

Codesign for Extreme Heterogeneity: Integrating Custom Hardware With Commodity Computing Technology to Support Next-Generation HPC Converged Workloads

James A. Ang , Kevin J. Barker, Draguna L. Vrabie , and Gokcen Kestor, *Pacific Northwest National Laboratory, Richland, WA, 99354, USA*

The future of high-performance computing (HPC) will be driven by the convergence of physical simulation, artificial intelligence, machine learning, and data science computing capabilities. While computational performance gains afforded by technology scaling, as predicted by Moore's Law, have enabled large-scale HPC system design and deployment using commodity CPU and GPU processing components, emerging technologies will be required to effectively support such converged workloads. These emerging technologies will integrate commodity computing components with custom processing and networking accelerators into ever-more heterogeneous architectures resulting in a diverse ecosystem of industry technology developers, university, and U.S. Government researchers. In this article, we describe efforts at the U.S. Department of Energy's Pacific Northwest National Laboratory to construct an end-to-end codesign framework that lays a groundwork for such an ecosystem, including notable outcomes, remaining challenges, and future opportunities.

Accelerating scientific discovery in fields like chemistry, materials science, and biology, or controlling complex engineered systems like the power grid, demands *converged* computing capabilities that integrate high-performance computing (HPC) simulation, artificial intelligence, and machine learning (AI/ML), and data science. Current commodity processors are optimized for either CPU-based general-purpose computation and transaction processing, or GPU-based high value commercial tasks like artificial neural networks for AI/ML. Since the 1990s, large-scale HPC platforms, such as those constructed for the

U.S. Department of Energy (DOE), have successfully integrated commodity processors, culminating in 2022 with the current highest performing supercomputer, the *Frontier* system at Oak Ridge National Laboratory. This achievement builds on nearly a decade of DOE investment in exascale hardware, software technologies, and applications development. Looking ahead, supercomputers built using only commodity processors will be suboptimal for diverse converged workloads. Fortunately, emerging technologies (including heterogeneous silicon-in-package [SIP] capabilities for integration of commodity CPU and GPU cores with custom chiplet accelerators, design automation and synthesis tools, performant reconfigurable and heterogeneous architectures, and runtime software tools) are making end-to-end codesign a viable path to achieve the required computing capabilities. Here, we describe efforts to construct an end-to-end codesign framework that uses converged application workloads to develop

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see <https://creativecommons.org/licenses/by/4.0/>
 Digital Object Identifier 10.1109/MIC.2022.3217423
 Date of publication 2 November 2022; date of current version 3 February 2023.

heterogeneous computing design concepts that leverage these emerging technologies, and perform test and evaluation on heterogeneous and reconfigurable architecture testbeds. We close with a discussion of notable outcomes, remaining challenges, future opportunities, and next steps.

MOTIVATION

The increasing complexities of mission and safety-critical infrastructure systems such as the smart power grid require advanced computational technologies to address system specification, design, verification, and validation requirements. Disruptions to infrastructure systems can result in changes in the observability of the states, the measurement structure, and data properties (e.g., as affected by adversarial cyber influences), the decision space (e.g., loss of controllability), or the uncertainty. Similar challenges associated with limited observations, sparse datasets, sensing design, graphical data representations, inaccurate and incomplete models of systems and processes, and uncertainty are also found in chemistry, catalysis, biology, materials design, and other scientific discovery domains, e.g.^{1,2,3}

Concurrent with these challenges, the rapid expansion of digital data collection devices, the ever-increasing volume, and diversity of data that are generated and stored, and the rapid development of methods and tools for data analysis offer transformational opportunities for scientific discovery. Advances in data analytics technologies and computing capabilities are outpacing nominal progress rates in engineering and science. These advances hold the promise of an unprecedented remake of the approaches and workflows of scientific enquiry and offer opportunities to advance novel application domains, such as cyber-physical security, where traditional model-driven approaches have limited impact.⁴ Workflows for scientific discovery or control of complex engineering systems will increasingly depend on converged applications. Rather than treat these workflows as a sequence of serial tasks, there is a great opportunity to integrate computational and experimental science with AI/ML and uncertainty quantification to increase confidence in the predictive capabilities of HPC applications. In the long run, realizing this opportunity will define a new way to think about HPC performance metrics and create improvements that can support and sustain an *annual doubling* of computing performance.

Pacific Northwest National Laboratory (PNNL) established the Data-Model Convergence (DMC) Initiative in 2019 to 1) advance computational domain

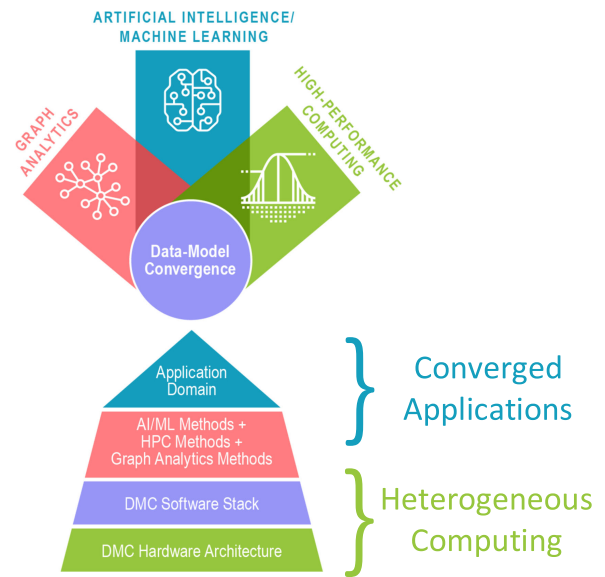


FIGURE 1. DMC vision: integration of computing modalities supported by a common system stack.

science through *converged* workloads that integrate traditional simulations, AI/ML models, and data sciences, and 2) develop heterogeneous computing capabilities that accelerate converged application workloads through multidisciplinary codesign collaborations. We aim to address shortcomings in the current state of the art in converged applications—that are supported with separate system stacks, with different programming models, tools, and even computer architectures to support distinct needs. As a result, convergence of these modalities is hampered, with resulting sequential computational workflows assembled in an *ad hoc* process by domain scientists introducing significant cognitive and computational overheads. The *status quo* for scientific workflows requires large amounts of associated data that must be converted from one representation to another and transferred between separate systems using vastly different programming paradigms and cognitive models.

DMC's goal is to *seamlessly integrate* these computing paradigms to obtain orders of magnitude improvement in computational efficiency using custom architectural concepts, to enable fundamentally new and transformational science at unprecedented scale (Figure 1). Our vision is to help shape a technology road map that defines long-term goals capable of sustaining over 1,000,000× performance improvements and will drive computing technology innovation for the next two decades—a new Moore's Law. We will achieve these longer term goals by developing codesign skills

that can be reapplied on future heterogeneous computing design concepts that will integrate commodity processors with non-von Neumann architectures like neuromorphic or quantum computing accelerators. While metrics of improvement today center on performance and energy efficiency, future work could add metrics pertaining to cybersecurity, resilience, and others.

CODESIGNING THE NEXT GENERATION OF COMPUTING SYSTEMS

Hardware/software *codesign*, the concept that architectures, algorithms, and applications are designed *in concert* with one another, provides an opportunity to extend system performance scaling beyond the limits of Moore's Law device scaling. While this concept is not new, emerging technologies and capabilities are finally making codesign feasible and driving its adoption, while enabling computational scientists to achieve more effective use of increasingly heterogeneous architectures.⁵

To address these challenges, PNNL is investing in foundational technologies in both application and computing system spaces, leveraging a long history in multidisciplinary domain science, simulation models, graph analytics, system software, compiler toolchains built using open-source frameworks such as LLVM⁶ and MLIR,⁷ design automation, high-level synthesis, advanced architecture testbeds, and more recent expertise in physics-informed deep neural models, and differentiable optimization. Our efforts have been focused in two research thrust areas.

The first thrust, *Converged Applications*, develops diverse application-driven workloads that integrate HPC modeling and simulation codes, data analytics on structured domain data, and the most recent AI/ML advances in deep learning. A prototypical example of such an application is the Active Learning Framework illustrated in Figure 2. This application workflow integrates molecular dynamics simulations used to generate datasets to train a machine learning model of energy potentials. The resulting domain-specific applications lead to transformation across diverse research communities, demonstrating the general applicability and impact of converged computing workloads on a broad cross section of science and engineering domains.

The second thrust area, *Heterogeneous Computing*, includes computer science activities that focus on novel hardware conceptual design and system software development. Emerging open-source design automation and synthesis tools, together with open-source hardware

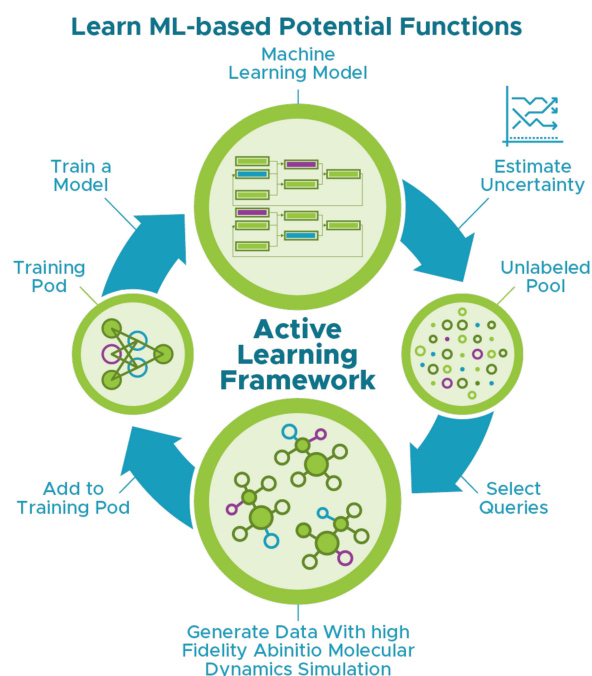


FIGURE 2. Notional diagram of a converged application workflow that integrates molecular dynamics simulations used for generating datasets to inform the training of a machine learning model with energy potentials in an active learning framework.

standards and performant reconfigurable computational substrates, are bringing hardware into the codesign loop. The *rapidly shrinking timescale* from kernel identification and analysis to hardware layout is making it feasible to prototype purpose-designed hardware to accelerate key kernels in converged applications. Through “open innovation” collaborations between academia, industry, and national laboratories and driven by open source tools, the codesign loop spanning workflow analysis, design space exploration, design automation, and synthesis can be fully realized. A key challenge remains to “close the loop” with workflows: application-level software drives requirements on architectural specification, while architectural capabilities should affect algorithmic choices at the higher software levels. For example in Figure 2, the heterogeneous computing thrust has developed design automation and synthesis tools to seamlessly generate hardware acceleration specifications from software kernels, memory profiling, and analysis tools to optimize data layout, and compiler frameworks to enable high-performance code generation targeting diverse heterogeneous computing elements.

Deep collaboration across these thrust areas is key to success in codesign. Our codesign process is designed to address two key issues impeding the

development of converged application workflows spanning both thrust areas: hardware designs unable to efficiently execute such workflows (due to characteristics such as irregularity or sparsity) and disparate software ecosystems leading to inefficient couplings between workflow components. Our codesign approach is developing system software tools that support multiple programming languages and library ecosystems using a common compilation framework while generating optimized code for numerous computational back-ends, some of which may be custom-designed using our design automation and synthesis tools. *Proxy workflows* capturing elements from physical simulation, data analytics, and machine learning into an integrated workflow distill full-scale applications into composable components and are effective tools for communication between domain computational scientists and system architects. These are covered in more detail below, while the development of proxy workflows is itself a challenge and an area of ongoing research.⁸

CONVERGED APPLICATIONS

High-fidelity modeling of complex dynamic systems has inherent gaps. For instance, physics-based simulations may require solving equations that contain unknown parameters that need to be estimated with measurement data. Similarly, developing highly accurate data-driven models requires extensive and high-quality datasets, which are not always available, especially in scientific domains. When left unfilled, these gaps result in models that have poor predictive accuracy. The perception of ML models as a black box, ignoring existing domain knowledge and requiring massive datasets, often discourages their application in scientific domains. The adoption and use of scientific ML is still a very active research area. Recent work led by PNNL on physics-informed ML⁹ demonstrates that integrating physics laws as prior knowledge in the architecture and learning methods of deep learning models increases model accuracy and requires smaller training datasets. Generalized approaches to embedding domain knowledge into machine learning models are a primary goal toward scalable machine reasoning.

The converged applications thrust investigates new theory and practice needed for the integration of scientific knowledge—either in structural or functional forms—with AI/ML-based descriptive (i.e., discovery), predictive (i.e., forecast), and prescriptive (i.e., optimization and control) analytics. This will reduce data requirements and accelerate model training while preserving learning ability and predictive accuracy. As a result, domain-aware models could be used as surrogates for costly high-fidelity simulations, thereby

significantly increasing research productivity without compromising scientific rigor. The research effort spanned several application areas, including power grid operation, molecular dynamics, and biological systems, selected based on PNNL's mission areas, their need for DMC computing capability, and the availability of relevant data, software, and subject matter expertise. This new capability will enable scientific discovery and scalable machine reasoning to define new performance measures for DOE applications.

The broad challenges associated with making decisions and enabling discovery under uncertainty have been explored at the convergence of data/graph analytics, HPC, and AI/ML by algorithmic and software development in the following research areas:

- ▶ *Embedding of domain knowledge in data-driven models*¹⁰: The goal is to develop scientific and technological principles for domain-aware ML. We focus on 1) knowledge representations for ML, including research into topics such as functional and structural knowledge representations, domain-aware latent spaces, knowledge across multiple scales, representation semantics, and bias; 2) integration of domain knowledge (e.g., hard or soft constraints, governing equations, physical laws, and model structure) into different ML regimes focusing on unsupervised, supervised, and semisupervised learning; and 3) data-driven scientific discovery (e.g., the discovery of new physics from models developed using incomplete physics).
- ▶ *Accelerate solutions of simulated complex dynamic systems*¹¹: The dynamic evolution of infrastructure systems is described by large-scale, hybrid (e.g., continuous and discrete), stiff, and differential-algebraic equations. The effects of contingencies are traditionally modeled by changing the equations' structure or parameters. The recent work by Nandanoori et al.¹¹ introduced a machine learning framework that considers both the event-based nature of cascading failures and transient behavior that emerges in the order of seconds in the power grid. This allows the development of predictive models that can robustly predict the outcomes of failures by real-time inference at the edge with more than three orders of magnitude speedups than by high-fidelity simulation. This provides a stepping stone for further algorithmic advances that can be executed in a distributed manner to predict the emergent transient behavior. The accelerated and accurate prediction will then inform decision algorithms to prevent, stop, and provide robust contingency recovery.

- › *Optimal decisions and control for dynamic systems are represented by scientific ML¹²*: We developed new methods to reduce the time to decision by leveraging domain-aware ML and novel hardware and software technologies developed by DMC. The emphasis is on 1) leveraging domain knowledge to learn from limited labeled data, focusing on semi-supervised learning, synthetic data generation, and active learning for optimal experiment design; 2) differentiable optimization methods that offer convergence and stability guarantees for learned decision policies; and 3) scalable meta-learning and automatic ML, focusing on the use of domain knowledge to accelerate the screening and identification of optimal model parameters and the synthesis of ML workflows for specific applications.
- › *Hybridized algorithms for multiagent decision systems with underlying graphical topology¹³*: This can be accomplished by integrating data-informed optimization, dynamic system decomposition methods, hierarchical ML models and algorithms that leverage infrastructure graph architecture knowledge, and data-driven control algorithms, such as reinforcement learning. The design of high-assurance learning-based systems and optimization-based verification of learning systems raise further research directions.

Attaining the above objectives requires a synergistic holistic codesign approach with the heterogeneous computing thrust area.

HETEROGENEOUS COMPUTING

The converged workflows targeted by DMC are diverse in their makeup and requirements, and commodity accelerators such as GPUs are suboptimal for tasks such as data analytics, graph algorithms, and other forms of memory-centric computing. However, effective use of custom heterogeneous architectures requires extensive programming and runtime software support to meet current and future domain requirements. Key science drivers, including the requirement to reliably store and process data generated by instruments such as advanced light sources have ushered in a new era of data analytics and data-driven science and demand ever-increasing computing performance and memory storage. At the same time, the new generation of researchers are used to the productivity inherent in high-level tools, languages, and programming models (e.g., Python, TensorFlow, and PyTorch in the DL domain) that remove the burden of low-level software

development and architectural awareness. While “high performance” has traditionally been achieved by low-level, *ad hoc* programming solutions, “portability and productivity” is the domain of high-level, domain-specific languages. Bridging the gap between the productivity of high-level programming environments and the performance of low level, architecture-specific solutions for future systems is a key objective of DMC.

The DMC codesign strategy for heterogeneous systems is based on several key principles. First, explore tradeoffs and causal relationships in the early stages of system design to achieve the expected performance and efficiency. Second, study full converged applications and applications workflows in a heterogeneous environment, where customized architectural concepts are integrated with general purpose CPUs and GPUs. Third, develop novel hardware concepts using an agile flow that translates algorithmic specifications into hardware models and designs. Fourth, support domain scientist usage of novel hardware accelerators without needing to know all the low-level architectural details. DMC, therefore, tackles the codesign and programming of extremely heterogeneous systems at all levels of the hardware and software stack, providing a truly holistic codesign environment for HPC, data analytics, and AI/ML converged applications.

Specifically, DMC is developing fundamental technologies in the areas of

- › *Compilers and Languages¹⁴*: Domain scientists often prefer to use expressive high-level, domain-specific languages that match the algorithmic description of their applications. However, emerging architectures often lack full software support and only target one or two domains (mostly AI/ML). Manually mapping M programming environments to N architectures is a daunting task that needs to be repeated for each domain language/architecture pair. Instead, DMC is developing compiler technologies that implement the principle of “write once, run everywhere.” The DMC compiler (COMET) is an MLIR-based compiler that converts different language frontends (Python, COMET Domain Specific Language, Rust) to a unified IR, hence different components of the converged applications can be implemented using different languages. The unified IR retains the application semantics information and is then optimized and lowered to architecture-specific IRs (e.g., CPU, GPUs, FPGA, AI engines).
- › *Runtime and system software¹⁵*: Complementary to code generation is execution of computational tasks on available heterogeneous devices and

orchestration of data movement. While computer vendors provide a runtime system that manages computation, data transfer, and command execution for their architecture (e.g., NVIDIA CUDA runtime), a few standards have emerged to abstract from application-specific implementations. However, most of these standards do not meet DMC requirements for converged applications and application workflows. Additionally, current runtimes do not often offer the possibility of integrating novel architectural models or simulation into the software ecosystem. DMC is developing technologies that facilitate integration of custom heterogeneous architectures in the system and provide a seamless programming environment for current and future extremely heterogeneous systems. The DMC Minos Computing Library (MCL)¹⁵ runtime manages computing and memory resources and is capable of exploiting data locality on heterogeneous devices. The runtime also manages communication and data/control dependencies among the tasks of the converged applications, even if those tasks belong to different applications.

- *Software-defined and reconfigurable architectures*¹⁶: Customized processors offer an efficient way of computing at the fraction of the time and power of general-purpose solutions. However, developing highly specialized architectures generally requires great time and effort and incurs wasted area and power if (some part of) the application cannot leverage such devices. Instead, DMC is developing novel capabilities to produce architectural models from high-level software specifications. Not only is DMC extending traditional high-level synthesis flows to include high-level programming languages but is also developing architectural models for coarse-grained reconfigurable architectures that can mitigate the drawbacks of application-defined architectures by reconfiguring the architecture to suit application's needs.

When combined, the technologies and capabilities developed in DMC will define a new codesign flow and an integrated programming environment to support converged applications on heterogeneous systems.

HETEROGENEOUS AND RECONFIGURABLE TESTBEDS

A crucial component in validating our codesign approach and resulting software and hardware are testbed platforms. While testbeds have long been employed to evaluate approaches to system and software design,

within a codesign framework they take on an additional role. The testbeds used by DMC include reconfigurable hardware to enable rapid prototyping of advanced architectural concepts and testing of design automation, synthesis, and compiler toolchains. Such platforms are also extremely heterogeneous, including processor types such as hard cores, AI/ML accelerators, and reconfigurable logic, making them excellent proving grounds for our system and application software targeting heterogeneous processing capabilities.

DMC's efforts benefit from PNNL's Center for Advanced Technology Evaluation (CENATE)¹⁷ project, funded by the ASCR program since 2016. CENATE's role has been to evaluate early technologies and quantify their potential on future system design using metrics of performance, power efficiency, and security. Current CENATE testbed systems are used to explore heterogeneity across architecture types and reconfigurable data-flow architectures. Our latest example and a key platform for the development of DMC software and architecture concepts is the *Junction* cluster, a first-of-its-kind, mid-size (50 node) cluster, with nodes that incorporate multicore AMD CPUs; AMD Instinct GPUs; Xilinx Versal Adaptive Compute Acceleration Platform (ACAP) accelerators containing ARM cores, AI accelerator cores, and programmable logic; and two network interface cards, including Xilinx Alveo SN1000 SmartNICs. Junction is used to demonstrate compiler-level optimizations, high-level synthesis tool chains, and customized accelerator architectures developed using our codesign approach.

FUTURE OPPORTUNITIES AND NEXT STEPS

We believe the future of computing across all scales will have several characteristics in common, including increasing heterogeneity and customization. This is driven by several factors, including the slowing of general-purpose computing performance due to the slowdown and subsequent ending of device scaling as predicted by Moore's law, the need for performance (e.g., latency or throughput) that is not achievable by general-purpose computing devices, and a need for energy efficiency.

The first casualty of the slowing of Moore's law is the declining role of the general-purpose, multicore CPU processor versus the advent of specialized accelerators. Many commercial efforts are focused on the development of special-purpose accelerators as stand-alone chips (e.g., GPUs, FPGAs, and AI engines). In the DMC Initiative, we are pursuing the holistic codesign of customized kernel accelerators, but we do not

see these as standalone accelerators. Instead, these can be thought of as hardware implementations of intrinsic kernel functions that are designed as separate intellectual property (IP) blocks that can be manufactured as chiplets and integrated into silicon-in-package (SIP) designs with CPU and GPU cores. In this context, the DMC Initiative assumes *computer and processor vendors will be receptive to implementing and integrating architectural innovations they did not design in-house*. The reason for this optimism is the growth in two open-source System-on-Chip (SoC) ecosystems (ARM and RISC-V) in which hardware architects are accustomed to the idea of integration of IP blocks that may be licensed from external entities.

A critical prerequisite to achieve this vision of customized and heterogeneous computing and enabling holistic codesign is multidisciplinary collaboration. Our selection of converged applications to drive the codesign of the heterogeneous computing software stack and hardware architecture concepts will make the resulting prioritizations and design tradeoffs generally useful for other converged applications. This assumption is based on the observation that most technical applications are not computation bound, so codesigning improvements in converged application performance are generalizable to improvements in data access and data movement performance.

ACKNOWLEDGMENTS

The authors acknowledge the efforts of the DMC project PIs. Within the constraints of this overview paper, we can only scratch the surface of accomplishments of their project teams. A full listing of the PIs, their teams, and DMC publications and software repositories is available on the initiative website.¹⁸ Finally the authors also thank Mark Raugas, an initial member of the DMC leadership team, whose energy and enthusiasm helped us organize the codesign collaborations among projects across the DMC Initiative. This work was supported by the Data-Model Convergence Initiative at Pacific Northwest National Laboratory. It was conducted under the Laboratory Directed Research and Development Program at PNNL, a Multiprogram National Laboratory Operated by Battelle for the U.S. Department of Energy under Contract DE-AC05-76RL01830.

REFERENCES

1. P. Pernot, "The long road to calibrated prediction uncertainty in computational chemistry," *J. Chem. Phys.*, vol. 27, 2022, Art. no. 114109.
2. E. A. Pidko, "Toward the balance between reductionist and systems approaches in computational catalysis: Model vs. method accuracy for the description of catalytic systems," *ACS Catalysis*, vol. 7, no. 7, pp. 4230–4234, 2017.
3. S. R. Spurgeon et al., "Towards data-driven next-generation transmission electron microscopy," *Nature Mater.*, vol. 20, pp. 274–279, 2021.
4. H. Lin, Z. Yan, Y. Chen, and L. Zhang, "A survey on network security-related data collection technologies," *IEEE Access*, vol. 6, pp. 18345–18365, 2018.
5. J. A. Ang et al., "Report on basic research needs for reimagining codesign for advanced scientific computing: Unlocking transformational opportunities for future computing systems for science," United States Dept. Energy, Office Sci. Tech. Inform. (OSTI), WA, DC, USA, Tech Rep. 1822199, 2021, doi: [10.2172/1822199](https://doi.org/10.2172/1822199).
6. C. Lattner and V. Adve, "LLVM: A compilation framework for lifelong program analysis and transformation," in *Proc. Int. Symp. Code Gener. Optim.*, 2004, pp. 75–86.
7. C. Lattner et al., "MLIR: Scaling compiler infrastructure for domain specific computation," in *Proc. IEEE/ACM Int. Symp. Code Gener. Optim.*, 2021, pp. 2–14.
8. S. Ghosh, N. R. Tallent, M. Minutoli, M. Halappanavar, R. Peri, and A. Kalyanaraman, "Single-node partitioned-memory for huge graph analytics: Cost and performance trade-offs," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, 2021, pp. 1–14.
9. P. Stinis, "Enforcing constraints for time series prediction in supervised, unsupervised, and reinforcement learning," in *Proc. AAAI Spring Symp. Combining Artif. Intell. Mach. Learn. Phys. Sci., CEUR Workshop*, 2020, Art. no. 2587.
10. S. Guan, H. Ma, S. Choudhury, and Y. Wu, "GEDet: Detecting erroneous nodes with a few examples," in *Proc. 47th Int. Conf. Very Large Data Bases*, 2021, pp. 2875–2878.
11. S. Nandanoori et al., "Graph neural network and Koopman models for learning networked dynamics: A comparative study on power grid transients prediction," *IEEE Access*, vol. 10, pp. 32337–32349, 2022.
12. E. King et al., "Koopman-based differentiable predictive control for the dynamics-aware economic dispatch problem," in *Proc. Amer. Control Conf.*, 2022, pp. 2194–2201.
13. S. P. Nandanoori, S. Pal, S. Sinha, S. Kundu, K. Agarwal, and S. Choudhury, "Data-driven distributed learning of multi-agent systems: A Koopman operator approach," in *Proc. 60th IEEE Conf. Decis. Control*, 2021, pp. 5059–5066.
14. E. Mutlu et al., "COMET: A domain-specific compilation of high-performance computational chemistry," in *Proc. 33rd Int. Workshop Lang. Compilers Parallel Comput.*, 2020, pp. 87–103.

15. R. Gioiosa, B. Mutlu, S. Lee, J. S. Vetter, G. Picierro, and M. Cesati, "The minos computing library: Efficient parallel programming for extremely heterogeneous systems," in *Proc. 13th Workshop Gen. Purpose Process. Using GPUs*, 2020, pp. 1–10.
16. C. Tan, C. Xie, A. Li, K. J. Barker, and A. Tumeo, "OpenCGR: An open-source unified framework for modeling, testing, and evaluating CGRAs," in *Proc. IEEE 38th Int. Conf. Comput. Des.*, 2020, pp. 381–388.
17. N. R. Tallent et. al., "Assessing advanced technology in CENATE," in *Proc. IEEE Int. Conf. Netw., Archit., Storage*, 2016, pp. 1–2.
18. 2020. [Online]. Available: <https://www.pnnl.gov/projects/dmc>


JAMES A. ANG serves as PNNL's sector lead for the DOE Office of Science Advanced Scientific Computing Research (ASCR) Program, Richland, WA, 99354, USA. Ang received his Ph.D. degree in mechanical engineering from the University of California, Berkeley. Contact him at ang@pnnl.gov.

KEVIN J. BARKER serves as the group lead for PNNL's High Performance Computing Group, leads the ASCR Center for Advanced Technology Evaluation program, Richland, WA, 99354, USA, and as a chief scientist for the Data Model Convergence

laboratory initiative. His research interests include hardware/software codesign, performance analysis and optimization, and advanced architecture concepts and testbeds. Barker received his Ph.D. degree in computer science from the College of William and Mary in Virginia. Contact him at kevin.barker@pnnl.gov.

DRAGUNA L. VRABIE is chief data scientist in the Data Sciences and Machine Intelligence Group where she serves as team leader for the Autonomous Learning and Reasoning team. Her current focus is on deep learning methodologies and algorithms for design and operation of high-performance, cyber-physical systems. Vrabie holds a doctorate in electrical engineering from the University of Texas at Arlington. Contact her at draguna.vrabie@pnnl.gov.


GOKCEN KESTOR is a senior research scientist at Pacific Northwest National Laboratory, Richland, WA, 99354, USA. Her research interests include HPC system software, extreme-scale application and system characterization, developing methodologies and tools to understand the impact of hardware, system software, and application characteristics on key HPC system design factors: performance, power consumption, and portability. Contact her at gokcen.kestor@pnnl.gov.




www.computer.org/cga

IEEE Computer Graphics and Applications bridges the theory and practice of computer graphics. Subscribe to CG&A and

- stay current on the latest tools and applications and gain invaluable practical and research knowledge,
- discover cutting-edge applications and learn more about the latest techniques, and
- benefit from CG&A's active and connected editorial board.

 **IEEE COMPUTER SOCIETY**

 **IEEE**