and Multicore Architectures (MAGMA) project further enables and expands our CUDA-based software library efforts, especially in the area of high-performance scientific computing.

The GPU COE designation also led to the establishment of a productive long-term collaboration between ICL and NVIDIA. As part of the collaboration and GPU COE designation, ICL has continuously received hardware, financial support, and other resources from NVIDIA.

THE UNIVERSITY OF TENNESSEE KNOXVILLE

CENTER FOR INFORMATION TECHNOLOGY RESEARCH

Administrative and technical support for ICL is provided by the Center for Information Technology Research (CITR).

CITR is supported in part by the University of Tennessee Office of Research and Engagement.

INNOVATIVE COMPUTING LABORATORY

Suite 203 Claxton
1122 Volunteer Blvd
Knoxville, TN 37996-3450

INNOVATIVE COMPUTING LABORATORY

FIND OUT MORE AT http://icl.utk.edu/