THE UNIVERSITY OF TENNESSEE KNOXVILLE

INNOVATIVE COMPUTING LABORATORY

CELEBRATING 30 YEARS

2018/19 REPORT

ICL / 2018/19 REPORT

DESIGNED BY **David Rogers** EDITED BY **Sam Crawford**

INNOVATIVE
COMPUTING LABORATORY
2018/19 REPORT

ICL/FROM THE DIRECTOR

If you wanted to pick something to symbolize this past year at the Innovative Computing Laboratory (ICL)—something that brings together the multiplicity of narrative threads connecting ICL's past, present, and future—Oak Ridge National Laboratory's (ORNL's) Summit supercomputer, and its ascent to number one on the TOP500 list of the world's fastest supercomputers, would be a good choice. For one thing, ICL has a close and enduring relationship with ORNL and its long line of superb high-performance computing (HPC) projects, in which Summit is just the latest installment. Second, ICL has developed and supported the TOP500 list and its LINPACK benchmark for more than three decades. Summit also scored number one on the relatively new and important High Performance Conjugate Gradients (HPCG) benchmark, which we have led the effort to establish over the last few years. But at a more fundamental level, the arrival of Summit represents a critical stage in the US Department of Energy's (DOE's) Exascale Computing Project (ECP), and ECP was very much the focus of the work of all ICL groups throughout 2018. Access to Summit has been and will continue to be essential to the success of our ECP projects.

Pursuing those projects gave 2018 a very distinctive character. Looking back over the history of ICL, I am inclined to think that there has never been a time when all of ICL's researchers and staff—collectively—have been as focused and "heads down" as we were this year under ECP's management discipline. Watching our ICL teams consistently deliver high-quality results on our seven ECP projects—SLATE, Exa-PAPI, DTE, PEEKS, Open MPI-X, xSDK4ECP, and CEED—reinforced and intensified my sense (which I remarked upon in last year's letter) of the incredibly high level of professionalism we have achieved.

That professionalism was most fully on display in our efforts to scout, evaluate, recruit, and integrate the kind of highly talented people that we need to continue our ECP successes. This year those efforts bore fruit, beginning with Jamie Finney who joined SLATE as a developer at the end of 2017. Tony Castaldo joined the Exa-PAPI effort the following spring, and then Ali Charara joined SLATE at the beginning of the fall. To carry out the newly added fast Fourier transform subproject (ECP-FFT), Alan Ayala was also added to the SLATE team as a Postdoctoral Research Associate this fall. Finally, at the beginning of the new year, Nuria Losada, whom we know well from her internship with us in 2016, will be joining the DTE project. It is no accident that we have been able to bring so many high-caliber people into ICL so quickly. Aside from all of the hard work, quality people attract quality people. This gives me great confidence going forward in both our ECP work and in all of our other research efforts.

As we discussed at our annual retreat last summer, the world of opportunity that our future research will have to explore is changing in profound and unprecedented ways. Two dimensions of that change stood out in 2018: first, the rise of machine/deep learning and its incorporation into scientific computing as a revolutionary new component in the methodology of science; second, the dramatic proliferation of new data sources (e.g., Internet of things, smart cities, and cyber-physical systems), which will require revolutionary innovation in distributed cyberinfrastructure. In fact, these two themes are connected because at least some of the processing required in the "data periphery" will certainly involve machine/deep learning. Both of these extraordinary developments were front and center at various meetings and workshops this year, including the International Conference for High Performance Computing, Networking, Storage, and Analysis (SC18) and the latest Big Data and Extreme-scale Computing (BDEC) meeting in Bloomington, Indiana. The Bloomington workshop was the first in a new series of six international workshops—BDEC2—which will extend to the end of 2020, and in which we are playing a leadership role. Since the goal of BDEC2 is to draft a design for the new distributed computing platform that science and engineering will require to engage the pervasive "digital continuum" that is now emerging, ICL should be well positioned to attack research opportunities in these exciting new areas.

Since we have all marked our calendars for the anniversary celebration in August, we all know that 2019 will mark ICL's 30th year. Through all the hard work and changing circumstances, the great successes and painful disappointments, we have charted a remarkable path and have traveled it all too quickly. But as that anniversary celebration will remind us—if we need reminding—the main testament to our success over the last three decades is the vital community of loyal and wonderful people, both here in Knoxville and spread out around the world, that ICL has built up and maintained over that time. Of course, I am grateful to the many government, industry, and private sponsors whose support made that success possible. And as always, my special thanks and congratulations go to ICL researchers, staff, and students for their creativity, skill, dedication, and tireless efforts to keep ICL on the leading edge of scientific computing worldwide.

Jack Dongarra
Director, ICL

ICL/INTRODUCTION

Situated in the heart of the University of Tennessee campus and at the nexus of academia, government, and industry, ICL impacts the world as a leader in advanced scientific computing and HPC through research, education, and collaboration.

The unique challenges of today's computational research are characterized by large datasets and the need for greater performance, energy conservation, and resilience. ICL's cutting-edge efforts, which now span 30 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 multi-core, many-core, 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 supercomputing-based discoveries in areas like life sciences, climate science, earthquake prediction, energy exploration, combustion and turbulence, advanced materials science, drug design and more.

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.

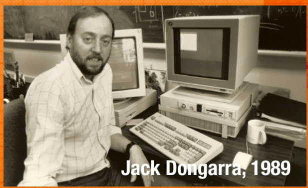
PERFORMANCE ANALYSIS AND BENCHMARKING

Performance evaluation and benchmarking are vital to developing science and engineering applications that run efficiently in an HPC environment. ICL's performance evaluation tools enable 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. These correlations are 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 HPC 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 two decades, and the lab has numerous projects in this arena under active development.

ICL/HISTORY


Jack Dongarra, 1989

1989

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

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

1992

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

1995

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

1997

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

NetSolve (GridSolve) was a client-server system that enabled users to solve complex scientific problems using remote resources.


ATLAS & NetSolve R&D100 Awards

2002

Fault Tolerant MPI (FT-MPI) was an MPI plugin for HARNESS that provided support for fault-tolerant applications crucial for large, long-running simulations.

2003

HPC Challenge was developed for the Defense Advanced Research Projects Agency (DARPA) and consisted of four benchmarks: HPL, Streams, RandomAccess, and PTRANS.

LAPACK for Clusters was developed in the framework of self-adapting numerical software to leverage the convenience of existing sequential environments bundled with the power and versatility of highly tuned parallel codes executed on clusters.

2006

Fault-Tolerant Linear Algebra (FT-LA) is a research effort to develop and implement algorithm-based fault tolerance in commonly used dense linear algebra kernels.

2010

Distributed Parallel Linear Algebra Software for Multi-core Architectures (DPLASMA) is a linear algebra package that enables sustained performance for distributed systems, where each node features multiple sockets of multi-core processors and, if applicable, accelerators like GPUs or Intel Xeon Phi.

2011

The **Parallel Ultra Light Systolic Array Runtime (PULSAR)** project developed a simple programming model for large-scale, distributed-memory machines with multi-core processors and hardware accelerators to automate multithreading, message passing, and multistream multi-GPU programming.

2012

The **Parallel Runtime Scheduling and Execution Controller (PaRSEC)** provides a generic framework for architecture-aware scheduling and management of microtasks on distributed, many-core heterogeneous architectures.

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

2014

Argo is an initiative to develop a new exascale operating system and runtime (OS/R) designed to support extreme-scale scientific computation.

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


ICL 25th Anniversary

2015

PAPI-EX extends PAPI with measurement tools for changing hardware and software paradigms.

Data-driven Autotuning for Runtime Execution (DARE) provides application-level performance tuning capabilities to the end user.

2016

The **Production-ready, Exascale-enabled Krylov Solvers for Exascale Computing (PEEKS)** project will explore the redesign of solvers and extend the DOE's Extreme-scale Algorithms and Solver Resilience (EASIR) project.

The **Software for Linear Algebra Targeting Exascale (SLATE)** project will converge and consolidate previous ICL efforts with LAPACK and ScaLAPACK into a dense linear algebra library that will integrate seamlessly into the ECP ecosystem.

2017

The **Batched BLAS (BBLAS)** effort will create an API for numerical computing routines that process batches of either uniformly sized or varying-size matrices or vectors and will serve as a working forum for establishing this strategy as the next official BLAS standard.


Batched BLAS Workshop

Prof. Jack Dongarra established ICL in 1989 when he received a dual appointment as a Distinguished Professor at the University of Tennessee, Knoxville (UTK) and as a Distinguished Scientist at ORNL. Thirty years later, ICL has grown into an internationally recognized research laboratory specializing in numerical linear algebra, distributed computing, and performance evaluation and benchmarking.

As we look back on the lab's body of work, which now spans three decades, it is important to remember the milestones that shaped the research and direction of ICL. To this end, we present the following projects and initiatives, all of which have special historical significance to ICL and our collaborators.

1992

Still developed today, the **Linear Algebra Package (LAPACK)** is a standard software library for numerical linear algebra.

1993

The **TOP500** was launched to improve and renew the Mannheim supercomputer statistics, which—at the time—had been in use for seven years

1994

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

PVM R&D100 Award

1999

The **Heterogeneous Adaptable Reconfigurable Networked SystemS (HARNESS)** was a pluggable, lightweight, heterogeneous, and distributed virtual machine environment.

Still in active development, the **Performance Application Programming Interface (PAPI)** is a standardized, easy-to-use interface that provides access to hardware performance counters on most major processor platforms.

PAPI R&D100 Award

2000

High-Performance LINPACK (HPL) Benchmark is a benchmark for distributed-memory computers that solves a (random) dense linear system in double-precision (64-bit) arithmetic.

2006

Four institutions merged efforts in 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 Multi-core Architectures (MAGMA) is a linear algebra library that enables applications to exploit the power of heterogeneous systems of multi-core CPUs and multiple GPUs or coprocessors.

Parallel Linear Algebra Software for Multi-core Architectures (PLASMA) is a dense linear algebra package designed to deliver the highest possible performance from a system of multiple sockets of multi-core CPUs.

2009

The **International Exascale Software Project (IESP)** brought together representatives of the global HPC community to plan and create a new software infrastructure for the extreme-scale systems that represent the future of computational science.

2013

The **Big Data and Extreme-scale Computing (BDEC)** workshop was initiated to map out and account for the ways in which the major issues associated with big data intersect with national (and international) plans being laid out for achieving exascale computing.

BDEC

The **Bench-testing Environment for Automated Software Tuning (BEAST)** project enables writing of tunable high-performance kernels by unleashing the power of heuristic autotuning.

The **High Performance Conjugate Gradients (HPCG)** benchmark is designed to measure performance that is representative of modern HPC capability by simulating patterns commonly found in real science and engineering applications.

2015

The **SparseKaffe** project establishes fast and efficient sparse direct methods for platforms with multi-core processors with one or more accelerators.

The **Task-based Environment for Scientific Simulation at Extreme Scale (TESSE)** uses an application-driven design to create a general-purpose software framework focused on programmer productivity and portable performance for scientific applications on massively parallel hybrid systems.

2016

ICL won seven awards through DOE's Exascale Computing Project (ECP) during the fall of 2016 and is the lead institution on four of these projects:

The **Distributed Tasking for Exascale (DTE)** project will extend the capabilities of the PaRSEC framework.

The **Exascale Performance Application Programming Interface (Exa-PAPI)** project builds on PAPI-EX and extends it with performance counter monitoring capabilities for new and advanced ECP hardware and software technologies.

2017

The **MAtrix, TEnsor, and Deep-learning Optimized Routines (MATEDOR)** team is performing the research required to define a standard interface for batched operations (BBLAS) and provide a performance-portable software library that demonstrates batching routines for a significant number of linear algebra kernels.

2018

The goal of **BDEC2**, a follow-on to BDEC and IESP, is to stage six international workshops to enable research communities in a wide range of disciplines to converge on a common platform in order to meet the daunting challenges of achieving exascale computing in the wake of a surging "data tsunami."

BDEC2

The main objective of the ECP **Fast Fourier Transform (ECP-FFT)** project is to design and implement a fast and robust 2-D and 3-D FFT library that targets large-scale heterogeneous systems with multi-core processors and hardware accelerators and to do so as a co-design activity with other ECP application developers.

ICL/HIGHLIGHTS

Thirty Years of Innovative Computing

The coming year marks the 30th anniversary of ICL. Over the past three decades, members of ICL have been conducting innovative research in HPC and cyberinfrastructure while making significant contributions to the HPC community along the way. What started as two graduate students and two postdocs has now grown into a world-renown research laboratory with a staff of over 40 researchers, students, and administrators. This milestone will be celebrated with a 30-year reunion and workshop on August 8–9, 2019.

Getting started in Ayres Hall, ca. 1989.

Jack Dongarra receives ACM-IEEE Computer Society Ken Kennedy Award, November 2013.

Past and present members of ICL attend ICL's 25th anniversary workshop, April 2015.

SIAM/ACM Prize in Computational Science and Engineering

ICL founder and director Jack Dongarra has been selected to receive the SIAM/ACM Prize in Computational Science and Engineering. This prestigious accolade is awarded every two years by SIAM and ACM in the area of computational science in recognition of outstanding contributions to the development and use of mathematical and computational tools and methods for the solution of science and engineering problems.

Prof. Dongarra will receive this award for his key role in the development of software and software standards, software repositories, performance and benchmarking software, and in community efforts to prepare for the challenges of exascale computing—especially in adapting linear algebra infrastructure to emerging architectures.

The award will be presented to Prof. Dongarra on February 25, 2019 at the SIAM Conference on Computational Science and Engineering in Spokane, Washington.

Thirty Years of SC

The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC) celebrated its 30th year in November 2018. ICL's Jack Dongarra is among those who have attended every SC event, starting in 1988. Prof. Dongarra and his fellow "SC Perennials" are pictured at right.

From left to right: Mike Bernhardt, Anne Marie Kelly, Horst Simon, Ralph McEldowney, Kenichi Miura, Ralph Roskies, John Gustafson, Fred Johnson, Steve Wallach, Steve Finn, Jack Dongarra, Steve Poole, Allen Malony, Vito Bongiorno, Jim Bottum, Lennart Johnsson, John Levesque, Quentin Stout, and Maxine Brown.

Best Paper Award

ICL Research Director Thomas Herault and his coauthors earned a Best Paper Award for their entry in the 20th Workshop on Advances in Parallel and Distributed Computational Models held in Vancouver, BC, Canada in May 2018.

The paper, "Optimal Cooperative Checkpointing for Shared High-Performance Computing Platforms," discussed how, in HPC environments, input/output (I/O) traffic from various sources often contends for limited bandwidth—especially when using checkpoint/restart (CR) to avoid failures. Herault et al. proposed a cooperative scheduling policy that optimizes the overall performance of concurrently executing CR-based applications.

Herault, T., Y. Robert, A. Bouteiller, D. Arnold, K. Ferreira, G. Bosilca, and J. Dongarra, **"Optimal Cooperative Checkpointing for Shared High-Performance Computing Platforms,"** *20th Workshop on Advances in Parallel and Distributed Computational Models (APDCM 2018)*, Best Paper Award, Vancouver, BC, Canada, IEEE, May 2018.

Best Poster Award

In June 2018, Azzam Haidar and his coauthors earned a Best Poster Award at the ISC High Performance 2018 conference in Frankfurt, Germany.

The poster, "Using GPU FP16 Tensor Cores Arithmetic to Accelerate Mixed-Precision Iterative Refinement Solvers and Reduce Energy Consumption," shows the results of using a mixed-precision iterative refinement technique to develop architecture-specific algorithms and highly tuned implementations for GPU Tensor Cores (TCs). What they found is that using FP16-TC arithmetic can provide up to a 4× speedup over regular double-precision arithmetic while retaining the double-precision accuracy and improving energy consumption by up to 5×. In their experiments, Haidar et al. were able to achieve the equivalent of 74 double-precision gigaFLOP/s per Watt on an NVIDIA V100 GPU.

All developments described in the poster will be made available through ICL's Matrix Algebra on GPU and Multi-core Architectures (MAGMA) library.

Haidar, A., S. Tomov, A. Ahmad, M. Zounon, and J. Dongarra, **"Using GPU FP16 Tensor Cores Arithmetic to Accelerate Mixed-Precision Iterative Refinement Solvers and Reduce Energy Consumption,"** *ISC High Performance (ISC18)*, Best Poster Award, Frankfurt, Germany, June 2018.

ICL/RESEARCH

What originally began 30 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. ICL has evolved and expanded our research agenda to accommodate the heterogeneous computing revolution and focus on algorithms and libraries for multi-core 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.

In the fall of 2016, ICL won an array of seven awards from the DOE's Exascale Computing Project (ECP). In doing so, ICL earned a place among an elite set of researchers from DOE laboratories who will create the software infrastructure for the nation's first exascale machines. On the following pages, we provide brief summaries of some of our efforts in these areas.

Distributed Tasking for Exascale

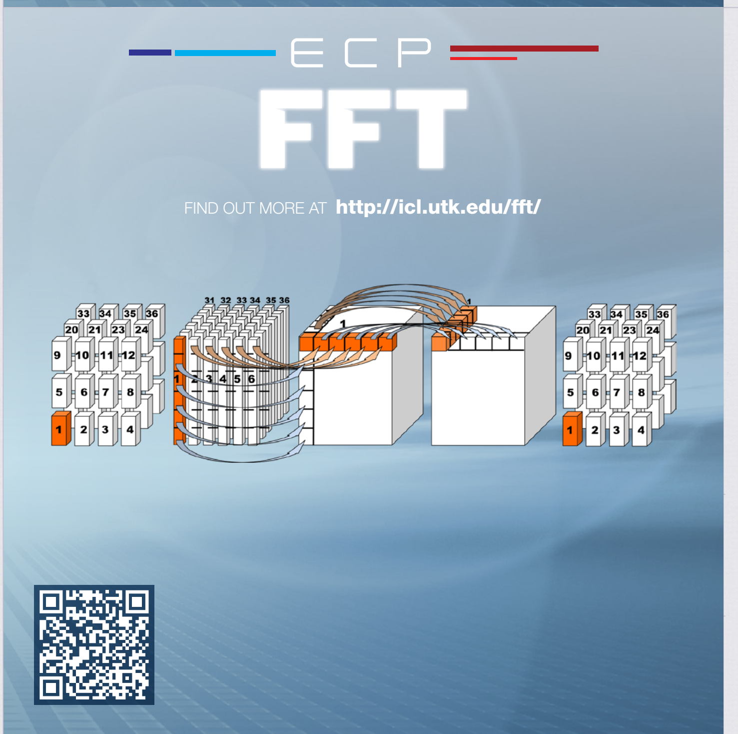
The Distributed Tasking for Exascale (DTE) project will extend the capabilities of ICL's Parallel Runtime and Execution Controller (PaRSEC) project—a generic framework for architecture-aware scheduling and management of microtasks on distributed, many-core, heterogeneous architectures. The PaRSEC environment also provides a runtime component for dynamically executing tasks on heterogeneous distributed systems along with a productivity toolbox and development framework that supports multiple domain-specific languages and extensions and tools for debugging, trace collection, and analysis.

With PaRSEC, applications are expressed as a direct acyclic graph (DAG) of tasks with edges designating data dependencies. This DAG dataflow paradigm attacks both sides of the exascale challenge: managing extreme-scale parallelism and maintaining the performance portability of the code. The DTE award is a vital extension and continuation of this effort and will ensure that PaRSEC meets the critical needs of ECP application communities in terms of scalability, interoperability, and productivity.

Exascale Performance Application Programming Interface

The Exascale Performance Application Programming Interface (Exa-PAPI) award builds on ICL's Performance Application Programming Interface (PAPI) project and extends it with performance counter monitoring capabilities for new and advanced ECP hardware and software technologies. PAPI provides 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.

Exa-PAPI extends this effort 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 like 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—all through one consistent interface.



Fast Fourier Transform

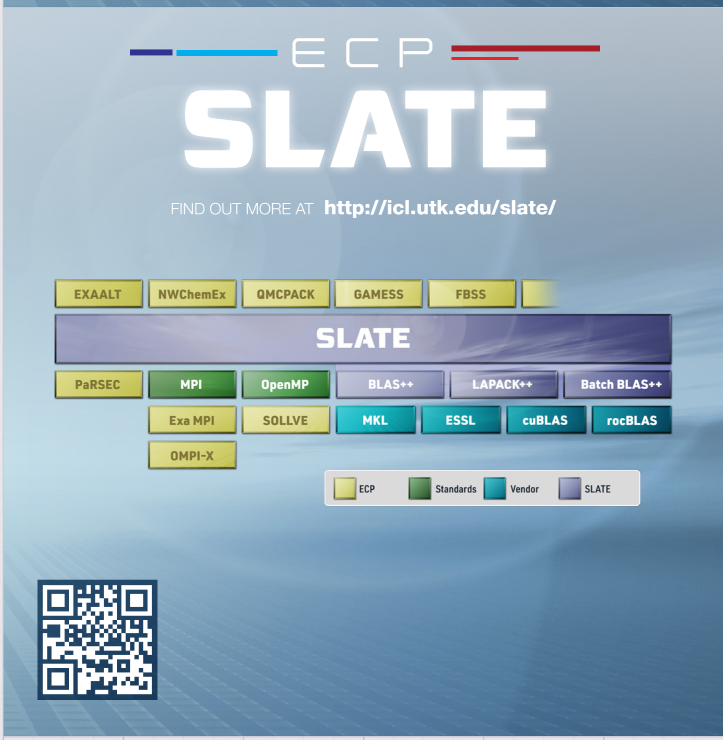
The fast Fourier transform (FFT) is used in many domain applications—including molecular dynamics, spectrum estimation, fast convolution and correlation, signal modulation, and wireless multimedia applications—but current state-of-the-art FFT libraries are not scalable on large heterogeneous machines with many nodes.

The main objective of the ECP FFT project is to design and implement a fast and robust 2-D and 3-D FFT library that targets large-scale heterogeneous systems with multi-core processors and hardware accelerators and to do so as a co-design activity with other ECP application developers. The work involves studying and analyzing current FFT software from vendors and open-source developers in order to understand, design, and develop a 3-D ECP FFT library that could benefit from these existing optimized FFT kernels or will rely on new optimized kernels developed under this framework.

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 extend the DOE's Extreme-scale Algorithms and Solver Resilience (EASIR) project. Many large-scale scientific applications rely heavily on preconditioned iterative solvers for large linear systems. For these solvers to efficiently exploit extreme-scale hardware, both the solver algorithms and the implementations must be redesigned to address challenges like extreme concurrency, complex memory hierarchies, costly data movement, and heterogeneous node architectures.

The PEEKS effort aims to tackle these challenges and advance the capabilities of the ECP software stack by making the new scalable algorithms accessible within the Trilinos software ecosystem. Targeting exascale-enabled Krylov solvers, incomplete factorization routines, and parallel preconditioning techniques will ensure successful delivery of scalable Krylov solvers in robust, production-quality software that can be relied on by ECP applications.



Software for Linear Algebra Targeting Exascale

For decades, ICL has applied algorithmic and technological innovations to the process of pioneering, implementing, and disseminating dense linear algebra software—including the Linear Algebra PACKage (LAPACK) and Scalable Linear Algebra PACKage (ScaLAPACK) libraries. The Software for Linear Algebra Targeting Exascale (SLATE) project will converge and consolidate that software into a dense linear algebra library that will integrate seamlessly into the ECP ecosystem.

For context, ScaLAPACK was first released in 1995. In the past two decades, HPC has witnessed tectonic shifts in the hardware technology, followed by paradigm shifts in the software technology, and a plethora of algorithmic innovation in scientific computing. At the same time, no viable replacement for ScaLAPACK emerged. SLATE is meant to be this replacement, boasting superior performance and scalability in the modern, heterogeneous, distributed-memory environments of HPC.

ECP
CEED
FIND OUT MORE AT
http://ceed.exascaleproject.org/

CEED Co-Design Center

The Lawrence Livermore National Laboratory (LLNL)–led Center for Efficient Exascale Discretizations (CEED) co-design effort will develop next-generation discretization software and algorithms—which deliver a significant performance gain over conventional low-order methods—to enable a wide range of DOE and National Nuclear Security Administration (NNSA) applications to run efficiently on future exascale hardware. CEED is a research partnership involving 30+ computational scientists from two DOE labs and five universities, including UTK.

For UTK's part, ICL will be instrumental in identifying, developing, and optimizing tensor contractions that are essential building blocks for these kinds of DOE/NNSA applications. The ICL team will also play an integral role in co-designing application programming interfaces (APIs) with the LLNL scientists, external partners, and vendors, and will deliver a high-performance tensor contractions package through the Matrix Algebra on GPU and Multi-core Architectures (MAGMA) library.

ECP
OMPI-X
FIND OUT MORE AT
http://www.icl.utk.edu/research/ompi-x/

Open MPI for Exascale

The Open MPI for Exascale (OMPI-X) project focuses on preparing the Message Passing Interface (MPI) standard—and its implementation in Open MPI—for exascale through improvements in scalability, capability, and resilience. Since its inception, the MPI standard has become ubiquitous in high-performance parallel computational science and engineering, and Open MPI is a widely used, high-quality, open-source implementation of the MPI standard. Despite their history and popularity, however, neither Open MPI nor the MPI standard itself is currently ready for the changes in hardware and software that will accompany exascale computing.

To mitigate this concern, OMPI-X will address a broad spectrum of issues in both the standard and the implementation by ensuring 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.

ECP
xSDK4ECP
FIND OUT MORE AT
https://xsdk.info/ecp/

The Extreme-Scale Scientific Software Development Kit for the Exascale Computing Project (xSDK4ECP) is a collaboration between Argonne National Laboratory (ANL), ICL, Lawrence Berkeley National Laboratory (LBNL), LLNL, Sandia National Laboratories (SNL), and the University of California at Berkeley (UCB). The project aims to enable seamless integration and combined use of diverse, independently developed software packages for ECP applications. Currently, this includes a wide range of high-quality software libraries and solver packages that address the strategic needs to fulfill the mission of DOE's Office of Science.

ICL's MAGMA project was integrated into the xSDK 0.3 release, and ICL's Parallel Linear Algebra Software for Multi-core Architectures (PLASMA) package was added to xSDK 0.4. To ensure consistency of naming conventions, runtime behavior, and installation procedure, xSDK informs the project development process by providing requirements and guidelines that are influential throughout the software development phase. Additionally, as part of the inclusion process—and to lighten the burden on system administrators and application developers—each xSDK package provides a Spack installation script that can be invoked independently or through the installation of the xSDK's Spack package.

AsyncIS

FIND OUT MORE AT
http://www.icl.utk.edu/research/asyncis/

Asynchronous Iterative Solvers for Extreme-scale Computing

The Asynchronous Iterative Solvers for Extreme-Scale Computing (AsyncIS) project aims to explore more efficient numerical algorithms by decreasing their overhead. AsyncIS does this by replacing the outer Krylov subspace solver with an asynchronous optimized Schwarz method, thereby removing the global synchronization and bulk synchronous operations typically used in numerical codes.

AsyncIS, a DOE-funded collaboration between Georgia Tech, UTK, Temple University, and SNL, also focuses on the development and optimization of asynchronous preconditioners (i.e., preconditioners that are generated and/or applied in an asynchronous fashion). The novel preconditioning algorithms that provide fine-grained parallelism enable preconditioned Krylov solvers to run efficiently on large-scale distributed systems and many-core accelerators like GPUs.

BATCHED BLAS

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

Batched Basic Linear Algebra Subprograms

The Batched Basic Linear Algebra Subprograms (BBLAS) effort, an international collaboration between INRIA, Rutherford Appleton Laboratory, Umeå University, the University of Manchester, and UTK, will create an API for numerical computing routines that process batches of either uniformly sized or varying-size matrices or vectors. This will go beyond the original Basic Linear Algebra Subprogram (BLAS) standard by specifying a programming interface for modern scientific applications, which produce large numbers of small matrices at once.

Individually, the small sizes of the inputs obviate the potential benefits of using BLAS but are a perfect fit for BBLAS. The BBLAS project will also serve as a working forum for gathering ideas and working out a plan for establishing the consensus for the next official standard that will serve the scientific community and be supported by hardware vendors.

BONSAI

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

BEAST OpeN Software Autotuning Infrastructure

The goal of the BEAST OpeN Software Autotuning Infrastructure (BONSAI) project is to develop a software infrastructure for using scalable, parallel hybrid systems to carry out large, concurrent autotuning sweeps in order to dramatically accelerate the optimization process of computational kernels for GPU accelerators and many-core coprocessors.

Recent developments include a distributed benchmarking engine—currently in active development—capable of scaling to tens of thousands of nodes. This will enable benchmarking millions of kernel configurations for different problem sizes and many input datasets while collecting hundreds of performance metrics—including time, energy consumption, cache misses, and memory bandwidth.

CAARES

FIND OUT MORE AT
http://www.icl.utk.edu/research/caares/

Cross-layer Application-Aware Resilience at Extreme Scale

The Cross-layer Application-Aware Resilience at Extreme Scale (CAARES) project, a collaborative effort between ICL, Rutgers University, and Stony Brook, aims to provide a theoretical foundation for multi-level fault management techniques and provide a clear understanding of existing obstacles that could obstruct generic and efficient approaches for fault management at scale. This effort is vital for large-scale science, because, as extreme-scale computational power enables new and important discoveries across all science domains, the current understanding of fault rates is casting a grim shadow and revealing a future where failures are not exceptions but the norm.

By studying combinations of fault tolerance techniques instead of studying them in isolation from each other, CAARES seizes the opportunity to identify moldable techniques at the frontier of known approaches and highlight a composition of methodologies that inherit their individual benefits but do not exhibit their drawbacks, leading to the development of resilience techniques able to bridge the gap between fault tolerance ergonomics and efficiency.

CORES

FIND OUT MORE AT
http://www.icl.utk.edu/research/cores/

Convex Optimization for Realtime Embedded Systems

The Convex Optimization for Real-time Embedded Systems (CORES) project aims to develop highly efficient, real-time convex optimization algorithms and toolsets for solving important engineering problems on hierarchical and heterogeneous embedded system architectures. Though recent advances in optimization solvers have enabled the solution of optimization problems on low-cost embedded systems, the size of the problems that can be solved in real time is still limited.

The CORES project, a collaboration between ICL and Michigan Technological University, works to address this limitation. The ICL team's main responsibility is the design and development of higher-performance, structure-aware linear solvers that would enable us to solve, in real time, the convex optimization problems that have significantly higher performance—and are orders of magnitude greater in size—compared to current state-of-the-art solvers.

DARE

FIND OUT MORE AT
http://www.icl.utk.edu/research/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 for the PLASMA, MAGMA, and SLATE libraries to ensure maximum performance on modern, GPU-accelerated hybrid systems. DARE's software architecture combines three components: hardware analysis, kernel modeling, and workload simulation.

The hardware analysis block builds a detailed model of the hardware, its computational resources, and its memory system; the kernel modeling block builds accurate performance models for the computational kernels involved in the workload; and the workload simulation block rapidly simulates a large number of runs to find the best execution conditions using the information provided by the other two blocks. The objective is to develop performance modeling techniques and accurate run time prediction capabilities—under different execution conditions—to tune compute-intensive numerical codes for modern GPU-accelerated architectures.

Distributed Parallel Linear Algebra Software for Multi-core Architectures

The Distributed Parallel Linear Algebra Software for Multi-core 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 multi-core processors and, if available, accelerators like GPUs or Intel Xeon Phi coprocessors. DPLASMA achieves this objective by deploying PLASMA algorithms on distributed-memory systems using 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 operate transparently on local data or introduce implicit communications to resolve dependencies, thereby removing the burden of initial data reshuffle and providing the user with a novel approach to address load balance.

Evolve, a collaborative effort between ICL and the University of Houston, expands the capabilities of Open MPI to support the National Science Foundation's (NSF's) critical software-infrastructure missions. Core challenges include: extending the software to scale to 10,000–100,000 processes; ensuring support for accelerators; enabling highly asynchronous execution of communication and I/O operations; and ensuring resilience. Part of the effort involves careful consideration of modifications to the MPI specification to account for the emerging needs of application developers on future extreme-scale systems.

So far, Evolve efforts have involved exploratory research for improving different performance aspects of the Open MPI library. Notably, this has led to an efficiency improvement in multi-threaded programs using MPI in combination with other thread-based programming models (e.g., OpenMP). A novel collective communication framework with event-based programming and data dependencies was investigated, and it demonstrated a clear advantage in terms of aggregate bandwidth in heterogeneous (shared memory + network) systems. Support for MPI resilience following the User-Level Failure Mitigation (ULFM) fault-tolerance proposal was released based on the latest Open MPI version and will soon be fully integrated into Open MPI.

High Performance Conjugate Gradients

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 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.

The 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 positioned HPCG to even better represent modern partial differential equation (PDE) solvers and made it easier to run 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. The current HPCG performance list was released at SC18 and now features over 140 supercomputing sites. The HPCG score has also been tracked by the TOP500 list since June 2017.

HPL

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

High Performance LINPACK

The High Performance LINPACK (HPL) benchmark solves a dense linear system in double precision (64-bit) arithmetic on distributed-memory computers. HPL is written in a portable ANSI C and requires an MPI implementation and either BLAS or the Vector Signal and Image Processing Library (VSIPL). HPL is often one of the first programs to run on large HPC installations, producing a result that can be submitted to the TOP500 list of the world's fastest supercomputers.

The major focus of HPL 2.3, released in 2018, was to improve the accuracy of reported benchmark results and ensure easier configuration and building on modern HPC platforms. HPL now features more detailed reporting of the solution's scaled residual and of the achieved performance number. Another addition is a software configuration tool based on Autotools and the removal of deprecated MPI functions. The LINPACK app for iOS achieved over 8 gigaFLOP/s on the iPhone X. For the November 2018 TOP500 list, an optimized version of the HPL code achieved nearly 150 petaFLOP/s.

LAPACK
ScaLAPACK

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

The Linear Algebra PACKage

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 multi-core BLAS library. ScaLAPACK is parallel, distributed, and relies on the BLAS, LAPACK, MPI, and BLACS libraries.

LAPACK 3.8.0, released in November 2017, includes level-3 BLAS communication-avoiding, symmetric-indefinite factorizations with Aasen's triangular tridiagonalization using the two-stage algorithm. 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 multiple relatively robust representations (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.

MAGMA

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

Matrix Algebra on GPU and Multi-core Architectures

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

MAGMA 2.5 features LAPACK-compliant routines for multi-core CPUs enhanced with NVIDIA GPUs (including the Volta V100). MAGMA now 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. A MagmaDNN package was launched to provide high-performance data analytics, including functionalities for machine learning applications that use MAGMA as their computational back end. The MAGMA Sparse and MAGMA Batched packages have been included since MAGMA 1.6.

MATEDOR

FIND OUT MORE AT
http://www.icl.utk.edu/research/matedor/

MAtrix, TEnsor, and Deep-learning Optimized Routines

The MAtrix, TEnsor, and Deep-learning Optimized Routines (MATEDOR) project will perform the research required to define a standard interface for batched operations and provide a performance-portable software library that demonstrates batching routines for a significant number of kernels. This research is critical, given that the performance opportunities inherent in solving many small batched matrices often yield more than a 10× speedup over the current classical approaches.

Working closely with affected application communities, along with ICL's Batched BLAS initiative, MATEDOR will define modular, optimizable, and language-agnostic interfaces that can work seamlessly with a compiler. This modularity will provide application, compiler, and runtime system developers with the option to use a single call to a routine from the new batch operation standard and would allow the entire linear algebra community to collectively attack a wide range of small matrix or tensor problems.

OPEN MPI

FIND OUT MORE AT
https://www.open-mpi.org/

Open-Source Message Passing Interface

The Open MPI Project is an open-source Message Passing Interface (MPI) implementation 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) in order to build the best MPI library available.

A completely new MPI 3.2–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 many-core environments, and architecture-aware capabilities—such as adaptive shared memory behaviors and dynamic collective selection—making it ready for next-generation exascale challenges.

PAPI

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

Performance Application Programming Interface

The Performance Application Programming Interface (PAPI) 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. Industry liaisons with AMD, Bull, Cray, Intel, IBM, NVIDIA, and others ensure seamless integration of PAPI with new architectures at or near their release. As the PAPI component architecture becomes more populated, performance tools that interface with PAPI automatically inherit the ability to measure these new data sources.

In 2018, ICL, together with the University of Maine, worked on PAPI-EX to build support for performance counters available in the latest generations of CPUs and GPUs, develop support for system-wide hardware performance counter monitoring, and strengthen the sampling interface in PAPI. PAPI-EX will also incorporate a counter inspection toolkit (CIT) designed to improve the understanding of low-level hardware events. Since 2016, the PAPI effort has been bolstered through the ECP Exa-PAPI project.

Parallel Runtime Scheduling and Execution Controller

The Parallel Runtime Scheduling and Execution Controller (PaRSEC) is a generic framework for architecture-aware scheduling and management of microtasks on distributed many-core heterogeneous architectures. Applications considered are expressed as a DAG of tasks with edges designating the 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 computational threads to the cores, overlaps communications and computations, and uses a dynamic, fully distributed scheduler. PaRSEC includes a set of tools to generate the DAGs and integrate them into legacy codes, a runtime library to schedule the microtasks 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.

Parallel Linear Algebra Software for Multi-core Architectures

The Parallel Linear Algebra Software for Multi-core Architectures (PLASMA) package implements a set of fundamental linear algebra routines using the Open Multi-Processing (OpenMP) standard. PLASMA includes, among others, routines for solving linear systems of equations, linear least square problems, parallel BLAS, and parallel matrix norms.

Over the last decade, PLASMA—which has been deployed on a variety of systems using Intel processors (including Xeon Phi coprocessors), IBM POWER processors, and ARM processors—has served as a tremendous research vehicle for the design of new dense linear algebra algorithms and has paved the way for new developments, including the new ECP SLATE project, which will ultimately deliver these capabilities at exascale.

PAPI Unifying Layer for Software-defined Events

The PAPI Unifying Layer for Software-defined Events (PULSE) project focuses on enabling cross-layer and integrated monitoring of whole application performance by extending PAPI with the capability to expose performance metrics from key software components found in the HPC software stack. Up to this point, the abstraction and standardization layer provided by PAPI has been limited to profiling information generated by hardware only. Information about the behavior of the underlying software stack had to be acquired either through low-level binary instrumentation or through custom APIs.

To overcome this shortfall, PULSE is extending the abstraction and unification layer that PAPI has provided for hardware events to also encompass software events. On one end, PULSE offers a standard, well-defined and well-documented API that high-level profiling software can utilize to acquire performance information about the libraries used by an application and present it to the application developers. On the other end, it provides standard APIs that library and runtime writers can utilize to communicate information about the behavior of their software to higher software layers.

SMURFS

FIND OUT MORE AT
http://www.icl.utk.edu/research/smurfs/

Simulation and Modeling for Understanding Resilience and Faults at Scale

The Simulation and Modeling for Understanding Resilience and Faults at Scale (SMURFS) project seeks to acquire the predictive understanding of the complex interactions of a given application, a given real or hypothetical hardware and software environment, and a given fault-tolerance strategy at extreme scale.

SMURFS is characterized by two facets: (1) medium- and fine-grained predictive capabilities and (2) coarse-grained fault tolerance strategy selection. Accordingly, 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. Also, based on a comprehensive set of studies using several application benchmarks, proxies, full applications, and several different fault tolerance strategies, ICL will gather valuable insights about application behavior at scale.

SparseKaffe

FIND OUT MORE AT
http://www.icl.utk.edu/research/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 (SparseKaffe) project creates fast and efficient sparse direct methods for platforms with multi-core 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 energy requirements.

The SparseKaffe project is a collaboration between UTK, the University of Florida, and Texas A&M University. ICL's work on the project concentrates on kernel designs and performance tuning, as well as on dynamic runtime scheduling using a dataflow model. This work 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 DARE project.

TESSE

FIND OUT MORE AT
http://www.iacs.stonybrook.edu/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) 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 massively parallel, hybrid, many-core systems of today and tomorrow.

The TESSE team is composed of researchers from Stony Brook, Virginia Tech, and UTK, who have designed a system that uses DAG-based data flow as the basis of the software. This capability, with the extensions being explored by the TESSE team, 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.

Ranking the 500 fastest computers in the world

With three decades of tracking supercomputing progress, the TOP500 list continues to provide a reliable historical record of supercomputers around the world. The list clearly lays out critical HPC metrics across all 500 machines and draws a rich picture of the state of the art in terms of performance, energy consumption, and power efficiency. The TOP500 now features an HPCG ranking, which measures a machine's performance using irregular accesses to memory and fine-grain recursive computations—factors that dominate real-world, large-scale scientific workloads.

In November 2018, the 52nd TOP500 list was unveiled at SC18 in Dallas, TX. The United States remained on top with the Oak Ridge Leadership Computing Facility's Summit machine. Summit updated its HPL benchmark result for the November 2018 list and achieved 143.5 petaFLOP/s (vs. 122.3 petaFLOP/s in June 2018).

User Level Failure Mitigation

User Level Failure Mitigation (ULFM) is a set of new interfaces for 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 where the 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. A wide range of application types and middlewares are already building on top of ULFM to deliver scalable and user-friendly fault tolerance. Notable recent additions include the CoArray Fortran language and SAP databases. ULFM software is available in recent versions of both MPICH and Open MPI.

ICL/PUBLICATIONS

Evidence of our research and our contributions to the HPC community might be best exemplified by the numerous publications we produce every year. Here is a listing of our most recent papers, including journal articles, book chapters, and conference proceedings. Many of these are available for download from our website.

FIND OUT MORE AT **https://www.icl.utk.edu/publications/**

Abdelfattah, A., A. Haidar, S. Tomov, and J. Dongarra, **"Analysis and Design Techniques towards High-Performance and Energy-Efficient Dense Linear Solvers on GPUs,"** *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, issue 12, pp. 2700–2712, December 2018.

Abdelfattah, A., A. Haidar, S. Tomov, and J. Dongarra, **"Batched One-Sided Factorizations of Tiny Matrices Using GPUs: Challenges and Countermeasures,"** Journal of Computational Science, vol. 26, pp. 226–236, May 2018.

Abdelfattah, A., A. Haidar, S. Tomov, and J. Dongarra, **"Optimizing GPU Kernels for Irregular Batch Workloads: A Case Study for Cholesky Factorization,"** *IEEE High Performance Extreme Computing Conference (HPEC'18)*, Waltham, MA, IEEE, September 2018.

Abdelfattah, A., M. Gates, J. Kurzak, P. Luszczek, and J. Dongarra, **"Implementation of the C++ API for Batch BLAS,"** *SLATE Working Notes*, no. 7, ICL-UT-18-04: Innovative Computing Laboratory, University of Tennessee, June 2018.

Anzt, H. and J. Dongarra, **"A Jaccard Weights Kernel Leveraging Independent Thread Scheduling on GPUs,"** *SBAC-PAD*, Lyon, France, September 2018.

Anzt, H., E. Chow, and J. Dongarra, **"ParILUT – A New Parallel Threshold ILU,"** *SIAM Journal on Scientific Computing*, vol. 40, issue 4: SIAM, pp. C503–C519, July 2018.

Anzt, H., I. Yamazaki, M. Hoemmen, E. Boman, and J. Dongarra, **"Solver Interface & Performance on Cori,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-18-05: University of Tennessee, June 2018.

Anzt, H., J. Dongarra, G. Flegar, and E. S. Quintana-Ortí, **"Variable-Size Batched Gauss–Jordan Elimination for Block-Jacobi Preconditioning on Graphics Processors,"** *Parallel Computing*, January 2018.

Anzt, H., J. Dongarra, G. Flegar, and T. Gruetzmacher, **"Variable-Size Batched Condition Number Calculation on GPUs,"** *SBAC-PAD*, Lyon, France, September 2018.

Anzt, H., J. Dongarra, G. Flegar, N. J. Higham, and E. S. Quintana-Ortí, **"Adaptive Precision in Block-Jacobi Preconditioning for Iterative Sparse Linear System Solvers,"** *Concurrency Computation: Practice and Experience*, March 2018.

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,"** *The International Journal of High Performance Computing Applications*, vol. 32, no. 2, pp. 220–230, March 2018.

Anzt, H., T. Gruetzmacher, E. Quintana-Orti, and F. Scheidegger, **"High-Performance GPU Implementation of PageRank with Reduced Precision based on Mantissa Segmentation,"** *8th Workshop on Irregular Applications: Architectures and Algorithms*, 2018.

Anzt, H., T. Huckle, J. Bräckle, and J. Dongarra, **"Incomplete Sparse Approximate Inverses for Parallel Preconditioning,"** *Parallel Computing*, vol. 71, pp. 1–22, January 2018.

Asch, M., T. Moore, R. M. Badia, M. Beck, P. Beckman, T. Bidot, F. Bodin, F. Cappello, A. Choudhary, B. R. de Supinski, et al., **"Big Data and Extreme-Scale Computing: Pathways to Convergence - Toward a Shaping Strategy for a Future Software and Data Ecosystem for Scientific Inquiry,"** *The International Journal of High Performance Computing Applications*, vol. 32, issue 4, pp. 435–479, July 2018.

Aupy, G., A. Benoit, B. Goglin, L. Pottier, and Y. Robert, **"Co-Scheduling HPC Workloads on Cache-Partitioned CMP Platforms,"** *Cluster 2018*, Belfast, UK, IEEE Computer Society Press, September, 2018.

Aupy, G., A. Benoit, S. Dai, L. Pottier, P. Raghavan, Y. Robert, and M. Shantharam, **"Co-Scheduling Amdhal Applications on Cache-Partitioned Systems,"** *International Journal of High Performance Computing Applications*, vol. 32, issue 1, pp. 123–138, January 2018.

Aupy, G., and Y. Robert, **"Scheduling for Fault-Tolerance: An Introduction,"** *Topics in Parallel and Distributed Computing: Springer International Publishing*, pp. 143–170, 2018.

Balaprakash, P., J. Dongarra, T. Gamblin, M. Hall, J. Hollingsworth, B. Norris, and R. Vuduc, **"Autotuning in High-Performance Computing Applications,"** *Proceedings of the IEEE*, vol. 106, issue 11, pp. 2068–2083, November 2018.

Benoit, A., A. Cavelan, F. Cappello, P. Raghavan, Y. Robert, and H. Sun, **"Coping with Silent and Fail-Stop Errors at Scale by Combining Replication and Checkpointing,"** *Journal of Parallel and Distributed Computing*, vol. 122, pp. 209–225, December 2018.

Benoit, A., A. Cavelan, Y. Robert, and H. Sun, **"Multi-Level Checkpointing and Silent Error Detection for Linear Workflows,"** *Journal of Computational Science*, vol. 28, pp. 398–415, September 2018.

Benoit, A., S. Perarnau, L. Pottier, and Y. Robert, **"A Performance Model to Execute Workflows on High-Bandwidth Memory Architectures,"** *The 47th International Conference on Parallel Processing (ICPP 2018)*, Eugene, OR, IEEE Computer Society Press, August 2018.

Bernholdt, D. E., S. Boehm, G. Bosilca, M. Grentla Venkata, R. E. Grant, T. Naughton, H. P. Pritchard, M. Schulz, and G. R. Vallee, **"A Survey of MPI Usage in the US Exascale Computing Project,"** *Concurrency Computation: Practice and Experience*, September 2018.

Bosilca, G., A. Bouteiller, A. Guermouche, T. Herault, Y. Robert, P. Sens, and J. Dongarra, **"A Failure Detector for HPC Platforms,"** *The International Journal of High Performance Computing Applications*, vol. 32, issue 1, pp. 139–158, January 2018.

Bouteiller, A., G. Bosilca, T. Herault, and J. Dongarra, **"Data Movement Interfaces to Support Dataflow Runtimes,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-18-03: University of Tennessee, May 2018.

Caniou, Y., E. Caron, A. Kong Win Chang, and Y. Robert, **"Budget-Aware Scheduling Algorithms for Scientific Workflows with Stochastic Task Weights on Heterogeneous IaaS Cloud Platforms,"** *2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, Vancouver, BC, Canada, IEEE, May 2018.

Casanova, H., J. Herrmann, and Y. Robert, **“Computing the Expected Makespan of Task Graphs in the Presence of Silent Errors,”** *Parallel Computing*, vol. 75, July 2018.

Castain, R., J. Hursey, A. Bouteiller, and D. Solt, **“PMIx: Process Management for Exascale Environments,”** *Parallel Computing*, vol. 79, pp. 9–29, January 2018.

Chow, E., H. Anzt, J. Scott, and J. Dongarra, **“Using Jacobi Iterations and Blocking for Solving Sparse Triangular Systems in Incomplete Factorization Preconditioning,”** *Journal of Parallel and Distributed Computing*, vol. 119, pp. 219–230, November 2018.

Dong, T., A. Haidar, S. Tomov, and J. Dongarra, **“Accelerating the SVD Bi-Diagonalization of a Batch of Small Matrices using GPUs,”** *Journal of Computational Science*, vol. 26, pp. 237–245, May 2018.

Dongarra, J., M. Gates, A. Haidar, J. Kurzak, P. Luszczek, S. Tomov, and I. Yamazaki, **“The Singular Value Decomposition: Anatomy of Optimizing an Algorithm for Extreme Scale,”** *SIAM Review*, vol. 60, issue 4, pp. 808–865, November 2018.

Gates, M., A. Charara, J. Kurzak, A. YarKhan, I. Yamazaki, and J. Dongarra, “**Least Squares Performance Report**,” *SLATE Working Notes*, no. 9, ICL-UT-18-10: Innovative Computing Laboratory, University of Tennessee, December 2018.

Gates, M., S. Tomov, and J. Dongarra, **“Accelerating the SVD Two Stage Bidiagonal Reduction and Divide and Conquer Using GPUs,”** *Parallel Computing*, vol. 74, pp. 3–18, May 2018.

Ghysels, P., S. Li, A. YarKhan, and J. Dongarra, “**Initial Integration and Evaluation of SLATE and STRUMPACK**,” *Innovative Computing Laboratory Technical Report*, no. ICL-UT-18-11: University of Tennessee, December 2018.

Haidar, A., A. Abdelfattah, M. Zounon, P. Wu, S. Pranesh, S. Tomov, and J. Dongarra, **“The Design of Fast and Energy-Efficient Linear Solvers: On the Potential of Half-Precision Arithmetic and Iterative Refinement Techniques,”** *International Conference on Computational Science (ICCS 2018)*, vol. 10860, Wuxi, China, Springer, pp. 586–600, June 2018.

Haidar, A., A. Abdelfattah, M. Zounon, S. Tomov, and J. Dongarra, **“A Guide for Achieving High Performance with Very Small Matrices on GPUs: A Case Study of Batched LU and Cholesky Factorizations,”** *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, issue 5, pp. 973–984, May 2018.

Haidar, A., H. Jagode, P. Vaccaro, A. YarKhan, S. Tomov, and J. Dongarra, **“Investigating Power Capping toward Energy-Efficient Scientific Applications,”** *Concurrency Computation: Practice and Experience*, vol. 2018, issue e4485, pp. 1–14, April 2018.

Haidar, A., S. Tomov, A. Abdelfattah, M. Zounon, and J. Dongarra, **“Using GPU FP16 Tensor Cores Arithmetic to Accelerate Mixed-Precision Iterative Refinement Solvers and Reduce Energy Consumption,”** *ISC High Performance (ISC’18)*, Best Poster, Frankfurt, Germany, June 2018.

Haidar, A., S. Tomov, J. Dongarra, and N. J. Higham, **“Harnessing GPU Tensor Cores for Fast FP16 Arithmetic to Speed up Mixed-Precision Iterative Refinement Solvers,”** *The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC18)*, Dallas, TX, IEEE, November 2018.

Han, L., L-C. Canon, H. Casanova, Y. Robert, and F. Vivien, **“Checkpointing Workflows for Fail-Stop Errors,”** *IEEE Transactions on Computers*, vol. 67, issue 8, pp. 1105–1120, August 2018.

Han, L., V. Le Fèvre, L-C. Canon, Y. Robert, and F. Vivien, **“A Generic Approach to Scheduling and Checkpointing Workflows,”** *The 47th International Conference on Parallel Processing (ICPP 2018)*, Eugene, OR, IEEE Computer Society Press, August 2018.

Herault, T., Y. Robert, A. Bouteiller, D. Arnold, K. Ferreira, G. Bosilca, and J. Dongarra, **“Optimal Cooperative Checkpointing for Shared High-Performance Computing Platforms,”** *2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, Best Paper Award, Vancouver, BC, Canada, IEEE, May 2018.

Jagode, H., A. Danalis, and J. Dongarra, **“Accelerating NWChem Coupled Cluster through Dataflow-Based Execution,”** *The International Journal of High Performance Computing Applications*, vol. 32, issue 4, pp. 540–551, July 2018.

Jagode, H., A. Danalis, R. Hoque, M. Faverge, and J. Dongarra, **“Evaluation of Dataflow Programming Models for Electronic Structure Theory,”** *Concurrency and Computation: Practice and Experience: Special Issue on Parallel and Distributed Algorithms*, vol. 2018, issue e4490, pp. 1–20, May 2018.

Kurzak, J., M. Gates, A. YarKhan, I. Yamazaki, P. Luszczek, J. Finney, and J. Dongarra, **“Parallel Norms Performance Report,”** *SLATE Working Notes*, no. 6, ICL-UT-18-06: Innovative Computing Laboratory, University of Tennessee, June 2018.

Kurzak, J., M. Gates, A. YarKhan, I. Yamazaki, P. Wu, P. Luszczek, J. Finney, and J. Dongarra, **“Parallel BLAS Performance Report,”** *SLATE Working Notes*, no. 5, ICL-UT-18-01: University of Tennessee, April 2018.

Kurzak, J., M. Gates, I. Yamazaki, A. Charara, A. YarKhan, J. Finney, G. Ragghianti, P. Luszczek, and J. Dongarra, **“Linear Systems Performance Report,”** *SLATE Working Notes*, no. 8, ICL-UT-18-08: Innovative Computing Laboratory, University of Tennessee, September 2018.

Le Fèvre, V., G. Bosilca, A. Bouteiller, T. Herault, A. Hori, Y. Robert, and J. Dongarra, **“Do moldable applications perform better on failure-prone HPC platforms?”** *11th Workshop on Resiliency in High Performance Computing in Clusters, Clouds, and Grids*, Turin, Italy, Springer Verlag, August 2018.

Luo, X., W. Wu, G. Bosilca, T. Patinyasakdikul, L. Wang, and J. Dongarra, **“ADAPT: An Event-Based Adaptive Collective Communication Framework,”** *The 27th International Symposium on High-Performance Parallel and Distributed Computing (HPDC ‘18)*, Tempe, Arizona, ACM Press, June 2018.

Marques, O., J. Demmel, and P. B. Vasconcelos, **“Bidiagonal SVD Computation via an Associated Tridiagonal Eigenproblem,”** *LAPACK Working Note*, no. LAWN 295, ICL-UT-18-02: University of Tennessee, April 2018.

Masliah, I., A. Abdelfattah, A. Haidar, S. Tomov, M. Baboulin, J. Falcou, and J. Dongarra, **“Algorithms and Optimization Techniques for High-Performance Matrix-Matrix Multiplications of Very Small Matrices,”** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-18-09: Innovative Computing Laboratory, University of Tennessee, September 2018.

Sun, J., J. Fu, J. Drake, Q. Zhu, A. Haidar, M. Gates, S. Tomov, and J. Dongarra, **“Computational Benefit of GPU Optimization for Atmospheric Chemistry Modeling,”** *Journal of Advances in Modeling Earth Systems*, vol. 10, issue 8, pp. 1952–1969, August 2018.

Tomov, S., A. Haidar, D. Schultz, and J. Dongarra, **“Evaluation and Design of FFT for Distributed Accelerated Systems,”** *ECP WBS 2.3.3.09 Milestone Report*, no. FFT-ECP ST-MS-10-1216: Innovative Computing Laboratory, University of Tennessee, October 2018.

Yamazaki, I., A. Abdelfattah, A. Ida, S. Ohshima, S. Tomov, R. Yokota, and J. Dongarra, **“Analyzing Performance of BiCGStab with Hierarchical Matrix on GPU Clusters,”** *IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Vancouver, BC, Canada, IEEE, May 2018.

Yamazaki, I., J. Kurzak, P. Wu, M. Zounon, and J. Dongarra, **“Symmetric Indefinite Linear Solver using OpenMP Task on Multicore Architectures,”** *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, issue 8, pp. 1879–1892, August 2018.

YarKhan, A., G. Ragghianti, J. Dongarra, M. Cawkwell, D. Perez, and A. Voter, **“Initial Integration and Evaluation of SLATE Parallel BLAS in LATTE,”** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-18-07: Innovative Computing Laboratory, University of Tennessee, June 2018.

Zaitsev, D., S. Tomov, and J. Dongarra, **“Solving Linear Diophantine Systems on Parallel Architectures,”** *IEEE Transactions on Parallel and Distributed Systems*, October 2018.

ICL/EVENTS

January 23-25 / Gaithersburg, MD
The Extreme Heterogeneity Workshop

February 5-9 / Knoxville, TN
ECP 2nd Annual Meeting

February 26-March 2 / Portland, OR
MPI Forum

March 5-10 / Tokyo, Japan
SIAM Conference on Parallel Processing for Scientific Computing (PP18)

March 20-23 / Dallas, TX
Open MPI Developer Meeting

March 25-26 / Chicago, IL
BDEC Workshop

Macrh 26-29 / San Jose, CA
GPU Technology Conference 2018

April 11 / Bloomington, IN
High-Performance Systems and Analytics for Big Data

April 15-18 / Baltimore, MD
2018 Spring Simulation Multi-Conference

April 16-19 / Barcelona, Spain
8th JLESC Workshop

April 30-May 1 / Washington, DC
2018 NSF SI2 PI Meeting

May 21-25 / Vancouver, BC, Canada
IPDPS 2018

May 24 / Washington, DC
PETTT Workshop: Performance Portability Libraries for DoD Applications

June 4-8 / Boulder, CO
GPU Hackathon Boulder

June 11-15 / Tempe, AZ
The ACM International Symposium on High-Performance Parallel and Distributed Computing (HPDC)

June 18–20 / Berkeley, CA
13th Scheduling for Large Scale Systems Workshop

April 16-19 / Frankfurt, Germany
ISC High Performance 2018

June 27-29 / Zürich, Switzerland
10th International Workshop on Parallel Matrix Algorithms and Applications (PMAA18)

July 2-4 / Basel, Switzerland
2018 Platform for Advanced Scientific Computing Conference (PASC18)

July 9-13 / Portland, OR
2018 SIAM Annual Meeting

July 23-27 / Newfoundland, Canada
25th International Conference on Domain Decomposition Methods (DD25)

August 8-10 / Boulder, CO
CEED 2nd Annual Meeting

August 15-17 / Richmond, VA
MolSSI Workshop and ELSI Conference

August 27-31 / Turin, Italy
24th International European Conference on Parallel and Distributed Computing (Euro-Par 2018)

Every year, members of our research staff attend national and international conferences, workshops, and seminars. These meetings provide opportunities to present our research, share our knowledge, and exchange ideas with leading computational science researchers from around the world. Participating in the intellectual life of the scientific community in this way is an essential part of the research process.

August 28-30 / Gatlinburg, TN

Smoky Mountains Computational Sciences and Engineering Conference (SMC 2018)

September 4-7 / Lyon, France

Workshop on Clusters, Clouds, and Data for Scientific Computing (CCDSC 2018)

April 16-19 / Bellport, NY

GPU Hackathon Brookhaven

September 23-26 / Barcelona, Spain

Euro MPI 2018

September 25-27 / Waltham, MA

2018 IEEE High Performance Extreme Computing Conference (HPEC '18)

October 16-18 / San Jose, CA

Open MPI Developer Meeting

October 25-26 / Paris, France

RTC4AO5: Real Time Control for Adaptive Optics

October 16-18 / San Jose, CA

The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC18)

November 28-30 / Bloomington, IN

BDEC2 Indiana

December 3-6 / San Jose, CA

MPI Forum

The International Conference for High Performance Computing Networking, Storage, and Analysis (SC), now in its 30th year, is a staple of ICL's November itinerary. SC is vital to the growth and evolution of HPC 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 high-performance and scientific computing.

SC18 was held in Dallas, Texas, on November 11–16. This year, four computational science research centers from the University of Tennessee—the Bredesen Center, the Global Computing Laboratory, the ICL, and the SimCenter—represented the university by anchoring a newly minted University of Tennessee booth. As usual, ICL had a significant presence at SC, with faculty, research staff, and students giving talks, presenting papers, and leading "Birds of a Feather" sessions.

ICL/PARTNERSHIPS

ICL fosters relationships with many academic institutions and research centers and has proactively built enduring partnerships with HPC vendors and industry leaders in the United States 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

Argonne National Laboratory	Emory University	Georgia Tech	Indiana University	Iowa State University
JICS Joint Institute for Computational Sciences	Berkeley Lab	Lawrence Livermore National Laboratory	Los Alamos National Laboratory	Michigan Technological University
NCSA	NICS	Northwestern University	Oak Ridge National Laboratory	Pacific Northwest National Laboratory
Rice	Rutgers, The State University of New Jersey	SDSC San Diego Supercomputer Center	Sandia National Laboratories	Stony Brook University
Texas A&M University	Berkeley, University of California	UC Riverside	The University of Chicago	University of Colorado Denver
University of Florida	University of Houston	Illinois, University of Illinois at Urbana-Champaign	The University of Maine	Manchester 1824, The University of Manchester
The University of New Mexico	USC University of Southern California	The University of Utah	University of Washington	Virginia Tech

Industry

AMD	Atos	Cray, The Supercomputer Company	Engility	IBM
IERUS Technologies	intel	MathWorks	nag	NVIDIA

Global Computing Laboratory

In June 2018, Prof. Michela Taufer joined UTK's Department of Electrical Engineering and Computer Science. In doing so, Prof. Taufer also relocated her Global Computing Laboratory to the Min H. Kao Electrical Engineering and Computer Science Building. Members of the Global Computing Laboratory, including Prof. Taufer's students and postdocs, engage in the design and testing of efficient computational algorithms and adaptive scheduling policies for scientific computing on GPUs, cloud computing, and volunteer computing. Interdisciplinary in its mission, the Global Computing Laboratory focuses on various aspects of HPC and scientific computing—including computational chemistry and chemical engineering, pharmaceutical sciences, seismology, and mathematics.

International Collaborators

Barcelona Supercomputing Center Barcelona, Spain	**Central Institute for Applied Mathematics** Jülich, Germany	**Doshisha University** Kyoto, Japan	**École Normale Supérieure de Lyon** Lyon, France	**École Polytechnique Federale de Lausanne** Lausanne, Switzerland
ETH Zurich Zurich, Switzerland	**European Centre for Research and Advanced Training in Scientific Computing** Toulouse, France	**European Exascale Software Initiative** European Union	**Forschungszentrum Jülich** Jülich, Germany	**High Performance Computing Center Stuttgart** Stuttgart, Germany
Hokkaido University Sapporo, Japan	**INRIA** France	**Karlsruhe Institute of Technology** Karlsruhe, Germany	**Kasetsart University** Bangkok, Thailand	**King Abdullah University of Science and Technology** Thuwal, Saudi Arabia
Laboratoire d'Informatique de Paris 6 Paris, France	**Moscow State University** Moscow, Russia	**National Institute of Advanced Industrial Science and Technology** Tsukuba, Japan	**Prometeus GmbH** Mannheim, Germany	**Regionales Rechenzentrum Erlangen** Erlangen, Germany
RIKEN Kobe/Wako, Japan	**Rutherford Appleton Laboratory** Chilton, Oxfordshire, England	**Soongsil University** Seoul, South Korea	**Technische Universitaet Wien** Vienna, Austria	**Technische Universität Dresden** Dresden, Germany
Tokyo Institute of Technology Tokyo, Japan	**Umeå University** Umeå, Swede	**Université Claude Bernard Lyon 1** Lyon, France	**University of Bordeaux** Bordeaux, France	**University of Cape Town** Cape Town, South Africa
University of Manchester Manchester, England	**University of Paris-Sud** Paris, France	**University of Picardie Jules Verne** Amiens, France	**University of Tsukuba** Tsukuba, Japan	

ICL/LEADERSHIP

While leading-edge research and high-impact software are hallmarks of the ICL mission, the lab also cultivates a strong core of leadership. To this end, ICL actively engages the HPC and computational research communities through impactful efforts like those outlined below.

THE UNIVERSITY OF TENNESSEE KNOXVILLE
CENTER FOR INFORMATION TECHNOLOGY RESEARCH

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

The Center for Information Technology Research (CITR) was established in 2001 to drive the growth and development of leading-edge information technology research at UTK. CITR's primary objective is to develop a thriving, well-funded community in basic and applied information technology research to help the university capitalize on the rich supply of 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. CITR has also provided secondary support for other UTK centers.

JLESC

JLESC Meeting Group Photo

ICL is now an Associate Partner of the Joint Laboratory for Extreme Scale Computing (JLESC). JLESC, founded in 2009 by the French Institute for Research in Computer Science and Automation (INRIA) and the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign, is an international, virtual organization that aims to enhance the ability of member organizations and investigators to overcome software challenges found in extreme scale, high-performance computers.

JLESC engages computer scientists, engineers, application scientists, and industry leaders to ensure that the research facilitated by the joint laboratory addresses science and engineering's most critical needs and takes advantage of the continuing evolution of computing technologies. Other partners include Argonne National Laboratory, the Barcelona Supercomputing Center, Jülich Supercomputing Center, and the RIKEN Center for Computational Science.

BDEC2 Indiana Group Photo

BDEC2

BIG DATA AND EXTREME-SCALE COMPUTING 2

In the past decade, the United States, the European Union, Japan, and China have each moved aggressively to develop their own plans for achieving exascale computing in the wake of a surging “data tsunami.” Focusing on scientific research and building on the previous International Exascale Software Project (IESP) and Big Data and Extreme-scale Computing (BDEC) efforts, the goal of the BDEC2 project is to stage six international workshops to enable research communities in a wide range of disciplines to converge on a common platform to meet the challenge of computing in the era of exascale and big data.

In 2018, ICL was instrumental in organizing and staging the first BDEC2 workshop in Bloomington, Indiana. Along with Jack Dongarra, several members of ICL’s CITR staff—including Terry Moore, Tracy Rafferty, and David Rogers—played essential roles in making the first BDEC2 workshop a major success. The next BDEC2 workshop, in which ICL is also heavily involved, is planned for the spring of 2019 in Kobe, Japan.

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

IGMCS

Interdisciplinary Graduate Minor in Computational Science

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

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. The IGMCS curriculum, requirements, and policies are governed by a program committee composed of faculty members from participating IGMCS academic units and departments.

ICL/PEOPLE

ICL Retreat, August 2018

As the HPC landscape continues to evolve rapidly, 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.

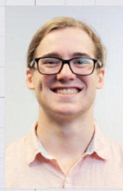
As part of an engineering college at a public research university, we have a responsibility to combine exemplary teaching with cutting-edge research. As such, we regularly employ bright and motivated graduate and undergraduate students. We have been, and will continue to be, very proactive in securing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

Staff and Students

Ahmad Abdelfattah
Research Scientist I

Hartwig Anzt
Research Consultant

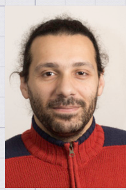
Alan Ayala
Post Doctoral Research Associate



Daniel Barry
Graduate Research Assistant

John Batson
Technical Editing Intern

George Bosilca
Research Assistant Professor

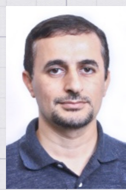


Aurelien Bouteiller
Research Director

Qinglei Cao
Graduate Research Assistant

Earl Carr
Program Administrator

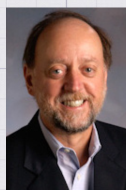
Tony Castaldo
Research Scientist II



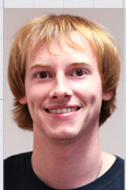
Ali Charara
Research Scientist I

Sam Crawford
Information Specialist II

Anthony Danalis
Research Director



Jack Dongarra
University Distinguished Professor
Director of ICL



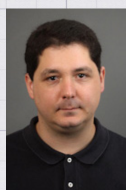
David Eberius
Graduate Research Assistant

Teresa Finchum
Administrative Specialist II

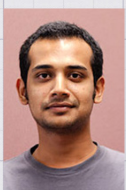
Jamie Finney
Software Engineer

Mark Gates
Research Scientist II

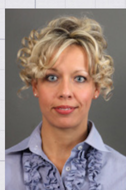
Damien Genet
Research Scientist I



Thomas Herault
Research Director

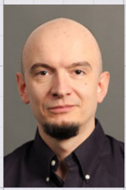


Reazul Hoque
Graduate Research Assistant



Heike Jagode
Research Assistant Professor

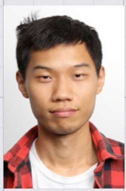
Anara Kozhokanova
Graduate Research Assistant



Jakub Kurzak
Research Assistant Professor

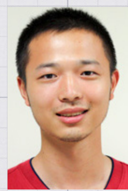
Julie Langou
Research Leader

Tracy Lee
Financial Specialist I



Yicheng Li
Graduate Research Assistant

Jiali Li
Graduate Research Assistant



Xi Luo
Graduate Research Assistant

Piotr Luszczek
Research Assistant Professor

Terry Moore
Associate Director

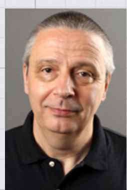
Phil Mucci
Research Consultant

Thananon Patinyasakdikul
Graduate Research Assistant

Yu Pei
Graduate Research Assistant

Tracy Rafferty
Business Manager

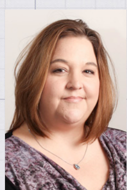
Gerald Ragghianti
Research Leader



Yves Robert
Visiting Scholar

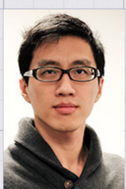
David Rogers
IT Specialist III

Daniel Schultz
Graduate Research Assistant

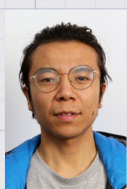


Leighanne Sisk
Administrative Specialist I

Stanimire Tomov
Research Assistant Professor



Yaohung Tsai
Graduate Research Assistant



Xiaoyang Wang
Graduate Research Assistant

Frank Winkler
Consultant

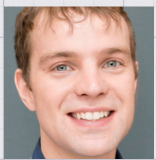
Ichitaro Yamazaki
Research Scientist II

Asim YarKhan
Research Scientist II

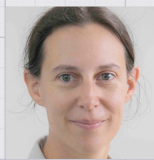
Dong Zhong
Graduate Research Assistant

ICL/VISITORS

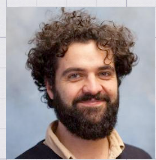
ICL has a long-standing tradition of hosting visitors from all over the world. Some stay only briefly to give insightful seminars or presentations, while others remain with us for as long as a year to collaborate, teach, and learn. Our connection to these researchers enables us to leverage an immense array of intellectual resources and work with the best and brightest people in the HPC community.



Maksims Abalenkovs
University of Manchester, UK



Anne Benoit
ENS Lyon and Georgia Tech

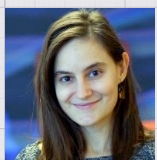


Pierre Blanchard
University of Manchester, UK

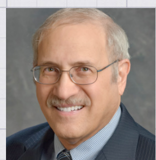
Terry Cojean
Karlsruhe Institute of Technology (KIT), Germany

Scott Emrich
University of Tennessee, Knoxville

Edgar Gabriel
University of Houston



Ana Gainaru
Vanderbilt University

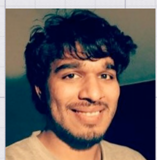


Lou Gross
NIMBioS
University of Tennessee

Travis Johnston
Oak Ridge National Laboratory

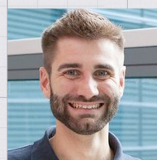
John Levesque
Cray

Yuechao Lu
Osaka University, Japan



Pratik Nayak
Karlsruhe Institute of Technology (KIT), Germany

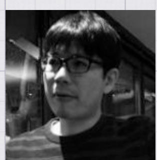
Srikara Pranesh
University of Manchester, UK



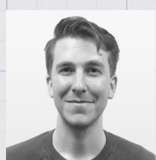
Joseph Schuchart
High Performance Computing Center, Stuttgart (HLRS), Germany

Hejer Shaiek
ENSEEIHT, France

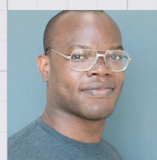
Jakub Šístek
University of Manchester, UK



Keita Teranishi
Sandia National Laboratories



Michael Wyatt
Global Computing Laboratory



Mawussi Zounon
University of Manchester, UK

ICL/ALUMNI

ICL has attracted many research scientists and students from a variety of backgrounds and academic disciplines. Many of these experts came to UTK specifically to work with Prof. Dongarra—beginning a long list of research talent to pass through ICL and move on to make exciting contributions at other institutions and organizations.

Maksims Abalenkovs
Carolyn Aebischer
Bivek Agrawal
Sudesh Agrawal
Emmanuel Agullo
Jennifer Allgeyer
Wes Alvaro
Ed Anderson
Daniel Andrzejewski
Thara Angskun
Papa Arkhurst
Dorian Arnold
Cedric Augonnet
Marc Baboulin
Matthew Bachstein
Zhaojun Bai
Ashwin Balakrishnan
Richard Barrett
Alex Bassi
David Battle
Micah Beck
Dulceneia Becker
Daniel Becker
Adam Beguelin
Annamaria Benzoni
Tom Berry
Vincent Berthoux
Scott Betts
Nikhil Bhatia
Laura Black
Noel Black
Susan Blackford
Wesley Bland
Kartheek Bodanki
David Bolt
Fernando Bond
Carolyn Bowers
Barry Britt
Randy Brown
Murray Browne
Bonnie Browne
Cynthia Browne
Giuseppe Bruno
Antonin Bukovsky
Greg Bunch
Alfredo Buttari
Anthony Canino
Domingo Gimenez Canovas
Chongxiao “Shawn” Cao
Henri Casanova
Cedric Castagnede
Ramkrishna Chakrabarty
Sharon Chambers
Zizhong Chen
Jaeyoung Choi
Wahid Chrabakh
Eric Clarkson
Andy Cleary
Michelle Clinard
Vincent Cohen-Addad
Matthias Colin
Charles Collins
Stephanie Moreaud Cooper
Tom Cortese
Camille Coti
Jason Cox
David Cronk
Javier Cuenca
Manoel Cunha
Yuanshun (Shaun) Dai
Cricket Deane
Remi Delmas
Frederic Desprez
Ying Ding
Jin Ding
Jun Ding
Martin Do
Simplice Donfack
Tingxing “Tim” Dong
Leon Dong
Nick Dongarra
David Doolin
Joe Dorris
Andrew Downey
Mary Drake
Julio Driggs
Brian Drum
Peng Du
Eduardo Echavarria
Victor Eijkhout
Brett Ellis
Shawn Ericson
Zachary Eyler-Walker
Lisa Ezzell
Christoph Fabianek
Graham Fagg
Mathieu Faverge
Diana Fayad
Shengzhog Feng
Don Fike
Salvatore Filippone
Anna Finchum
Mike Finger
Markus Fischer
Len Freeman
Xiaoquan Fu
Erika Fuentes
Karl Fuerlinger
Megan Fuller
Edgar Gabriel
Tracy Gangwer
Lynn Gangwer
Kelley Garner
Nathan Garner
Tina Garrison
Adriana Garties
Peter Gaultney
Christoph Geile
Jean Patrick Gelas
Boris Gelfend
Jonathan Gettler
Scott Gibson
Eric Greaser
Stan Green
Alice Gregory
Amina Guermouche
Jason Gurley
Bilel Hadri
Hunter Hagewood
Azzam Haidar
Christian Halloy
Sven Hammarling
J. Mike Hammond
Hanumantharayappa
Hidehiko Hasegawa
Satomi Hasegawa
Chris Hastings
Blake Haugen
David Henderson
Greg Henry
John Henry
Julien Herrmann
Holly Hicks
Alexandra Hicks-Hardiman
Sid Hill
Tomoyuki Hiroyasu
George Ho
Josh Hoffman
Jeff Horner
Mitch Horton
Yan Huang
Harry Hughes
Aurelie Hurault
Chris Hurt
Paul Jacobs
Emmanuel Jeannot

Weizhong Ji
Yulu Jia
Weicheng Jiang
Song Jin
Patrick Johansson
Matt Johnson
Aral Johnson
Sean Jolly
Jan Jones
Kim Jones
Vijay Joshi
Khairul Kabir
Venkata Kakani
Ajay Kalhan
Balajee Kannan
Madhuri Kasam
Kiran Kumar Kasichayanula
Ajay Katta
David Katz
Joshua Kelly
Supriya Kilambi
Myung Ho Kim
Youngbae Kim
Jenya Kirshtein
Cindy Knisley
Michael Kolatis
Moritz Kreutzer
Chandra Krintz
Tilman Kuestner
Krerkchai Kusolchu
Coire Kyle
Amanda Laake
Xavier Lacoste
Julien Langou
Jeff Larkin
Brian LaRose
Frank Lauer
Valentin Le Fevre
DongWoo Lee
Pierre Lemarinier
Todd Letsche
Sharon Lewis
Yinan Li
Weiran Li
Xiang Li
Chaoyang Liu
Kevin London
Matt Longley
Florent Lopez
Nuria Losada
Hatem Ltaief
Daniel Lucio
Richard Luczak
Teng Ma
Sticks Mabakane
Robert Manchek
Gabriel Marin
Tushti Marwah
Theo Mary
Ian Masliah
Donald McCasland
Tyler McDaniel
Paul McMahan
Eric Meek
James Meyering
Jeremy Millar
Michelle Miller
Cindy Mitchell
Stuart Monty
Keith Moore
Shirley Moore
Erik Moore
Robert Morgan
Kishan Motheramgari
Steven Moulton
Daichi Mukunoki
Matthew Nabity
Shankar Narasimhaswami
Rajib Nath
Fernando Navarro
John Nelson
Donnie Newell
Peter Newton
Jonas Nilsson
Jakob Oestergaard
Caroline Papadopoulos
Leelinda Parker
Thibault Parpaite
Dilip Patlolla
Andy Pearson
Paul Peltz
Theresa Pepin
Antoine Petitet
Peter Pham
Gregoire Pichon
Vlado Pjesivac
Jelena Pjesivac-Grbovic
James S. Plank
Tim Poore
Roldan Pozo
Farzona Pulatova
Martin Quinson
Tammy Race
Sangamesh Ragate
James Ralph
Ganapathy Raman
Kamesh Ramani
Mei Ran
Arun Rattan
Sheri Reagan
Mike Reynolds
George Rhinehart
Jon Richardson
Stephen Richmond
Ken Roche
Andrew Rogers
Tom Rothrock
Tom Rowan
Narapat (Ohm) Saengpatsa
Kiran Sagi
Evelyn Sams
Ken Schwartz
Keith Seymour
Farial Shahnaz
Hejer Shaiek
Brian Sheely
Zhiao Shi
Sergei Shinkarev
Majed Sidani
Shilpa Singhal
Matt Skinner
Peter Soendergaard
Raffaele Solca
Gwang Son
Fengguang Song
Thomas Spencer
Jeffrey Steill
Erich Strohmaier
Xiaobai Sun
Martin Swany
Daisuke Takahashi
Judi Talley
Ronald Tam
Chunyan Tang
Yuan Tang
Yusuke Tanimura
Keita Teranishi
Dan Terpstra
Joe Thomas
John Thurman
Francoise Tisseur
Jude Toth
Bernard Tourancheau
Volodymyr Turchenko
Lauren Vaca
Phil Vaccaro
Sathish Vadhiyar
Robert van de Geijn
Chad Vawter
Eugene Vecharynski
Scott Venckus
Antoine Vernois
Reed Wade
Michael Walters
Robert Waltz
Mike Waltz
Jerzy Wasniewski
Vince Weaver
Scott Wells
David West
R. Clint Whaley
Jody Whisnant
James White
Scotti Whitmire
Susan Wo
Felix Wolf
Stephen Wood
Wei Wu
Jiayi Wu
Panruo Wu
Qiu Xia
Tinghua Xu
Tao Yang
Erlin Yao
Kevin Ye
Jin Yi
Haihang You
Lamia Youseff
Brian Zachary
Dmitry Zaitsev
Omar Zenati
Yuanlei Zhang
Junlong Zhao
Yong Zheng
Luke Zhou
Min Zhou

ACKNOWLEDGMENTS

For 30 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, and we acknowledge the following agencies for supporting our efforts:

AFOSR AIR FORCE OFFICE OF SCIENTIFIC RESEARCH

NSF

OAK RIDGE National Laboratory

U.S. DEPARTMENT OF ENERGY | Office of Science

U.S. DEPARTMENT OF ENERGY | NNSA National Nuclear Security Administration

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

Atos

ENGILITY

intel

MathWorks

NVIDIA

UT-BATTELLE

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://www.icl.utk.edu/