Scalable Ecosystems for Data Science (SEDS*)*

P. Beckman, A. Choudhary, J. Dongarra, G. Fox, W. Gropp, and D. Reed

1 Introduction

New generations of scientific instruments and sensors are producing unprecedented amounts of data in domains as diverse as environmental modeling and disaster response, critical infrastructure management, smart grids and intelligent transportation systems, astronomy and astrophysics, chemistry and biology, and human-computer interaction. Rich tools and techniques for data analysis have emerged in each of these domains, which a recent NRC report described as "Seven Computational Giants of Massive Data Analysis" that span basic statistics, generalized N-body problems, graph-theoretic computations, linear algebra, optimizations, integration, and alignment problems. Although many of these domains exploit HPC systems, they do so in a largely separate ecosystem, disjoint from the software ecosystem of traditional HPC domains. We believe that *a Scalable Ecosystems for Data Science (SEDS) is needed to enable and accelerate data-driven scientific inquiry by integrating "best of breed" techniques from machine learning, big data analytics, and HPC, augmented with innovative new algorithms and software.*

Technical and economic synergies exist among the challenges facing data-intensive scientific research and high-performance computational modeling, and advances in both are needed for future breakthroughs. An integrated environment would further scientific understanding, improve the performance of machine learning algorithms and software, stimulate creation of new tools and techniques to facilitate deployment of a common cyber infrastructure.

The SEDS approach is founded on the premise that algorithms should combine best practices in HPC and data science, because many algorithms are expressed in the mathematics of linear algebra and, for graph problems, sparse matrices. A key approach is combining the best elements of HPC, clouds, and data-intensive platforms with new middleware. Further, SEDS approach envisions that a process for research that implements a virtuous cycle that creates a series of *Discovery Appliances* embodying research ideas and innovations from algorithms, applications, and hardware. We believe that only by addressing the entire stack—applications, algorithms, programming and runtime systems, and hardware—can one succeed in revolutionizing computational and data science.

2 SEDS Strategy

Current HPC platforms are architected and configured for floating-point-intensive applications with regular memory access patterns, tightly coupled interprocessor communication, and modest input/output demands. By contrast, big data platforms target applications requiring high-capacity storage and I/O, along with a larger fraction of integer operations and more irregular memory access patterns. HPC designs minimize communication latency with expensive networks for tight coupling, whereas for big data analysis economics have largely dictated use of commodity networks with higher communication latency. The SEDS strategy is based on a simple premise and conviction: *a revolution in computational and data science is possible only by integrating applications, algorithms, programming and runtime systems and hardware in a common ecosystem.* These *Discovery Appliances*, the primary software packaging and distribution mechanism for the results of SEDS approach, can be structured as software *containers*

2.1 System Software and Infrastructure

Today's HPC systems have software stacks, system architectures, and operational policies that often limit their applicability to a narrow range of computationally intensive applications. Consequently, users with data-analysis tool chains and complex workflows all too often simply ignore the HPC community and focus on readily available tools, many of which are not designed for scalability to large platforms or petabytes of data. To date, neither community has fully embraced the shifting nature of algorithm optimization, driven by manycore energy constraints and data movement costs. There is an opportunity to re-imagine a convergence environment by leveraging major hardware and software technology shifts to create a new "software defined scientific computer" for the larger research community.

2.1.1 Convergence Architectures

Classic HPC architectures focus on floating point performance, memory and interconnect bandwidth, file system resilience, and a very low mean-time to interrupt for parallel computations, whereas systems designed for data analysis often focus on aggregate I/O operations and trade raw computing performance for data capacity. Fortuitously, several technology shifts are bringing these two architectures closer together. First, the NVRAM capacity, reliability and power now make it economically feasible to include high-capacity NVRAM in all nodes, allowing *in situ* analysis. Second, shared node address spaces allow adaptive mapping of computations to maximize performance. Third, software-defined networking now supports data center network traffic shaping and prioritization, allowing bandwidth allocation among competing network data flows. At the same time, HPC architectures offer very high performance interconnects with low latency. Finally, operating system virtualization via containers now supports creation of tailored, domain-specific software stacks that can be co-resident on a single node. Combining these technologies from HPC systems and data analysis systems will provide revolutionary new capabilities.

2.1.2 Hybrid, Customized Software

Most current HPC systems operate as space-shared resources. Furthermore, the software stack is mostly static; packages are updated only when system administrators respond to community pressure. These two operational choices make interactive data analysis, persistent services, and domain-specific software stacks difficult. Data scientists need shared access to handfuls of nodes for data exploration and long-term access for continuous processing of real-time data streams and providing persistent data services to constituent scientific community. In addition, analysis workflows require web technologies, Java, Python, and other tools from the big data middleware ecosystem (e.g., tools like Zookeeper, Storm, Hadoop, Spark, Pig (MapReduce) and Hbase (NOSQL)) that are often not supported in HPC systems.

2.1.3 Elastic Data Management

Data sharing is a key part of data science workflows; teams load, clean, analyze, and then publish data sets. These data sets are no longer simple file sets, but rather active query environments, from which scientists request data slices or annotate a database by adding new values. To enable this always-on analysis model, scalable data management technologies for heterogeneous and unstructured data are needed. Examples include MongoDB, Elastic Search, SciDB, various forms of Hadoop/HDFS that must interoperate with their HPC counterparts: parallel file systems, MPI and data formats such as HDF and NetCDF. Query processing and analysis libraries atop these layers need to be developed, which in turn will drive analytics and data mining algorithms.

2.1.4 Storage and File System Consistency

One of the keys to high performance file systems is matching the data locality and consistency model provided by the file system to application needs. In contrast to HPC systems, cloud service operators and big data analysis software systems have adopted weaker consistency models [8], which have provided much better performance and resilience. By contrast, HPC models have application-oriented consistency models for I/O more in line with application needs, and some *ad hoc* systems provide weaker consistency but higher performance.

2.2 Application and Algorithm Co-design

Achieving the SEDS goal will require a co-design effort in which domain experts in applications, algorithms, and software work cooperatively resulting in a continuous virtuous co-design cycle that enables exploration and prototyping of hardware and software architectures with variety of applications and algorithms.

2.2.1 Illustrative Domain Engagement and Representative Applications

Internet of Things: Data streaming is exemplified most clearly by the Internet of Things (IoT). Estimates suggest there will be over 20 billion such Internet-connected devices by 2020. IoT applications include identifying machine faults, managing traffic flow, optimizing energy production and distribution (smart grids), understanding urban dynamics and services (smart cities), agricultural and environmental ecosystems, and personal monitoring. Internet-connected scientific instruments such as light sources, telescopes, and satellites bring different challenges, as the data originates from a smaller number of larger instruments.

Materials Science and Engineering: Materials science and engineering innovations enable more efficient batteries, stronger and lighter materials, newer materials for mobile devices and sensors, and new drugs and medical delivery mechanisms. Traditionally, materials research has been experimentally focused, though simulations are now critical to identifying newer materials and structures. Advanced materials science research instruments such as the Advanced Photon Source (APS) now produce terabytes of data in single experiment. Deep machine learning and predictive techniques are critical to deriving insights from this data.

Astronomy/Cosmology: Modern astronomy combines massive data inputs from both new experiments and outputs from high-resolution simulations, and then apply additional computing to draw accurate and testable conclusions. To quantify cosmological constraints, it must reliably detect, measure, and classify billions of sources from imaging data. Next, the galaxy distances must be accurately estimated, and distribution and temporal evolution modeled. Finally, Large numbers of simulated, synthetic galaxy catalogs, similar in size and complexity to the actual galaxy catalog, must be generated and processed to quantify any systematic biases in the data reduction pipeline and to allow reliable error measurements. The required spatial resolution means that brute force techniques for computing even the simplest statistical quantities cannot be used. Scalable analytics kernels and functions must be developed

Computational Social Science and Network Science: The availability of big data, collected in real time from social media, smartphones and the web), along with powerful computational resources, now allow researchers to study human behavior in new and unprecedented ways. Blending techniques from machine learning, network science, natural language processing, and time-series analysis, allows researchers to model and predict how individuals behave, connect, produce and consume information, and how they make decisions affecting their online and offline worlds. Potential applications span national security, emergency management, crime tracking and fighting, marketing strategies, election forecasting, and health monitoring. To enable such applications, researchers must store and query heterogeneous historical data and train large, models with network interactions found by machine learning.