ASC™

Advanced Simulation and Computing

CO-DESIGN STRATEGY

NNSA
National Nuclear Security Administration

Notice: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information apparatus, product, or process disclosed, or represents that its use would not infringe on privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

SAND 2015-9821R
Issued by Sandia National Laboratories for NNSA's
Office of Advanced Simulation and Computing and
Institutional Research and Development Programs, NA-114

For more information, contact: Thuc Hoang at
Thuc.Hoang@nnsa.doe.gov

Advanced Simulation and Computing
Co-Design Strategy

Office of Advanced Simulation and Computing
and Institutional Research and Development Programs, NA-114

James A. Ang
Thuc T. Hoang
Suzanne M. Kelly
Allen McPherson
Rob Neely

January 2016

A Publication of the Office of Advanced Simulation and Computing
and Institutional Research and Development,
NNSA Defense Programs

ACKNOWLEDGMENTS

This ASC Co-design Strategy Plan was a group effort with invaluable writing and editing contribution from Bert Still (LLNL), Rob Hoekstra (SNL), Fred Johnson (Leidos), Sriram Swaminarayan (LANL) and Alexis Blanc (NNSA). Additional review assistance came from David Womble (SNL) and Ken Alvin (SNL), with publication expertise provided by Scott Kiefer (SNL) and Doug Prout (SNL). Lastly, this document would not be possible without my co-authors' tremendous involvement and unwavering support: Sue Kelly (SNL), Jim Ang (SNL), Al McPherson (LANL) and Rob Neely (LLNL).

Thuc T. Hoang, Program Manager
Office of Advanced Simulation and Computing
and Institutional Research and Development Programs
Office of Defense Programs
National Nuclear Security Administration

CONTENTS

Acknowledgements ii
Foreword iv
Executive Summary v
Introduction 1
Co-design Overview 2
Co-design Principles 3
Technical Challenges 4
The Co-design Continuum 6
Scope 7
Engagements 8
Co-design Mechanisms 9
Proxies as the Tool of Co-design 12
Conclusion 13
References 14
Acronyms 14
Appendix: ASC National Work Breakdown Structure 15

FOREWORD

The Advanced Simulation and Computing (ASC) Program is essential to the mission of the National Nuclear Security Administration (NNSA)'s Office of Defense Programs (DP), which is to maintain the safety, security, and effectiveness of the U.S. nuclear weapons stockpile without nuclear testing. ASC's success depends on the ability to provide the next generation of stockpile stewards with the simulation tools that accurately and efficiently model the complex physics involved in a nuclear weapon explosion, along with providing engineering analysis of the weapons delivery and complex accident scenario environments. High-performance computer simulations inform critical DP stockpile stewardship decisions through detailed behavior prediction, uncertainty quantification, and validation through comparison with comprehensive experimental results and historical tests.

To carry out its mission, ASC must overcome demanding challenges in the coming years. Specifically, we are faced with potential application performance challenges as future ASC platforms will incorporate many-core and heterogeneous computing architectures to solve the problem of power efficiency. Moreover, ASC must position the NNSA national laboratories such that they can continue to deliver on NNSA's current nuclear security mission needs, while adapting to radical technology changes, and continue running the most demanding applications necessary to support weapons certification and the research of underlying weapon sciences.

The simulation environment of the future will be transformed by new computer architectures and new programming techniques will need to be developed to capitalize on these advances. Within this context, ASC applications must transition to the new simulation environment or risk stagnation. The difficulty of successfully transitioning the code base should not be underestimated. Our national security mission requires ever-increasing simulation capabilities to continue the move from confirmative to predictive capabilities. To that end, co-design is emerging as an important strategy to provide ASC a system-level, holistic approach in our attempt to optimize the utilization of next-generation, advanced technologies to meet the stockpile mission requirements.

This ASC Co-design Strategy document discusses the technological challenges in depth and describes the initial steps toward a new era of technical opportunities for predictive simulation. Partnerships with industry, other DOE and U.S. agencies, and academia are a key part of our approach. Agility and adaptability are essential to ensure that future generations of ASC simulation capabilities and resources will continue to underpin our nation's nuclear deterrent and resolve to forgo nuclear testing.

Douglas Wade, Acting Director
Office of Advanced Simulation and Computing
and Institutional Research and Development Programs
Office of Defense Programs
National Nuclear Security Administration

EXECUTIVE SUMMARY

This ASC Co-design Strategy lays out the full continuum and components of the co-design process, based on what we have experienced thus far and what we need to meet the program's mission of providing high performance computing (HPC) and simulation capabilities for NNSA to carry out its stockpile stewardship responsibility.

This document starts by presenting key co-design principles that serve to guide the ASC program toward taking the necessary steps to support the transition of the ASC application code base to the new programming environment brought about by incoming advanced architectures. This transition must be executed, as efficiently as possible, with portability, performance, and usability in mind. These principles are:

1. **Mission:** Enable nuclear weapons codes to support the Stockpile Stewardship Program (SSP) and the Annual Assessment Review (AAR) to certify that the U.S. stockpile is safe, secure, and effective through efficient utilization of advanced computational resources.
2. **Vendor Engagement:** Partner with the U.S. computer industry to influence vendor hardware and software capabilities, and gain a deeper understanding of architectural trends and their implications for the nuclear weapons code base.
3. **Research:** Develop a focused research agenda among designers of hardware, applications, and programming environments to tackle the interdependent challenges that next-generation, extreme-scale platforms present to ASC applications.
4. **Partnerships:** Leverage the strengths of vendors, academia, and the national laboratories in pursuit of a sustainable HPC ecosystem.

The primary driver for co-design is the paradigm shift in computer architectures that is necessary to continue advancing realized performance improvements under constraints of portability, power, reliability, and usability. That shift, in turn, puts significant pressure on the application teams responsible for delivering on the ASC mission and necessitates a coordinated effort among hardware, system software, and application developers. This triad of coordination summarizes the concept that is co-design.

The degree of influence of the co-design process spans a continuum that encompasses reactive, proactive, and transformative co-design. The level of co-design is dependent upon both the level of resources available and the amount of time ASC is given to execute it properly. The recently established ASC Advanced Technology Development and Mitigation (ATDM) program element aims to complement the DOE FastForward and DesignForward research efforts, and various foundational software stack research activities within DOE, by providing a concrete driver in the form of "clean slate" applications to drive co-design requirements. Co-design also plays a critical role in moving the current production applications in ASC's Integrated Codes program element toward a more efficient execution on next-generation architectures. After careful assessment, successful co-design results will be inserted into existing production applications in the form of new algorithms and implementations. This will be done based on lessons learned through the use of proxy applications and the design and implementation of abstraction layers that provide access to additional levels of parallelism and portability across a wide range of architectures. While the software/hardware computing industry is critically important and central to co-design, the co-design participation space has evolved to include a broad range of contributors, including the NNSA and DOE national labs, our sister HPC program in DOE Office of Science (SC), academia, and international partners. Additionally, we will collaborate with other U.S. government agencies to collectively formulate complementary strategies within the framework of the whole-of-government effort for the National Strategic Computing Initiative (NSCI) Executive Order.

The mechanisms by which the ASC co-design effort is executed are discussed in this document and include:

- open proxy applications and proxy architectures,
- performance simulation and abstract machine models,
- academic engagements,
- hack-a-thons bridging application requirements with early research endeavors,
- Centers of Excellence and non-Recurring Engineering contracts with system vendors,
- standards committees for MPI, OpenMP®, and
- emerging software standards to complement and enhance the programming environment.

This document is intended to provide the reader with a high-level view of the ASC co-design strategy as it currently exists, and will be a living document as new engagement mechanisms and priorities evolve.

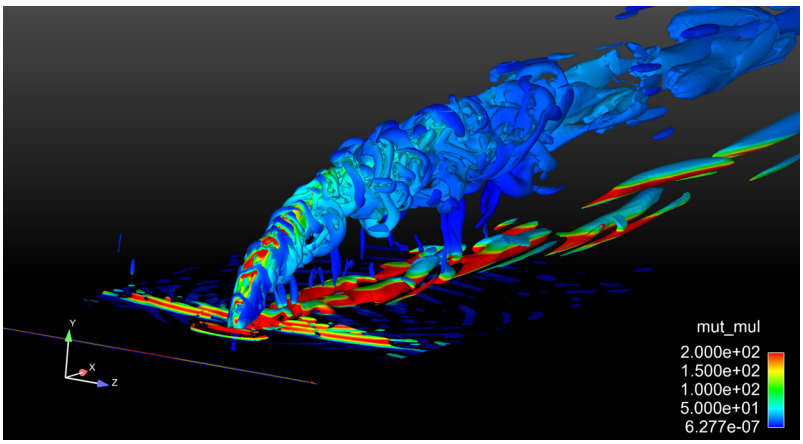
INTRODUCTION

The ASC program is approaching twenty years of operation in providing HPC and simulation tools to the NNSA SSP. Throughout the years, both the complexity and capability of simulation software have grown immensely and kept pace with the approximately 20,000-fold increase in peak computing speeds available on ASC platforms. The growth has been in large part due to a relatively stable message passing programming model utilizing large-scale, distributed memory commodity processors. To meet the ever-increasing simulation demands coming from the weapons program, ASC has a well aligned strategy of procuring systems that provide the bulk of the computing resources for the program in a cost-effective manner, via the Commodity Technology System (CTS) acquisitions, while also tracking with industry on the advanced technology front via the Advanced Technology System (ATS) deployments [1].

The introduction of many-core processor architectures in HPC systems in recent years has destabilized and caused turmoil to the ASC simulation and computing environment. The extreme-scale, or exascale, challenges caused by the substantial architectural shifts are recognized and well documented by the HPC community [2-7]. The ASC program recognizes that the simulation environment of the future will be transformed by new computer architectures. New programming paradigms (typically referred to as "programming models") will be needed to take advantage of these architectural advances. Transitioning the ASC integrated codes to a more modern, efficient, and effective code base will be a complex process requiring several years to complete before high performance and productivity can be achieved as a steady state. Since mid-FY14, ASC has implemented its application transition strategy by standing up the ATDM program element and embarking on the co-design journey to ensure its application code base has a successful transition to the exascale computing paradigm that will arrive in the next decade. By exploring technology-driven approaches to new programming algorithms in consultation with vendors and computer scientists, developers working on ATDM next-generation application codes will gain important experience to help identify potential paths forward for ASC's current production codes. Also, using the wide variety of co-design mechanisms, developers working on current ASC codes in the other program elements (i.e., Integrated Codes, Physics and Engineering Models, and Validation and Verification) will gain insight into the degrees of modification required to keep the existing codes viable on future hardware.

ASC is incorporating co-design practices into the program to address incompatibilities between new hardware technologies and weapons application code structures that will reduce application performance.

Co-design is expected to have a large role in enabling ASC codes to perform well on the advanced computer architectures being introduced over the next decade. The goal is to integrate co-design practices into the ASC program to achieve the best possible outcome for addressing impending performance issues of ASC codes. The codes embody the best understanding of the physics and engineering of the nuclear stockpile and are an important contributor to achieving SSP missions.



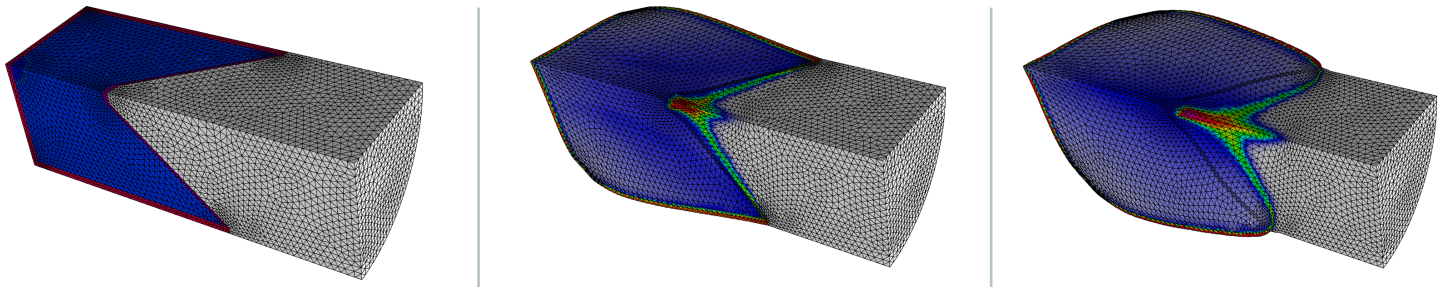
Iso-surfaces of Q-Criterion colored by eddy-viscosity from a Hybrid RANS-LES simulation of a jet-in-crossflow flow field. The image shows the complex interacting vortex structures in the flow field. The algorithms were influenced by information on new hardware changes.

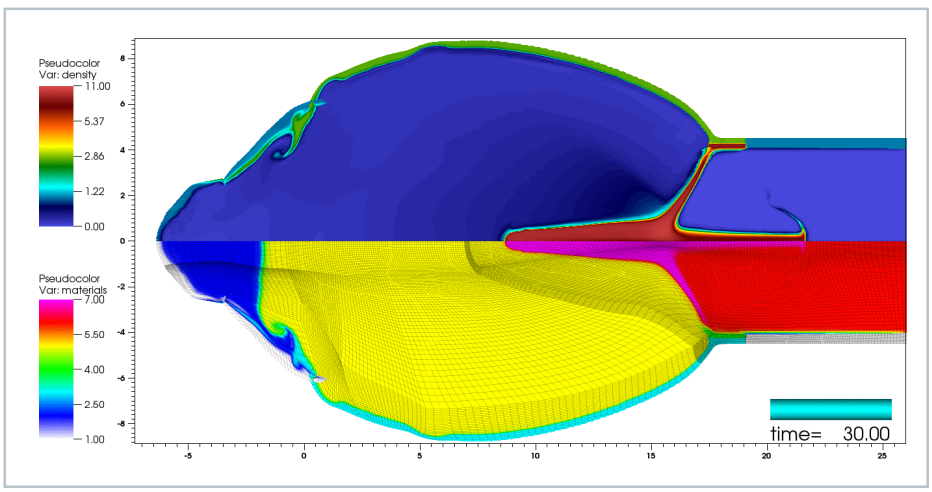
CO-DESIGN OVERVIEW

Co-design is a process already implemented by DOE Office of Science and NNSA in recent years to engage with computer industry in response to the HPC ecosystem being impacted by technology changes. This is further complicated by DOE's aggressive exascale computing goals within the next decade. Co-design, at its core, embodies the concept that HPC systems should be produced with application software influencing hardware design tradeoffs, while also recognizing that applications and supporting software must be developed in anticipation of hardware changes. The co-design process is not unique to HPC and has been used in many disciplines to develop designs that encourage all participants to find solutions within the context of the total system. Each stakeholder may have a different set of requirements and a different time horizon. For the NNSA some teams, such as those maintaining large and complex nuclear weapons applications with hundreds of developer-years of effort, desire a mechanism for incremental adaptation of their codes, also known as Engineering and Physics Integrated Codes (EPICs). Based on the enormous ASC code investment for the past two decades, this community is a key player, but a design based solely on their requirements could miss opportunities for better future performance and most likely delay needed code base changes to ensure long-term code viability. Likewise, hardware vendors must consider their broader market base and the long timeline for product development; their solution may be too far afield for the application developers' needs. The co-design process encourages all parties ranging from the system designers, computer architects, application software and tools developers, facilities, etc., to jointly design and optimize the system specifications via open communication and collaboration.

Co-design is used in many industries to bring together disparate stakeholders to consider the entire solution space. Communication between hardware vendors and ASC application developers will be facilitated by co-design processes.

Several years of experience has taught us that co-design requires a significant amount of planning and effort to be effective. In addition, because both hardware and ASC application software require many years between concept, design, and production, co-design must be executed as early as possible in a symbiotic partnership with users, application developers, vendors, and technology researchers.





These are simulation images made with a high-order finite element code (BLAST), which has been developed and optimized using co-design principles. The ASC program has added HE capability to the (research) hydrocode in order to do the shaped charge calculations.

CO-DESIGN PRINCIPLES

The ASC Program's Co-design Strategy is based on four principles outlined below, from which success will be defined. Foremost of these is the ASC mission of providing the necessary computational capabilities and maintaining the nuclear weapons codes to be continually viable for NNSA to sustain a safe, secure, and effective nuclear deterrent through the application of science, technology, engineering, and manufacturing. A properly scoped applied research program is needed to preserve the investment in the ASC codes, as significant, disruptive architectural changes are occurring. When necessary, the codes must adapt to the hardware and software architectural changes uncovered through co-design interactions with the computer vendors. Since co-design is only effective if it is a two-way communication, the code teams are required to identify and prioritize architectural changes and new hardware/software capabilities that help preserve the investment ASC has made. Partnerships that leverage the respective strengths of vendors, academia, and the national labs are a proven model of success for co-design processes and research activities.

1. **Mission:** Enable nuclear weapons codes to support the Stockpile Stewardship Program and the Annual Assessment Review (AAR) to certify that the U.S. stockpile is safe, secure, and effective through efficient utilization of advanced computational resources.

   Integrated multi-physics, multi-scale simulations on powerful ASC HPC systems are key to supporting the annual assessment of the U.S. stockpile systems; resolving Significant Finding Investigations (SFIs); accomplishing upcoming Life Extension Program (LEP) goals with safety and surety features; and for supporting qualification of hostile environments, safety calculations of abnormal environments, and gravity and reentry simulations. The LEPs, in particular, depend on simulation from initial planning through final certification. Moreover, the demands on ASC capabilities grow as the nuclear stockpile moves further from the nuclear test base, either through aging or through LEPs. The ultimate measure of success of the ASC co-design effort is the deployment of increasingly predictive simulations with quantified uncertainties in order to continuously advance our ability to certify the stockpile without nuclear testing.

2. **Vendor Engagement:** Partner with U.S. computer industry to influence vendor hardware and software capabilities and gain a deeper understanding of architectural trends and their implications for the nuclear weapons code base.

   In-depth, technical communications with computer vendors allow ASC scientists to consider current application characteristics as well as guide new applications, algorithms, and system software through a deep understanding of long-term, emerging hardware trends. Since system software is typically provided by HPC vendors, it is an important area for co-design with a goal of bridging our EPIC code base from its current user environment to a new setting that could eventually be equally performant. By coherently expressing current and future code performance requirements to vendors, ASC has a tremendous opportunity to impact future hardware and software features.

3. **Research:** Develop a focused research agenda among designers of hardware, applications, and programming environments to tackle the interdependent challenges that next-generation, extreme-scale platforms present to ASC applications.

   The broader HPC community is working to maximize the value of next-generation platforms for scientific applications. Certain areas, such as programming models, are critical to ASC applications, while others such as domain specific languages and cyber research are better addressed by other research communities. ASC must confront software and other challenges associated with advances in hardware architectures and identify which should be researched internally and which can be adopted from the broader community. As ASC's applied research and development (R&D) products are assessed to be "technology transfer ready", they will be incorporated into the implementation of a robust production environment upon which mission-critical applications can be deployed.

4. **Partnerships:** Leverage the strengths of academia, other U.S. agencies, and the DOE national laboratories in pursuit of a sustainable HPC ecosystem.

   While ASC has some mission-related requirements that necessitate an internal applied R&D program

among the NNSA laboratories, there are numerous technical areas in which ASC will definitely benefit from partnerships with external communities. The ASC program must expand our engagements beyond our usual partners to include other U.S. government agencies, academia, other leading non-US computer vendors, and key international partners to ensure performance portability of our codes. ASC must maintain a close working relationship with this broader community to ensure that features essential to our needs are continuously identified, investigated, integrated, and supported in future products. Many algorithmic and system-level software challenges are best studied and solved together by our laboratory researchers and external collaborators with the appropriate expertise

TECHNICAL CHALLENGES

Computer chip vendors continue to increase transistor density at Moore's Law rates, which specifies a doubling of transistors in a dense integrated circuit approximately every two years. This increase in transistor density, combined with increased system sizes through scalable interconnects, has been the primary driver for two decades of accelerated growth in HPC capabilities. Historically, the accompanying increases in Central Processing Unit (CPU) clock frequencies translated directly to improved application performance without any changes to the code. Due to heat dissipation and other physical device constraints, frequency is no longer increasing along with transistor density. Similarly, reductions in power consumption per transistor are slowing, and threaten to stall as CMOS technology reaches physical limitations. These factors, shown in Figure 1, have stagnated the development of higher-performance fat cores, which have higher CPU frequencies, a high tolerance to memory latency, and high memory capacity such as those present in many current commodity-based servers.

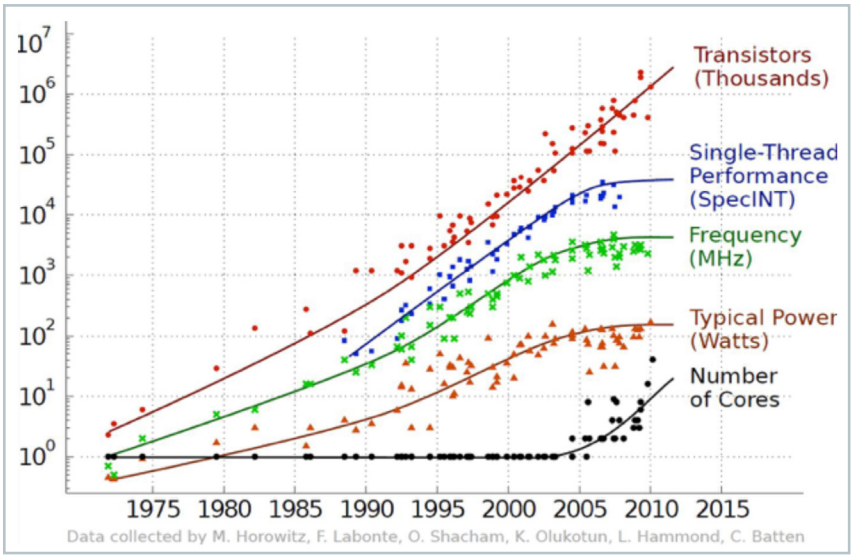


Figure 1: Traditional sources for performance improvements have stagnated

To sustain the performance increase trend and to use the massive amount of transistors available, vendors have switched to a *many-core* approach, leading to rapidly increasing numbers of weaker, or lightweight, cores that have less memory capacity and modest CPU frequencies in exchange for a greater number of cores in a CPU socket. The many-core architecture trades per-core application performance for increased scalability. Applications must take advantage of both increased parallelism via message passing, as well as explicit fine-grained concurrency within the node. With aggregate system power constraints and memory capacity limiting the upside potential of relying solely on adding more nodes to the system for performance gains, the requirement for applications to expose much greater fine-grained concurrency is a key requirement for future architectures. This paradigm shift, from fat-core to a many-core approach, places extreme demands on programming models used in the codes because the application developers must now consider how to effectively utilize the massive parallelism inherent in these many-core architectures which did not exist before.

An added challenge is that aggregate memory bandwidth (how quickly memory can be fed to the processing unit) is not keeping pace with CPU demands, and memory latencies (the time for a memory request to be fulfilled) have stagnated. Since computations require data upon which to compute, if the data is not available to a core the CPU must pause and wait for that data to be delivered. This in turn places a new emphasis on optimizing algorithms based on their use of the memory subsystem over traditional concerns of optimizing use of the floating point units.

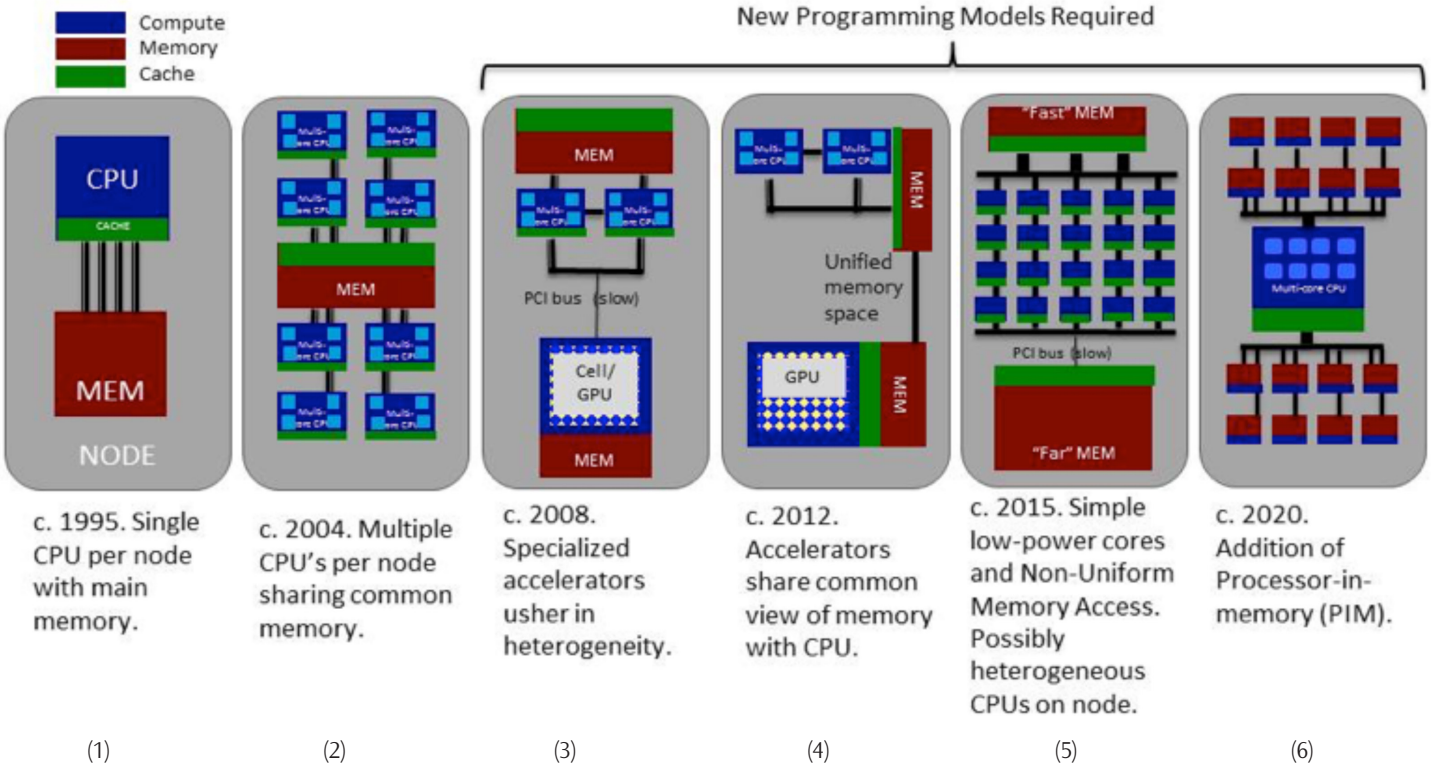


Figure 2: Node architectures are increasingly more complex and require new programming models to extract their performance potential

Indeed, current industry trends driven by big data and data analytics workloads reflect a move from a compute-centric to data-centric view, of which the determining factor on performance is also data motion. As traditional HPC simulations shift toward a more data-centric approach, co-design will help augment industry trends to the mutual benefit of these two communities as their requirements increasingly overlap.

The path forward is unclear due to multiple, competing new architectures and programming models. New constraints, such as power efficiency, introduce additional challenges.

Adding to these complex challenges, a variety of competing approaches to new architectures and programming models have led to even greater uncertainty. The traditional computing paradigm is conceptually represented by a compute element connected to a large and fast memory subsystem, as seen in the first diagram of Figure 2. Performance was dictated by how efficiently computational operations (FLOPs) could be performed. Even as architectures changed to incorporate multi-core processors (Diagram 2, Figure 2), this paradigm held. However, this paradigm is now shifting. With machines like ASC Sequoia (many-core Blue Gene®/Q) and ASC Trinity (hybrid CPU and Xeon Phi™), performance is determined more by the ability to move data through the system than by the ability to compute on that data.

The four architectures shown in Diagrams 3-6 of Figure 2 are viable alternatives for the next decade. In every case, there are multiple distinct memory subsystems and each architecture carries out calculations in very different ways. The Graphics Processor Unit (GPU) accelerators in Diagrams 3 and 4 utilize simpler throughput-optimized cores, with traditional latency optimized multicore processors handling those parts of the calculation that are poorly suited to the GPUs. The many-core processor in Diagram 5 uses energy efficient cores which are similar to, but much less capable than, those of traditional CPUs. The architecture in Diagram 6 inverts the traditional paradigm by sending the computation to the data. In each instance, the approach to programming the system is quite different. Consequently, achieving performance portability across all architectures is critical.

Finally, the need for additional resiliency features further complicates the design space. As systems grow in size and complexity, the number of component failures will increase proportionally. Resiliency features at all levels of hardware and software will be needed to shield the impact of component failures from the applications.

These new architectures are driving a need for new programming models that can access the dramatically increased parallelism within the node. Access to this parallelism requires lightweight processes, heavy use of shared memory and vectorization, and minimization of data movement throughout the complex hierarchies of data storage. While fine-grained parallel programming interfaces have been available in the community for quite some time (e.g., OpenMP®, Pthreads, and various GPU interfaces), they have not harnessed the increasingly fine-grained hardware features in a way that has allowed our current codes to exploit them effectively without resorting to non-portable, vendor-specific solutions. This drives a key component of our co-design strategy, which is to *spur the development of new programming models that will allow us to abstract hardware details away from the programmers*, thus insulating them from the variety and uncertainty of hardware features. These programming models will allow easy access to parallelism and performance on the new architectures while creating some degree of portability when moving between different architectures. Without this approach, our large code base of applications would eventually be unsustainable in the future.

THE CO-DESIGN CONTINUUM

NNSA has been applying a co-design methodology for over five years. Our use of co-design has evolved along a continuum—from the early reactive approach, to the current proactive methodology, and towards our proposed transformative path. These three approaches of co-design have varying characteristics as described below.

Reactive Co-design: This was the early, more traditional, avenue down the co-design path. Many, if not most, HPC systems (both hardware and system architecture) were closely tied to the computer vendors' technology roadmaps. Application developers may have had a couple of years to anticipate what was coming, but their application development efforts were mainly focused on porting efforts to exploit the capabilities that were already established in the vendors' plans. Some opportunity for leverage could be found through investments in system software and algorithms to serve as a bridge between the pending hardware and system architecture and the much larger legacy application code base.

Proactive Co-design: This is the approach that our current ATDM projects are following. The emphasis is to leverage the DOE FastForward and DesignForward investments to influence directions and priorities in existing commodity computing technology roadmaps. The feedback from our current industry partners is that these investments already have the expected influence over future architectures to the benefit of ASC and the broader HPC simulation community. This approach also includes a strong software engineering component to develop new algorithms specifically designed to take advantage of new architectural features and programming models, as well as abstraction layers to provide access to the features and programming models without introducing unnecessary complexity or lack of portability into both current and future codes.

Transformative Co-design: This is the co-design path that would be enabled by a comprehensive, fully resourced Exascale Computing Project (ECP). While there can be a gray area between Proactive and Transformative Co-design, the key distinction is that the latter provides an opportunity to develop future hardware and system architecture designs that are unconstrained by current technology roadmaps. This does not mean that Transformative Co-design is constraint-free. For DOE, the flexibility that can be offered in off-roadmap hardware/system architectures will be used to address important new application requirements/constraints on performance and portability.

The different co-design paths are compared and contrasted in Figure 3.

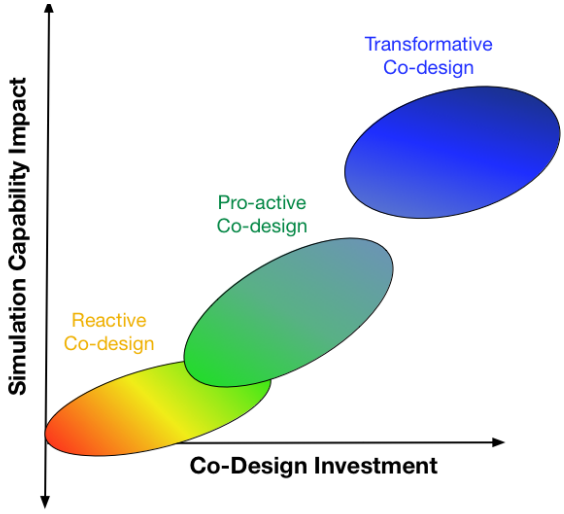
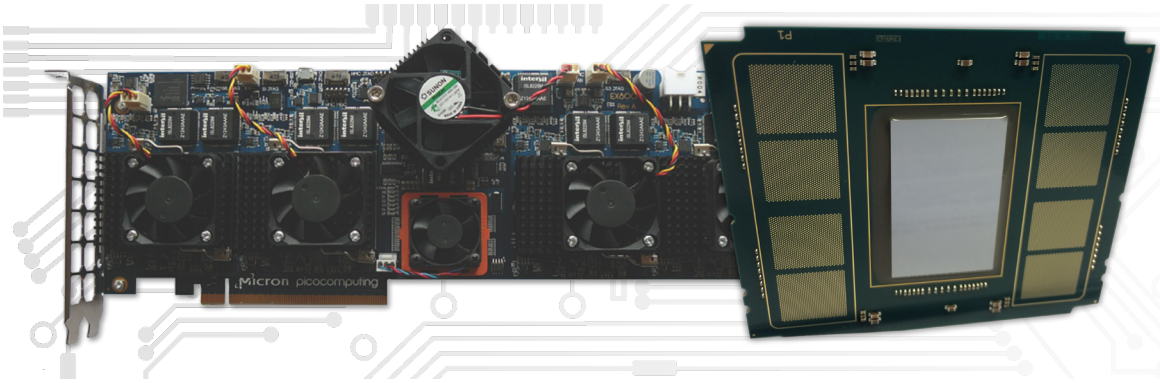


Figure 3: Simulation Capability versus Co-design investment for Reactive, Proactive and Transformative Co-design approaches.



Advanced memory and processor components can be evaluated to determine their usefulness to ASC applications. Shown are a Micron Hybrid Memory Cube and an Intel Knights Landing processor.

This plot is notional only, but it is intended to convey the different co-design activities that can be pursued with different levels of funding and resource commitment. At the lowest level, Reactive Co-design shows that there can be an adverse impact on simulation capability as new hardware architectures, such as multi-level memory, are introduced. As application codes are re-written and ported to leverage the new architectures, simulation capabilities and performance can improve but the amount of code reuse drops. With sufficient investment, Proactive Co-design can attain higher levels of simulation capability while preserving more of the existing application code base, but increased co-design investment still reduces the amount of code reuse. In Transformative Co-design, the highest levels of application code reuse can be obtained at the highest levels of simulation capability if this co-design approach is funded by a strongly supported ECP with a requirement to bridge to DOE's significant existing application portfolio.

SCOPE

Co-design is a process of end-to-end optimization – implemented through collaborative, multi-disciplinary teams that include ASC computational and computer scientists and representatives from HPC hardware and software vendors. Optimization goals include taking full advantage of computing resources under operational constraints such as available electrical power and facility conditions. A key goal of co-design is to include a strategy to maximize the ability of ASC codes to exploit performance efficiencies associated with potentially disruptive technology trends. Technical approaches that provide an on-ramp path for the existing ASC application portfolio to operate in the new advanced architecture environment are needed. ASC codes must not only prevent regression of their current performance levels but also have a viable path forward to exploit new architectural features. To this end, an important objective of co-design is to influence hardware and system architecture towards technology trends that are less, rather than more, disruptive to the ASC application portfolio.

The scope of the ASC co-design activity is determined by the precious resources (staff, budget) that we can allocate to the effort. The ASC ATDM program element adopts a primarily proactive, application-centric co-design approach. Within the constraints of the ATDM budget, this effort will invest in the initial development of new ASC applications that will utilize application frameworks, libraries, and system software innovations in order to perform well on hardware and system architectures that are expected to be available before 2020. There may be opportunities to influence future hardware, via the DOE FastForward and DesignForward investments, but those opportunities will largely be serendipitous in that they will have to be aligned with, and thus limited to, minor adjustments to existing vendor product roadmaps. Should the DOE ECP be funded at a sufficient budget level, a much deeper co-design opportunity will arise in which future hardware and system architectures are co-designed with an explicit intent to create or preserve an on-ramp for our existing application portfolio with robust and lasting solutions designed to insulate the codes from additional disruptive trends that follow. This holistic hardware, software, and application co-design engagement will not only leverage the accomplishments in ATDM but will also support much deeper activity with industry.

ASC's ability to influence vendor roadmaps will determine whether application developers will just react to imposed changes, aggressively/proactively modify the code, or make transformations that utilize new, designed-in, on-ramp features for the existing code.

ENGAGEMENTS

Co-design requires two-way, tightly coordinated, and symbiotic collaborations across multiple disciplines, teams and community sectors. To this end, one of the primary goals of the ASC program's co-design efforts is to engage in collaborations across the spectrum of the HPC ecosystem, as depicted in Figure 4. ASC's engagement strategy involves collaboration:

1) Within the ASC program among the three NNSA laboratories
2) Across the Department of Energy, in particular with the Office of Science programs
3) With HPC vendors to provide key technology and emerging solutions
4) With U.S. universities performing core research and building the next generation of HPC experts
5) Across U.S. federal agencies, ensuring a national perspective
6) Internationally, with strategic partners

These collaborations are meant to allow convergence of hardware and software solutions to optimize for the productivity and performance of ASC workloads on future generations of HPC platforms.

DOE Office of Science: The Department of Energy, through the NNSA's ASC program and the DOE Office of Science's Advanced Scientific Computing Research (ASCR) program, has long been the main driver in the nation's preeminence in high performance computing. Over the last 5 years, there has been a tremendous increase in collaboration between ASC and ASCR as evidenced by the many cooperative workshops, joint research activities, and joint procurements. Numerous NNSA laboratory scientists also participate in Office of Science projects, enhancing the collaboration relationship. A deeper multi-disciplinary relationship between the ASC code teams and the open computer science communities is crucial to ASC's success. The sharing of proxy applications and performance analysis has been a key indicator of this partnership.

HPC Vendors: DOE's engagement with HPC vendors is another high priority of co-design. The FastForward and DesignForward programs are primary vehicles for these collaborations. These efforts fund vendors to explore, develop, and potentially productize technologies of importance to DOE missions. If driven only by vendors' traditional customer bases, many of these required technologies would not otherwise come to market. Engagement within these efforts is focused on co-design of a set of open source proxy applications. These proxies, meant to represent key DOE workloads, provide both requirements and an experimental platform for vendors. DOE personnel work closely with the vendors on co-design optimization of these proxies on their particular hardware.

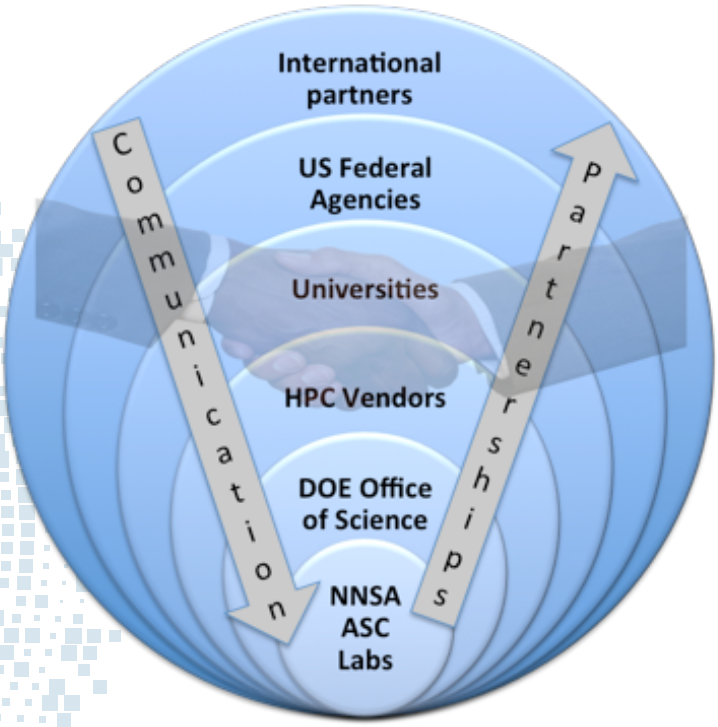


Figure 4 - The ASC co-design strategy strengthens and expands key partnerships across the HPC ecosystem

Universities: The current six Predictive Science Academic Alliance Program (PSAAP2) centers were started in 2013 to carry out R&D in scientific application model and code development, verification and validation, and extreme-scale computing. NNSA-funded students spend 10-week internships at the laboratories as part of their participation in PSAAP2. Co-design summer schools organized by NNSA labs have also proven to be an effective vehicle for collaboration with the academic community and as a successful vehicle for recruitment and hiring of new staff at the labs.

U.S. Federal Agencies: At the national level, high performance computing has become crucial to the nation's security. As the cyber-information age has dawned, protection and analysis of tremendous streams of data are one of the largest technical challenges the nation faces. The need for HPC-enhanced solutions to these tough data analytic problems has created an opportunity for our community to collaborate with other agencies, under the NSCI framework. Common solutions both in hardware and software across data analytics and scientific computing hold the promise of even greater leveraging of federal resources.

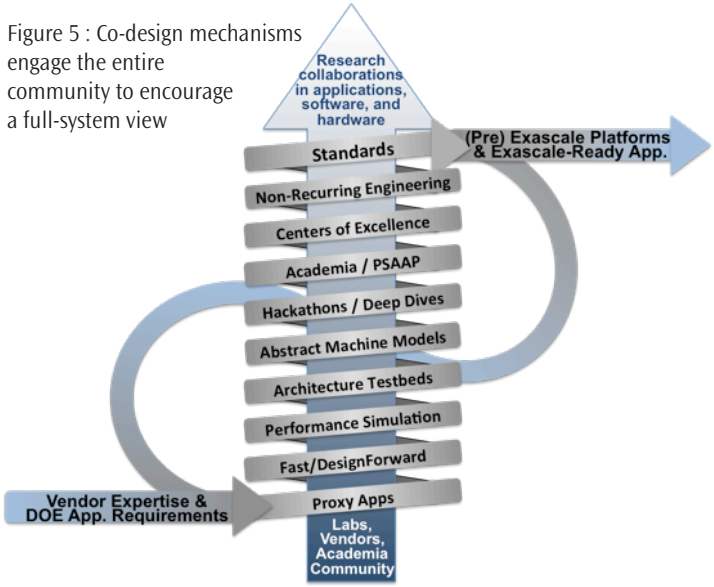
International Partners: International collaboration in co-design is enabled by bilateral cooperative agreements NNSA has with France and United Kingdom. Multiple efforts in joint collaboration are underway, typically enabled by use of each laboratory's proxy applications. Regular technical meetings are held to plan, coordinate, and review the results of these interactions.

Co-design is not simply a division of labor, with inputs and outputs being passed between independent teams. The community utilizes a rich set of co-design mechanisms described below to enhance interactions and understanding to achieve the final goal of applications taking advantage of, and influencing, the development of next-generation HPC systems.

CO-DESIGN MECHANISMS

As a collaboration process among vendors, laboratories and universities involving broad expertise from hardware architects and system software developers to domain scientists, computer scientists and applied mathematicians, co-design is both comprehensive and complex, requiring a wide range of tools and processes to achieve its goals. Each component in Figure 5 plays a role in co-design interaction. For example, some establish common baselines for discussion, analysis, or measurement, and others provide a forum for technical or strategic interactions.

Figure 5 : Co-design mechanisms engage the entire community to encourage a full-system view



Proxy Applications: Proxy applications have been described as the "language of co-design." Full-scale applications are too large and complex (and, for ASC, may be classified or export controlled) to serve as effective tools for communication with vendors, hardware architects and system software developers. Proxy applications are simplified representations of the algorithms, data layout and movement, and/or communication patterns suitable for use in trade-off evaluations in hardware and software design space. A proxy application is intended to represent specific aspects of a single, real weapons application and is expected to be modified during design studies. Thus, there may be multiple proxy applications for one production application code. These characteristics are quite different from benchmarks which generalize frequently used application behaviors and have a stable code base so that comparisons can be made across generations of platforms over time. Proxy applications need to be validated against the originating application as simplifications may result in unexpected behaviors. Since the ASC program has oversight over both the research/co-design activities and its production codes, it can ensure a coordinated calibration and validation effort. This coordination adds confidence to the value of the ASC proxy applications.

Proxy applications are an important communication vehicle between ASC Labs and the rest of the HPC community.

TEST BEDS

Figure 6: Advanced architecture test beds, such as the sample shown here, are used for studies in application performance, portability, power, programming models, and system software. From upper left and clockwise are 1) Catalyst, which is used to explore different approaches for utilizing large amounts of high performance non-volatile memory in HPC simulation and Big Data analytics, 2) Teller, which contains integrated CPU/GPU processors and custom power monitoring hardware, and 3) Darwin, which focuses on small-scale ATS/exascale node architectures and emerging programming alternatives.

FastForward/DesignForward: These projects are jointly funded by ASC and ASCR with the primary objective of establishing strategic partnerships with key computer vendors. The goal of these partnerships is to initiate and accelerate the research and development of node and memory architectures (FastForward) and exascale system designs (DesignForward), and to foster the commercialization of promising emerging technologies. These public-private partnerships between industry, ASCR, and ASC support the development of innovative technologies critical to constructing sustained-exaflop systems and lowering the economic and manufacturing barriers to their commercial productization.

Performance Simulators and Models: Hardware architecture simulators run models of potential future HPC technology. Models vary from detailed (e.g., cycle accurate) to coarse grain, depending on what aspect of application performance is being studied. Also, depending on the performance question, a model may only implement a specific characteristic of a potential system, such as network, memory, or processor. An important aspect of these simulators and models is the ability to "dial in" different values for the hardware feature(s) under analysis. By varying the parameters, one can determine an application's sensitivity, or lack thereof, to a hardware component. The proxy architectures can provide the plausible value ranges for the parameters in the simulation.

Advanced Architecture Test Beds: Test beds that contain the latest computing technology, such as those highlighted in Figure 6, provide a window into the status of evolving hardware and software trends. These state-of-the-art systems vary in size from a single node to a few racks of equipment and allow for performance testing of application and/or system software. With multiple test beds representing different vendor technologies, it is possible to determine the porting effort required to maintain performance portability. These systems are critical for programming model exploration. Since ATDM programmers use these systems in the development of next-generation applications that will be more suited to the new architectures, some test beds must be on private, secure networks to support the sensitivity level of the application codes. The test beds can also be used for system software R&D as individual node(s) can be dedicated to this secondary purpose.

Abstract Machine Models and Proxy Architectures: These serve as key tools for communication between hardware architects and application developers. An abstract machine

model is a schematic diagram of the key components and linkages associated with a potential future architecture. A proxy architecture augments an abstract machine model with speeds, feeds, and capacities to enable application designers to understand and reason about the suitability of the architecture for a specific application and to communicate potential improvements to the computing component and system designers.

Hackathons and Deep Dives: These multi-day sessions provide an opportunity for laboratory application experts and code developers to work closely with teams of vendors, academics and other DOE laboratory colleagues in an informal but focused environment to explore complex coding issues and to further the understanding of some of the key constraints and challenges of large-scale application development. These sessions are critical for the ATDM developers to obtain a deep understanding of architectural trends and sharing the code development issues with vendors (Figure 7).

Academic Partnerships: In addition to the PSAAP partnerships described above, the ASC laboratories work closely with a number of university research groups. Academic research is often the genesis for co-design solutions, particularly in the areas of programming models, tools, and algorithm development. To support these partnerships, the NNSA labs set aside a fixed amount of time on the large unclassified HPC resources for academic use, providing computing cycles critical for the demonstration of research at large scale. Perhaps most importantly, these academic partnerships also provide the laboratories with an important pipeline of students with the skills to come work at the laboratories upon graduation by introducing them to the ASC mission and laboratory culture.

ASC Centers of Excellence: In the recent ATS1 (Trinity) and ATS2 (Sierra) procurements, the ASC laboratories established collaborations with the respective vendors to enlist their skills in assisting ASC's preparation of its applications for deployment and production use on the ATS platforms in 2016 and 2018 respectively. Subject matter experts on the vendors' payroll who have security clearances will work side by side and embedded with the ASC application teams to perform targeted co-design (Figure 8). The lab teams benefit from the deep expertise of the vendors who understand their hardware and software environment better than anyone. In return, the vendors get a stronger understanding of NNSA application requirements beyond the published proxy applications and take that knowledge back into the design iteration process for the planned exascale systems.

Non-Recurring Engineering (NRE): The NRE activities associated with a specific system acquisition may be viewed as a concluding phase or successful result of the co-design process. The technical laboratory and vendor staff associated with NRE activities will benefit from a broad understanding of the co-design reasoning and decisions that culminate with the near-term delivery of the

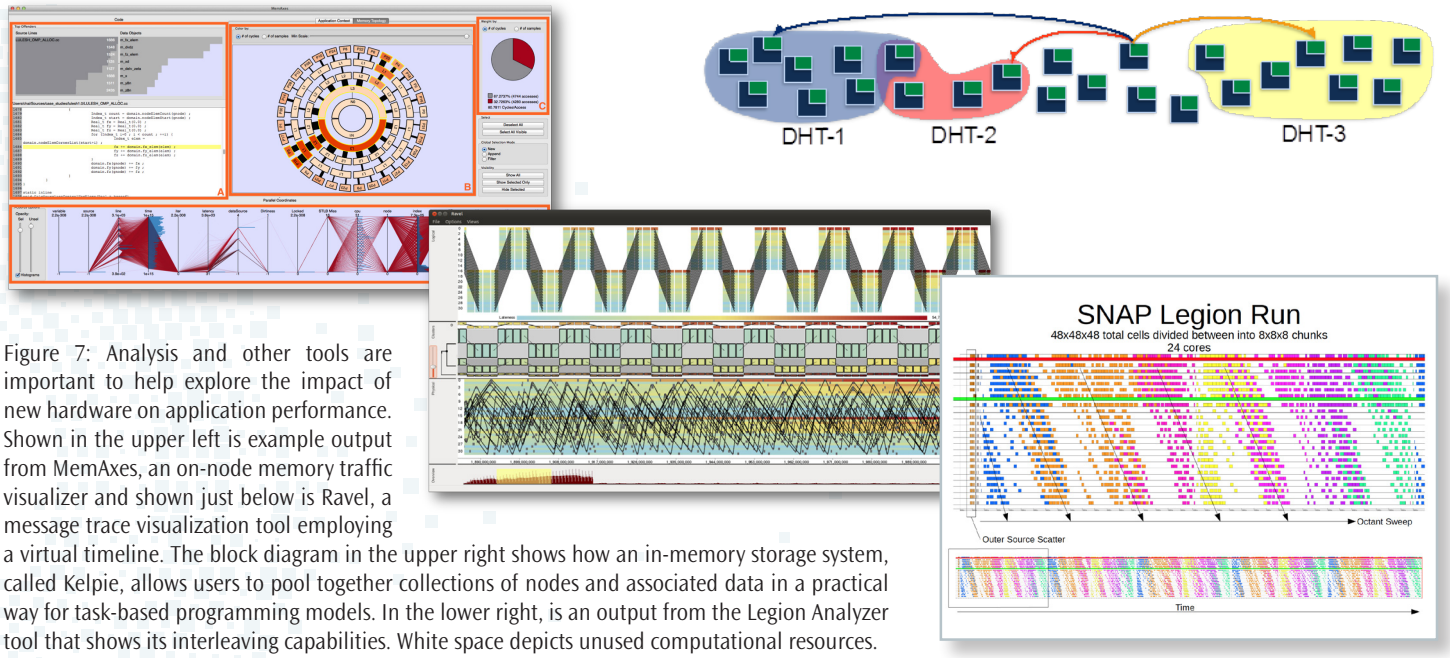


Figure 7: Analysis and other tools are important to help explore the impact of new hardware on application performance. Shown in the upper left is example output from MemAxes, an on-node memory traffic visualizer and shown just below is Ravel, a message trace visualization tool employing a virtual timeline. The block diagram in the upper right shows how an in-memory storage system, called Kelpie, allows users to pool together collections of nodes and associated data in a practical way for task-based programming models. In the lower right, is an output from the Legion Analyzer tool that shows its interleaving capabilities. White space depicts unused computational resources.

contracted system hardware and software technologies, which will then be integrated into a productive, useable HPC system. The aforementioned ASC Centers of Excellence are examples of some NRE activities.

Standards Groups: Through the years, members of the ASC program have participated in formal and de facto standards working groups for critical software technologies, such as programming languages (e.g., C, C++, and Fortran) and programming models (e.g., MPI and OpenMP®). While the level of effort has been modest, the timing is such that increased active participation, including proactive engagement, is warranted. These standard bodies form a co-design community within themselves as vendor, academic, industry, and national laboratory members come together to align their requirements and design solutions. Emerging software technologies, such as memory hierarchies, burst buffers, power monitoring and control, and resilience need champions to ensure portable interfaces. Resilience mechanisms are particularly important to ASC since the jobs running on advanced technology systems can take weeks or months to complete while the future systems' Mean Time Between Failure (MTBF) may be a few days.

By combining a number of these tools, we form a co-design ecosystem which can perform complementary evaluation of promising technologies.

Proxies as the Tool of Co-design

Figure 9 diagrams the synergy between some of the co-design mechanisms. Proxy applications, derived from full applications, can be run on either advanced architecture test beds that are harbingers of potential future HPC platforms or on architectural simulators that implement advanced architectures. Additionally, the set of proxy architectures informs the test bed community of which architectures to provide at small scale. Each proxy architecture specifies for the simulation environment tunable values (including number ranges and units) that

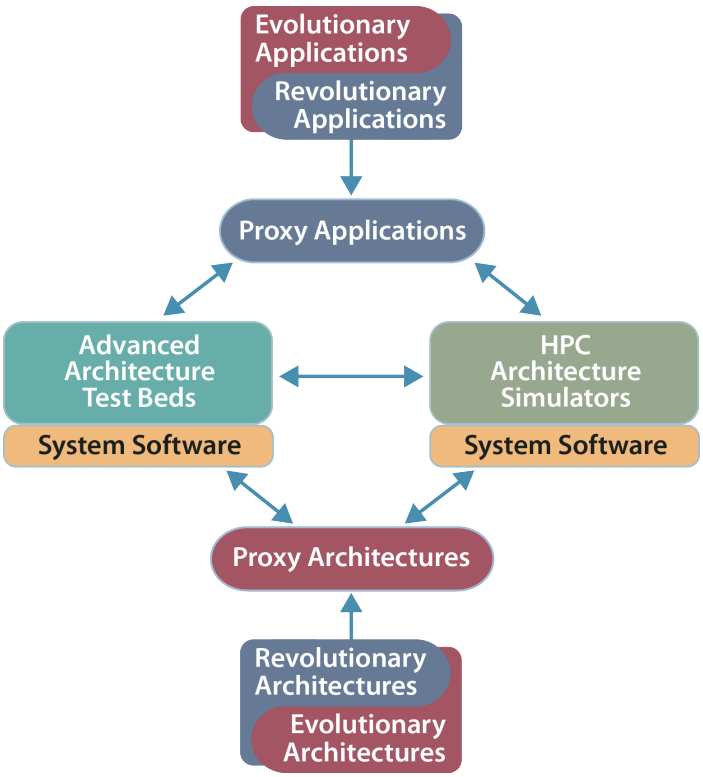


Figure 9: The combination of co-design tools increases our confidence in predicting application performance

Figure 8: The Centers of Excellence associated with the upcoming ASC pre-exascale platforms give cleared vendors the opportunity to work along side application developers to help each other understand the implication of hardware features on real codes.

have performance impacts. When a sufficient range of parameters is applied, the models may be shared openly between users, academics, and researchers. A specific parameter set that closely relates to a point design will likely be proprietary and, therefore, only shared within the appropriate disclosure constraints.

While proxy applications and architectures provide an important tool for research, the ultimate goal is to inform and guide the development of our future production applications and platforms to meet the program's code performance, portability and productivity requirements.

CONCLUSION

In summary, it is certain that the simulation environment of the future will be transformed by new computer architectures, and that new programming models will need to be developed to capitalize on those advances. Within this context, the ASC applications must transition to the new simulation environment or risk stagnation. Moreover, ASC must position the NNSA national laboratories such that they can continue to deliver on NNSA's current nuclear security mission needs, while adapting to radical technology changes, and continue running the most demanding applications necessary to support weapons certification and the research of underlying weapon sciences. Whether ASC will do co-design transformatively, proactively, or reactively is highly contingent on a number of factors. These include having a sustaining, steady-state budget over time; the ability to hire highly skilled staff in the areas such as applied math and computational and computer science; and a long enough time window that allows for careful execution on the co-design strategy laid out in this document. The ability of our large multi-scale, multi-physics applications to effectively adapt to next-generation, extreme-scale architectures remains uncertain. However, the ASC program and NNSA laboratory personnel are well poised and prepared to work on these exciting technical challenges and view them as unique opportunities to seek creative solutions in order to achieve success in the years ahead.

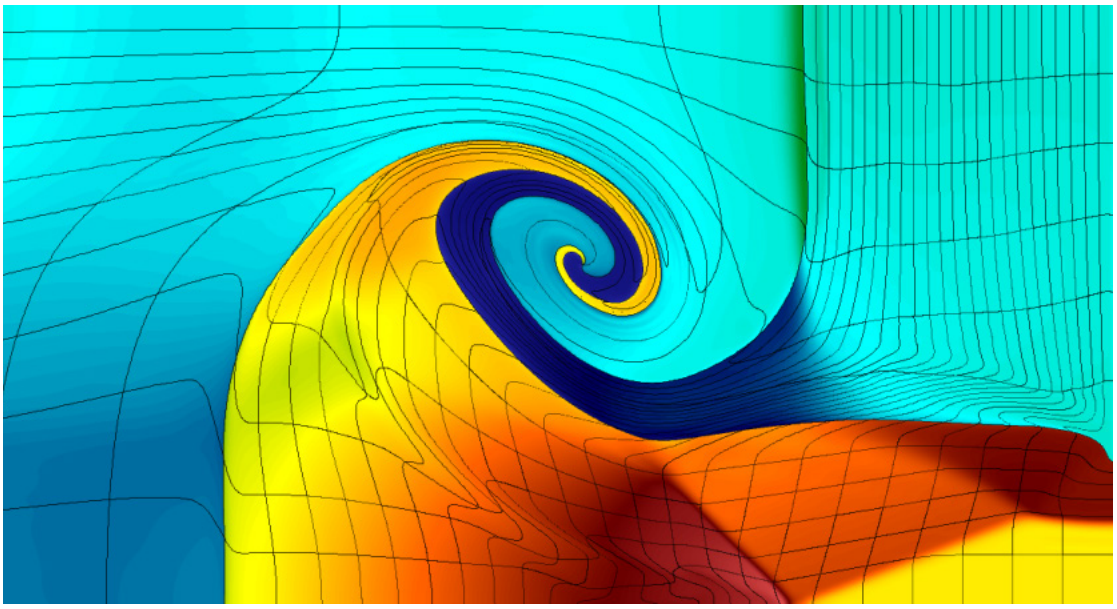


Image shows a triple-point shock wave hydrodynamics test problem for the new, co-designed BLAST code that highlights the effectiveness of high-order curvilinear mesh elements.

REFERENCES

1. *Advanced Simulation and Computing: Computing Strategy*, Ang, J. A., Henning, P. J., Hoang, T. T., and Neely, R. 2013. http://nnsa.energy.gov/sites/default/files/nnsa/05-13-inlinefiles/2013-05-23%20ASC-StratV9.pdf

2. *Top Ten Exascale Research Challenges* DOE ASCAC Subcommittee Report, February 10, 2014.

3. *Technical Challenges of Exascale Computing* JASON Report JSR-12-310, April 2013.

4. *Report of the Task Force on High Performance Computing of the Secretary of Energy Advisory Board*, August 10, 2014.

5. *The Future of Computing Performance, Game Over or Next Level*, National Research Council of the National Academies' report, 2011.

6. *Exascale Software Study: Software Challenges in Extreme Scale Systems* DARPA study released September 14, 2009.

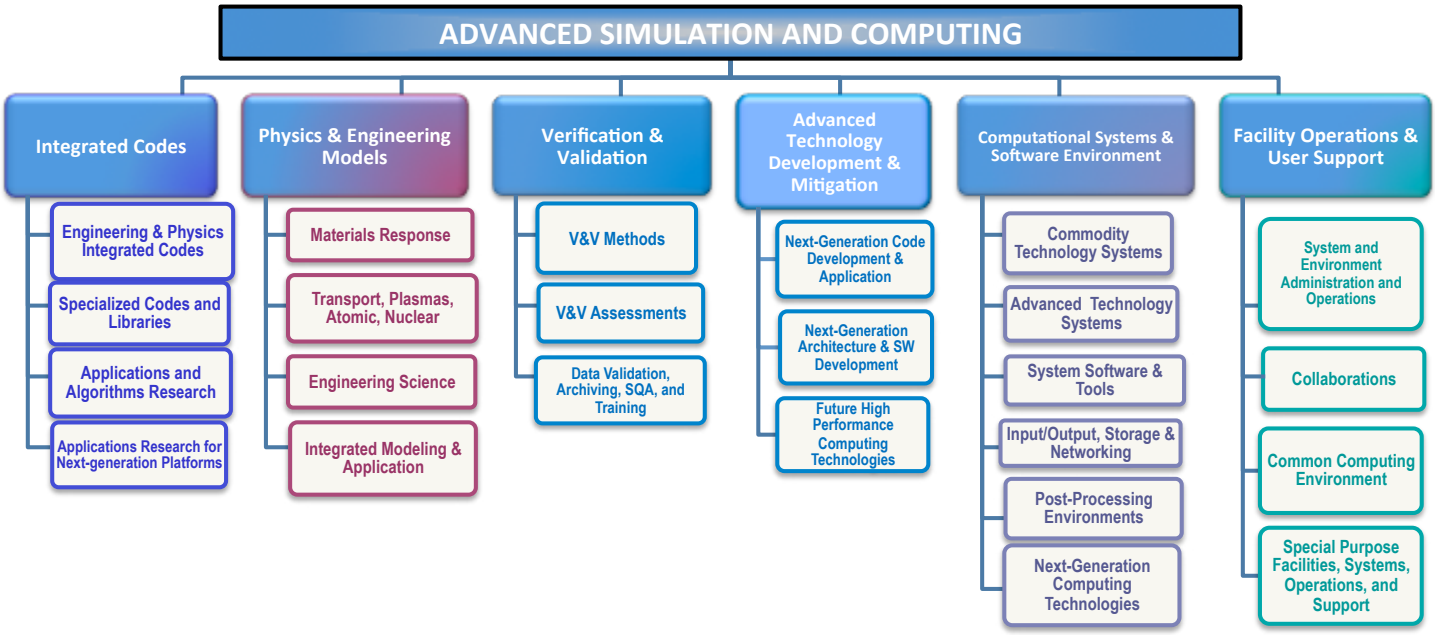
7. *Exascale Computing Study: Technology challenges in achieving exascale systems* DARPA Technical report, 2008.

ACRONYMS

AAR	Annual Assessment Review
ASC	Advanced Simulation and Computing
ASCR	Advanced Scientific Computing Research
ATS	Advanced Technology System
ATDM	Advanced Technology Development and Mitigation
CPU	Central Processing Unit
CTS	Commodity Technology System
DOE	Department of Energy
DP	Defense Programs
ECP	Exascale Computing Project
EPIC	Engineering and Physics Integrated Codes
FLOPS	FLoating point OPerations per Second
GPU	Graphics Processing Unit
HE	High Explosive
HPC	High Performance Computing
LANL	Los Alamos National Laboratory
LEP	Life Extension Program
LLNL	Lawrence Livermore National Laboratory
MPI	Message Passing Interface
MTBF	Mean Time Between Failure
NNSA	National Nuclear Security Administration
NRE	Non-Recurring Engineering
NSCI	National Strategic Computing Initiative
OpenMP®	Open Multi-Processing
PSAAP	Predictive Science Academic Alliance Program
Pthreads	Posix threads
R&D	Research and Development
SC	DOE Office of Science
SFI	Significant Finding Investigation
SNL	Sandia National Laboratories
SSP	Stockpile Stewardship Program

APPENDIX

ASC National Work Breakdown Structure



ASC™

NNSA
National Nuclear Security Administration