FINAL REPORT
LAB 14-1043
Scalable Analysis Methods and *In Situ* Infrastructure for Extreme Scale Knowledge Discovery

The DOE Award number DE-SC0012561

Project Nickname
SENSEI: Scalable Environments for Scientific Explorations In Situ

Project Summary Report
October 2014 through May 2017

Contents

1 Project Overview and Summary of Accomplishments

The primary challenge motivating this team's work is the widening gap between the ability to compute information and to store it for subsequent analysis. This gap adversely impacts science code teams, who are able to perform analysis only on a small fraction of the data they compute, resulting in the very real likelihood of lost or missed science, when results are computed but not analyzed.

Our approach is to perform as much analysis or visualization processing on data while it is still resident in memory, an approach that is known as *in situ* processing. The idea *in situ* processing was not new at the time of the start of this effort in 2014, but efforts in that space were largely *ad hoc*, and there was no concerted effort within the research community that aimed to foster production-quality software tools suitable for use by DOE science projects. In large, our objective was produce and enable use of production-quality *in situ* methods and infrastructure, at scale, on DOE HPC facilities, though we expected to have impact beyond DOE due to the widespread nature of the challenges, which affect virtually all large-scale computational science efforts. To achieve that objective, we assembled a unique team of researchers consisting of representatives from DOE national laboratories, academia, and industry, and engaged in software technology R&D, as well as engaged in close partnerships with DOE science code teams, to produce software technologies that were shown to run effectively at scale on DOE HPC platforms.

The main focal points of this team's activities were as follows:

- Show the viability of *in situ* methods and infrastructures, at scale, on contemporary DOE HPC platforms within the context of several DOE science examplars. Our approach consisted annual campaigns targeting runs, at increasing scale, on multiple DOE HPC platforms, during each year of the project. We used a combination of miniapplications and specific science codes, and focused on a limited set of design and execution patterns so as to limit problem scope (§2).
- To increase code portability through the development and use of a *generic* interface between data producers (i.e., simulations) and data consumers (i.e., *in situ* methods and infrastructures) (§3).
- Engage in close partnerships with DOE science code teams, whom provided science drivers for the software technology R&D and whom also benefited through one-on-one interactions to begin making use of *in situ* methods and infrastructures (§4).
- Through outreach, publications, and presentations, foster the cooordination and formation of a community of *in situ* developers, researchers, practitioners (§5).
- Produce and release software artifacts, consisting of *in situ* methods and infrastructures (§6).

An overview of the team's accomplishments is shown in timeline form in Fig. 1. This chart shows yearly campaigns where we push the limits of scalability, first to 64K ranks in early 2015, to 256K ranks in 2016, along with an unprecedented 1M ranks in 2016. Our outreach activities include the launching of a workshop at the annual SC conference series that focuses on *in situ* topics, along with activities that engaged a broader and international community with SIAM minisymposia in Paris, France.

Over the course of the project, the team has been highly productive in terms of generating publications (§7.1):

- 7 peer-reviewed journal articles (IEEE Visualization; Computer Graphics Forum; Parallel Computing; Computational Astrophysics and Cosmology);
- 19 peer-reviewed conference papers (SCXX, HPDC, IEEE Cluster, Grid, and Cloud Computing; eScience; Parallel CFD; ALENEX; IPDPS; and many others);

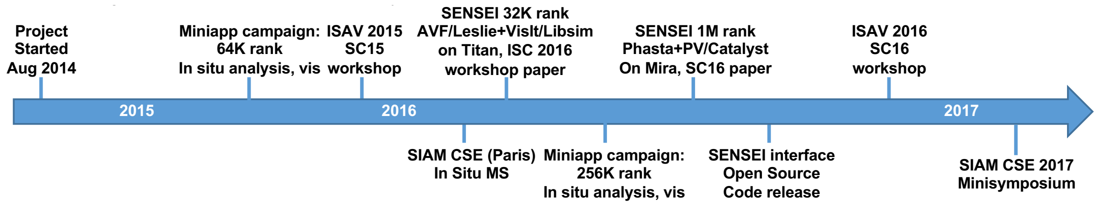


Figure 1: Timeline of SENSEI team accomplishments, beginning with the project start in August 2014, and going through May 2017.

- 8 peer-reviewed workshop papers;
- 4 peer-reviewed panel abstracts (LDAV, SCXX, SIAM PP16, SIAM CSE 2017;

The team has also been highly active in outreach and fostering interaction within the community of *in situ* researchers and practitioners outside this project (§7.3). We launched a new workshop at the SCXX annual conference series, In Situ Infrastructures for Enabling Extreme Scale Analysis and Visualization, which was held in 2015 and 2016, and has been accepted into the 2017 program. This workshop has been an overwhelming success, drawing the attention and participation of the worldwide HPC *in situ* community, with participation from academia, government labs, and industry. We have also organized multiple minisymposia at SIAM meetings, presented tutorials at SC in 2015 and 2016, engaged in direct outreach to undergraduates and underrepresented groups, and given about 55 invited presentations on various topics related to *in situ* technologies.

2 Scalability Studies

Problem

In order for software tools to be usable by DOE scientists on DOE HPC platforms, those software tools must undergo a process of verification that shows they work at scale on those platforms. While each of VisIt/Libsim, ParaView/Catalyst, ADIOS, and GLEAN were, to varying degrees, production-quality tools in their own right, they had not undergone the process of being proven to work at scale in a rigorous fashion on DOE HPC platforms. When such tests are successfully completed, the DOE science community—including science code teams as well as DOE HPC facilities personnel—will have greater confidence in the software infrastructure being used and deployed on their systems. Such confidence helps to promote more widespread adoption and use within the DOE science community.

Approach

In brief, our approach to this problem was to engage in organized scalability studies on all major DOE HPC facilities. These studies consisted of using a miniapplication as a data producer connected to *in situ* data consumers, and conducting runs at increasing scale over the lifetime of the project.

We opted for an approach similar to that adopted by many DOE HPC facilities for both system RFIs as well as system acceptance tests: the use of miniapplications that are representative of common scientific workloads. In our case, the problem space is a bit more complicated due to the fact that there is a data producer as well as a data consumer.

Running miniapplication campaigns at gradually increasing scale over the 3-year life of the project, across several platforms, allowed us to discover impediments to scalability, as well as to design and engineer solutions to those impediments. Those code improvements in turn found their way into production releases of the constituent *in situ* technologies, and would in turn benefit worldwide scientific community.

Results

Over the course of the project, we conducted two major scalability studies designed to push the limits of concurrency and performance on DOE HPC platforms. To prepare for these studies, we created and assembled the necessary software components, which consist of data producers, the *in situ* interface, and data consumers.

For a data consumer, we created a standalone MPI-based code, `oscillators`, that computes a time varying function on a 3D computational domain that models the behavior of a user-specific number of damped oscillators. This code uses a "shared nothing" form of parallelism, where the MPI ranks can each perform computations independently with no inter-rank communication required. As such, it is highly scalable to arbitrary concurrency. Over the lifetime of the project, we instrumented this miniapplication to use the generic *in situ* interface (§3). As with other types of miniapplications, ours is representative of many types of DOE science workloads, namely a bulk-synchronous parallel (BSP) computation built atop MPI. Our approach with the `oscillators` code allows us to replicate BSP computation at scale but at much lower computational cost compared to a production science code.

We created several *in situ* data consumers, or endpoints. Two of these are relatively simple computations. A `histogram` computation performed constant-width-bin histogram computation, and an `autocorrelation` code computed the autocorrelation calculation over a time-varying signal. While single timestep *in situ* calculations are important, our approach of also including a time-varying *in situ* calculation was a first, and one of relevance to many DOE science problems. Other *in situ* data consumers include visualization pipelines, implemented in VisIt/Libsim and ParaView/Catalyst, that create an image of a heatmap-colored slice extracted from a 3D volume, and rendered from a 3D viewpoint.

All of the software tools that we built for use in the miniapplication campaigns are released under an open source license and are available to the worldwide scientific community (§6).

Scalability study #1: Spring-Summer 2015. This campaign consisted of weak-scaling studies at concurrency levels ranging from 1K to 64K-way concurrency on Mira at ALCF, Edison at NERSC, and Titan at OLCF. In these tests, the `oscillators` code produced data at those levels of concurrency, we invoked the `histogram` and `autocorrelation` calculations via VisIt/Libsim and ParaView/Catalyst in true *in situ* fashion (i.e., no data movement) and via ADIOS and GLEAN in an *in transit* fashion (data moves from the computational cores to the analysis cores, but never lands on disk). These studies revealed impediments, such as potentially long initialization times due to the use of shared libraries. Also, we encountered problems with ADIOS on the Blue Gene platform, and ultimately decided to not invest resources in the engineering required for ADIOS to work on the Blue Gene platform, as that platform was known to not be on the list of next-generation systems at DOE HPC centers.

Scalability study #2: Spring-Summer 2016. As with the previous scaling study, we made use of the `oscillators` code as a data generator, but this time made use of the new SENSEI generic *in situ* interface (§3). This approach allowed us to connect the `oscillators` code to any of the infrastructures without modification to change from one to another. This weak scaling study pushed the limits out to 256K-way concurrency on Titan and Mira (NERSC did not have a machine that large at that time). This campaign went into significant depth to measure the impact of the SENSEI generic interface in terms of runtime and memory footprint at scale, as well as compared the cost of running a representative at-scale workload in *post hoc* fashion to *in situ* fashion. In addition to runs with the `oscillators` miniapplication code, we also engaged in at-scale runs of several different science codes at very high levels of concurrency. One of these runs, with the PHASTA code connected to ParaView/Catalyst, was at over 1M-way concurrency, which was an unprecedented level of scalability ever achieved for *in situ* processing. The results from this miniapplication study provided data for a technical paper that appeared in the SC16 Technical Program.

3 SENSEI Generic *In Situ* Interface

Problem

The key problem for DOE science codes is that the code instrumentation needed to take advantage of *in situ* methods limits the code to one particular approach and implementation. If a science code wants to take advantage of *in situ* processing capabilities, the code developers must modify that code to make calls to a particular *in situ* infrastructure. Over the course of the code's lifetime, if the developers want to try out a different *in situ* infrastructure or approach, then the code must be modified to make use of the new *in situ* infrastructure's application programming interface (API). This particular problem reflects the complexity and cost of instrumenting and maintaining simulation codes.

A related problem is the fact there exists a diversity of *in situ* infrastructures and methods. On this project alone, we have representatives from four different existing *in situ* infrastructure teams (VisIt/Libsim, ParaView/Catalyst, ADIOS, GLEAN) as well as researchers who develop new *in situ* analysis methods, such as for computational geometry and topology. The presence of so many different ways of making use of *in situ* methods and infrastructures is itself an impediment to their use, as well as an impediment to deployment of new innovation and methods that would be of use to the DOE science community. That impediment is reflected in the cost of code development and maintenance required to take advantage of new methods and technologies.

Approach

One of the central objectives for this project was to design and create a generic *in situ* interface, which enabled a science code to be instrumented once, but then be able to connect to and make use of any number of potential *in situ* infrastructures or methods.

A central challenge with this objective is rectifying between a simulation's particular data model and the data model of the *in situ* method or infrastructure. Our approach to this problem is to provide means for mapping from a simulation's data model, via *data adaptor*, into an *in situ* data bridge, and thence into the *in situ* method or infrastructure via an *analysis endpoint adaptor*.

Such an approach is highly flexible, for it is adapatable to data models common in DOE science applications. And it is also adaptable for use by a diverse set of *in situ* processing methods and infrastructures.

Results

We implemented the SENSEI generic *in situ* interface, the elements of which are shown in Fig. 2. One element is the adaptor between the producer and a bridge, where a data producer's data model is mepped into a generic data object. Another element is the adaptor on the data consumer end, where the contents of the generic data object are mapped into the consumer's native data model.

We instrumented to `oscillators` miniapplication code to make use of the SENSEI interface for the 2016 scalability studies (§2), and demonstrated running a data producer and connecting it to several different *in situ* methods and infrastructures. This accomplishment demonstrates us reaching our objective of having a data producer being able to make use of different *in situ* processing technologies through a single interface.

The SENSEI interface was the subject of a peer-reviewed publication in a workshop at SC16: In Situ Infrastructures for Enabling Extreme-scale Analysis and Visualization, held in Salt Lake

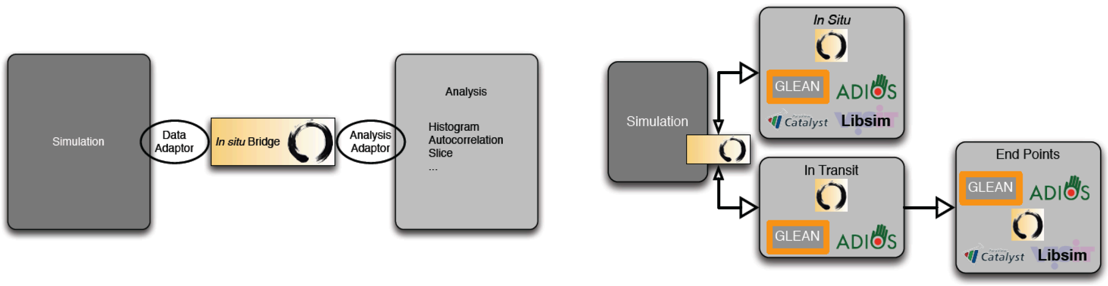


Figure 2: The SENSEI generic *in situ* interface. (Left) a data adaptor provides the means to map from a simulation's native data model into a generic data object, that goes through a bridge, where it is then mapped via an analysis adaptor into the native data model of an *in situ* analysis method. (Right) This approach allows us to instrument a simulation once with the interface, then make use of several different *in situ* methods and infrastructures.

City in November 2016.

Next Steps

While this approach and implementation proved useful, there is still a great deal of work remaining to be done. The initial implementation supports mapping 2D/3D structured arrays into and out of the generic data object, which covers a significant number of DOE science cases. However, there are many other data models in DOE science codes, such as adaptive mesh refinement, unstructured meshes, and so forth. Providing support for these more complex data models to be mapped requires additional engineering to encode the complex metadata describing those meshes/data models into the generic data object, and making it available to downstream consumers. This topic is the subject of future work.

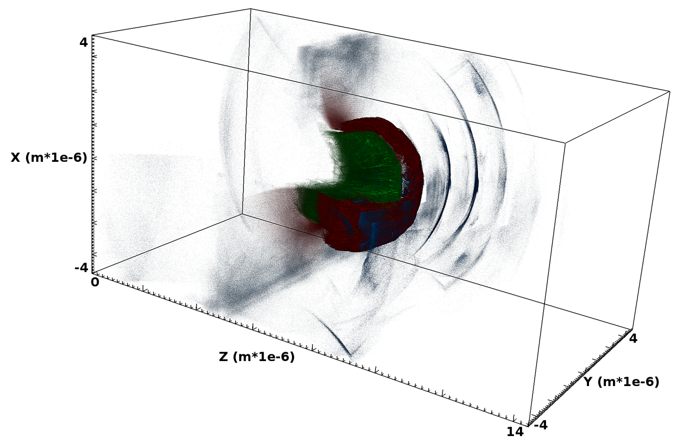
4 Science Code Team Collaborations

4.1 Laser Plasma Accelerator Physics: Ion Beam Theory and Dynamics

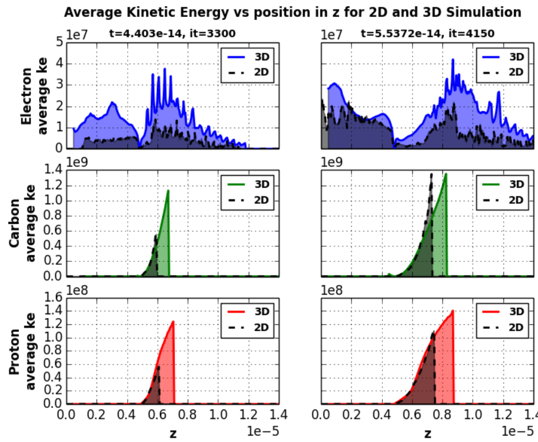
Science objective and stakeholders: The rapid development of laser technologies has made the interaction of high intensity laser pulses with matter a major focus of theoretical and experimental research. This interaction has attracted a lot of attention since it can potentially revolutionize a number of applications by making available compact sources of high energy beams of electrons and ions, and high frequency radiation. In particular, the beams of ions accelerated to the energies from several MeV to hundreds of MeV or even GeV are expected to be available from laser plasma interaction. Several lasers able to achieve 100 MeV levels of proton energy are already in operation and many more facilities are being built or planned. These ion beams have a wide range of applications, such as, radiography, deflectometry, cancer therapy, injection into conventional accelerators, fast ignition, isochoric heating of matter, positron emission tomography, nuclear physics and others.

Accelerator scientists at the Berkeley Lab Laser Accelerator (BELLA) center are using Warp—an advanced particle-in-cell (PIC) simulation framework—for computational modeling, design, and study of laser-based ion and electron acceleration to improve understanding of the complex acceleration processes and to optimize and develop new accelerators. PIC is a technique that uses a combination of particles and meshes to solve Vlasov-type equations and is widely used in computational studies where the modeling of kinetic effects is required.

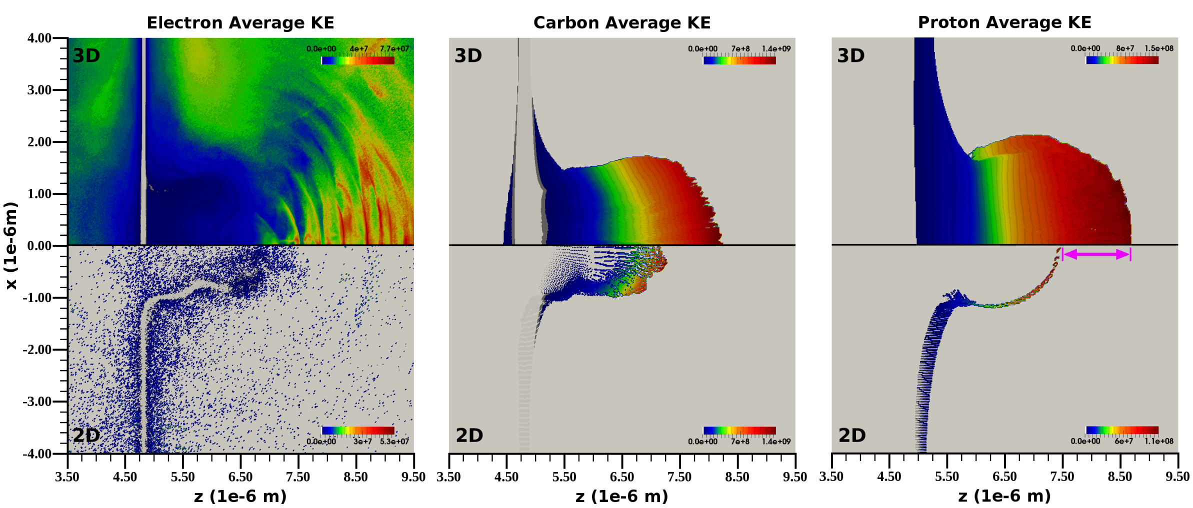
Accurate three-dimensional modeling of laser-based particle accelerators using PIC requires: i)



(a) Isosurfaces of kinetic energy



(b) Average kinetic energy over z.



(c) Average kinetic energy in x/z for the 3D ($x > 0$) and 2D ($x < 0$) simulation.

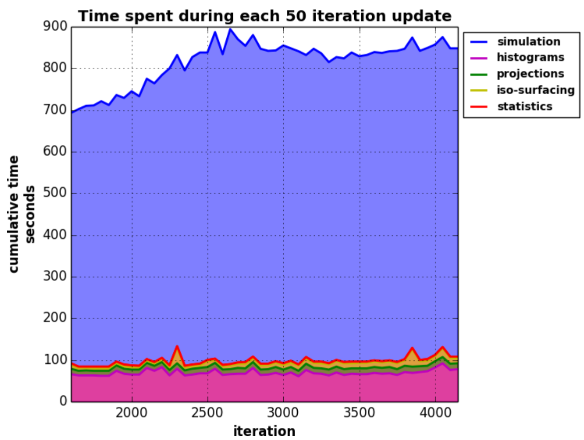
Figure 3: Examples from our *in situ* analysis. (a) *In Situ* rendering of 20 isosurfaces. (b) Histograms, and (c) projection extracts, rendered in post-process analysis.

high-resolution meshes with billions of cells to resolve the high plasma frequencies, ii) hundreds of millions and in many cases billions to trillions of particles to model the plasma, and iii) 10^4 to 10^6 timesteps to accurately resolve the laser and particle motions and interactions. At the same time, often only a small fraction of the particles form particle features of interest, such as, a beam. Understanding of the complex acceleration processes requires visualization and analysis at high temporal and spatial resolutions and the ability to study the relationships and interactions between multiple particle types and fields. The large data sizes and need for high temporal and spatial resolution are—in conjunction with recent lag in I/O bandwidth and storage capacity relative to growing computational capacity—making it increasingly prohibitive to collect all the data required for *post hoc* analysis in persistent storage.

Accomplishments: In this reporting period we adapted our phase space particle analysis and Yee-Cell re-meshing codes for use with WarpIV, an application that couples VisIt LibSim and Warp. We applied the newly developed analysis codes to the study of ion accelerator beam modeling. Specifically we analyzed a 2D and a 3D simulation with identical initial and boundary conditions to assess the value of the more complex and computationally costly 3D simulations.

The actual simulation box size is $(8\mu m \times 8\mu m \times 14\mu m)$ and the actual duration is $6.2710e^{-14}$ seconds. The temporal resolution is $r_{time} = 6400$ steps and the spatial mesh resolution is $r_{space} = (900 \times 900 \times 1800)$ in 3D and correspondingly $r_{space} = (900 \times 1800)$ in 2D. The 2D simulation contains 251,600 electron, 179,776 carbon, and 71,824 proton particles per timestep and the 3D simulation models the motions of more than 2.1 billion particles at each timestep; 1,069,251,640 electron, 809,557,568 carbon, and 259,694,072 proton particles. The 3D run made use of 3200 cores on 200 nodes of NERSC's Cray Edison for 16 hours.

For every simulation we computed a broad range of *in situ* visualizations and analytics, including, i) histograms of various quantities, ii) basic statistics of various quantities, e.g., mean, standard error, iii) filtering of particles to extract and analyze beam particles, iv) merging of particle species to define joint particle distributions, v) two and three-dimensional binning of the data to compute multi-dimensional histograms and derived spatial statistics, and vi) 3D renderings using transparent isosurfaces of binned quantities. For visualizations of derived data products, such as histograms and 2D data binnings, we typically save the reduced data product for *post hoc* rendering. For complex 3D visualizations, where saving the raw data is prohibitive, we store the images only.



(a) Run time in seconds by category at 50 iteration updates.

	Category	I/O Cost
in situ	histogram	4.49 MB
	projections	795.97 MB
	isosurfaces	40.77 MB
	statistics	3484 b
	total	**841.23 MB**
hypothetical *post hoc*	particle position and velocity	**3.24 TB**

(b) Total I/O cost and comparison with identical post-process analysis.

We instrumented WarpIV with a light weight, coarse grained, profiling API and have used it to study the performance of the $3D$ run presented above. For the performance analysis we grouped similar operations into one of five categories: i) "simulation", for the Warp solver computations; ii) "histograms", for the computation and I/O of 12 histograms; iii) "projections", for the computation and I/O of 18 $2D$ projections that were used to compare $2D$ and $3D$ wave fronts; iv) "isosurfaces", for the $3D$ isosurface computation, translucent rendering, and image I/O of the Ke and density fields; and v) "statistics", for a number of simple descriptive statistics computations and I/O.

The stack plot in Fig. 4(a) shows the time spent to complete each category at every 50 iteration simulation update. At each update the visualization, analysis, and I/O operations consumed approximately 11% to 15% of the total time, while the remainder of the time, i.e. $> 85\%$, was used by the simulation. This ratio of simulation vs. *in situ* data analysis is quite reasonable as the overall run time performance and scalability of the simulation has not been impacted drastically, in particular when considering the large range of analyses performed.

In addition to run time, another primary cost factor is I/O. As part of the *in situ* analyses, histograms are written in an ASCII CSV format, projections in VTK compressed binary VTR format, isosurfaces are rendered and written as PNG images, and statistics are output to standard error stream. Figure 4(b) shows the total I/O cost of the run by category. The total I/O cost in the run is approximately $841MB$. If one were to make the same analysis *post hoc*, one would need to save the particle positions and velocities for each species at each 50 iteration update, resulting in a total of approximately $3.2TB$ written to disk. Thus, by using *in situ* we have reduced our I/O cost by a factor of more than $4000\times$.

Future work: The analysis above revealed a number of interesting phenomena which have focused and the direction of the ion accelerator investigation. In the coming months we will continue to work with the science team on a new and more advanced ion acceleration simulation analysis. Analysis will be driven by the specific needs of the investigation, however it will build on the tools that we have developed in previous work, such as our Yee Cell re-mesher and phase space feature detection and analysis filters. Load balancing is a challenge that is particular to ion accelerator simulation. Initially the particles are concentrated in the target sheet leaving the vast bulk of the simulation domain empty. This creates a load imbalance that severely curtails scalability. For example the 3D runs reported above took 16 hours and we found that adding more cores did not decrease the run time. The science team is testing a dynamic load balancing scheme that will address the issue. We will need to modify the adapter layer in WarpIV to handle data from the new load balancing scheme. In addition the science team has been working with Cray and Intel on threading the simulation code through NERSC's NESAP program. Thus we also need to handle the new data layouts and prepare to run VisIt in its fledgling multi-threaded mode. Our experience with rendering transparent isosurfaces has sparked interest in auto tuned rendering algorithms, hybrid rendering schemes that allow one to explore or tune *in situ* rendered images *post hoc*, as well as improving the efficiency of extract based I/O in order to enable *post hoc* visualization. We are also interested in expanding WarpIV so that scientists could easily swap out any one of the back end *in situ* infrastructures. For instance this would enable *in transit* analysis.

4.2 Accelerator Modeling: IMPACT

Science objective and stakeholders: Accelerators are important tools in many DOE mission critical applications. They are expensive devices that require careful computer modeling in design, construction and operation. The computer modeling process generates a large amounts of data and is another good candidate for *in situ* analysis. The specific accelerator modeling code we have selected for this work is IMPACT, a code for Integrated Map and Particle Accelerator Tracking.

IMPACT is a parallel particle-in-cell code to model the dynamics of multiple charged particle beams in large accelerators. It solves the Vlasov/Poisson equations using a particle-based technique. Scientists use this code to study halo formation, coupling resonance in high intensity beams, microbunching instability in high brightness electron linear accelerators (LINACs), and other beam dynamics. These use cases involve DOE facilities as well as international collaborations, including LBNL, SLAC, ORNL (for SNS), MSU (for FRIB), CERN (for superconducting LINAC and proton synchrotron), J-PARC, among others. The current software development effort of IMPACT is supported partially by the BES advanced accelerator modeling project and partially by the HEP CAMPA project.

The primary outputs from IMPACT are in the form of checkpoint files. Given a simulation with 100 billion particles, each check point file will be 8 TB. It is common for a simulation to produce 1000 of such checkpoint files. A common analysis task is to count the number of *halo particles* outside the center of the particle beam. Counting particles in the halo using the checkpoint files is slow because the checkpoint files are quite large. Counting them *in situ* would make the information available immediately during a simulation run and could also reduce the number of checkpoint files produced.

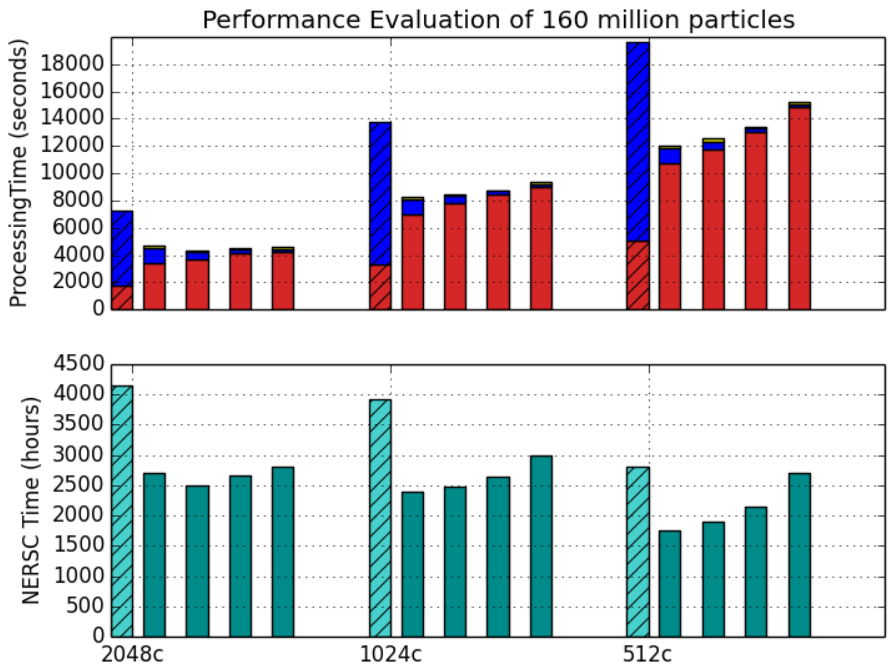


Figure 4: Computing (blue) and histograms (yellow) *in situ* reduces the overall elapsed time as well as the time charged by the computer centers.

Another set of *in situ* tasks that would be useful is computing statistics about the particles, such as various 1D-, 2D-, 3D- and 4D-histograms, quantiles, and statistical moments of various quantities. These statistics are relatively inexpensive to compute, and could be computed *in situ* at a much higher temporal frequency than using the *post hoc* approach. These high-resolution statistical quantities will allow the simulation results to be more directly and accurately compared to the experimental observations.

Accomplishments November 2016 through March 2017: IMPACT is a good test case of SENSEI framework for two special reasons. It is written in Fortran 90 and therefore exercises the compatibility of SENSEI *in situ* API with Fortran. In the recent months, we have converted the computation of dozens of diagnostic variables in IMPACT to be carried out *in situ*. Figure 4 shows that shipping the data to dedicated nodes to compute the diagnostics and histograms not only reduces the overall elapsed time compared to computing all of them inside IMPACT code, but also reduce the total core-hours charged by the computer center.

Future work: The developers of IMPACT are interested in building upon the work on *in situ* diagnostic capability to develop advanced analysis capabilities. Such capability would allow the physicists to explore more complex beam dynamics such as resonance and high-order symmetry.

4.3 Aircraft Design

Science objective and stakeholders: The design of aircraft wings is a complex process, especially when considering active flow control features such as injecting fluid into the boundary layer. Such flow control features can positively affect airfoil performance by re-attaching separated flow, reducing vibrations and virtual reshaping of the lifting surfaces. The first step in the design process is an understanding of these desired influences by scientists and engineers. The next step is trying to optimize airfoil performance through flow control injection placements and wing geometry. This can result in a large number of runs which each requires significant compute capacity in order to properly capture the purposefully created complex physics. In order to get insight in a timely manner, *in situ* methods can be used to reduce the cycle time in examining different designs for favorable performance. Since the design metrics are known *a priori*, initial costs for setting up the *in situ* analysis and visualization can be prorated over the entire set of runs.

A general purpose CFD code called PHASTA (https://github.com/PHASTA/phasta) was used by Ken Jansen and Michel Rasquin of UC Boulder to examine how such a workflow could be enhanced by HPC computing for Boeing. Much of the work focused on time-to-solution issues and in their work they showed that PHASTA scaled well to up to 3 million MPI ranks on Argonne's Mira BG/Q supercomputer. This was done by using 4 MPI ranks per core or 64 MPI ranks for node. Each node also has 16 GB of memory per node which results in on average 256 MB of memory available to each MPI process. The focus of their work was on a simulation of a realistic tail rudder assembly.

Our science stakeholders on this are Ken Jansen from UC Boulder, who is also part of the FASTMath SciDAC institute, and Michel Rasquin from Cenaero in Belgium and UC Boulder.

Figure 5: Zoomed slice through the wing of a PHASTA simulation.

Accomplishments: A major goal of this project is to develop production capabilities for *in situ* analysis and visualization at the leading edge of computing capacity. This means that our tools need to be run at the scales that the leading edge codes run at. Because of this, we chose PHASTA since we knew it was performant at the scales our *in situ* tools needed to work at and would be a good test of that. Additionally, memory consumption is a significant consideration for PHASTA when running on machines such as Mira since there is only 256 MB available per MPI rank. Thus, using PHASTA on Mira will require our tools to scale well while also being memory efficient.

PHASTA was one of the earliest codes to be instrumented with Catalyst. Because of this, it was straight-forward to modify PHASTA to use SENSEI's new API and that was done as part of this work. Another issue was that the original data adaptor was written before any of the zero-copy infrastructure was developed in VTK. While the memory for PHASTA's scalar arrays could be used directly in VTK's data arrays for the initial integration of the data adaptor, the velocity and point coordinate vector arrays had to be deep copied due to PHASTA using a *structure-of-arrays* memory layout. With the newly developed zero-copy functionality in VTK, we modified the data adaptor to use the new zero-copy infrastructure to reduce the *in situ* memory consumption. In order to run efficiently on Mira, we also built a Catalyst edition specific to providing SENSEI's slice analysis adaptor's functionality to reduce library size and thus memory footprint. The slice analysis computes a slice through the domain and then renders an image of the generated slice. An example of this is shown in Figure 5. The resulting static executable size when built with Catalyst

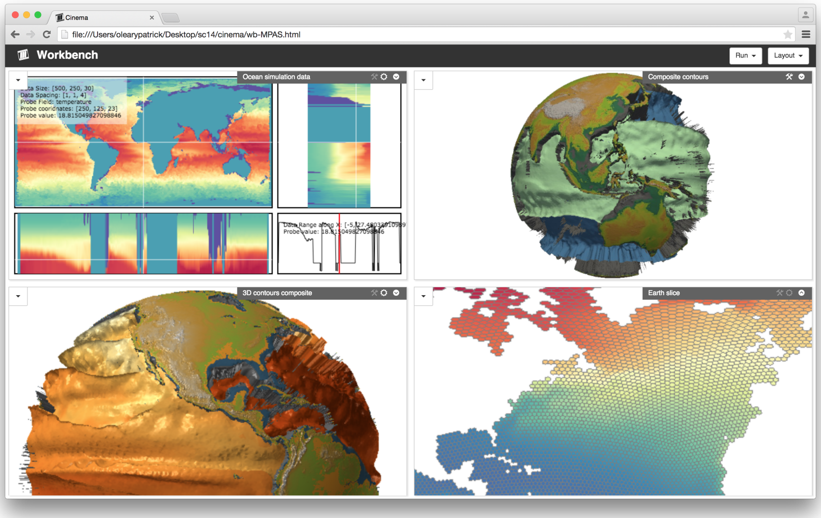


Figure 6: Cinema Workbench demonstrating several approaches for exploration visualization of *in situ* produced MPAS-O data products.

was 153 MB while the executable size without Catalyst was 87 MB.

Since benchmarking PHASTA runs on Mira for 256K and 1M MPI[1] rank runs, we have been analyzing the results and planning our collaborative FY 2017 work. We have reduced the file system introspections during ParaView Catalyst initialization by performing the file IO on the first MPI process and broadcasting that information to the other processes.

Future work: In order to run PHASTA on Mira as efficiently as possible, we will need to reduce the Catalyst memory footprint such that everything can fit within the 256 MB per MPI rank. This will allow us to scale to even larger process counts to help understand the SENSEI *in situ* behavior at full machine scale on Mira.

Ongoing discussions with the Ken Jansen have focused the work on a computational steering functionality in SENSEI.

Additionally, we have a Stretch goal of running full machine scale on Mira later in 2017.

4.4 Climate Science

Science objective and stakeholders: Accelerated Climate Modeling for Energy, or ACME, is designed to accelerate the development and application of fully coupled, state-of-the-science Earth system models for scientific and energy applications. The project initially will examine and try to answer "big science" questions in three areas that drive climate change: the water cycle, biogeochemistry and the cryosphere (areas of the earth where water exists as ice or snow). ACME's scientific program can only be achieved with DOE's Leadership Computing Facilities. They will be applying a wide range of high-performance computing approaches including developing new algorithms specially adapted to these architectures. Only through efficient use of these computers will they be able to model the many physical processes at the fidelity necessary to answer ACME's science questions.

[1]This particular run, exceeding 1M-way concurrency, is the largest-ever *in situ* run, and was published in our SC16 technical program paper.

CAM variables are stored in disk file with a different index order than the one in CPU resident memory because of parallel decomposition. Therefore, moving variables from CPU to disk requires an index reshuffle on the fly. This remap occurs on a distributed architecture (I/O nodes), and from there analysis products are computed when the data is written to disk. Although this scales far greater than using sequential I/O, this may not map well to the architectures of the next-generation DOE's Leadership Computing Facilities. ACME codes like MPS-Ocean and CAM 5.0 evolve 10s to 100s of parameters and are capable of producing 1000s of derived quantities. However, there exists a widening gap between FLOPs and I/O capacity which will make full-resolution, I/O-intensive *post hoc* analysis prohibitively expensive, if not impossible. We believe that by integrating ACME codes with *in situ* I/O mitigation capabilities, we will enable the ACME project to complete their science objectives.

The overall science stakeholder in this project is the are ACME and the Community Earth System Model (CESM), in particular Sandia National Laboratories' Mark Taylor the chief computational scientist for ACME and Argonne National Laboratory's computational climate scientist, Robert Jacob.

Accomplishments: We completing the Catalyst *in situ* adaptor work for the CESM/CAM 5 SE (spectral element) dynamic core last Fall 2015. Prior to this year, we completed development of Catalyst adapters for CAM 5 FD and MPAS–O use cases, preliminary results for visualization pipelines for both codes using Catalyst on up to 8K cores to create Cinema data sets. To date, the codes for ACME are now fully instrumented with Catalyst for *in situ* analysis and visualization.

Future work: For FY2017, we intend to migrate to the SENSEI infrastructure and test scaling of both analysis and visualization pipelines leveraging new multi-core parallel environments available in VTK (VTK SMP and VTK-m).

VTK SMP is a parallel programming environment tailored for scientific visualization algorithms that helps application programmers move to multi-core programming. The SMP (symmetric multiprocessing) framework provides an infrastructure for easy development of shared memory parallel algorithms in VTK. The SMP framework is essentially is a thin abstraction over a number of back ends. Currently, VTK SMP supports four back ends: Sequential (serial execution), Simple (uses vtkMultiThreader for simple parallelization), TBB (based on Intel's TBB) and Kaapi (based on Inria's XKaapi). Only the TBB and Kaapi back ends are suitable for parallel production use. Therefore, we will be leveraging the TBB back end, and using the Simple back end for debugging.

VTK-m is a toolkit of scientific visualization algorithms for emerging processor architectures, and high-performance computing with increasing dependency on accelerators. VTK-m supports the fine-grained concurrency for data analysis and visualization algorithms required to drive extreme scale computing by providing abstract models for data and execution that can be applied to a variety of algorithms across many different processor architectures. We will focus on the second generation of Intel® Xeon Phi™ Product Family, called Knights Landing (KNL) Many Integrated Core (MIC) Architecture provided in the Cori Phase II system.

It is our belief that this combined multi-core/*in situ* work will deliver substantial new capabilities for the ACME scientific program in the coming years.

4.5 Combustion Modeling

Science objective and stakeholders: The desire to scale simulation environments to be able to overlap with real-world experimental environments—in terms of physical size and temporal scale—has been a key component in the progression towards exascale. For many systems, however, one or both of these remains a problem. In the case of the intersection between computational and experimental evaluation of industrial-scale combustion systems, it has recently become possible to begin to directly overlap the two. A key difficulty is in determining whether the flame dynamics in the simulation accurately represent the dynamics of a particular experiment. In addition to the inherently chaotic nature of turbulent combustion, which guarantees that no point-by-point comparison will be possible, there are also issues related to uncertainty in the underlying chemical models and parameters that may cause the simulation and experiment to diverge.

The LESLIE3D code is used in design studies for gas turbine combustors and rocket engines. The simulations solve large eddy simulations of complete real geometry with chemistry and Lagrangian droplets for liquid to gas phase changes and combustion.

The investments from the SENSEI project allow us to impact a wide array of application science stakeholders from a spectrum of industries, labs, and universities including the following: Georgia Tech, United Technologies Research Center, Rocketydyne, Aerojet, Wright-Patterson AFB, Edwards AFB, NASA Glenn, Pratt-Whitney and Solar Turbines.

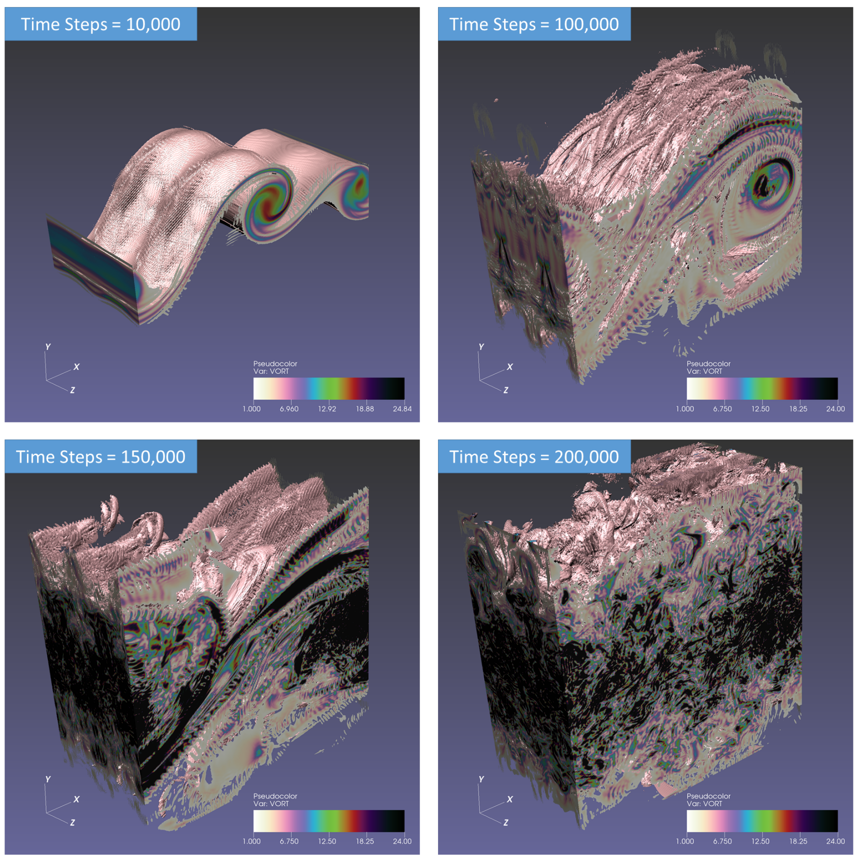


Figure 7: The Evolution of Temporal Mixing Layer from Initial to Vortex Breakdown

Accomplishments: We instrumented AVF-LESLIE with a SENSEI adaptor that calculates vorticity magnitude and exposes data array slices (to remove ghost cells). The SENSEI analysis adaptor was provided a VisIt session file to set up the visualization. The visualization consists of 3 isosurfaces and 3 slice planes of vorticity magnitude. Its purpose is to give a visual reference to the evolution of the turbulent flow features from initial mixing through homogeneous turbulence.

We conducted benchmarks on Titan at Oak Ridge Leadership Class Compute Facility. The scaling studies were performed on a Cartesian grid size of 1025^3 and physical non-dimensional domain sizes of 4π x 4π x 2π. The study used between 8192 and 131,072 cores, using all 16 cores per compute node.

The benchmarks study the strong-scaling characteristics of AVF-LESLIE, and the memory and computational overhead associated with the *in situ* methods and infrastructure.

Each run represents AVF-LESLIE running for 100 time steps, with SENSEI being called at each time step and Libsim analysis every 5 time steps. The study simulates unsteady dynamics of a temporally evolving planar mixing layer (TML). This type of fundamental flow mimics the dynamics encountered when two fluid layers slide past one another and is found in atmospheric and ocean fluid dynamics as well as combustion and chemical processing. The two sliding fluid layers are subject to inviscid instabilities and can evolve from largely 2-d laminar flow into fully developed, 3-d homogeneous turbulent flow as shown by Metcalfe [2]. Figure 7 presents visualizations of the TML flowfield at 10,000, 50,000, 100,000 and 200,000 time steps where the flow evolves from the initial flow field, vortex braids begin to form, wrap and then the flow breakdown leading to homogeneous turbulence, respectively.

Before *in situ* processing was implemented, AVF-LESLIE scaled well up to 16k cores, but efficiency degrades at higher core counts. After SENSEI/Libsim *in situ* rendering was added, the per-iteration time for AVF-LESLIE increased due to the time taken for *in situ* processing. The time taken to initialize SENSEI increases with processor count, largely due to one-time Libsim initialization costs

The analysis time which includes the time to expose data to SENSEI, read a session file, set up plots, perform data extraction, render geometry, create the composited image, and save the image, quickly exceeded the time spent in the solver due to the complexity of the visualization, as shown in Figure 8. *In Situ* analysis time is highly dependent on the complexity of the analysis, or in this case, the nature of visualizations being produced. Over the 100 time step run, the costs of calling Libsim to produce rendered images added an average of 1-1.5 seconds per time step to the solver runs over the numbers of cores tested.

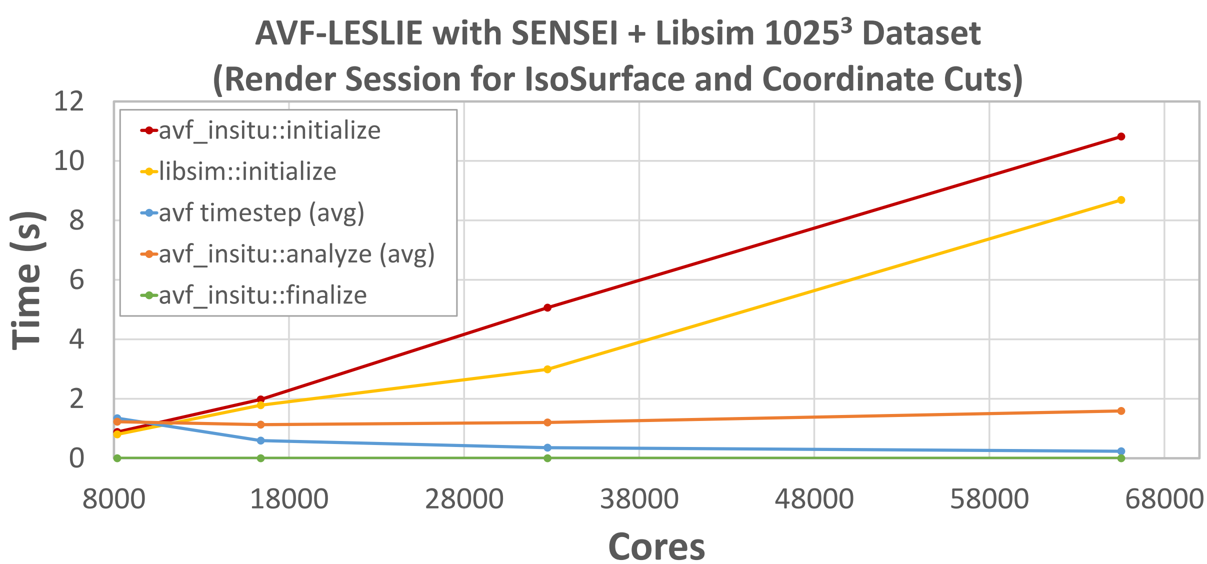


Figure 8: AVF-LESLIE Performance with SENSEI/Libsim *In Situ* Processing

Since the Libsim visualization was complex, it was executed one out of every 5 times in which

[2] Metcalfe, R. W., Orszag, S. A., Brachet, M. E., Menon, S., Riley, J. J. (1987). SECONDARY INSTABILITY OF A TEMPORALLY GROWING MIXING LAYER. Journal of Fluid Mechanics, 184, 207-243.

SENSEI was invoked by the solver. This means that 4/5 times, the SENSEI analysis time was low and the 1/5 times that Libsim analysis was invoked, the time was high. To see the actual costs of calling Libsim to produce the visualizations, AVF-LESLIE reported the time spent in SENSEI analysis for each time step.

It is important to contrast the *in situ* overhead to the traditional *post hoc* workflow. At 1025^3 and 65K core, AVF requires approximately 24 seconds to save a time step of volume data. Therefore, based upon the current overhead numbers, one can afford 3-4 times greater temporal resolution for visualization and analysis in comparison to writing out volume data for *post hoc* processing. This capability is very important for transient data and resolving turbulent flow characteristics. Future work is focused on reducing this *in situ* overhead further and applying it towards extracting further knowledge about the TML flowfield.

Milestones:

- Benchmarked weak and strong scaling of AVF-LESLIE up to 64k core on Edison and 131k core on Titan with SENSEI *in situ* infrastructure

Future work:

- Implement and benchmark autocorrelation *in situ* analysis pipeline with AVF-LESLIE-SENSEI

- Utilize autocorrelation in a large scale combustion and/or temporal mixing layer problem

4.6 Cosmology

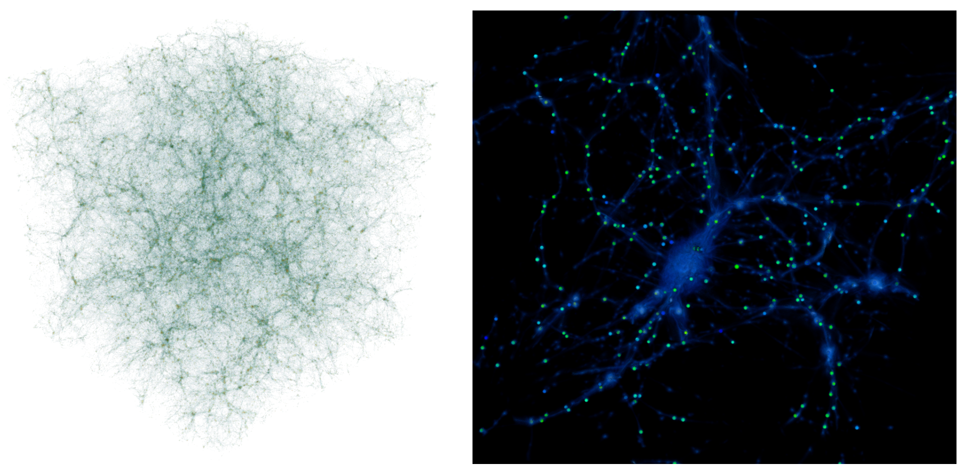


Figure 9: Volume rendered density distribution of cosmological simulation (left). *Halos* are over-densities, i.e., regions of locally higher density, shown as spheres (right).

Science objective and stakeholders: Cosmology aims to determine how the rich structure of today's universe—including stars, galaxies and clusters of galaxies—evolved from an initially almost homogeneous state. This large-scale structure depends on cosmological parameters, such as the properties of dark energy. To determine these cosmological parameters, cosmologists run numerical simulations of multiple-model universes and identify the model that best matches the observed matter distribution. One of the most important analysis tasks for cosmological simulations is identifying structures such as stars, galaxies and clusters of galaxies. These, structures show up as over-densities, or so called *halos*, in the simulation domain.

Simulations by our collaborators use the BoxLib-based Nyx code. These simulations typically produce plot files for *post hoc* analysis for approximately 10 time steps per run that are selected by best estimate. This strategy makes it likely to miss important and interesting events. Furthermore, writing only a few snapshots does not provide sufficient accuracy for feature tracking. *In Situ* analysis will enable analysis at full accuracy to recover a more accurate halo formation history and avoid the risk of missing important events. Furthermore, certain physical models, such as feedback from active galactic nuclei, cannot be correctly implemented without *in situ* halo finding making since subsequent simulation steps need access to detected halos.

The overall science stakeholder in this project is the computational cosmology community, in particular researchers of the LBNL Computational Cosmology Center (C^3) and the Computation-Driven Discovery for the Dark Universe SciDAC. Our work will also benefit observations from the Dark Energy Spectroscopic Instrument (DESI) and the Large Synoptic Survey Telescope (LSST).

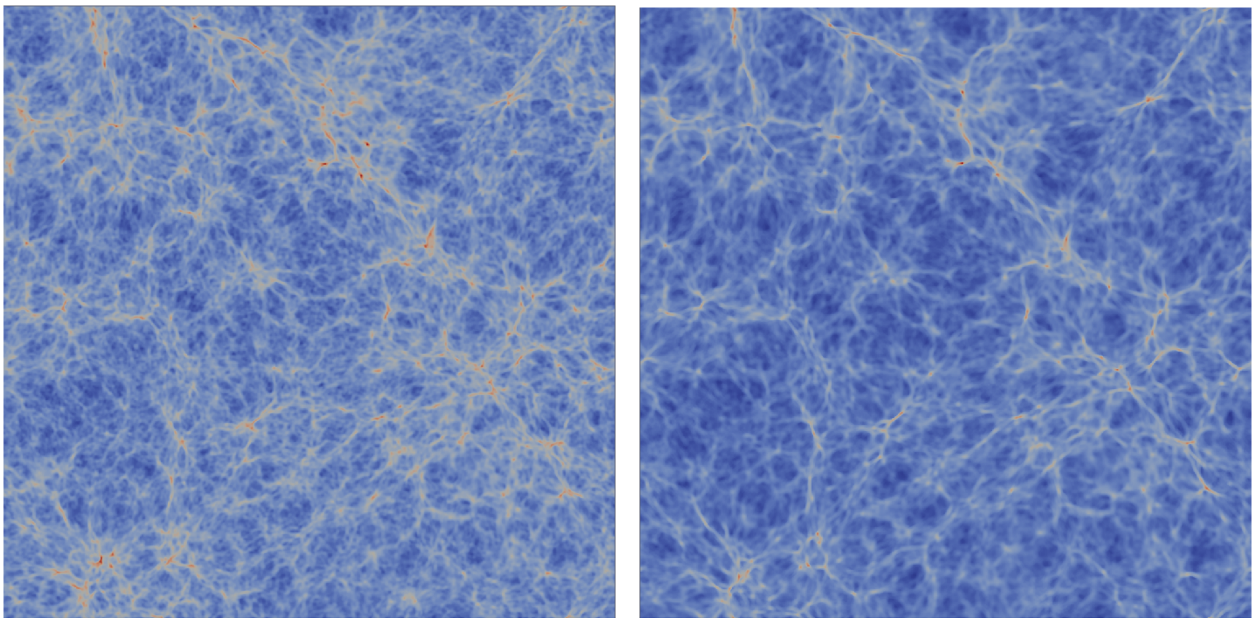


Figure 10: Time steps 200 and 300 of the 1024^3 Nyx Lyman α forest simulation. Simulations often only save every 100th time step. The difference between these time steps is considerable, hampering feature tracking.

Accomplishments:

- Worked with C^3 and Center for Computational Science and Engineering on publication on integrating halo finder into their "sidecar" analysis infrastructure and performing studies about the best split of compute cores between analysis and visualization. This effort resulted in a paper submitted "Computational Astrophysics and Cosmology".
- Developed an interface between Nyx and the SENSEI analysis framework—currently working only on single level data sets, and conducted an extensive study aimed at understanding the time and memory overhead associated with performing *in situ* operations: histogram and performing visualization of 2D slices extracted through 3D domains. The weak scaling study varies over three problem sizes, 1024^3, 2048^3, and 4096^3 mesh points, and at three levels of concurrency, 512, 4096, and 32768 cores. Among the results: the executable size is about 1.6x larger because of the *in situ* infrastructure; the additional runtime required for performing operations *in situ* was negligible; memory required for *in situ* operations ranged from minimal

(histogram) to modest (slice visualization); the biggest win for the *in situ* configuration is being able to entirely avoid costly I/O.
- Developed Henson, a code that relies on position-independent executables and coroutines to run multiple loosely coupled codes (separate executables) either in situ or in transit, depending on an external configuration. Adapted Gadget, a widely used N-body simulation code, to work with Henson and performed a set of analysis tasks (included lightcone construction, entropy estimation, and halo detection) both in situ and in transit.

Future work: During the reminder of the project, we will continue our work in the following areas:
- We will add the ability to return a list of halo positions and masses to our halo finder and investigate using these results to improve the accuracy of simulation results. Furthermore, we will release Reeber2 and our halo finder to the public.
- We will improve the bridge between Nyx/Boxlib and SENSEI, adding the ability to use full AMR hierarchies in VisIt- and ParaView-based analyses.
- We will investigate the feasibility of using time-varying topological analysis for the efficient calculation of merger trees, which track the evolution of halos over time.
- Add support for multiple AMR levels to Nyx/SENSEI interface (FY 2016).
- Test VisIt coupling to Nyx via SENSEI and use it for *in situ* analysis (FY 2016).
- Release Reeber2 to the public (FY 2016).
- Communicate halo finder results back to Nyx and investigate at least one use case for using halos in the simulation (FY 2016/2017).
- Investigate use of time-varying topological analysis for effective calculation of merger trees (FY 2017).

4.7 Materials Science

Science objective and stakeholders: This project is developing tools to gain insight into the 3D structure of block copolymer films that have been produced by directed self-assembly (DSA) on lithographically defined chemically nanopatterned surfaces. Molecular dynamic simulations produce morphologies which in turn are used to simulate scattering patterns. A comparison between experimental and simulated scattering is used, iteratively, to adjust the MD input parameters (materials, interface and boundary conditions and substrate design) until a match is obtained.

Our target code is a custom written molecular dynamic Monte Carlo simulation that implements a Theoretically-Informed Coarse-Grained (TICG) model.At the onset of our project the TICG output a VTK file for later post processing and this 3D VTK file was an ensemble average—variation is added "artificially" during *post hoc* processing. With the entire simulation and feedback loop now implemented with *in situ* analysis, we operate directly on the individual 3D models, and the variation "fits" with the physics. A covariance matrix adaptive (CMA) algorithm was implemented to "close the loop": the output from all the analysis cores is compared with the experimental data and the CMA step adjusts the parameters for the next loop. The TICG is currently running on up to 8000 cores on Mira—a BlueGene/Q system at ANL. The *in situ* analysis provides a mechanism to produce more physically realistic output, thus improving accuracy. Our *in situ* analysis has removed the need for intermediate file I/O. Stakeholders in this project are materials science and engineering as well as the semi-conductor/electronics industry.

Currently, the application team is migrating the code to use LAMMPS to run on larger scales with respect to problem size and systems. We will explore using *in situ* visualization to assist the scientists in understanding the relationship between the input material properties and the resulting scattering patterns.

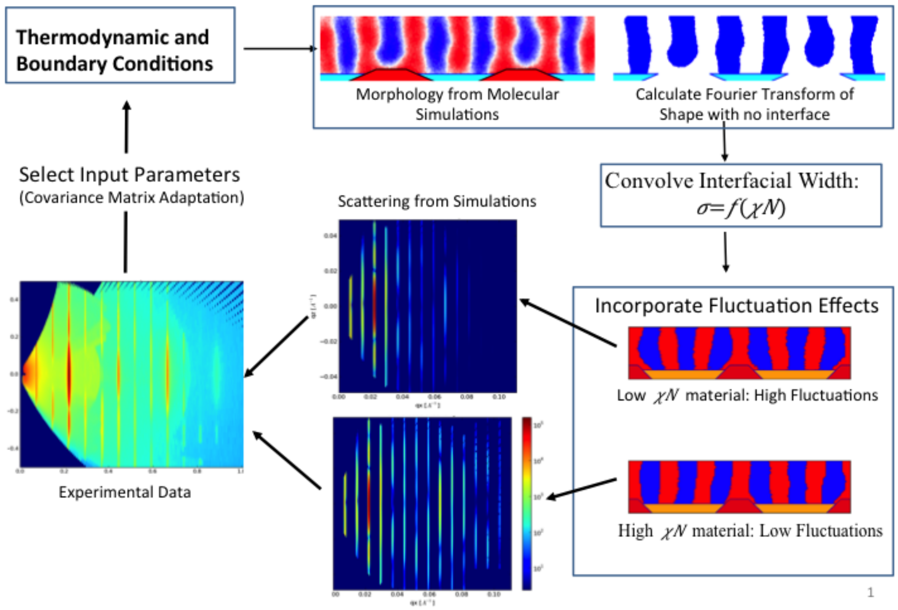


Figure 11: Overview of steps used for comparing simulated scattering to experimental data

Accomplishments:

- The codes has been scaled to tens of thousands of cores – the simulation time increases exponentially with the problem size (see figure 12).
- A paper describing the science is under review (see section 7.1). Using TICG we explored the impact of various material and processing parameters. In our paper, we demonstrate that interface parameters between the chemical pattern and the block copolymer were important to assemble through-film structures. We also show the impact of the width of the guiding strip on the resulting structure.
- (in progress) currently integrating TICG with SENSEI interface.

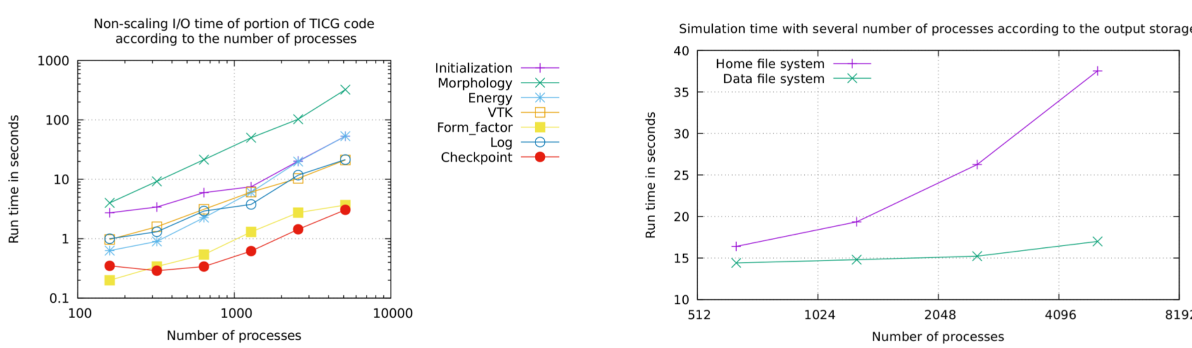


Figure 12: Scaling results for TICG analysis on MIRA at Argonne LCF.

Future work: During the reminder of the project, we will continue our work in the following areas:

- Explore visualization during analysis and additional *in situ* analyses (FY 2017). For example, the change in x-ray scattering as a function of input parameters would be a useful tool for materials scientists.
- Code re-written using available MD codes (using LAMMPS) which has domain decomposition built-in. (FY 2017)
- Complete MD implementation in LAMMPS for larger spatial scales (FY 2017).
- Introduce *in situ* visualization (FY2017).

4.8 Wind Plant Modeling

Science objective and stakeholders.

Electrical power generation via large scale wind turbine plants or farms continues to grow world wide as a cost-effective renewable energy resource. The proper placement (micrositing) of wind turbines within a given resource area must consider the optimal power capture and also the wake interference effects from turbine to turbine. The wake dispersion, momentum losses and turbulent mixing that results in recharging the inflow to each turbine is affected greatly by the inherent turbulence in the atmospheric boundary layer.

Prof. Dmitry Mavriplis at the University of Wyoming has been developing high order finite element discontinuous Galerkin methods for several years. His methodologies been applied to various aerodynamic applications such as airplanes, helicopters and now wind turbines. This methodology utilizes a dual-solver and dual-mesh overset grid framework that couples mixed-order computational fluid dynamics (CFD) solvers. An overset grid framework allows for meshes to be attached to component parts of a system to capture relative motion between these parts. This framework also allows for different solvers to be applied to the different meshes. For a wind turbine plant simulation consisting of several individual wind turbines, curvilinear grids are attached to the wind turbine towers and the individual rotor blades. These ”near body” meshes capture the relative motion of each turbine while resolving the boundary layer over each component. To connect the resulting wake generated by each wind turbine with minimum numerically generated diffusion and to capture the subsequent interactions with downstream turbines, an off body Cartesian mesh with mesh refinement (h-refinement) and order enrichment (p-enrichment) is used.

The objective of this science facing project is to predict the power generated by a wind turbine plant and to capture the unsteady fluid structure interactions occurring within each turbine with the plant within a realistic atmospheric boundary layer (ABL) inflow. The ABL is obtained by injecting an ABL obtained by a meso-scale model at the wind site. For the current case, the researchers utilized the Lillgrund Wind Farm which is located about 10 km off the coast of southern Sweden. This offshore wind farm consists of 48 Siemens Wind Power, each rated at 2.3MegaWatts with a 93m rotor diameter.

Accomplishments: During the current period, we worked with Prof. Mavriplis and Dr. Michael Brazell to instrument their solver with VisIt/Libsim. They have recently completed a study of the Liligrund Wind Plant with uniform inflow. The domain size is 10kmx10km, 1.1 billion degrees of freedom (DOFS) off-body and 340 million DOFS for the near body (7.1 million per turbine). The near-body solver is NSU3D, finite volume second order method. For the off-body grids they used the dg4est code a discontinuous Galerkin mixed p-enrichment of second/third order accurate with 12 levels of h-refinement. The simulations were performed on a new computer systems, Cheyenne, located at National Center for Atmospheric Research / University Corporation for Atmospheric Research (NCAR/UCAR) Computational and Information Systems Lab (CISL). Cheyenne is a 5.34 petaflop SGI ICE XA Cluster with 145,152 Intel Xeon processor cores in 4,032 dual-socket nodes (36 cores/node) and 313 TB of total memory. According to Dr. Brazell: “*In Situ* has helped

Figure 13: The Lilligrund Wind Plant

reduce the amount of 3d data that we need to store, Just the restart files take 10 TB on a 12 hour run. So we cant store volume VTK files. Automation has gone way up, we have scripts to go straight from simulation to animation. Also, *in situ* has helped with debugging, I can detect a problem area, move the slices and get quick viz and diagnose issues. This way I don't need to pull down 3d data and I can just use slices on my local machine."
Future work:

- Simulation of wind plant with atmospheric turbulent inflow boundary layer.
- Implement additional *in situ* pipelines for the wind plant studies.

5 Outreach Overview

Be sure to include forward references to publications (§7.1), presentations (§7.4), service (§7.2), and outreach (§7.3).

6 Software Products/Releases

7 Publications, Presentations, Service, Awards and Outreach

7.1 Publications

7.1.1 Peer-reviewed Journal Articles

1. Jonas Lukasczyk, Ross Maciejewski, Gunther H. Weber, Christoph Garth, and Heike Leitte. Nested tracking graphs. *Computer Graphics Forum (Special Issue, Proceedings Eurographics/IEEE Symposium on Visualization)*, 36(3), 2017. In press.

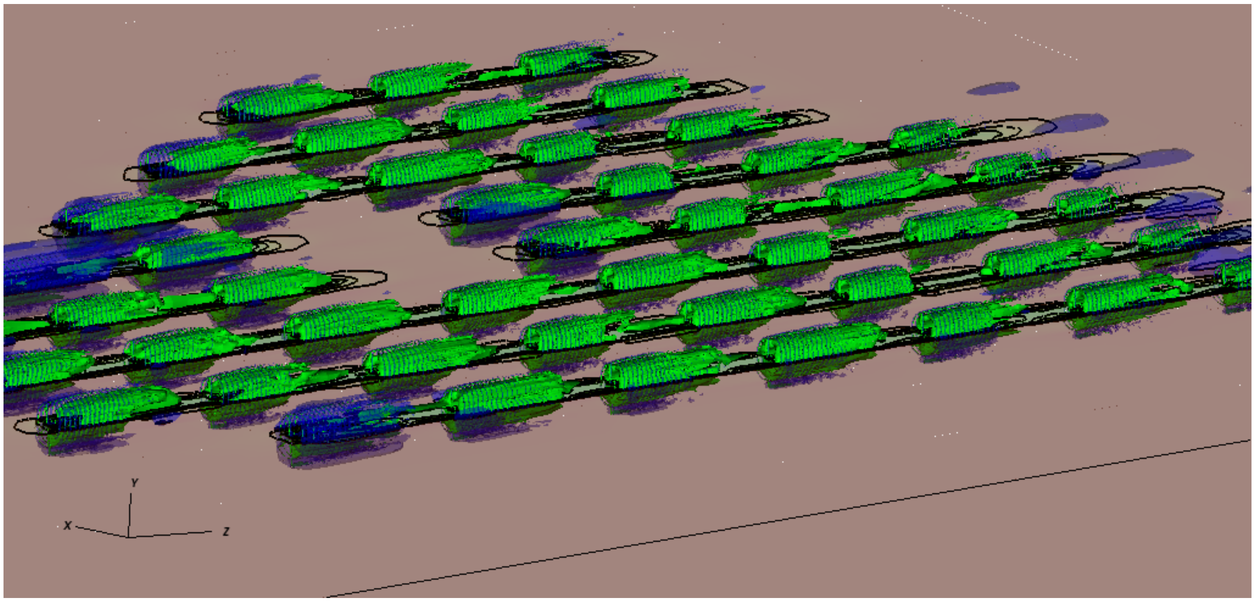


Figure 14: Iso Surfaces Pseudocolor Plots of Velocity Magnitude for the Lilligrund Wind Plant

2. Brad J. Whitlock and Earl P.N. Duque. In Situ Production of Extract Databases for Visualization. *Supercomputing Frontiers and Innovations*, 3(4), 2016.

3. Brian Friesen, Ann Almgren, Zarija Lukić, Gunther H. Weber, Dmitriy Morozov, Vincent Beckner, and Marcus Day. *In Situ* and In-transit Analysis of Cosmological Simulations. *Computational Astrophysics and Cosmology*, 3(4):1–18, August 2016. LBNL-1006104.

4. Andrew C. Bauer, Hasan Abbasi, James Ahrens, Hank Childs, Berk Geveci, Scott Klasky, Kenneth Moreland, Patrick O'Leary, Venkatram Vishwanath, Brad Whitlock, and E. Wes Bethel. *In Situ* Methods, Infrastructures, and Applications on High Performance Computing Platforms, a State-of-the-art (STAR) Report. *Computer Graphics Forum (Special Issue: Proceedings of EuroVis 2016)*, 35(3), June 2016. LBNL-1005709.

5. Oliver Rübel, Burlen Loring, Jean-Luc Vay, David P. Grote, Remi Lehe, Stepan Bulanov, Henri Vincenti, and E. Wes Bethel. WarpIV: In Situ Visualization and Analysis of Ion Accelerator Simulations. *IEEE Computer Graphics and Applications*, 36(3):22–35, May 2016. Selected for presentation in IEEE VIS session of selected Computer Graphica and Applications Papers. LBNL-1005718.

6. Patrick O'Leary, James Ahrens, Sébastien Jourdain, Scott Wittenburg, David H. Rogers, and Mark Petersen. Cinema image-based in situ analysis and visualization of mpas-ocean simulations. *Parallel Computing*, 2015.

7.1.2 Peer-reviewed Conference Papers

1. P. O'Leary, S. Jhaveri, A. Chaudhary, W. Sherman, K. Martin, D. Lonie, E. Whiting, J. Money, and S. McKenzie. Enhancements to VTK enabling scientific visualization in immersive environments. In *2017 IEEE Virtual Reality (VR)*, pages 186–194, March 2017.

2. Utkarsh Ayachit, Andrew Bauer, Earl P. N. Duque, Greg Eisenhauer, Nicola Ferrier, Junmin Gu, Kenneth E. Jansen, Burlen Loring, Zarija Lukić, Suresh Menon, Dmitriy Morozov, Patrick

O'Leary, Reetesh Ranjan, Michel Rasquin, Christopher P. Stone, Venkat Vishwanath, Gunther H. Weber, Brad Whitlock, Matthew Wolf, K. John Wu, and E. Wes Bethel. Performance analysis, design considerations, and applications of extreme-scale in situ infrastructures. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '16, pages 79:1–79:12, Piscataway, NJ, USA, 2016. IEEE Press.

3. E. Wes Bethel, Martin Greenwald, Kerstin Kleese van Dam, Manish Parashar, Stefan M. Wild, and H. Steven Wiley. Management, Analysis, and Visualization of Experimental and Observational Data – The Convergence of Data and Computing. In *Proceedings of the 2016 IEEE 12th International Conference on eScience*, Baltimore, MD, USA, October 2016.

4. Dmitriy Morozov and Zarija Lukić. Master of puppets: Cooperative multitasking for in situ processing. In *Proceedings of High-Performance Parallel and Distributed Computing (HPDC)*, 2016.

5. Brad J. Whitlock, James R. Forsythe, and Steve M. Legensky. In Situ Infrastructure Enhancements for Data Extract Generation. In *54th AIAA Aerospace Sciences Meeting, SciTech 2016*, pages 1–12, January 2016.

6. Anshuman Goswami, Yuan Tian, Karsten Schwan, Fang Zheng, Jeffrey Young, Matthew Wolf, Greg Eisenhauer, and Scott Klasky. Landrush: Rethinking in-situ analysis for gpgpu workflows. In *Cluster, Cloud and Grid Computing (CCGrid), 2016 16th IEEE/ACM International Symposium on*, pages 32–41. IEEE, May 2016.

7. Preeti Malakar, Venkatram Vishwanath, Christopher Knight, Todd Munson, and Michael E. Papka. Optimal Execution of Co-analysis for Large-scale Molecular Dynamics Simulations. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '16, pages 60:1–60:14, Piscataway, NJ, USA, 2016. IEEE Press.

8. Earl P.N Duque, Brad J. Whitlock, Steve M. Legensky, Christopher P. Stone, Reetesh Ranjan, and Suresh Menon. The impact of in situ data processing and analytics upon scaling of cfd solvers and workflows. In *Proceedings of the 27th International Conference on Parallel CFD (ParCFD 2015)*, 2015.

9. Patrick O'Leary, Mark Christon, Sebastien Jourdain, Chris Harris, Markus Berndt, and Andrew Bauer. Hpccloud: A cloud/web-based simulation environment. In *2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom)*, pages 25–33. IEEE, 2015.

10. Preeti Malakar, Venkatram Vishwanath, Todd Munson, Christopher Knight, Mark Hereld, Sven Leyffer, and Michael E. Papka. Optimal scheduling of in-situ analysis for large-scale scientific simulations. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '15, pages 52:1–52:11, New York, NY, USA, 2015. ACM.

11. Jai Dayal, Jay Lofstead, Greg Eisenhauer, Karsten Schwan, Matthew Wolf, Hasan Abbasi, and Scott Klasky. Soda: Science-driven orchestration of data analytics. In *e-Science (e-Science), 2015 IEEE 11th International Conference on*, pages 475–484. IEEE, 2015.

12. Michael Kerber, Dmitriy Morozov, and Arnur Nigmetov. Geometry helps to compare persistence diagrams. In *Proceedings of Algorithm Engineering and Experiments (ALENEX)*, 2016.

13. Penporn Koanantakool, Ariful Azad, Aydin Buluc, Dmitriy Morozov, Sang-Yun Oh, Leonid Oliker, and Katherine Yelick. Communication-avoiding parallel sparse-dense matrix-matrix multiplication. In *Proceedings of International Parallel and Distributed Processing Symposium (IPDPS)*, 2016.

14. Andrew Wissink, Jayanarayanan Sitaraman, Buvaneswari Jayaraman, Beatrice Roget, Vinod K. Lakshminarayan, Mark A. Potsdam, Rohit Jain, Andrew Bauer, and Roger Strawn. Recent Advancements in the Helios Rotorcraft Simulation Code. In *54th AIAA Aerospace Sciences Meeting, SciTech 2016*, pages 1–22, January 2016.

15. Ryan Lewis and Dmitriy Morozov. Parallel computation of persistent homology using the blowup complex. In *ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 323–331, Portland, OR, June 2015. LBNL-177165.

16. Manolis Doxastakis, Hyo Seon Suh, Xuanxuan Chen, Paulina A. Rincon Delgadillo, Lingshu Wan, Lance Williamson, Zhang Jiang, Joseph Strzalka, Jin Wang, Wei Chen, Nicola Ferrier, Abelardo Ramirez-Hernandez, Juan J. de Pablo, Roel Gronheid, and Paul Nealey. Grazing-incidence small angle x-ray scattering studies of nanoscale polymer gratings. In *Proc. SPIE 9424, Metrology, Inspection, and Process Control for Microlithography XXIX*, March 2015.

17. J. Dayal, K. Schwan, M. Wolf, G. Eisenhauer, J. Lofstead, H. Abbasi, and S. Klasky. Soda: Science-driven orchestration of data analytics. In *Proceedings of the IEEE International Conference on eScience (eScience'15)*, 2015.

18. J. Ahrens, S. Jourdain, P. O'Leary, J. Patchett, D. H. Rogers, , and M. Petersen. An image-based approach to extreme scale in situ visualization and analysis. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC '14)*, pages 424–434, Piscataway, NJ, USA, 2014. IEEE Press.

19. J. Ahrens, S. Jourdain, P. O'Leary, J. Patchett, D. H Rogers, M. Petersen, A. Bauer, P. Fasel, F. Samsel, and B. Boeckel. In situ mpas-ocean image-based visualization. In *Visualization & Data Analytics Showcase, International Conference for High Performance Computing, Networking, Storage and Analysis (SC '14)*, 2014. http://sc14.supercomputing.org/program/sc14-archive.

7.1.3 Peer-reviewed Workshop Papers

1. Erich Lohrmann, Zarija Lukic, Dmitriy Morozov, and Juliane Mueller. Programmable in situ system for iterative workflows. In *Proceedings 21st Workshop on Jobs Scheduling Strategies for Parallel Processing (JSSPP 2017)*, May/June 2017. In press.

2. Hamish A. Carr, Gunther H. Weber, Christopher M. Sewell, and James P. Ahrens. Parallel peak pruning for scalable SMP contour tree computation. In *Proceedings of the 6th IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, pages 75–84, October 2016. Best paper award.

3. Chris Harris, Patrick O'Leary, Michael Grauer, Aashish Chaudhary, Chris Kotfila, and Robert O'Bara. Dynamic Provisioning and Execution of HPC Workflows Using Python. In *Proceedings of the 6th Workshop on Python for High-Performance and Scientific Computing*, PyHPC '16, pages 1–8, Piscataway, NJ, USA, 2016. IEEE Press.

4. Utkarsh Ayachit, Brad Whitlock, Matthew Wolf, Burlen Loring, Berk Geveci, David Lonie, and E. Wes Bethel. The SENSEI Generic in Situ Interface. In *Proceedings of the 2nd Workshop on In Situ Infrastructures for Enabling Extreme-scale Analysis and Visualization*, ISAV '16, pages 40–44, Piscataway, NJ, USA, November 2016. IEEE Press.

5. Andrey Ovsyannikov, Melissa Romanus, Brian Van Straalen, Gunther H. Weber, and David Trebotich. Scientific Workflows at Datawarp-speed: Accelerated Data-intensive Science Using NERSC's Burst Buffer. In *Proceedings of the 1st Joint International Workshop on Parallel Data Storage & Data Intensive Scalable Computing Systems*, PDSW-DISCS '16, pages 1–6, Piscataway, NJ, USA, 2016. IEEE Press.

6. D. Pugmire, J. Kress, J. Choi, S. Klasky, T. Kurc, R. M. Churchill, M. Wolf, G. Eisenhower, H. Childs, K. Wu, A. Sim, J. Gu, and J. Low. Visualization and Analysis for Near-Real-Time Decision Making in Distributed Workflows. In *2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 1007–1013, May 2016.

7. Utkarsh Ayachit, Andrew Bauer, Berk Geveci, Patrick O'Leary, Kenneth Moreland, Nathan Fabian, and Jeffrey Mauldin. ParaView Catalyst: Enabling in situ data analysis and visualization. In *Proceedings of ISAV 2015: In Situ Infrastructures for Enabling Extreme-scale Analysis and Visualization*, Austin, TX, November 2015.

8. Wahid Bhimji, Debbie Bard, Melissa Romanus, David Paul, Andrey Ovsyannikov, Brian Friesen, Matt Bryson, Joaquin Correa, Glenn K. Lockwood, Vakho Tsulaia, Suren Byna, Steve Farrell, Doga Gursoy, Chris Daley, Vince Beckner, Brian Van Straalen, David Trebotich, Craig Tull, Gunther H. Weber, Nicholas J. Wright, Katie Antypas, and Prabhat. Accelerating Science with the NERSC Burst Buffer Early User Program. In *CUG2016 Proceedings*, 2016. Best paper award, in press.

7.1.4 Panels – Peer Reviewed Abstracts

1. Andrew C. Bauer, E. Wes Bethel, Brad Whitlock, and Erich Lohrmann. In Situ Methods and Infrastructures: Fast Insight Through Smarter Computing. In *SIAM Computational Sciences and Engineering (CSE) Minisymposium 74*, Atlanta, GA, USA, February 2017.

2. E. Wes Bethel (organizer, moderator, panelist), Hank Childs, Ken Moreland, Dave Pugmire, Matt Larsen, and Matthieu Dorier. In Situ Efforts and Challenges in Large Data Analysis and Visualization. In *IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, Baltimore, MD, USA, October 2016.

3. E. Wes Bethel, Patrick O'Leary, Venkat Vishwanath, and Matthew Wolf. *In Situ* Methods and Infrastructure: Answers Without All the I/O. In *SIAM Parallel Processing 2016 (PP16) Minisymposium MS66*, Paris, France, April 2016. (Organizer: Bethel).

4. E. Wes Bethel (organizer), Patrick O'Leary, John Clyne, Venkat Vishwanath, and Jacqueline Chen. In Situ Methods: Hype or Necessity? In *IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC15)*, Austin, TX, USA, November 2015. (Organizer: Bethel).

7.1.5 DOE Workshop Reports

1. E. Wes Bethel and Martin Greenwald (eds.). Report of the DOE Workshop on Management, Analysis, and Visualization of Experimental and Observational data – The Convergence of

Data and Computing. Technical report, Lawrence Berkeley National Laboratory, Berkeley, CA, USA, 94720, May 2016. LBNL-1005155.

7.1.6 Peer-reviewed Book Chapters

1. Patrick Oesterling, Christian Heine, Gunther H. Weber, Dmitriy Morozov, and Gerik Scheuermann. Computing and visualizing time-varying merge trees for high-dimensional data. In *Topological Methods in Data Analysis and Visualization IV (Proc. TopoInVis 2015)*. 2015. Accepted with minor revisions. Received best paper award at workshop.

7.1.7 Publications Pending Publication, In Review, Submissions

1. Gurdaman Khaira, Manolis Doxastakis, Alec Bowen, Jiaxing Ren, Hyo Seon Suh, Tamar Segal-Peretz, Chun Zhou, Xuanxuan Chen, Adam Hannon, Nicola Ferrier, Venkatram Vishwanath, Daniel Sunday, Roel Gronheid, Joe Kline, Juan de Pablo, and Paul Nealey. Derivation of multiple co-varying material and process parameters using physics-based modeling of x-ray data. (under review), 2017.

7.2 Service

7.2.1 Conference/Workshop Chair or Co-chair

1. ISAV 2017: In Situ Infrastructures for Enabling Extreme Scale Analysis and Visualization. SC17 workshops, November 2017, Denver, Colorado, USA. (Bethel, Duque, Ferrier, O'Leary, Weber, Wolf) ***Any updates?***

2. Posters Co-Chair, IEEE Visualization (SciVis), Phoenix, Arizon, USA, October 2017. (Weber)

3. ISC Workshop on In-Situ Visualization 2017, Frankfurt, Germany, June 2017 (Weber).

4. ISAV 2016: In Situ Infrastructures for Enabling Extreme Scale Analysis and Visualization. SC16 workshops, November 2016, Salt Lake City, Utah, USA. (Bethel, Duque, Ferrier, O'Leary, Weber, Wolf)

5. Posters Co-Chair, IEEE Visualization (SciVis), Baltimore, Maryland, USA, October 2016. (Weber)

6. Minisymposium on In-Situ Processing for Computational Fluid Dynamics Data Analysis and Visualization, 29th International Conference on Parallel Computational Fluid Dyanamics 2017 (PARCFD2017), Glasgow, Scotland, UK, May 15-17 2017. (Duque and Bethel)

7. High Performance Data Analysis and Visualization (HPDAV), an International Parallel and Distributed Processing Symposium Workshop (IPDPS), May 23, 2016. Chicago, IL, USA. (Bethel)

8. DOE Computer Graphics Forum. April 26-28, 2016, Pacific Grove, CA. Site chair. (Bethel)

9. Visualization Co-Chair, 11th International Symposium on Visual Computing, Las Vegas, Nevada, December 2015. (Weber)

10. ISAV 2015: In Situ Infrastructures for Enabling Extreme Scale Analysis and Visualization. SC15 workshops, Monday 16 Nov 2015, Austin, TX, USA. (Bethel, Vishwanath, Weber, Wolf)

11. LDAV 2015: IEEE Visualization Large Scale Data Analysis and Visualization Symposium, Chicago, Illinois, October 2015. General Chair, 2015. (Vishwanath)

12. SIAM PP16 Minisymposium MS66: *In Situ* Methods and Infrastructure: Answers Without All the I/O. April 15, 2016, Paris, France. (Bethel, O'Leary, Vishwanath, Wolf)

13. Eurographics Symposium on Parallel Graphics and Visualization. June 6–7, 2016, Gröningen, the Netherlands. (Bethel)

14. Data Management, Visualization, and Analysis of Experimental and Observational Data. DOE/ASCR Workshop on Data Management, Visualization, and Analysis of Experimental Data. September 29 – October 1, 2015. Bethesda, MD, USA. (Bethel)

15. Visualization and Data Analysis (VDA) Conference (sponsor: IS&T/SPIE). Conference co-chair, February 2015. San Francisco, CA, USA. (Bethel)

16. U. S. Department of Energy, Office of Science, Advanced Scientific Computing Research. Scientific Data Management, Analysis, and Visualization, Program-wide PI Meeting. Jan 13-15, 2015, Walnut Creek, CA. (Bethel)

17. Workshop Co-Chair, Workshop Visualization in Environmental Sciences (EnvirVis), Cagliari, Sardinia, Italy, May 2015. (Weber)

7.2.2 Program Committee

1. Siggraph Asia 2017, Symposium on Visualization. (Bethel)

2. SC17 Visualization Showcase. (Bethel)

3. Topology-Based Methods in Visualization (TopoInVis) 2017, Tokyo, Japan, February 2017. (Weber)

4. Future Trends in Nuclear Physics Computing. Thomas Jefferson National Laboratory, Newport News, VA. May 2017. (Bethel)

5. EnvirVis 2017, part of EuroVis 2017. June 2017. (Bethel)

6. EuroVis Short Papers, Barcelona, Spain, June 2017. (Weber)

7. IEEE International Conference on High Performance Computing, Data, and Analytics (HiPC) 2016 Workshop "High Performance Computing and Big Data in Molecular Engineering," December 2016. (Weber)

8. ISAV 2016: In Situ Infrastructures for Enabling Extreme Scale Analysis and Visualization. SC16 workshops, November 2016, Salt Lake City, UT, USA. (Ayachit, Loring, Morozov, Sim, Whitlock, Wu)

9. SC:IEEE/ACM International Conference for High Performance Computing, Networking, Storage and Analysis (SC 2016), Salt Lake City, Utah, USA, November 2016. (Vishwanath)

10. The 7th International Workshop on Data-intensive Distributed Computing (DIDC'16), Kyoto, Japan, June 1, 2016. (Vishwanath)

11. IEEE Symposium on Large Data Analysis and Visualization 2016 (Bethel, O'Leary).

12. Program Committee, ISC Workshop on In-Situ Visualization 2016, Frankfurt, Germany, June 2016. (Weber)

13. International Program Committee, EuroVis Papers & Short Papers, Groningen, Netherlands, June 2016. (Weber)

14. 9th IEEE/ACM International Conference on Utility and Cloud Computing (UCC 2016), Shanghai, China, December 2016. (Sim)

15. Fifth IEEE International Workshop on Cloud Computing Systems, Networks, and Applications (CCSNA'16), in conjunction with IEEE Global Communications Conference, Washington D.C., USA, December 2016. (Sim)

16. 33rd International Conference on Massive Storage Systems and Technology, May 2017. (Wolf)

17. 37th IEEE International Conference on Distributed Computing Systems. June 2017 (Wolf)

18. Supercomputing16 Birds-of-a-Feather (Bauer)

19. Program Committee, Supercomputing 2016, Salt Lake City, UT, USA, November 2016. (Vishwanath)

20. Program Committee, International Conference on Massive Storage Systems and Technology, 2016 (Wolf)

21. IEEE Visualization Large Scale Data Analysis and Visualization Symposium, International Program Committee, 2015. (Bethel, O'Leary, Ayachit)

22. EuroVis/EnvrVis: Visualization in Environmental Sciences: 2015. (Bethel)

23. IEEE Visualization Technical Program Committee, 2015. (Bethel)

24. ACM Siggraph Asia, Visualization in High Performance Computing Technical Program Committee, 2015. (Bethel, Weber)

25. Visualization and Data Analysis, 2016. (Morozov)

26. IEEE GC 2015 Workshop on Cloud Computing Systems, Networks, and Applications - Fourth International, CCSNA15. (Sim)

27. IEEE International Conference on Utility and Cloud Computing, UCC2015. (Sim)

28. SC15 Scientific Visualization Showcase, Austin, Texas, November, 2015. (Ayachit)

29. ISAV 2015: In Situ Infrastructures for Enabling Extreme Scale Analysis and Visualization. SC15 workshops, Monday 16 Nov 2015, Austin, TX, USA. (Ayachit, Duque, Ferrier, Loring, Morozov, O'Leary, Sim, Whitlock, Wu)

30. SSDBM 2015 27th International Conference on Scientific and Statistical Database Management. June 29 - July 1, 2015, San Diego, CA. (Wu)

31. IEEE/ACM BDC 2015, December 7-10, 2015 – St. Raphael Resort, Limassol, Cyprus (Wu)

32. International Parallel and Distributed Processing Symposium (IPDPS) 2015. (Bethel)

33. EuroVis/EnvrVis: Visualization in Environmental Sciences: 2015. (Bethel)
34. Topology-Based Methods in Visualization (TopoInVis), 2015 (Morozov, Weber).
35. EuroVis, Cagliari, Sardinia, Italy, May 2015 (Weber).
36. EuroVis Short Papers Program Committee, Cagliari, Sardinia, Italy, May 2015 (Weber).
37. International Conference on High Performance Computing (HiPC) 2015 (Vishwanath).

7.2.3 Editorial Board Member/Reviewer

1. Eurovis 2017 technical program (Bethel).
2. IEEE Transactions on Visualization and Computer Graphics. 2015, 2017 (Weber).
3. Computer Graphics Forum: 2016 (Weber)
4. ACM SIGGRAPH Asia: 2016. (Bethel)
5. IEEE Visualization Technical Program: 2016. (Bethel, Weber)
6. International Journal of Grid and Distributed Computing: 2016 (Sim)
7. Foundations of Computational Mathematics: 2016 (Morozov)
8. ACM–SIAM Symposium on Discrete Algorithms: 2015, 2017 (Morozov)
9. SIAM Meeting on Algorithm Engineering and Experiments: 2017 (Morozov)
10. SIAM Journal on Applied Algebra and Geometry (Morozov)
11. IEEE Transactions on Visualization and Computer Graphics: 2015 (Weber)
12. Computational Geometry: Theory and Applications, 2015–2017 (Morozov)
13. Neural Information Processing Systems (NIPS'15) (Morozov)
14. International Journal of Grid and Distributed Computing (Sim)
15. Journal of Parallel and Distributed Computing (Sim)
16. ACM Siggraph Asia: 2015, 2016. (Bethel)
17. IEEE Visualization Technical Program: 2015, 2016. (Bethel, Weber)
18. IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPS): 2015.
19. IEEE Visualization Symposium on Large Scale Data Analysis and Visualization: 2015. (Bethel)
20. Journal of Neuroscience Methods. (Morozov)
21. Nature Communications. (Morozov)
22. IEEE Pacific Visualization, 2015 (Morozov, Weber).
23. IEEE VisWeek, Scientific Visualization, 2015 (Weber).

24. BMC Bioinformatics 2014 (Weber).
25. International Journal of HPC Applications 2015 (Wolf).
26. Pattern Recognition Letters, 2014 (Morozov).
27. ACM Symposium on Computational Geometry, 2015 (Morozov).
28. Visualization and Data Analysis (VDA), 2015 (Morozov).
29. International Symposium on Symbolic and Algebraic Computation (ISSAC), 2015 (Morozov).

7.2.4 Journal Editor

1. IEEE Computer Graphics and Applications, Special Issue on High Performance Visualization and Analysis. May-June 2016. E. Wes Bethel and Kelly Gaither, editors.
2. Journal of Applied and Computational Topology (Morozov)

7.2.5 Proposal Reviewer

1. Department of Energy Early Career Research Program, Reviewer, 2017. (Weber)
2. NSF IIS Review Panel, January 2017 (Bethel).
3. Austrian Science Fund: 2016 (Bethel).
4. NSF Panel: 2016 (Wolf)
5. Department of Energy Early Career Research Program, Reviewer, 2016 (Weber).
6. National Science Foundation, Division of Information & Intelligent Systems, Panelist, 2016 (Weber).
7. National Science Foundation, Panelist, 2016 (Wolf)
8. NSF Panelist, March 2015. (Ferrier)
9. NSF Panelist, Spring 2015. (Morozov)
10. NSF Panelist, Spring 2015. (Wolf)

7.2.6 Service to DOE

1. DOE/ASCR Exascale Requirements Review Series. Rockville MD, September 2016. Report release target: Spring 2017. (Bethel)
2. DOE/NP Future Trends in Nuclear Physics Computing, May 2–5 2017, Thomas Jefferson National Accelerator Facility, Newport News, VA. (Bethel, organizing committee and speaker)
3. Sustainable Pathways Program, Reviewer. December 2016. (Bethel)
4. DOE Workshop on Exascale Requirements Review for Nuclear Physics. June 15-17, 2016, Gaithersburg MD. Bethel role: observer and participant, contributed to discussions in plenary and breakout sessions. Used opportunity to forge closer ties with the NP community for the purposes of forming collaborations and informing NP colleagues of capabilities and directions in ASCR.
5. DOE INCITE Reviewer, 2016 (Vishwanath).

7.2.7 External Program Review

1. DOE/BER Earth Systems Grid Federation Program Review. Potomac, MD. June 2017. (Bethel)

7.2.8 Student Thesis/Dissertation Committee

1. PhD Committee member/examiner for Ms. Hoa Nguyen, University of Utah. Fall 2016. (Bethel)
2. PhD committee member (examiner) for Pawin Vongmasa, Stanford University, September 2015 (Morozov).
3. PhD committee member for Lingfei (Teddy) Wu, College of William and Mary, December 2015 (Wu)
4. Ms. Hoa Nguyen, PhD Qualifying Examination Committee, Computer Science, University of Utah, April 2016. (Bethel)
5. Ryan Lewis, PhD Examination Committee, Institute for Computational and Mathematical Engineering, Stanford University, May 2016. (Morozov)
6. PhD committee member (chair) for Liting Hu, Georgia Tech, May 2016 (Wolf)
7. PhD committee member (chair) for Dipanjan Sengupta, Georgia Tech, May 2016 (Wolf)
8. PhD Qualifying Examination Committee Member, Murugesan, S., Computer Science, University of California, Davis, August 2015 (Weber).

7.3 Outreach

7.3.1 SC17 ISAV 2017 Workshop

Continuing on the success of the SC15 and 2016 workshops, we will have a similar workshop at SC17. The workshop proposal was submitted and the workshop has been accepted for Sunday November 12, 2017. The workshop details are available at the following url http://vis.lbl.gov/Events/ISAV-2017/.

7.3.2 SC17 SENSEI *In Situ* Tutorial Proposal

In addition to the SC17 workshop, we submitted a SC17 tutorial proposal entitled "SENSEI Cross-Platform View of In Situ Analytics". The presenters include Bethel, Bauer, Vishwanath, Whitlock and Wolf from the SENSEI project.

7.3.3 Parallel CFD 2017 Minisymposium

We organized a minisymposium on *in situ* topics to be held at the Parallel CFD 2017 event, located in Glasgow, Scotland during 15-17 May 2017. This audience is of interest to the SENSEI project because of our work with CFD code team projects (e.g., Phasta), which focus on topics aligned with the DOE mission and national interests, such as the engineering of airframes and heavy launch vehicles. It is an opportunity to further engage the international HPC PCFD community.

SENSEI members Duque and Bethel are co-organizers, and the minisymposium will include speakers from SENSEI (W. Bethel, E. Duque), from JAXA, the Japan Aerospace Exploration Agency (S. Tsutsumi), from INRIA (G. Antoniu), and Oak Ridge National Laboratory (D. Pugmire).

Further information about the event is available online at https://www.strath.ac.uk/engineering/parcfd2017/.

7.3.4 SIAM CSE 2017 Minisymposium

We organized a minisymposium at the 2017 SIAM Conference on Computational Science and Engineering, held 27 February – 3 March 2017 in Atlanta, GA. The audience at this venue is well aligned with SENSEI for it is a meeting of computational science researchers and practitioners, many of whom design and run codes on large-scale HPC resources and whom can benefit from a better understanding of issues related to *in situ* processing. Our minisymposium covered topics ranging from an overview of issues motivating *in situ*, a survey of current tools and capabilities, examples of *in situ* being used in practice, and an introduction to the SENSEI *in situ* interface.

From the SENSEI team, A. Bauer and W. Bethel were co-organizers, and speakers included A. Bauer, W. Bethel, E. Duque, and E. Lohrmann (from Georgia Tech).

Information about CSE 2017 is located at https://www.siam.org/meetings/cse17/; information about our specific minisymposium is located at http://meetings.siam.org/sess/dsp_programsess.cfm?SESSIONCODE=61113

7.3.5 SC16 ISAV Workshop

Continuing on the success of the SC15 workshop, we held a workshop at SC16 on Sunday November 13, 2016. Based on feedback from SC15, the SC16 workshop was extended to a full day. As in SC15, we included an "Early Career Program Committee" to support the next generation of researchers. The workshop details are available at the following url http://vis.lbl.gov/Events/ISAV-2016/.

The ISAV-16 workshop opened with a keynote speaker. Over the course of the day, nine short papers and four lightenting talks were presented. The day concluded with a panel discussion. All sessions were well attended and there was active discussion and questions after the talks and panel.

The workshop attendance was at room capacity for most of the day. The evaluations were generally positive. The most frequent complaint was the lack of space. Most evaluations were complimentary of all elements of the format (longer and shorter talks, panel).

The program is available at the URL above and the proceedings are published in the ACM Digital Library.

7.3.6 SC16 ParaView Tutorial

An introductory full-day tutorial called "Large Scale Visualization with ParaView" was given at SC16. This tutorial included a section on using ParaView Catalyst.

7.3.7 SIAM PP 2016 Minisymposium

We organized a minisymposium at the 17th SIAM Conference on Parallel Processing for Scientific Computing held in Paris, France from April 12 - 15, 2016. Our minisymposium was titled "In Situ Methods and Infrastructure: Answers Without All the I/O" and was held on April 15, 2016. We had two talk presented by Wes Bethel (LBNL) and Julien Jomie (Kitware).

The details are available at the following url http://meetings.siam.org/sess/dsp_programsess.cfm?SESSIONCODE=22113.

7.3.8 SC15 Workshop

In an effort to reach out to a broad cross-section of the scientific HPC community, we submitted a workshop proposal to SC15, the focus of which is *in situ* infrastructures, methods, and their applicability to science. The workshop proposal was accepted, and was held on, Monday 16 November 2015, in the afternoon.

The workshop name is "ISAV 2015: First Workshop on In Situ Infrastructures for Enabling Extreme-scale Analysis and Visualization", and the workshop url is http://vis.lbl.gov/Events/ISAV-2015/.

This workshop brought together researchers, developers and practitioners from industry, academia, and government laboratories who use *in situ* methods in extreme-scale, high performance computing. The goal is to present existing in-situ infrastructures, reference examples in a range of science and engineering applications, to discuss topics like opportunities presented by new architectures; existing infrastructure needs, requirements, and gaps; and experiences to foster and enable *in situ* analysis and visualization. The program is available at the URL above and the proceedings are published in the ACM Digital Library.

We used this opportunity to train the next generation of reserchers (students and postdocs) in reviewing papers by creating an "Early Carreer Program Committee" and having in-depth discussion of reviews with the committee.

7.3.9 SC15 *In Situ* Panel Proposal

In addition to the SC15 workshop, we submitted a SC15 panel proposal entitled "In Situ Methods: Hype or Necessity?". The panelists include Bethel, O'Leary, and Vishwanath from the SENSEI project, Jacqueline Chen (SNL-NM) to represent a "why computational science needs *in situ*" perspective, and John Clyne (NCAR) to represent the "why *in situ* is over-hyped" perspective. The panel was held on, Friday 21 November 2015, in the morning (10:30a – noon).

7.3.10 SC15 Libsim BoF

To build community for the Libsim *in situ* infrastructure, we held a birds of a feather session at SC15 called "Scalable In Situ Data Analysis and Visualization Using VisIt/Libsim". This session presented successful uses of Libsim and shared *in situ* experiences and lessons learned. The format consisted of lightning talks that speak to Libsim features and usage, followed by community discussions of user experiences, applications, and best practices.

7.3.11 SC15 ParaView Catalyst Tutorial

An introductory half-day tutorial called "Getting Started with In Situ Analysis and Visualization Using ParaView Catalyst" was given at SC15. This tutorial was focused on users aspects of *in situ* technologies and how to leverage them efficiently in parallel. Hands-on exercises were used to show attendees how to generate Python scripts for outputting data extracts, images and a Cinema database from Catalyst.

7.3.12 University at Buffalo CDSE Days, SC15 and IEEE VIS 2015 ParaView Tutorials

Introductory half-day tutorials called on ParaView were given at University at Buffalo's Computational and Data-Enabled Science and Engineering Days workshop, IEEE VIS 2015 and SC15. The tutorials were focused on new to intermediate ParaView users and provided a hands-on approach to learning how to use the ParaView GUI and Python API. Additionally, information on running ParaView efficiently in parallel was presented in each.

7.3.13 Outreach for under-represented groups in CS

In 2015 and 2016, N. Ferrier was a mentor for Argonne Outreach events "Introduce a Girl to Engineering" and "Science Careers in Search of Women" that bring middle school and high school girls to Argonne to learn about research in science and engineering.

7.4 Talks and Invited Presentations

1. Patrick O'Leary. Poised for a disruptive transformation. In *University of Idaho/Idaho National Laboratory and the Center for Advanced Energy Studies Colloquium*, Idaho Falls, Idaho, March 2017.

2. Gunther H. Weber. Topological analysis at the extreme scale: Finding features in large data sets/re-thinking visualization for the exascale (keynote). In *Visualization and Data Analysis 2017, 2017 IS&T International Symposium on Electronic Imaging (EI 2017)*, Burlingame, California, February 2017.

3. E. Wes Bethel. In Situ Processing Overview and Relevance to the HPC Community. In *SIAM Conference on Computational Science and Engineering, MS74: In Situ Methods and Infrastructures: Faster Insight Through Smarter Computing*, Atlanta, GA, USA, February 2017.

4. E. Wes Bethel. Towards Exascale: High Performance Visualization and Analytics. In *2017 ASCR Computer Science Principal Investigators Meeting (CS PI Meeting)*, Bethesda, MD, USA, March 2017.

5. E. Wes Bethel. Scalable Analysis Methods and In Situ Infrastructure for Extreme Scale Knowledge Discovery. In *2017 ASCR Computer Science Principal Investigators Meeting (CS PI Meeting)*, Bethesda, MD, USA, March 2017.

6. E. Wes Bethel. Management, Analysis, and Visualization of Experimental and Observational Data: The Convergence of Data and Computing. In *2017 ASCR Computer Science Principal Investigators Meeting (CS PI Meeting)*, Bethesda, MD, USA, March 2017.

7. E. Wes Bethel. Navigating the Landscape of Scientific Data Challenges and Opportunities. In *Office of Science Data Working Group and Facilities Working Group Seminar*, Germantown, MD, USA, January 2017.

8. E. Wes Bethel. Convergence of Computing and Data: An Opportunity for Long-term Impact on SC Science Programs, and Beyond. In *Office of Advanced Scientific Computing Research Seminar*, Germantown, MD, USA, January 2017.

9. E. Wes Bethel. Performance Analysis, Design Considerations, and Applications of Extreme-scale *In Situ* Infrastructures. In *ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC16)*, Salt Lake City, UT, USA, November 2016.

10. E. Wes Bethel. Towards Adoption and Deployment of In Situ Methods and Infrastructure. In *ISAV 2016: In Situ Infrastructures for Enabling Extreme-scale Analysis and Visualization, a SC16 workshop*, Salt Lake City, UT, USA, November 2016. Invited panel presentation.

11. E. Wes Bethel. The SENSEI Generic In Situ Interface. In *IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, Baltimore, MD, USA, October 2016. Invited panel presentation.

12. E. Wes Bethel. Management, Analysis, and Visualization of Experimental and Observational Data – The Convergence of Data and Computing. In *National Institute of Standards and Technology, Applied and Computational Mathematics Division Seminar Series*, Gaithersburg, MD, USA, October 2016.

13. Gunther H. Weber. Time-varying merge trees. In *Division for Media and Information Technology (MIT), Campus Norrköping of Linköping University*, Norrköping, Sweden, September 2016.

14. Gunther H. Weber. *In Situ* Methods and Infrastructure: Answers Without All the I/O. In *ASTRONUM 2016—the 11th International Conference on Numerical Modeling of Space Plasma Flows*, Monterey, CA, USA,, June 2016.

15. E. Wes Bethel. *In Situ* Methods, Infrastructures, and Applications on High Performance Computing Platforms, a State-of-the-art (STAR) Report. In *Eurovis 2016*, Gröningen, Netherlands, June 2016.

16. E. Wes Bethel. Management, Analysis, and Visualization of Experimental and Observational Data – The Convergence of Data and Computing. In *U.S. Department of Energy Advanced Scientific Computing Advisory Committee (ASCAC) Meeting*, Washington, DC, USA, September 2016.

17. E. Wes Bethel. Management, Analysis, and Visualization of Experimental and Observational Data – The Convergence of Data and Computing. In *Thomas Jefferson National Laboratory Colloquium*, Newport News, VA, USA, August 2016.

18. Dmitriy Morozov. Practical Guide to Persistent Homology using Dionysus. In *Bootcamp at the Institute for Computational and Experimental Research in Mathematics, Brown University*, Providence, RI, USA, September 2016.

19. Andrew Bauer. In Situ Analysis and Visualization: Background, State of the Art and Examples Using ParaView Catalyst. In *Air Force Research Laboratory, U.S. Air Force*, Daython, OH, USA, July 2016.

20. Andrew Bauer. ParaView Catalyst: Effective In Situ Analysis and Visualization. In *Army HPC User Group Review, Engineering Research and Development Center, U.S. Army*, Vicksburg, MS, USA, August 2016.

21. Brad J. Whitlock and Atsushi Toyoda. VisIt Prime training at Tokyo University of Science. In *Tokyo University of Science*, Tokyo, Japan, August 2016.

22. Brad J. Whitlock. Scalable In Situ Processing using VisIt and Libsim. In *JAXA In Situ Workshop*, Chofu, Japan, August 2016.

23. Gunther H. Weber. Scientific Visualization of Big Data. In *Learning from other domains – Big Data and Visualization, 4th Workshop on Next-Generation Analytics for the Future Power Grid*, Richland, WA, USA, September 2015.

24. Gunther H. Weber. IsoFind: Halo Finding Using Merge Trees. In *Department of Energy Computer Graphics Forum (DOE CGF)*, Monterey, CA, USA, May 2016.

25. E. Wes Bethel. *In Situ* Methods and Infrastructure: Answers Without All the I/O. In *SIAM Parallel Processing 2016 (PP16)*, Paris, France, April 2016.

26. E. Wes Bethel. In Situ Methods: Hype or Necessity? In *IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC15)*, Austin, TX, USA, November 2015.

27. E. Wes Bethel. Visualization, Analysis, and Exascale: Trouble or Triumph? In *The Ultrascale Visualization Workshop, IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC15)*, Austin, TX, USA, November 2015.

28. E. Wes Bethel. HPC Visualization and Analysis at the Exascale: Big Headaches, Big Opportunities. In *Siggraph Asia Symposium on Visualization in High Performance Computing*, Kobe, Japan, November 2015.

29. E. Wes Bethel. In Situ 2020: Back to the Future, Again. In *IEEE Symposium on Large Data Analysis and Visualization Symposium, co-located with IEEE Visualization 2015*, Chicago, IL, USA, October 2015.

30. E. Wes Bethel. Exascale Computing Challenges and Volume Rendering Optimizations for Advanced Architectures. In *Computer Science Department Colloqium Seris, Old Dominion University*, Norfolk, VA, USA, April 2016.

31. Nicola J. Ferrier. Multi-modal 3D Characterization of Materials. Invited talk at the APS Horizons Workshop: Challenges in Integrating Data Science, Computational Modelling and Advanced Characterization, May 2016.

32. Nicola J. Ferrier. Multi-scale, multi-modal Dynamic 3D Imaging. Keynote presentation at Multimodal Data Analysis Hackathon, April 2016.

33. P. O'Leary. Responsive large data analysis and visualization with the paraview ecosystem. In *NVIDIA GPU Technology Theater SC'15 the International Conference for High Performance Computing Networking, Storage and Analysis*, Austin, TX, November 2015.

34. P. O'Leary and S. Jourdain. Is the web ready for visualization? In *Workshop on Visualization Technologies at SC'15 the International Conference for High Performance Computing Networking, Storage and Analysis*, Austin, TX, November 2015.

35. Brad J. Whitlock. Libsim Overview and Recent Developments, November 2015.

36. Brad J. Whitlock. VisIt 2.10 Improvements and Harnessing In Situ, November 2015.

37. Brad J. Whitlock. In Situ Infrastructure Enhancements for Data Extract Generation. In *54th AIAA Aerospace Sciences Meeting, SciTech 2016*, January 2016.

38. Brad J. Whitlock. In Situ: How to Instrument Your Solver with VisIt and Libsim, January 2016.

39. Brad J. Whitlock. Libsim Improvements to Enable Better In Situ Workflows, April 2016.

40. Brad J. Whitlock. In Situ Production of Extract Databases for Visualization. In *ISC Workshop on In Situ Visualization*, June 2016.

41. Andrew Bauer. ParaView Catalyst: Scalable In Situ Processing. In *2015 Air Force HPC User Forum: Visualization Session*, Dayton, OH, July 2015.

42. Andrew Bauer. In Situ Analysis and Visualization for Rotor Aeromechanics Simulations. In *ISAV 2015: Lightning Talk*, Austin, TX, November 2015.

43. Dmitriy Morozov. Parallel computation of persistent homology using the blowup complex. In *ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, Portland, OR, June 2015.

44. E. P.N. Duque. Facing the Knowledge Extraction and Visualization Challenges of the NASA CFD 2030 Vision. In *AIAA Aviation 2015*, Dallas, TX, USA, June 2015.

45. E. P.N. Duque. Accelerating the Post-Processing of Large Scale Unsteady CFD Applications via In Situ Data Reduction and Extracts. In *Department of Aeronautical Engineering Graduate Seminar*, Embry-Riddle University, Daytona Beach, FL, USA, April 2015.

46. Drew A. Boyuka, Xiaocheng Zou, Nagiza Samatova, Junmin Gu, Kesheng Wu, Norbert Podhorszki, and Scott Klasky. Adios query interface design. Invited presentation at Supercomputing Frontiers 2015, Singapore, 2015.

47. Gunther H. Weber. Computing and visualizing time-varying merge trees for high-dimensional data. In *TopoInVis 2015*, Annweiler, Germany, May 2015. Best paper award.

48. E. Wes Bethel. Brook No Delay: In Situ Visualization and Analytics on HPC Platforms. In *BIS 2015, Inria@SiliconValley, Panel on Big Data Science: Data Analytics Meets High Performance Computing*, Berkeley, CA, USA, May 2015.

49. E. P.N. Duque. Accelerating the Post-Processing of Large Scale Unsteady CFD Applications via In Situ Data Reduction and Extracts. In *Virginia Polytechnic University Department of Aerospace and Ocean Engineering Graduate Seminar*, Virginia Polytechnic University, Blacksburg, VA, USA, September 2015.

50. J. Forsythe and B. Whitlock. Big Data Challenge: Landing in a Ship Airwake. In *Intelligent Light Webinar*, Rutherford, NJ, May 2015.

51. B. Whitlock. Promoting In Situ with VisIt, Libsim, and FieldView. In *DOE Computer Graphics Forum*, Newport, RI, 2015.

52. E. Wes Bethel. Scientific Visualization. In *Invited lecture, MSIM 742, Synthetic Environments and Advanced Visualization*, Old Dominion University, Norfolk, VA, USA, February 2015.

53. D. Morozov. Computing topology in parallel. In *Topological and Geometric Data Analysis seminar, Ohio State University*, Columbus, OH, April 2015.

54. D. Morozov. Wrinkles on everest: Persistence and stability in an omniscalar world. In *Department of Computer Science colloquium, Tulane University*, New Orleans, LA, November 2014.

55. D. Morozov. Distributed computation of persistent homology using the blowup complex. In *Computational Topology and Geometry workshop at the Foundations of Computational Mathematics (FOCM) conference*, Montevideo, Uruguay, December 2014.

8 Collaborations

Science Area	*Name(s) of Collaborator*	*Name(s) of SENSEI Personnel*
Astrophysics	Ann Almgren, Brian Friesen, Zarija Lukić, Peter Nugent (LBNL)	Gunther Weber, Dmitriy Morozov
Aircraft Design	Ken Jansen (Professor at University of Colorado at Boulder) and Michel Rasquin (Post-Doctorate at Cenaero in Belgium and University of Colorado at Boulder)	Andrew Bauer, Patrick O'Leary, Venkat Vishwanath, Utkarsh Ayachit
Climate Science	Mark Taylor (SNL), Robert Jacob (ANL)	Patrick O'Leary, Venkat Vishwanath, Utkarsh Ayachit
Combustion	Suresh Menon (Professor and Director Georgia Tech Computational Combustion Laboratory), Reetesh Ranjan (Post-Doctorate Georgia Tech Computational Combustion Laboratory), Steve M Legensky (General Manager and Founder, Intelligent Light), Christopher P. Stone (President, Computer Science and Engineering LLC)	Earl Duque, Matthew Wolf, Brad Whitlock
Accelerator Modeling	Ji Qiang	John Wu, Alex Sim
Materials Science	Juan DePablo, Paul Nealey (IME, UChicago)	Nicola Ferrier, Venkat Vishwanath
Accelerator Modeling Warp	Jean-Luc Vay (LBNL)	Burlen Loring