INNOVATIVE COMPUTING LABORATORY

2020/21 REPORT

THE UNIVERSITY OF TENNESSEE KNOXVILLE

INNOVATIVE COMPUTING LABORATORY

2020/21 REPORT

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

2020/21 ICL REPORT

FROM THE DIRECTOR

Jack Dongarra
Director, ICL

You will not be surprised to learn that writing this year's "Letter From the Director" presented some unique challenges. How can one give a brief commentary on what happened at ICL in a year like 2020, when the global Covid-19 pandemic so completely disrupted all our familiar ways of being, living, and working together? The fact that in 2019 we celebrated our thirtieth anniversary, which meant that our sense of the history of life and work at ICL was fresh in our memories, makes the sense of radical discontinuity we have been experiencing in 2020 all the more jarring. With that in mind, I want to begin this letter where I often end it—by saying how proud and grateful I am to ICL's researchers, students, and staff for the way they have persevered in facing the challenges of this exceedingly difficult year.

Of course, meeting our commitments to the Department of Energy's (DOE) Exascale Computing Project (ECP) was our main priority in 2020, as it has been for the last several years. By engaging with and supporting ECP's mission-critical application communities, and by facilitating the integration of our components (e.g., SLATE, PAPI, PaRSEC, PEEKS, heFFTe, OpenMPI, xSKD, CEED) into ECP's portfolio, we continue to play a vital role in creating the robust software ecosystem on which future exascale computing efforts around the world will depend. The investments we have made over many years to build up our proficiency in translating research innovation into production-quality software are serving us well now, as we strive in this uncanny time to sustain our leadership.

By contrast with ECP, where our roles and schedules of work are well established, the pandemic has made the problem of developing new research collaborations and opportunities more challenging. Building collaborations is a social process, and the whole social metabolism of our HPC community has been upended by the need to do nearly everything virtually through a Zoom window. Fortunately, we have some long-standing collaborations that we can still build on to do new things.

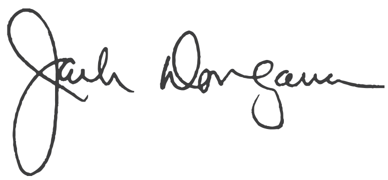
For example, this year, the National Science Foundation (NSF) funded the Basic ALgebra LIbraries for Sustainable Technology with Interdisciplinary Collaboration (BALLISTIC) project, which continues our ongoing research on numerical libraries with our colleagues at the University of California, Berkeley and the

University of Colorado Denver. The goal of BALLISTIC is to create a layered package of software components that provides scientific applications with state-of-the-art linear algebra algorithms, numerics, and performance that can run at every level of the platform deployment pyramid—updating LAPACK and ScaLAPACK. Also, in the important field of artificial intelligence (AI), we built on our long-running collaborations with Indiana University, Argonne National Laboratory (ANL), and Rutgers University to develop the Surrogates Benchmark Initiative (SBI), which was funded by DOE early this summer. Computational science is being revolutionized by AI-generated surrogates that can replace all or part of traditional simulations to achieve incredible performance improvements. SBI will create a community repository and open data ecosystem for HPC application surrogate benchmarks, including data, code, and all the collateral artifacts that science and engineering communities need to create these surrogates.

Of course, the creativity, expertise, and industry of ICL's people are what make our projects successful, and the pandemic has complicated the challenge of bringing good new people on board. This year, Joseph Schuchart and Wissam Sid Lakhdar joined our Distributed Computing and Linear Algebra groups, respectively; but because of the pandemic, they are currently working for us through a contractor, the former from Germany and the latter from Algeria. Closer to home, Paul Bagwell, who joined our Linear Algebra group as a software engineer, is also working remotely from Seymour, TN. Finally, at the beginning of December, we were fortunate to be able to hire Deborah Penchoff, a talented and accomplished professional and team leader, with expertise in computational chemistry, who has been working for UT's Howard Baker Center for Public Policy. Over the next few months, she will prepare to step into the position of Associate Director, when Terry Moore retires at the end of February 2021. As you know, Terry has ably filled that role for more than two decades, bringing a unique combination of skill, creativity, dedication, good humor, and (need I say it?) philosophical perspective to the job. Helping Deborah manage that transition under the current conditions adds an extra degree of difficulty to an already challenging task.

But even with all of the difficult challenges that our current situation presents, my optimism for the future of ICL is undiminished. Our funding is solid, and our research teams have continued to be productive, despite the pains of the social isolation we are all suffering through. Our administrative and technical support team members—Joan, Teresa, Tracy, Leighanne, Earl, David, Sam, Geri, and Terry—have shown tremendous resolve and resilience in keeping ICL services running smoothly. Our collaborations remain strong, not only nationally, but also locally, with Michela Taufer and her group in the Global Computing Laboratory. We all know that the struggle with the pandemic will continue well into 2021, and we all miss the vitality and camaraderie that has always been such an important part of life at ICL. But I am proud and thankful for the true mettle that our group has shown through the troubles of 2020, and it gives me every confidence that our creativity and determination will restore us to much better conditions next year.

As always, I am grateful to the many government, industry, and private sponsors whose support has helped make that success possible, but especially to the strong and ongoing support of the University of Tennessee administration and the Tickle College of Engineering.



INTRODUCTION

Located in the heart of the University of Tennessee's Knoxville campus, ICL is a computer science research and development laboratory with a rich, 30-year history that has spanned radical advancements in computing hardware and parallelism and the explosion of data-intensive science and computation wedged against the need for ever-increasing energy efficiency and resilience.

ICL's work, which has evolved and expanded to address these challenges, 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.

AREAS OF RESEARCH

NUMERICAL LINEAR ALGEBRA

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

PERFORMANCE EVALUATION & 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 require runtime systems that can exploit all of 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 over two decades, and the lab has numerous projects in this arena under active development.

HISTORY

Prof. Jack Dongarra established ICL in 1989 when he received a dual appointment as a Distinguished Professor at UTK and as a Distinguished Scientist at Oak Ridge National Laboratory (ORNL). Over 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 over 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.

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 parallel networking tool that enabled a user to leverage a network of heterogeneous Unix and Windows machines as a single distributed parallel processor.

1992 The **Basic Linear Algebra Communication Subprograms (BLACS)** project was created to make linear algebra applications easier to program and more portable.

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 super-computer 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. MPI has since become the de facto standard for communication in parallel distributed computing systems.

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.

1996 The **Programming Environment and Training (PET)** effort was a component of the US Department of Defense (DoD) High Performance Computing Modernization Program and was initiated to allow full-capacity use of DoD resources and extend the range of applicability to DoD technical problems through training, support for software development, and technology transfer.

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.

The **Repository in a Box (RIB)** was developed as a toolkit for creating and maintaining web-based, interoperable metadata repositories.

1999 **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.

2000 **High-Performance Linpack (HPL)** is a benchmark for distributed-memory computers that solves a (random) dense linear system in double-precision (64-bit) arithmetic. HPL is often one of the first programs to run on large HPC machines, producing a result that can be submitted to the TOP500 list of the world's fastest supercomputers.

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. FT-MPI was later merged with LA-MPI, LAM/MPI, and PACX-MPI to form the Open MPI project.

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.

Four institutions merged efforts in the **Open Source Message Passing Interface (Open MPI)**: FT-MPI from UTK/ICL, LA-MPI from LANL, and LAM/MPI from Indiana University, with contributions from PACX-MPI 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.

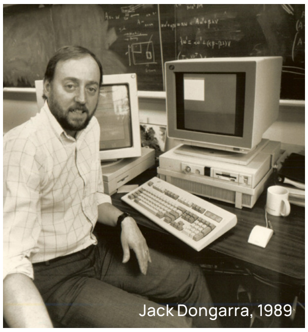
2009 The **International Exascale Software Project (IESP)** brought together representatives of the global HPC community to plan and create a new software infrastructure for the extreme-scale systems that represent the future of computational science.

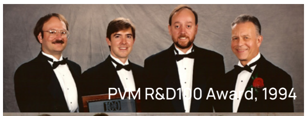
2010 **Distributed Parallel Linear Algebra Software for Multi-core Architectures (DPLASMA)** is a linear algebra package that enables sustained performance for distributed systems, where each node features multiple sockets of multi-core CPUs and, if applicable, accelerators like GPUs or Intel Xeon Phi.

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 multi-stream, 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.


Jack Dongarra, 1989


PVM R&D100 Award, 1994


Atlas & NetSolve R&D100 Awards, 1997


PAPI R&D100 Award, 1999

ICL 25th Anniversary, 2015


ICL 30th Anniversary, 2019

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.

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.

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.

2015 **Data-driven Autotuning for Runtime Execution (DARE)** provides application-level performance tuning capabilities to the end user.

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

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 landed seven awards through the DOE's ECP program 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.

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.

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."

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.

2019 ICL celebrated its 30th year in 2019.

The **Ecosystem for Programming and Executing eXtreme Applications (EPEXA)**, aims to create a software framework that implements high-performance methods for irregular and dynamic computations that are poorly supported by current programming paradigms.

The **Scalable Run Time for Highly Parallel, Heterogeneous Systems (ScaRT)** project aims to increase the scientific throughput of existing and future cyberinfrastructure platforms by reducing communication overheads and better matching the functionality of communication libraries to modern communication adapters.

The **Development of Exascale Software for Heterogeneous and Interfacial Catalysis (DESC)** project focuses on understanding the relationship between algorithms and hardware platforms and on jointly optimizing the software and hardware to achieve efficient implementations of materials science, chemistry, and physics applications.

2020 ICL began work on two new projects in 2020:

The **Surrogate Benchmark Initiative (SBI)** aims to provide benchmarks and tools for assessing deep neural network "surrogate" models. A surrogate model can imitate part or all of a simulation and produce the same outcomes while requiring less resources. Tools developed under SBI will evaluate these surrogate models to measure progress and inform the codesign of new HPC systems to support their use.

Basic ALgebra LIbraries for Sustainable Technology with Interdisciplinary Collaboration (BALLISTIC) will create software components that deliver access to the most up-to-date algorithms, numerics, and performance via Sca/LAPACK interfaces; make available advanced algorithms, numerics, and performance capabilities; and provide a well-engineered conduit for new developments to be channeled to the science and engineering applications that depend on high-performance linear algebra libraries.

YEAR IN REVIEW

Perseverance

In a year unlike any other in recent memory, ICL is very fortunate to have the infrastructure and support of the University of Tennessee and the Tickle College of Engineering, both of which are encouraging laboratories like ICL to conduct research and work remotely when possible. And while working from home is not a novel idea, this kind of en masse, off-site productivity is unprecedented. Nevertheless, ICL's efforts in 2020 have yielded significant results, and our hard work is being recognized within UTK and beyond.

Prof. Jack Dongarra and 2020 graduate Dr. Yaohung "Mike" Tsai.

High-Performance Hardware

In late 2020, NVIDIA awarded ICL one of their new DGX A100 compute nodes. This new system is equipped with 2 AMD 64-core Rome CPUs in a dual-socket configuration with 2 TB of system memory flanked by 8 state-of-the-art NVIDIA A100 (Ampere) GPUs with a total of 640 GB of GPU memory. The new A100 GPUs have a set of unique processing cores (Tensor Cores) that can utilize NVIDIA's Tensor Float 32 arithmetic for significant computational speedups in applications like machine learning and AI without sacrificing the accuracy of more traditional floating point arithmetic.

The new DGX A100 will join ICL's machine rack alongside an NVIDIA DGX-1 compute node, which was awarded in 2019. These awards stem from a longstanding relationship with NVIDIA, including ICL's designation as an NVIDIA CUDA Center of Excellence. Overall, the University's acquisition of HPC hardware has been exceptional in recent years, and in 2019, UTK was also awarded a POWER9-based compute cluster from IBM that uses nodes similar to those in ORNL's Summit supercomputer.

ACCOLADES

2020 IEEE Computer Society Computer Pioneer Award

In May of 2020, ICL Director and UTK Distinguished Professor Jack Dongarra received the IEEE Computer Society Computer Pioneer Award for his "leadership in the area of high-performance mathematical software." The Computer Pioneer Award was established in 1981 by the IEEE Computer Society to recognize and honor the efforts of those who contributed to the creation and continued vitality of the computer industry.

UTK EECS Gonzalez Family Award for Excellence in Research

In April of 2020, UTK's Min H. Kao Department of Electrical Engineering & Computer Science (EECS) presented Prof. Jack Dongarra with the 2020 Gonzalez Family Award for Excellence in Research for his leadership in ICL's numerical linear algebra, distributed computing, and performance analysis and benchmarking research efforts.

IEEE-CS Charles Babbage Award

In May of 2020, ICL collaborator and frequent visitor Prof. Yves Robert was awarded the 2020 IEEE Computer Society Charles Babbage Award for "contributions to parallel algorithms and scheduling techniques." The award consisted of a $1,000 honorarium, certificate, and an invitation to present at the 34th annual IEEE International Parallel and Distributed Processing Symposium (IPDPS 2020).

Best Paper Award

Recent ICL graduate Dr. Xi Luo and his coauthors earned a Best Paper Award at IEEE Cluster 2020, which was held virtually in lieu of an on-the-ground meeting in Kobe, Japan.

The paper, "HAN: A Hierarchical AutotuNed Collective Communication Framework," lays out a strategy for adapting to the increased scale and heterogeneity of new and upcoming HPC platforms and the challenges that these trends present to the design of MPI libraries.

With a task-based and modular design, HAN can easily swap out abstracted submodules—while keeping tasks intact—when adapting to new hardware. This strategy provides strong and flexible support for future HPC systems as the hardware inevitably changes underneath. Owing to its autotuning component, the HAN approach also significantly improves the default Open MPI implementation and achieves decent speedups against state-of-the-art MPI implementations in a battery of application tests.

RESEARCH

FROM LEADING-EDGE RESEARCH TO HIGH-IMPACT SOFTWARE

What originally began over 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, including eight projects sponsored by DOE's Exascale Computing Project (ECP). 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.

RESEARCH AREA	SOFTWARE	VERSION	ASSOCIATED PROJECTS	AVAILABLE AT
Numerical Linear Algebra	BLAS	3.8.0	LAPACK, ScaLAPACK, HPL	http://github.com/Reference-LAPACK
	Ginkgo	1.3.0	PEEKS, MAGMA	https://github.com/ginkgo-project/ginkgo/releases
	heFFTe	2	MAGMA, heFFTe (ECP FFT)	https://bitbucket.org/icl/heffte/
	LAPACK	3.9.0	Linear Algebra PACKage (LAPACK)	http://github.com/Reference-LAPACK/lapack
	MAGMA	2.5.4	PEEKS, CORES, MATEDOR	https://bitbucket.org/icl/magma/src/master/
	MagmaDNN	1.2	MAGMA, MATEDOR	https://bitbucket.org/icl/magmadnn/
	PLASMA	20.9.20	PLASMA	https://bitbucket.org/icl/plasma
	ScaLAPACK	2.1.0	Scalable LAPACK (ScaLAPACK)	https://github.com/Reference-ScaLAPACK/scalapack/
	SLATE	2020.10.00	SLATE	https://bitbucket.org/icl/slate
Performance Evaluation and Benchmarking	HPCG	3.1	HPL, TOP500	https://github.com/hpcg-benchmark/hpcg/
	HPL	2.3	TOP500, HPL-AI	https://www.netlib.org/benchmark/hpl/
	HPL-AI	2020.2.25	HPL	https://bitbucket.org/icl/hpl-ai/
	PAPI	6.0.0	Exa-PAPI, PULSE, PAPI-ex	https://bitbucket.org/icl/papi/
Distributed Computing	DPLASMA	2.0.0	PaRSEC, DTE	https://bitbucket.org/icldistcomp/dplasma/
	PaRSEC	2.0.0	DTE, DPLASMA	https://bitbucket.org/icldistcomp/parsec/
	Open MPI	4.0.5	ULFM, Evolve, OMPI-X	https://github.com/open-mpi/ompi
	ULFM	4.0.2u1	Open MPI, Evolve	https://bitbucket.org/icldistcomp/ulfm2

AsyncIS

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

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.

BALLISTIC

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

Basic ALgebra LIbraries for Sustainable Technology with Interdisciplinary Collaboration (BALLISTIC) is an NSF-funded effort to create new software components capable of running at every level of the platform pyramid by delivering seamless access to the most up-to-date algorithms, numerics, and performance via familiar Linear Algebra PACKage (LAPACK) and Scalable Linear Algebra PACKage (ScaLAPACK) interfaces; by making advanced algorithms, numerics, and performance capabilities available through new interface extensions; and by providing a well-engineered conduit for channeling new developments to science and engineering applications that depend on high-performance linear algebra libraries.

Scientific software libraries have long provided a large and growing resource for high-quality, reusable software components upon which applications from science and engineering can be rapidly constructed. The BALLISTIC project, through the leading-edge research it channels into its software deliverables, will lead to the introduction of tools that will simplify the transition to the next generation of extreme-scale computer architectures. The main impact of the project will be to develop, release, and deploy software into the scientific community to make it competitive on a world-wide scale and to contribute to standardization efforts in the area.

Batched BLAS

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

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 application programming interface (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 establishing the consensus for the next official standard that will serve the scientific community and ensure support from hardware vendors.

CAARES

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

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 are 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.

CEED

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

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 is 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 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 Multicore Architectures (MAGMA) library.

CORES

AVAILABLE IN
MAGMA 2.5.4

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

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.

DESC

FIND OUT MORE AT
https://hetcat-ccs.github.io/

The Development of Exascale Software for Heterogeneous and Interfacial Catalysis (DESC) project focuses on understanding the relationship between algorithms and hardware platforms and how to jointly optimize the software and hardware to achieve efficient implementations for applications in materials science, chemistry, and physics. DESC is a joint effort between ICL/UTK, DOE's Ames Laboratory, EP Analytics, Inc., Georgia Tech, Old Dominion University, and Virginia Tech and is funded by the DOE Computational Chemical Sciences project.

ICL's contribution focuses on expressing GAMESS computational chemistry algorithms in the form of a dataflow graph and subsequently mapping the DAG representation of the kernels to the hardware platforms. This representation allows for capturing the essential properties of the algorithms (e.g., data dependencies) and computation at extreme scale by utilizing the hardware components (e.g., CPU or GPU) best suited for the type of computational task under consideration. The dataflow-based form of these algorithms makes them compatible with next-generation task scheduling systems like PaRSEC, StarPU, and Legion.

DPLASMA

VERSION
2.0.0

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

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.

DTE

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

The Distributed Tasking for Exascale (DTE) project will extend the capabilities of ICL's PaRSEC runtime—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 (DSLs) and extensions and tools for debugging, trace collection, and analysis.

PaRSEC also enables fast prototyping DSLs to express the dependencies between tasks and provides a stable, scalable, and efficient distributed runtime so they can run on any execution platform at any scale. The underlying 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.

EPEXA

FIND OUT MORE AT
https://www.icl.utk.edu/research/epexa/

A collaborative project involving Virginia Tech, Stony Brook, and ICL/UTK, the Ecosystem for Programming and Executing eXtreme Applications (EPEXA) aims to create a software framework that implements high-performance methods for irregular and dynamic computations that are poorly supported by current programming paradigms. Employing science-driven codesign, the EPEXA team will harden a successful research prototype into an accessible, production-quality programming model that will leverage DSLs to improve accessibility and accelerate the adoption of high-performance tools for computer scientists and domain scientists.

The project bridges the so-called "valley of death" between a successful proof of concept and an implementation with enough quality, performance, and community support to motivate application scientists and other researchers to adopt it and push for its community use. Specifically, the new powerful data-flow programming model and associated parallel runtime directly address multiple challenges faced by scientists as they leverage rapidly changing computer technologies—including current massively parallel, hybrid, and many-core systems.

Evolve

Evolve, a collaborative effort between ICL and the University of Houston, expands the capabilities of the Open Message Passing Interface (Open MPI) to support the 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., Open Multi-Processing [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.

Exa-PAPI

AVAILABLE IN
PAPI 6.0.0

FIND OUT MORE AT
https://icl.utk.edu/exa-papi/

The Exascale Performance Application Programming Interface (Exa-PAPI) project is developing a new C++ Performance API (PAPI++) software package from the ground up that offers a standard interface and methodology for using low-level performance counters in CPUs, GPUs, on/off-chip memory, interconnects, and the I/O system—including energy/power management. PAPI++ is building upon classic-PAPI functionality and strengthening its path to exascale with a more efficient and flexible software design—a design that takes advantage of C++'s object-oriented nature but preserves the low-overhead monitoring of performance counters and adds a vast testing suite.

In addition to providing hardware counter–based information, a standardizing layer for monitoring software-defined events (SDE), which exposes the internal behavior of runtime systems and libraries (e.g., communication and math libraries) to the applications, is being incorporated. As a result, the notion of performance events is broadened from strictly hardware-related events to also include software-based information. Enabling monitoring of both hardware and software events provides more flexibility to developers when capturing performance information.

FFT

AVAILABLE IN
heFFTe 2

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

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 develop a Highly Efficient FFTs for Exascale (heFFTe) library that provides fast and robust multidimensional FFTs for large-scale heterogeneous systems with multi-core processors and hardware accelerators. HeFFTe collects and leverages existing FFT capabilities while building a sustainable FFT library that minimizes data movements, optimizes MPI communications, overlaps computations with communications, and autotunes performance on various architectures and large scale-platforms. The current heFFTe v2.0 release achieves very good scalability on pre-exascale systems and performance that is close to 90% of the roofline peak.

HPCG

VERSION
3.1

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

The High Performance Conjugate Gradients (HPCG) benchmark is designed to measure performance that is representative of modern scientific applications. It does so by exercising the computational and communication patterns 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 3.1 reference code was released in March of 2019. 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 or source code releases from AMD, ARM, Intel, and NVIDIA, which are carefully optimized for the vendors' respective hardware platforms. The current HPCG performance list was released at SC20 and now features 160 supercomputing entries. HPCG rankings have also been tracked by TOP500.org since June of 2017.

HPL

VERSION
2.3

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

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 machines, producing a result that can be submitted to the TOP500 list of the world's fastest supercomputers. Carefully optimized versions of HPL are available from major HPC hardware vendors.

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 GNU 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 2020 TOP500 list, an optimized version of the HPL code achieved over 442 petaFLOP/s on the Fugaku supercomputer at RIKEN, Japan.

HPL-AI

VERSION
2020.2.25

FIND OUT MORE AT
https://icl.bitbucket.io/hpl-ai/

The High Performance LINPACK for Accelerator Introspection (HPL-AI) benchmark seeks to highlight the convergence of HPC and AI workloads based on machine learning (ML) and deep learning (DL) by solving a system of linear equations using novel, mixed-precision algorithms that exploit modern hardware. While traditional HPC focuses on simulation runs for modeling phenomena in a variety of scientific disciplines, the mathematical models that drive these computations mostly require 64-bit accuracy. However, the ML/DL methods that fuel advances in AI can achieve the desired results at 32-bit or lower precisions. This lesser demand for working precision fueled a resurgence of interest in new hardware platforms that deliver a mix of unprecedented performance levels and energy savings to achieve the classification and recognition fidelity afforded by higher-accuracy formats on classic hardware.

HPL-AI strives to unite these two realms by connecting its solver formulation to the decades-old HPL framework of benchmarking supercomputers. A number of large-scale HPC installations—including some machines on the TOP500—have now been benchmarked with HPL-AI, starting with Oak Ridge National Laboratory's Summit machine in 2019 and now including RIKEN's Fugaku supercomputer, which achieved 2 exaFLOP/s in mixed-precision performance.

LAPACK/ScaLAPACK

LAPACK 3.9.0 http://www.netlib.org/lapack/
ScaLAPACK 2.1.0 http://www.netlib.org/scalapack/

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

LAPACK 3.9.0, released in November 2019, adds a QR-preconditioned QR SVD method and an LAPACK Householder reconstruction routine. 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 2.1.0, which includes a new robust ScaLAPACK routine for computing the QR factorization with column pivoting along with improved accuracy of the Frobenius norm, was released in November 2019.

MAGMA

VERSION
2.5.4

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

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 features LAPACK-compliant routines for multi-core CPUs enhanced with NVIDIA or AMD GPUs. MAGMA 2.5.4 now includes more than 400 routines that cover one-sided dense matrix factorizations and solvers, two-sided factorizations, and eigen/singular-value problem solvers, as well as a subset of highly optimized BLAS for GPUs. A MagmaDNN package has been added and further enhanced to provide high-performance data analytics, including functionalities for ML applications that use MAGMA as their computational back end. The MAGMA Sparse and MAGMA Batched packages have been included since MAGMA 1.6.

MATEDOR

AVAILABLE IN
MAGMA 2.5.4

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

The MAtrix, TEnsor, and Deep-learning Optimized Routines (MATEDOR) project is performing 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 BBLAS 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.

OMPI-X

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

The Open MPI for Exascale (OMPI-X) project focuses on preparing the 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.

Open MPI

VERSION
4.0.5

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

The Open MPI Project is an open-source 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 (e.g., 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

VERSION
6.0.0

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

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, 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 2020, ICL and collaborators at the University of Maine worked on PAPI-Ex to build support for performance counters available in the latest generations of CPUs and GPUs (NVIDIA and AMD), develop support for system-wide hardware performance counter monitoring, and strengthen the sampling interface in PAPI. PAPI 6.0.0 was released on March 4, 2020. This release includes a new API for SDEs, a major revision of the "high-level API," and several new components, including ROCM and ROCM_SMI (for AMD GPUs), powercap_ppc and sensors_ppc (for IBM POWER9 and later), SDE, and the I/O component (exposes I/O statistics exported by the Linux kernel). PAPI 6.0.0 also ships with a new Counter Analysis Toolkit (CAT) that assists with native performance counter disambiguation through micro-benchmarks.

PaRSEC

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

The Parallel Runtime Scheduling and Execution Controller (PaRSEC) is a generic framework for architecture-aware scheduling and management of 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.

PEEKS

AVAILABLE IN
Gingko 1.3.0

FIND OUT MORE AT
https://icl.utk.edu/peeks/

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, heterogeneous node architectures, and the increasing adoption of low-precision processor technology.

The Production-ready, Exascale-Enabled Krylov Solvers (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 Ginkgo software ecosystem. The PEEKS algorithms focus on communication-minimizing Krylov solvers, parallel incomplete factorization routines, and parallel preconditioning techniques, as these building blocks form the numerical core of many complex application codes. Ginkgo provides native support for NVIDIA GPUs, AMD GPUs, and Intel GPUs to ensure successful delivery of scalable Krylov solvers in robust, production-quality software that can be relied on by ECP applications.

PLASMA

VERSION
20.9.20

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

The Parallel Linear Algebra Software for Multi-core Architectures (PLASMA) implements a set of fundamental linear algebra routines using the latest updates to the 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 has been used on a variety of systems using Intel CPUs and coprocessors, IBM POWER processors, and ARM processors. As a research vehicle, PLASMA continues as an example of modern design for new dense linear algebra algorithms. At the same time, PLASMA benefits from the continuous evolution of the OpenMP standard that now includes off-load functionality and enables porting to hardware accelerators. The latest PLASMA release, v20.9.20 from September 2020, added support for MAGMA, Apple's macOS, and fixed user-reported bugs related to Fortran and POSIX threads.

PULSE

AVAILABLE IN
PAPI 6.0.0

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

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.

ScaRT

FIND OUT MORE AT
https://www.icl.utk.edu/research/scart

The Scalable Run Time for Highly Parallel, Heterogeneous Systems (ScaRT) project aims to increase the scientific throughput of existing and future cyberinfrastructure platforms by reducing communication overheads; by improving the match between modern, parallel-computing frameworks and the applications running on top; and by better matching the functionality of the underlying communication library to the capabilities of modern communication adapters.

To this end, SCaRT brings together a multidisciplinary team to (1) design and implement a communication library with new communication primitives; (2) accelerate multiple task-based runtimes (e.g., Legion and PaRSEC) and communication libraries (e.g., MPI and GasNET); (3) port key components to a programmable NIC; and (4) deliver improvements and extensions to mainstream communication libraries to provide the new functionality.

SLATE

VERSION
2020.10.00

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

For decades, ICL has applied algorithmic and technological innovations to the process of pioneering, implementing, and disseminating dense linear algebra software—including the LAPACK and 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, some 25 years ago. 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 innovations 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.

SMURFS

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

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.

SBI

FIND OUT MORE AT
https://www.icl.utk.edu/research/surrogates

The DOE-funded Surrogate Benchmark Initiative (SBI) is a collaborative effort involving Indiana University, UTK/ICL, and Rutgers University that aims to provide new benchmarks and tools for assessing deep neural network "surrogate" models. Trained on data produced by ensemble runs of a given HPC simulation, a surrogate model can imitate—with high fidelity—part or all of that simulation and produce the same outcomes for a given set of inputs while requiring far less time and energy.

At present, however, there are no accepted benchmarks to evaluate these surrogate models, and there is no easy way to measure progress or inform the codesign of new HPC systems to support their use. SBI aims to address this fundamental problem by creating a community repository and a Findable, Accessible, Interoperable, and Reusable (FAIR) data ecosystem for HPC application surrogate benchmarks, including data, code, and all relevant collateral artifacts that the science and engineering community needs to use and reuse these data sets and surrogates.

TOP500

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

With over three decades of tracking the progress of supercomputing, 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 2020, the 56th TOP500 list was unveiled during the International Conference for High Performance Computing, Network, Storage, and Analysis (SC20), which was held virtually. Japan took the crown with Fugaku, their new ARM-based machine built by Fujitsu. The system's 158,976 A64FX SoCs propelled Fugaku to 442 petaFLOP/s in the HPL benchmark, making it the fastest supercomputer in the world by a factor of 2.8× (over the United States' Summit machine) and the first ARM system to achieve the number one spot on the TOP500.

ULFM

VERSION
4.0.2u1

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

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 wherein 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.

xSDK4ECP

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

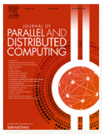
The Extreme-Scale Scientific Software Development Kit for the Exascale Computing Project (xSDK4ECP) is a collaboration between Argonne National Laboratory (ANL), ICL, Lawrence Berkeley National Laboratory (LBNL), LLNL, SNL, and the University of California, Berkeley. The project aims to enable seamless 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 requirements of DOE's Office of Science.

To ensure the consistency of naming conventions, runtime behavior, and installation procedures, xSDK informs the project development process by providing requirements and guidelines that are influential throughout the software development phase. xSDK lightens the burden on system administrators and application developers, because each xSDK package provides a Spack installation script that can be invoked independently or through the installation of the xSDK's Spack package. In addition, xSDK now ships with a set of curated examples that show potential integrations of packages into application exemplars. ICL's MAGMA, PLASMA, SLATE, and heFFTe libraries are now all included in the most recent release, xSDK 0.6.

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.

http://www.icl.utk.edu/publications



Abdelfattah, A., S. Tomov, and J. Dongarra, **"Matrix Multiplication on Batches of Small Matrices in Half and Half-Complex Precisions,"** *Journal of Parallel and Distributed Computing*, vol. 145, pp. 188-201, November 2020.

Abdelfattah, A., S. Tomov, and J. Dongarra, **"Matrix Multiplication on Batches of Small Matrices in Half and Half-Complex Precisions,"** *Journal of Parallel and Distributed Computing*, vol. 145, pp. 188-201, November 2020.

Abdelfattah, A., T. Costa, J. Dongarra, M. Gates, A. Haidar, S. Hammarling, N. J. Higham, J. Kurzak, P. Luszczek, S. Tomov, et al., **"A Set of Batched Basic Linear Algebra Subprograms,"** *ACM Transactions on Mathematical Software*, October 2020.

Abdelfattah, A., S. Tomov, and J. Dongarra, **"Investigating the Benefit of FP16-Enabled Mixed-Precision Solvers for Symmetric Positive Definite Matrices using GPUs,"** *International Conference on Computational Science (ICCS 2020)*, Amsterdam, Netherlands, Springer, Cham, June 2020.

Abdelfattah, A., H. Anzt, E. Boman, E. Carson, T. Cojean, J. Dongarra, M. Gates, T. Gruetzmacher, N. J. Higham, S. Li, et al., **"A Survey of Numerical Methods Utilizing Mixed Precision Arithmetic,"** *SLATE Working Notes*, no. 15, ICL-UT-20-08: University of Tennessee, July 2020.

Anzt, H., T. Cojean, C. Yen-Chen, J. Dongarra, G. Flegar, P. Nayak, S. Tomov, Y. M. Tsai, and W. Wang, **"Load-Balancing Sparse Matrix Vector Product Kernels on GPUs,"** *ACM Transactions on Parallel Computing*, vol. 7, issue 1, March 2020.

Anzt, H., T. Cojean, Y-C. Chen, F. Goebel, T. Gruetzmacher, P. Nayak, T. Ribizel, and Y-H. Tsai, **"Ginkgo: A High Performance Numerical Linear Algebra Library,"** *Journal of Open Source Software*, vol. 5, issue 52, August 2020.

Anzt, H., Y. M. Tsai, A. Abdelfattah, T. Cojean, and J. Dongarra, **"Evaluating the Performance of NVIDIA's A100 Ampere GPU for Sparse and Batched Computations,"** *2020 IEEE/ACM Workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems (PMBS)*: IEEE, November 2020.

Archibald, R., E. Chow, E. D'Azevedo, J. Dongarra, M. Eisenbach, R. Febbo, F. Lopez, D. Nichols, S. Tomov, K. Wong, et al., **"Integrating Deep Learning in Domain Sciences at Exascale,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-10: University of Tennessee, August 2020.

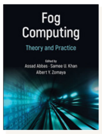
Ayala, A., S. Tomov, A. Haidar, and J. Dongarra, **"heFFTe: Highly Efficient FFT for Exascale,"** *International Conference on Computational Science (ICCS 2020)*, Amsterdam, Netherlands, June 2020.

Barry, D., A. Danalis, and H. Jagode, **"Effortless Monitoring of Arithmetic Intensity with PAPI's Counter Analysis Toolkit,"** *13th International Workshop on Parallel Tools for High Performance Computing*, Dresden, Germany, Springer International Publishing, September 2020.

Bathie, G., L. Marchal, Y. Robert, and S. Thibault, **"Revisiting Dynamic DAG Scheduling under Memory Constraints for Shared-Memory Platforms,"** *22nd Workshop on Advances in Parallel and Distributed Computational Models (APDCM 2020)*, New Orleans, LA, IEEE Computer Society Press, May 2020.

Beams, N., A. Abdelfattah, S. Tomov, J. Dongarra, T. Kolev, and Y. Dudouit, **"High-Order Finite Element Method using Standard and Device-Level Batch GEMM on GPUs,"** *2020 IEEE/ACM 11th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA)*: IEEE, November 2020.

Beck, M., T. Moore, P. Luszczek, and A. Danalis, **"Interoperable Convergence of Storage, Networking, and Computation,"** *Advances in Information and Communication: Proceedings of the 2019 Future of Information and Communication Conference (FICC)*, no. 2: Springer International Publishing, pp. 667-690, 2020.



Beckman, P., J. Dongarra, N. Ferrier, G. Fox, T. Moore, D. Reed, and M. Beck, **"Harnessing the Computing Continuum for Programming Our World,"** *Fog Computing: Theory and Practice*: John Wiley & Sons, Inc., 2020.

Benoit, A., V. Le Fèvre, P. Raghavan, Y. Robert, and H. Sun, **"Design and Comparison of Resilient Scheduling Heuristics for Parallel Jobs,"** *22nd Workshop on Advances in Parallel and Distributed Computational Models (APDCM 2020)*, New Orleans, LA, IEEE Computer Society Press, May 2020.

Brown, C., A. Abdelfattah, S. Tomov, and J. Dongarra, **"Design, Optimization, and Benchmarking of Dense Linear Algebra Algorithms on AMD GPUs,"** 2020 IEEE High Performance Extreme Computing Virtual Conference: IEEE, September 2020.

Brown, C., A. Abdelfattah, S. Tomov, and J. Dongarra, **"Design, Optimization, and Benchmarking of Dense Linear Algebra Algorithms on AMD GPUs,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-12: University of Tennessee, August 2020.

Cao, Q., Y. Pei, K. Akbudak, A. Mikhalev, G. Bosilca, H. Ltaief, D. Keyes, and J. Dongarra, **"Extreme-Scale Task-Based Cholesky Factorization Toward Climate and Weather Prediction Applications,"** *Platform for Advanced Scientific Computing Conference (PASC20)*, Geneva, Switzerland, ACM, June 2020.

Cao, Q., G. Bosilca, W. Wu, D. Zhong, A. Bouteiller, and J. Dongarra, **"Flexible Data Redistribution in a Task-Based Runtime System,"** *IEEE International Conference on Cluster Computing (Cluster 2020)*, Kobe, Japan, IEEE, September 2020.

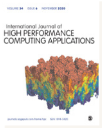
Demmel, J., J. Dongarra, J. Langou, J. Langou, P. Luszczek, and M. Mahoney, **"Prospectus for the Next LAPACK and ScaLAPACK Libraries: Basic ALgebra LIbraries for Sustainable Technology with Interdisciplinary Collaboration (BALLISTIC),"** *LAPACK Working Notes*, no. 297, ICL-UT-20-07: University of Tennessee.

Dongarra, J., **"Report on the Fujitsu Fugaku System,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-06: University of Tennessee, June 2020.

Dongarra, J., L. Grigori, and N. J. Higham, **"Numerical Algorithms for High-Performance Computational Science,"** *Philosophical Transactions of the Royal Society A*, vol. 378, issue 2166, 2020.

Dongarra, J., M. Gates, P. Luszczek, and S. Tomov, **"Translational Process: Mathematical Software Perspective,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-11, August 2020.

Du, Y., L. Marchal, G. Pallez, and Y. Robert, **"Robustness of the Young/Daly Formula for Stochastic Iterative Applications,"** *49th International Conference on Parallel Processing (ICPP 2020)*, Edmonton, AB, Canada, ACM Press, August 2020.



Farhan, M. Al, A. Abdelfattah, S. Tomov, M. Gates, D. Sukkari, A. Haidar, R. Rosenberg, and J. Dongarra, **"MAGMA Templates for Scalable Linear Algebra on Emerging Architectures,"** *The International Journal of High Performance Computing Applications*, vol. 34, issue 6, pp. 645-658, November 2020.

Gainaru, A., B. Goglin, V. Honoré, P. Raghavan, G. Pallez, P. Raghavan, Y. Robert, and H. Sun, **"Reservation and Checkpointing Strategies for Stochastic Jobs,"** *34th IEEE International Parallel and Distributed Processing Symposium (IPDPS 2020)*, New Orleans, LA, IEEE Computer Society Press, May 2020.

Gates, M., A. Charara, A. YarKhan, D. Sukkari, M. Al Farhan, and J. Dongarra, **"Performance Tuning SLATE,"** *SLATE Working Notes*, no. 14, ICL-UT-20-01: Innovative Computing Laboratory, University of Tennessee, January 2020.

Gates, M., A. Charara, J. Kurzak, A. YarKhan, M. Al Farhan, D. Sukkari, and J. Dongarra, **"SLATE Users' Guide,"** *SLATE Working Notes*, no. 10, ICL-UT-19-01: Innovative Computing Laboratory, University of Tennessee, July 2020.

Goebel, F., H. Anzt, T. Cojean, G. Flegar, and E. S. Quintana-Orti, **"Multiprecision Block-Jacobi for Iterative Triangular Solves,"** *European Conference on Parallel Processing (Euro-Par 2020)*: Springer, August 2020.

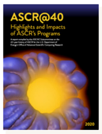
Haidar, A., H. Bayraktar, S. Tomov, J. Dongarra, and N. J. Higham, **"Mixed-Precision Solution of Linear Systems Using Accelerator-Based Computing,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-05: University of Tennessee, May 2020.

Haidar, A., H. Bayraktar, S. Tomov, J. Dongarra, and N. J. Higham, **"Mixed-Precision Iterative Refinement using Tensor Cores on GPUs to Accelerate Solution of Linear Systems,"** *Proceedings of the Royal Society A*, vol. 476, issue 2243, November 2020.

Han, L., L-C. Canon, J. Liu, Y. Robert, and F. Vivien, **"Improved Energy-Aware Strategies for Periodic Real-Time Tasks under Reliability Constraints,"** *40th IEEE Real-Time Systems Symposium (RTSS 2019)*, York, UK, IEEE Press, February 2020.

Han, L., Y. Gao, J. Liu, Y. Robert, and F. Vivien, **"Energy-Aware Strategies for Reliability-Oriented Real-Time Task Allocation on Heterogeneous Platforms,"** *49th International Conference on Parallel Processing (ICPP 2020)*, Edmonton, AB, Canada, ACM Press, 2020.

Hendrickson, B., P. Messina, B. Bland, J. Chen, P. Colella, E. Dart, J. Dongarra, T. Dunning, I. Foster, R. Gerber, et al., **"ASCR@40: Four Decades of Department of Energy Leadership in Advanced Scientific Computing Research,"** *Advanced Scientific Computing Advisory Committee (ASCAC)*, US Department of Energy, August 2020.



Hendrickson, B., P. Messina, B. Bland, J. Chen, P. Colella, E. Dart, J. Dongarra, T. Dunning, I. Foster, R. Gerber, et al., **"ASCR@40: Highlights and Impacts of ASCR's Programs,"** *US Department of Energy's Office of Advanced Scientific Computing Research*, June 2020.

Hori, A., K. Yoshinaga, T. Herault, A. Bouteiller, G. Bosilca, and Y. Ishikawa, **"Overhead of Using Spare Nodes,"** *The International Journal of High Performance Computing Applications*, February 2020.

Jagode, H., A. Danalis, and D. Genet, **"Roadmap for Refactoring Classic PAPI to PAPI++: Part II: Formulation of Roadmap Based on Survey Results,"** *PAPI++ Working Notes*, no. 2, ICL-UT-20-09: Innovative Computing Laboratory, University of Tennessee, July 2020.

Jagode, H., A. Danalis, and J. Dongarra, **"Formulation of Requirements for New PAPI++ Software Package: Part I: Survey Results,"** *PAPI++ Working Notes*, no. 1, ICL-UT-20-02: Innovative Computing Laboratory, University of Tennessee Knoxville, January 2020.

Kolev, T., P. Fischer, A. Abdelfattah, S. Ananthan, V. Barra, N. Beams, R. Bleile, J. Brown, R. Carson, J-S. Camier, et al., **"CEED ECP Milestone Report: Improve Performance and Capabilities of CEED-Enabled ECP Applications on Summit/Sierra,"** *ECP Milestone Reports*: Zenodo, May 2020.

Krzhizhanovskaya, V., G. Závodszky, M. Lees, J. Dongarra, P. Sloot, S. Brissos, and J. Teixeira, eds., **"Twenty Years of Computational Science,"** *International Conference on Computational Science (ICCS 2020)*, Amsterdam, Netherlands, June 2020.

Krzhizhanovskaya, V., G. Závodszky, M. Lees, J. Dongarra, P. Sloot, S. Brissos, and J. Teixeira, eds., **"Computational Science – ICCS 2020: 20th International Conference, Amsterdam, The Netherlands, June 3–5, 2020, Proceedings, Part I–VII,"** *Lecture Notes in Computer Science*, 1, no. 12137–12143: Springer International Publishing, June 2020.

Lindquist, N., P. Luszczek, and J. Dongarra, **"Replacing Pivoting in Distributed Gaussian Elimination with Randomized Techniques,"** *2020 IEEE/ACM 11th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA)*: IEEE, November 2020.

Lindquist, N., P. Luszczek, and J. Dongarra, **"Improving the Performance of the GMRES Method using Mixed-Precision Techniques,"** *Smoky Mountains Computational Sciences & Engineering Conference (SMC2020)*, August 2020.

Lopez, F., E. Chow, S. Tomov, and J. Dongarra, **"Asynchronous SGD for DNN Training on Shared-Memory Parallel Architectures,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-04: University of Tennessee, Knoxville, March 2020.

Lopez, F., and T. Mary, **"Mixed Precision LU Factorization on GPU Tensor Cores: Reducing Data Movement and Memory Footprint,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-13: University of Tennessee, September 2020.

Lopez, F., E. Chow, S. Tomov, and J. Dongarra, **"Asynchronous SGD for DNN Training on Shared-Memory Parallel Architectures,"** *Workshop on Scalable Deep Learning over Parallel And Distributed Infrastructures (ScaDL 2020)*, May 2020.

Losada, N., P. González, M. J. Martín, G. Bosilca, A. Bouteiller, and K. Teranishi, **"Fault Tolerance of MPI Applications in Exascale Systems: The ULFM Solution," Future Generation Computer Systems**, vol. 106, pp. 467-481, May 2020.

Lu, Y., I. Yamazaki, F. Ino, Y. Matsushita, S. Tomov, and J. Dongarra, **"Reducing the Amount of out-of-core Data Access for GPU-Accelerated Randomized SVD,"** *Concurrency and Computation: Practice and Experience*, April 2020.

Luo, X., W. Wu, G. Bosilca, Y. Pei, Q. Cao, T. Patinyasakdikul, D. Zhong, and J. Dongarra, **"HAN: A Hierarchical AutotuNed Collective Communication Framework,"** *IEEE Cluster Conference*, Kobe, Japan, Best Paper Award, IEEE Computer Society Press, September 2020.

Luszczek, P., Y. Tsai, N. Lindquist, H. Anzt, and J. Dongarra, **"Scalable Data Generation for Evaluating Mixed-Precision Solvers,"** *2020 IEEE High Performance Extreme Computing Conference (HPEC)*: IEEE, September 2020.

Pei, Y., Q. Cao, G. Bosilca, P. Luszczek, V. Eijkhout, and J. Dongarra, **"Communication Avoiding 2D Stencil Implementations over PaRSEC Task-Based Runtime,"** *2020 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, New Orleans, LA, IEEE, May 2020.

Slaughter, E., W. Wu, Y. Fu, L. Brandenburg, N. Garcia, W. Kautz, E. Marx, K. S. Morris, Q. Cao, G. Bosilca, et al., **"Task Bench: A Parameterized Benchmark for Evaluating Parallel Runtime Performance,"** *International Conference for High Performance Computing Networking, Storage, and Analysis (SC20)*, November 2020.

Tomov, S., A. Ayala, A. Haidar, and J. Dongarra, **"FFT-ECP API and High-Performance Library Prototype for 2-D and 3-D FFTs on Large-Scale Heterogeneous Systems with GPUs,"** *ECP Milestone Report*, no. FFT-ECP STML13-27: Innovative Computing Laboratory, University of Tennessee, January 2020.

Tsai, Y. M., T. Cojean, and H. Anzt, **"Sparse Linear Algebra on AMD and NVIDIA GPUs—The Race is On,"** *ISC High Performance*: Springer, June 2020.

Winkler, F., **"Redesigning PAPI's High-Level API,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-03: University of Tennessee, February 2020.

Wong, K., S. Tomov, and J. Dongarra, **"Project-Based Research and Training in High Performance Data Sciences, Data Analytics, and Machine Learning,"** *The Journal of Computational Science Education*, vol. 11, issue 1, pp. 36-44, January 2020.

Wyrzykowski, R., E. Deelman, J. Dongarra, and K. Karczewski, eds., **"Parallel Processing and Applied Mathematics: 13th International Conference, PPAM 2019, Bialystok, Poland, September 8–11, 2019, Revised Selected Papers, Part I–II,"** *Lecture Notes in Computer Science*, no. 12043–12044: Springer International Publishing, March 2020.

YarKhan, A., M. Al Farhan, D. Sukkari, M. Gates, and J. Dongarra, **"SLATE Performance Report: Updates to Cholesky and LU Factorizations,"** *Innovative Computing Laboratory Technical Report*, no. ICL-UT-20-14: University of Tennessee, October 2020.

Zaitsev, D., and P. Luszczek, **"Docker Container Based PaaS Cloud Computing Comprehensive Benchmarks using LAPACK,"** *Computer Modeling and Intelligent Systems CMIS-2020*, Zaporizhzhoa, March 2020.

Zhong, D., Q. Cao, G. Bosilca, and J. Dongarra, **"Using Advanced Vector Extensions AVX-512 for MPI Reduction,"** *EuroMPI/USA '20: 27th European MPI Users' Group Meeting*, Austin, TX, September 2020.

Zhong, D., P. Shamis, Q. Cao, G. Bosilca, and J. Dongarra, **"Using Arm Scalable Vector Extension to Optimize Open MPI,"** *20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID 2020)*, Melbourne, Australia, IEEE/ACM, May 2020.

EVENTS

IN-PERSON EVENTS

JANUARY 13-17 **Master in High Performance Computing Program (MHPC)** TRIESTE, ITALY

FEBRUARY 2-7 **Exascale Computing Project Annual Meeting (ECP 2020)** HOUSTON, TX

FEBRUARY 12-15 **SIAM Conference on Parallel Processing for Scientific Computing (SIAM PP)** SEATTLE, WA

FEBRUARY 18-21 **MPI Forum** PORTLAND, OR

FEBRUARY 25-27 **NSF Workshop on Smart Cyberinfrastructure** CRYSTAL CITY, VA

MARCH 1-6 **Dagstuhl Seminar on Resiliency in Numerical Algorithm Design for Extreme Scale Simulations** WADERN, GERMANY

VIRTUAL EVENTS

MARCH 10-11 **ECP Industry Council Meeting**

MARCH 23-26 **NVIDIA GPU Technology Conference (GTC2020)**

MAY 7-8 **Variable Precision in Mathematical and Scientific Computing Workshop**

MAY 18-22 **21st IEEE International Workshop on Parallel and Distributed Scientific and Engineering Computing (PDSEC 2020)**

JUNE 10 **Big Data and Extreme-Scale Computing (BDEC2)**

JUNE 22-25 **ISC High Performance 2020**

JULY 6-17 **SIAM Annual Meeting 2020**

JULY 10 **Big Data and Extreme-Scale Computing (BDEC2)**

JULY 27-31 **Practice and Experience in Advanced Research Computing (PEARC20)**

AUGUST 11-12 **CEED 4th Annual Meeting**

AUGUST 26-28 **Smoky Mountains Computational Sciences & Engineering Conference (SMC2020)**

SEPTEMBER 1 **Big Data and Extreme-Scale Computing (BDEC2)**

SEPTEMBER 8-10 **11th JLESC Workshop**

SEPTEMBER 14-17 **IEEE Cluster 2020**

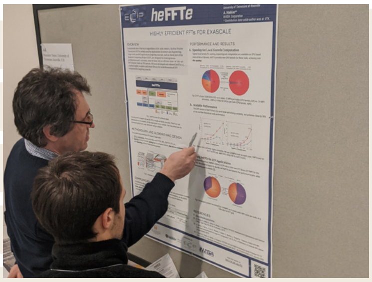
SEPTEMBER 21-24 **EuroMPI 2020**

SEPTEMBER 21-25 **IEEE High Performance Extreme Computing Conference (HPEC 2020)**

SEPTEMBER 28-30 **MPI Forum**

OCTOBER 15 **Big Data and Extreme-Scale Computing (BDEC2)**

NOVEMBER 9-19 **International Conference for High Performance Computing Networking, Storage, and Analysis (SC20)**

SIAM PP20

The Society for Industrial and Applied Mathematics 2020 Conference on Parallel Processing for Scientific Computing (SIAM PP20) was held in Seattle, Washington on February 12–15, 2020, making it the last major conference to be held in person this year. The SIAM conference series is one of ICL's most heavily attended events—second only to SC—and SIAM PP20 was in keeping with this tradition.

In total, 13 research scientists, post docs, and students from all three of ICL's research groups made the journey to present material on: mixed-precision developments in the MAGMA library; sustainability strategies in SLATE; sparse matrix vector product advancements in PEEKS; evaluations of new systems on the HPL, HPCC, HPCG, and HPL-AI benchmarks; updates to the ULFM extension; and optimizations to the PaRSEC runtime.

SC20

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

As with many workshops and conferences this year, SC20 moved to an all-digital platform for 2020. Expanding the conference dates an additional week to accommodate this new medium, SC20 ran from November 9 to November 19. Five computational science research centers from the University of Tennessee—the Bredesen Center, the Global Computing Laboratory, the Innovative Computing Laboratory, the Joint Institute for Computational Sciences, and Chattanooga's SimCenter—represented the university by anchoring the University of Tennessee's virtual booth. As usual, ICL had a significant presence at SC, with faculty, research staff, and students giving talks, hosting tutorials, presenting papers, and leading "Birds of a Feather" sessions through SC20's web conferencing platform.

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 & ACADEMIC PARTNERSHIPS

- Argonne National Laboratory
- Ames 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
- Old Dominion University
- 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
- The University of New Mexico
- USC University of Southern California
- The University of Utah
- University of Washington
- Virginia Tech

Global Computing Laboratory

GCLab
Dr. Michela Taufer

In June 2018, Prof. Michela Taufer joined UTK's Department of Electrical Engineering and Computer Science and relocated the Global Computing Laboratory to the Min H. Kao Electrical Engineering and Computer Science Building. The Global Computing Laboratory focuses on various aspects of HPC and scientific computing—including computational chemistry and chemical engineering, pharmaceutical sciences, seismology, and mathematics.

INDUSTRY PARTNERSHIPS

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 Fédérale de Lausanne
LAUSANNE, SWITZERLAND

ETH Zürich
ZÜRICH, 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

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 (LIP6)
PARIS, FRANCE

Moscow State University
MOSCOW, RUSSIA

National Institute of Advanced Industrial Science and Technology TSUKUBA, JAPAN

Parallel and HPC Application Software Exchange TSUKUBA, JAPAN

Prometeus GmbH
MANNHEIM, GERMANY

Regionales Rechenzentrum Erlangen (RRZE)
ERLANGEN, GERMANY

RIKEN
WAKŌ, JAPAN

Rutherford Appleton Laboratory
OXFORD, ENGLAND

Soongsil University
SEOUL, SOUTH KOREA

Technische Universität Wien
VIENNA, AUSTRIA

Technische Universität Dresden
DRESDEN, GERMANY

Tokyo Institute of Technology
TOKYO, JAPAN

Umeå University
UMEÅ, SWEDEN

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

LEADERSHIP

THE UNIVERSITY OF TENNESSEE KNOXVILLE
CENTER FOR INFORMATION TECHNOLOGY RESEARCH

Center for Information Technology Research

The Center for Information Technology Research (CITR) was established in 2001 to drive the growth and development of leading-edge information technology research at 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.

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

IGMCS

Interdisciplinary Graduate Minor in Computational Science

Addressing the need for a curriculum in computational science, CITR worked with faculty and administrators from several departments and colleges in 2007 to help establish a university-wide program that supports advanced degree concentrations in this growing field. With the Interdisciplinary Graduate Minor in Computational Science (IGMCS), students pursuing advanced degrees in a variety of fields of science and engineering can enhance 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.

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

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, BDEC2 staged four international workshops and four virtual meetings to enable research communities in a wide range of disciplines to converge on a common platform to meet the daunting challenges of computing in the era of exascale and big data.

In 2020, CITR was instrumental in organizing and staging four virtual BDEC2 workshops. Along with Jack Dongarra, several members of ICL's CITR staff—including Terry Moore and Joan Snoderly—played essential roles in making these BDEC2 workshops a major success. A summary of the BDEC2 effort—in the form of a comprehensive report—is underway, and publication is expected in early 2021.

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

Joint Laboratory for Extreme Scale Computing

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, the Jülich Supercomputing Center, and the RIKEN Center for Computational Science.

FIND OUT MORE AT https://jlesc.github.io/

PEOPLE

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.

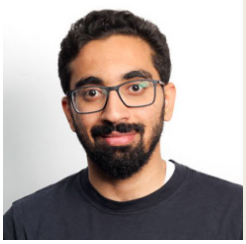
In 2020, three of ICL's Graduate Research Assistants earned their PhDs from UTK under the guidance and mentorship of Prof. Dongarra and ICL's research scientists. Congratulations to:

David Eberius
Now at Oak Ridge National Laboratory
"Providing Insight into the Performance of Distributed Applications Through Low-Level Metrics"

Xi Luo
Now at Stanford University
"Optimization of MPI Collective Communication Operations"

Yaohung Tsai
Now at Facebook
"Mixed Precision Numerical Linear Algebra Algorithms: Integer Arithmetic Based LU Factorization and Iterative Refinement for Symmetric Eigenvalue Problem"

Ahmad Abdelfattah
RESEARCH SCIENTIST II



Mohammed Al Farhan
POST DOCTORAL RESEARCH ASSOCIATE

Hartwig Anzt
CONSULTANT

Alan Ayala
POST DOCTORAL RESEARCH ASSOCIATE

Paul Bagwell
SOFTWARE ENGINEER

Daniel Barry
GRADUATE RESEARCH ASSISTANT

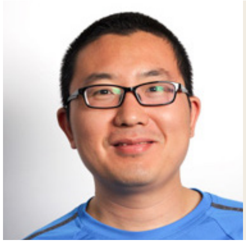
John Batson
INFORMATION SPECIALIST

Natalie Beams
RESEARCH SCIENTIST I

George Bosilca
RESEARCH ASSISTANT PROFESSOR

Aurelien Bouteiller
RESEARCH ASSISTANT PROFESSOR

Cade Brown
UNDERGRADUATE RESEARCH ASSISTANT



Qinglei Cao
GRADUATE RESEARCH ASSISTANT

Earl Carr
PROGRAM ADMINISTRATOR

Tony Castaldo
RESEARCH SCIENTIST II

Sebastien Cayrols
POST DOCTORAL RESEARCH ASSOCIATE

Sam Crawford
TECHNICAL SERVICES LEADER

Anthony Danalis
RESEARCH ASSISTANT PROFESSOR

Jack Dongarra
DIRECTOR AND DISTINGUISHED PROFESSOR

Teresa Finchum
ADMINISTRATIVE SPECIALIST II

Mark Gates
RESEARCH ASSISTANT PROFESSOR

Damien Genet
RESEARCH SCIENTIST I

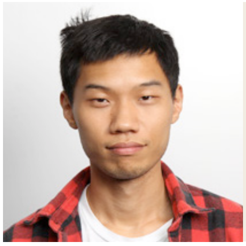
Thomas Herault
RESEARCH ASSISTANT PROFESSOR

Heike Jagode
RESEARCH ASSISTANT PROFESSOR

Julie Langou
RESEARCH LEADER

Tracy Lee
FINANCIAL SPECIALIST I

Jiali Li
GRADUATE RESEARCH ASSISTANT



Yicheng Li
GRADUATE RESEARCH ASSISTANT

Neil Lindquist
GRADUATE RESEARCH ASSISTANT

Florent Lopez
POST DOCTORAL RESEARCH ASSOCIATE

Nuria Losada
POST DOCTORAL RESEARCH ASSOCIATE

Piotr Luszczek
RESEARCH ASSISTANT PROFESSOR

Maksim Melnichenko
GRADUATE RESEARCH ASSISTANT

Terry Moore
ASSOCIATE DIRECTOR

Phil Mucci
CONSULTANT

Yu Pei
GRADUATE RESEARCH ASSISTANT

Deborah Penchoff
ASSOCIATE DIRECTOR

Gerald Ragghianti
RESEARCH LEADER

David Rogers
IT SPECIALIST III

Joseph Schuchart
RESEARCH ASSOCIATE II

Daniel Schultz
GRADUATE RESEARCH ASSISTANT

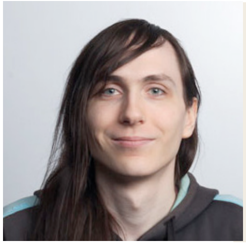
Wissam Sid Lakhdar
RESEARCH SCIENTIST I

Leighanne Sisk
ADMINISTRATIVE SPECIALIST I

Joan Snoderly
ASSISTANT DIRECTOR

Dalal Sukkari
POST DOCTORAL RESEARCH ASSOCIATE

Stanimire Tomov
RESEARCH ASSISTANT PROFESSOR



Aaron Welch
NON-UT STUDENT ASSISTANT

Frank Winkler
CONSULTANT

Asim YarKhan
RESEARCH SCIENTIST II

Dong Zhong
GRADUATE RESEARCH ASSISTANT

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.

Richard Archibald
Oak Ridge National Laboratory

Richard Barrett
Sandia National Laboratories

Micah Beck
University of Tennessee

Sunita Chandrasekaran
University of Delaware

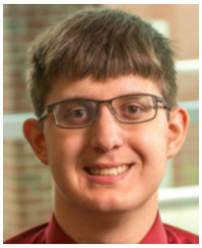
Chris Gropp
University of Tennessee

Azzam Haidar
NVIDIA

Hidehiko Hasegawa
University of Tsukuba

Clark Hathaway
Global Computing Laboratory

Jeff Larkin
NVIDIA



Ian Lumsden
Global Computing Laboratory

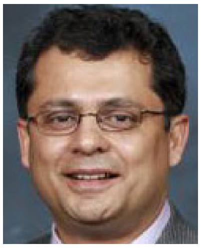
Sebastian Mobo
Global Computing Laboratory

Paula Olaya
Global Computing Laboratory

Yves Robert
ENS Lyon

Marcus Ritter
Technical University of Darmstadt

Chad Steed
Oak Ridge National Laboratory



Sudip Seal
Oak Ridge National Laboratory

Anthony Skjellum
UT Chattanooga SimCenter

Nigel Tan
Global Computing Laboratory

Michael Wyatt
Global Computing Laboratory

Felix Wolf
Technical University of Darmstadt

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

Ajay Kalhan
Balajee Kannan
Madhuri Kasam
Kiran Kumar Kasichayanula
Ajay Katta
David Katz
Joshua Kelly
Supriya Kilambi
Youngbae Kim
Myung Ho Kim
Jenya Kirshtein
Cindy Knisley
Michael Kolatis
Anara Kozhokanova
Moritz Kreutzer
Chandra Krintz
Tilman Kuestner
Jakub Kurzak
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
Xiang Li
Weiran Li
Yinan Li
Chaoyang Liu
Kevin London
Matt Longley
Florent Lopez
Hatem Ltaief
Yuechao Lu
Daniel Lucio
Richard Luczak
Xi Luo
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
Shirley Moore
Keith 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
Thananon Patinyasakdikul
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
Tracy Rafferty
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
Yuan Tang
Chunyan Tang
Yusuke Tanimura
Keita Teranishi
Dan Terpstra
Joe Thomas
John Thurman
Francoise Tisseur
Jude Toth
Bernard Tourancheau
Yaohung “Mike” Tsai
Volodymyr Turchenko
Lauren Vaca
Phil Vaccaro
Sathish Vadhiyar
Robert van de Geijn
Chad Vawter
Eugene Vecharynski
Scott Venckus
Antoine Vernois
Reed Wade
Michael Walters
Mike Waltz
Robert Waltz
Xiaoyang Wang
Bob Ward
Jerzy Wasniewski
Vince Weaver
Scott Wells
David West
R. Clint Whaley
Jody Whisnant
James White
Scotti Whitmire
Susan Wo
Felix Wolf
Stephen Wood
Jiayi Wu
Wei Wu
Panruo Wu
Qiu Xia
Tinghua Xu
Ichitaro Yamazaki
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 over 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:

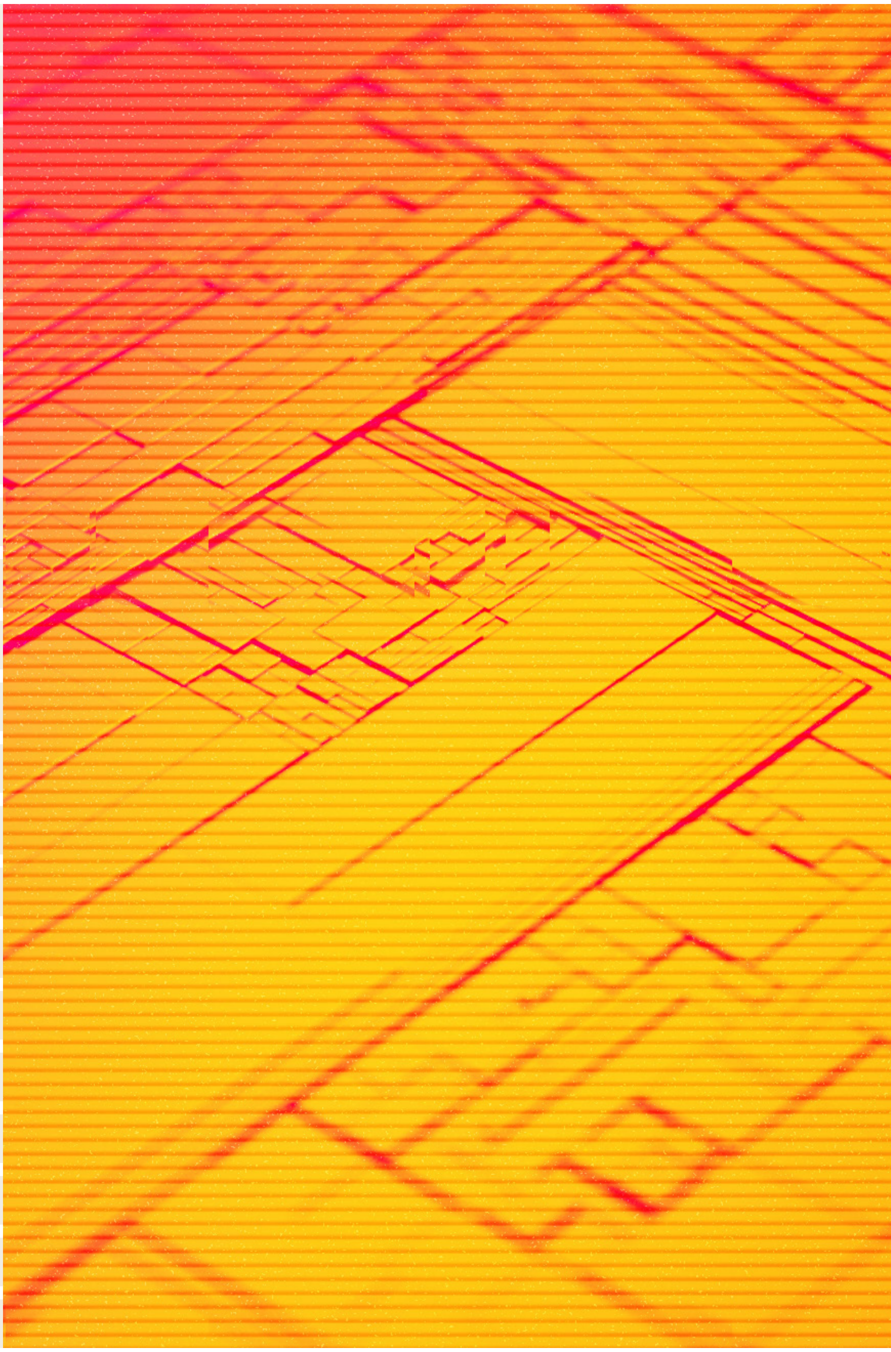
AFOSR Air Force Office of Scientific Research
U.S. Department of Energy Office of Science
U.S. Department of Energy NNSA National Nuclear Security Administration
NSF National Science Foundation
Oak Ridge National Laboratory

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

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.

The University of Tennessee Knoxville
Center for Information Technology Research



ICL INNOVATIVE COMPUTING LABORATORY

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