

SANDIA NATIONAL LABORATORIES

SNL ADTM

FY18Q3 report for ATDM ST projects to ECP

June 30, 2018

Prepared by: Aaron Pennington

Prepared for:

ECP Confluence updates

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology and Engineering Solutions of Sandia, LLC.

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



2.3.1.04 ATDM PMR

Accomplishments and Progress:

- **Q4: Kokkos back-end R&D and external collaborations**
 - A collaboration with AMD was continued to develop the ROCm backend for AMD GPU support. During a focused multi-day session with AMD personal on-site at Sandia the backend was improved to now pass the vast majority of Kokkos unit-tests.
 - The Kokkos team helped prepare proposals for and attended the ISO C++ committee meeting in Rapperswill, Switzerland. Kokkos's "atomic_ref" proposal was voted into the C++20 standard draft.
 - The team gave the keynote presentation at the HIPS workshop at IPDPS in Vancouver.
 - The Kokkos team attended an Intel ECP meeting, where development of an Intel ECP architecture backend for Kokkos was started.
 - Status: On Schedule

- **Q4: Kokkos R&D for SIMD types and remote memory spaces**
 - A Remote Memory Spaces prototype on the basis of SHMEM was implemented, enabling distributed data structures with integration into Kokkos View based data structures.
 - Status: On Schedule

- **Q4: Kokkos support for ASC applications and libraries, and project management requirements**
 - The team made progress towards a Kokkos 3.0 release slated for summer 2018.
 - ECP planning for FY19 was performed, including the definition of milestones and KPP goals.
 - Status: On Schedule

- **Q4: DARMA interoperability with Kokkos, OpenMP, and MPI**
 - Completed DARMA-MPI interoperability mini-apps with PIC move kernel in DARMA and solver in MPI adapted from HPCCG.
 - Status: On schedule

- **Q4: Development of agile components for backend development**
 - Updated all DARMA components to modern CMake system. DARMA backend now configurable with serialization, preprocessor, and metaprogramming libraries and provides universal CMake package interface for all backend implementations and configurations.
 - Status: On schedule

- Q4: *(Planned)* **Perform development work with performance analysis team on tools. Perform ATS-1 and ATS-2 tuning.**
 - DAT requests to be submitted for Mutrino and then Trinity in July for PIC and Contact apps

Activities:

- Continuing to sustain rigorous, high-quality multi-architecture and multi-compiler
- Continuing participation in ISO/C++ Standard
- Participated in ATDM deep dive at Sandia/NM

2.3.2.04 ATDM Tools

Accomplishments and Progress:

- **[DEV] Coordination and General DevOps support for ATDM**
 - Running Coordinated DevOps for ATDM process, management of ATDM DevOps backlog and Kanban board, running meetings with team representatives, following up on issues, etc.
 - Continued support for the ATDM JIRA site (addressing move to CEE LAN management, addressing permissions issue between ATDM JIRA projects and git repository browsing, notification emails, etc.)
 - Status: On schedule
- **[DEV] Deploy scalable build, integration and test framework**
 - SPARC is no longer interested in completing the TriBITS prototype system.
 - Increased effort needed to complete Trilinos testing and integration work will not allow funding and time to complete this.
 - Status: At Risk
- **[DEV] Trilinos integration and multi-platform CI testing for ATDM apps**
 - Set of Trilinos builds being done by EMPIRE now complete on all platforms but testing many more Trilinos packages and submitting to Trilinos CDash site so that Trilinos developers can help to maintain.
 - Doing continued evaluation and bug fix efforts with Kitware for new versions of CMake/CTest/CDash critical for reporting Trilinos testing results for ATDM and set of CDash site for ATDM APPs.
 - Status: On schedule
- **[PERF] Performance analysis and assessment of SPARC application on ATS platforms**

- Assessment of vectorization levels achieved in SPARC (with Trilinos solver stack) is continuing. Results for improved Kokkos Kernels (with manual SIMD-ization) show improvement in some key routines.
- Status: On schedule
- **[PERF] Performance analysis and assessment of EMPIRE application on ATS platforms**
 - Benchmarking of MiniEM on Mutrino (ATS-1) and Serrano (CTS-1) is underway with feedback provided to the Trilinos development teams.
 - Optimized TPL builds provided on Mutrino (ATS-1) for Haswell and are in development for Knights Landing. This will enable binaries for the full EMPIRE code to be built on Mutrino in preparation for future RAFT-copying to the SCN.
 - Status: On schedule
- **[PERF] Assess ATS-1 and ATS-2 development toolchains for tri-lab co-design L2 milestone**
 - Developing parameterized builds for CTH, Xyce and SPARC on Mutrino (ATS-1), IBM POWER9/Volta (ATS-2) and Cavium/ARM (future Vanguard) is completed. Work continues on parameterization of SIERRA/IC builds which may require changes to the bake system. CTH evaluation using compilers from GCC, Intel and IBM underway.
 - Status: On schedule

Activities:

- **[PERF] Deployment of LDMS-based continuous monitoring on ATS-1 and testbeds**
 - Contract placed for additions to LDMS monitoring daemons. Work is underway with LDMS team (and contractor support) to deploy on ASC test bed systems during Spring of 2018.
- **[PERF] Performance analysis and assessment of ASC/IC applications on ATS platforms**
 - CTH benchmarking assessment underway on ARM, POWER8, POWER9 and KNL systems
 - Early benchmark results using Xyce (UUR) on Intel Sky Lake and Cavium ARM systems have been produced.

2.3.3.04 ATDM Math Libraries

Accomplishments and Progress

- **Baseline performance data of existing linear solvers at scale and improve overall simulation time-to-solution by decreasing linear solver setup time by investigating multigrid methods, improving load balancing, and conducting cost-benefit analysis of effectiveness vs setup for a variety of solver capabilities (Technical Lead: Hu)**
 - James Elliott has improved executable size for EMPIRE on KNL mutrinos. This has led to improved simulation run times (up to 30%). He's obtained more modest improvements on SPARC on GPUs.
 - Chris Siefert has been profiling MueLu's usage of threaded KokkosKernels smoothers on Nvidia P100s. This has led to optimizations such that Kokkos utilization is above

- 80% for standard Jacobi, multithreaded Gauss-Seidel, and iterative ILU. Profiling and optimization of multigrid setup is ongoing.
 - Status: On schedule
- **Integrate sensitivity analysis and ROL optimization tools with Tempus in support of transient full-space optimization (Technical Lead: Phipps)**
 - Q3 portion for investigating and integrating pseudo-transient adjoint sensitivity methods is completed.
 - Progress on Q4 portion for connecting to ROL: A new transient interface for ROL has been developed and tested on the Burger's equation. The next step is to connect connect this with Tempus.
 - Status: On schedule
- **Create interoperability for Kokkoskernels with ASC, ATDM, and ASCR developed simulation codes via vector friendly data structures (Technical Lead: Rajamanickam)**
 - Delivered GPU based tridiagonal solver to SPARC. This improved SPARC GPU performance more than 8x. The GPU performance could be improved even further. This will be pursued as a continuation.
 - Started preliminary work on GPU based triangular solves. This got delayed due to the complications that arose due to compiler issues for SPARC (bullet 2).
 - Preliminary work started to support a deterministic coloring for MueLU setup on GPUs.
 - Status: On schedule
- **Strategic integration of agile components in SPARC and EMPIRE including finite element basis tools and discretization support, a LOCA continuation driver (for SPARC), time integration methods, and deployment of Kokkos (Technical Lead: Pawlowski)**
 - The blocked versions of the Panzer assembly tools have been converted to use Kokkos. This involved significant refactors of the volume and boundary condition gather/scatter operators and dof manager utilities. This resulted in a speedup of ~30x on HSW/KNL systems for RHS Assembly phase in EMPIRE. This work was required for the EMPIRE L2 and brings the RHS assembly times down to ~10% of overall runtime.
 - Development of Panzer projection operators for EMPIRE completed. Will be integrated into EMPIRE in early Q4 in support of EMPIRE L2.
 - Tempus is integrated into an EMPIRE-PIC branch and demonstrates first through fifth order convergence.
 - New time steppers added to Tempus: Trapezoidal method (a.k.a. Crank-Nicolson), DIRK version Backward Euler method, 1-Stage IRK Theta method, and 2-Stage EDIRK Theta Method.
 - Tempus Newmark steppers are being used in Albany/LCM to demonstrate new domain decomposition-based Schwartz coupling method.
 - Status: On schedule

Activities:

- **Demonstrate forward UQ and/or calibration for hypersonic fluid flows with no coupling to ablation, in support of L2 milestone.**
 - Developed a new method for estimating uncertain boundary conditions in double-cone hypersonic flow experiments being used to validate SPARC. It uses a self-similar formulation of the flow solution to ease the construction of surrogates and identify experimental measurement errors.
- **Dakota support for EMPIRE/PIC L2 milestone**
 - Interfaced Dakota to EMPIRE to evaluate the sensitivity of plasma frequency with respect to particle properties using a Langmuir oscillation problem.
- **Support creation of a combined SIMD/Ensemble scalar type**
 - Co-advising with KokkosKernels a student who is working on this

2.3.4.04 ATDM Data and Viz

Accomplishments and Progress:

- **Develop and deliver input/output support for SNL ATDM Applications. FY18 focus is on checkpoint/restart capability and development of IOSS to support SPARC and EMPIRE.**
 - Results:
 - **SPARC Checkpoint/Restart:** Code merged into SPARC and managed by SPARC team. Investigating burst buffer performance problems observed with CGNS.
 - **Empire Checkpoint/Restart:** Particle and mesh interfaces merged into EMPIRE. Currently developing restart support for L2 milestone.
 - **IOSS for SPARC:** Continued support for SPARC requirements
 - Status: On Schedule
- **Develop I/O capabilities for current and future ATS platform and ATDM application needs. FY18 focus is on burst-buffer support and I/O support for hybrid meshes**
 - Result:
 - **HDF burst buffer:** Consolidated functionality into IOSS. New extensions provide simple means for IOSS users to take advantage of burst buffer. Currently testing.
 - **IOSS hybrid mesh:** Merging structured/unstructured implementations into the hybrid mesh API and implementing parallel capability.
 - Status: On Schedule
- **Develop and deliver scalable tools and algorithms for analysis and visualization (both *in situ* and post-processing) for next-generation, many-core architectures.**

- Result: Enhancements on Q2's radix-k sorting in VTK-m postponed until Q4 due to staffing issues (see details).
- Status: On Schedule
- **Integrate, demonstrate, and deploy analysis and visualization libraries and tools for ATDM applications on the ASC Advanced Technology Systems (ATS).**
 - Result:
 - **Catalyst for SPARC:** Catalyst now integrated into SPARC I/O. Demonstrated generating in situ images on mutrino. Selecting additional SPARC examples.
 - **TuckerMPI in situ compression or SPARC:** Developed Paraview plugin to interface with parallel version of TuckerMPI.
 - Status: On Schedule

Additional ECP Milestones: *(If you have ECP milestones that are not aligned with PLAR deliverables)*

- **STDV04-21 [Q3] Prototype Catalyst/ParaView in-situ viz for unsteady RV flow on ATS-1**
 - Result: Working with the SPARC team, Catalyst is now integrated with the I/O layer in SPARC. Using a test problem, demonstrated performance generating in situ images on Mutrino. Ongoing work is scaling up a sample problem by the SPARC apps team.
 - Status: On Schedule/Complete – I feel comfortable checking this off once the Catalyst code is checked into the SPARC main repository. I believe we are waiting on a code review for this to happen. We hope to have this done by the end of Q3.
- **STDV04-20 [Q4] Document/Tune performance of VTK-m algorithms on ATS-1**
 - Result: Due to changes in staffing, we have not made much progress on this milestone. In the next month, we will be running some updated performance metrics on the external faces algorithm using updated radix-k sorting. We have also brought in Todd Kordenbrock to replace Ben Newton, who has moved to other projects, to resume work on improving vectorization.
 - Status: On Schedule
- **STDV04-25 [Q4] Kokkosize TuckerMPI**
 - Result: Deferred to FY19 based on funding availability
 - Status: Delayed
- **Undocumented-TODO [Q3] TuckerMPI compression of non-rectilinear meshes [Kolla]**

- Result: In the context of functional tensor approximation of non-rectilinear mesh data, a key first step is choosing the right basis along each mode/dimension that comprises the best functional form. Working on a method where the basis selection is informed by the data itself, whereby the data is first interpolated onto an underlying structured mesh, and the singular vectors along each mode, resulting from a higher-order SVD of the interpolated data, are used to guide the choice of the basis along that mode.
 - Status: On Schedule
- **STDV04-18 [Q4] Catalyst/TuckerMPI in situ compression for SPARC**
 - Result: Extended the Paraview plugin for compression, that interfaces with TuckerMPI library, to work in a parallel setting (was working for serial data as of Q2). Made modifications both in the TuckerMPI library and the plugin code to re-align the MPI domain decomposition to a Cartesian topology that is expected by TuckerMPI. Currently working on handling StructuredGrid and MultiBlock type datasets that are likely to be the case for SPARC. In good shape to demo in-situ compression for a SPARC data set through ParaView/Catalyst.
 - Status: On Schedule
- **STDV04-17 [Q4] Demonstrate HDF/DataElevator Burst Buffer capability**
 - Result: Initial Q1 work discovered an incompatibility between HDF/DataElevator and the way IOSS users write data. The work led to the insight that the capability could instead be implemented more efficiently inside of IOSS. The team has updated IOSS with mechanisms that enable users to take advantage of the burst buffer during I/O operations. The mechanisms provide users with a simple interface that can be extended to run on other platforms. The team is conducting experiments with different codes to ensure correctness and quantify performance.
 - Status: On Schedule
- **STDV04-16 [Q4] EMPIRE: Initial checkpoint/restart of PIC structure on ATS-1**
 - Result: FAODEL I/O handlers for Particle data have been constructed for EMPIRE and merged. Additional work will construct proper checkpoint/restart functionality and measure performance for the L2 milestone.
 - Status: On Schedule
- **STDV04-10 [Q1] SPARC: Demonstrate burst-buffer-based checkpoint/restart on ATS-1**
 - Result: An initial set of I/O writers/readers was presented to SPARC shortly after Q1. Following some refactoring, the SPARC team built a new checkpoint/restart interface and merged our HIO interface into the code. I/O team continues to investigate I/O problems, including a performance problem observed when using CGNS with the ATS1 burst buffer.

- Status: DELAYED – This work required a number of iterations with the application team. In hindsight, a fixed deliverable deadline for anything that is really “I/O support” is probably a bad idea.
- **STDV04-13 [Q2] SPARC: Production IOSS-Based partitioning for structured/unstructured grids at scale on ATS-1**
 - Result: Good progress on the parallel file-per-processor input and output options to complement the single-file output previously implemented. The parallel decomposition algorithm has been rewritten to provide more optimal decompositions and to be more robust at large processor counts. Unit testing has been improved to increase robustness.
 - Status: DELAYED – Clearly this is a level-of-effort task that does not have a deliverable.
- **STDV04-3 [Q4] SPARC: Demonstrate IOSS-based hybrid mesh**
 - Result: Merging the structured and unstructured implementations into the hybrid mesh API. Implementing the parallel capability.
 - Status: On Schedule
- **STDV04-6 [Q3] Empire: Initial checkpoint/restart of fluid code on ATS-1**
 - Result: Initial FAODEL I/O functions were constructed to write/read mesh data in EMPIRE/Fluid. While code to insert the I/O operations into the application’s graph of operations has been written and merged into EMPIRE/Fluid, additional work is required to implement the full checkpoint/restart capability. This work depends on the PIC/Fluid integration that is currently taking place.
 - Status: DELAYED into Q4 Progress has been made, but underlying codebase evolving
- **STDV04-7 [Q4] Transition I/O components to ATS-2**
 - Result: This work will take place later in Q4 as platforms become available. Small experiments have been conducted on Ride to help prepare for this work.
 - Status: On Schedule
- **Undocumented-TODO [Q4]** Explore/develop machine-learning methods to support mesh and geometry design, generation, and pre-processing.
 - Result: A new ML-based approach has been developed for preparing CAD models for analysis based upon a proposed set of targeted operations that are applied using the Cubit Geometry and Meshing Toolkit. We have demonstrated that by performing a localized geometric operation on a CAD model, the improvement in mesh quality can be measured and used to train machine learning models. Extraction of features that accurately characterize local topology and expected mesh quality have been the primary focus of this research during this quarter. As a result, several ML models have been developed based upon

training data extracted from real-world components. Prediction of mesh quality based upon geometric operations and localized features have shown promising results.

- Status: On Schedule
- **Undocumented-TODO [Q4]** Explore/develop machine-learning methods for adaptive solvers for large-scale PDE codes.
 - Results: Generated an expanded set of ~24,000 MueLu runs ("Mark 3b") for use with machine learning. Developing library-mode interface for Avatar machine learning software.
 - Status: On Schedule
- **Undocumented-TODO [Q4]** Explore/develop machine-learning methods to support HPC system software.
 - Result: The team has identified three research directions in which machine learning techniques may be able to impact HPC system software - adaptive P-State control for HPC workloads, job allocation for dragonfly networks, and memory management. They are now working to establish the infrastructure (both simulation and physical) to explore the impact reinforcement learning (RL) based ML approaches may provide. In developing a RL software base to apply to HPC system software, the team was able to publish a manuscript quantizing RL control costs at IJCNN. Additionally, they have submitted a chapter providing an overview of mathematical optimizations for deep learning to Cyber-Physical Systems and Security.
 - Status: On Schedule

Activities: *(these are additional activities that you are managing too but are either not captured in the key deliverables/PLATR and/or are too detailed for that level of reporting)*

- Active participation in the DOE Computer Graphics Forum
- Participation at the ASC PI Meeting
- Oldfield and Templeton working on a briefing on Machine Learning opportunities for ASC

2.3.5.04 ATDM SW ecosystem

Accomplishments and Progress:

- **(AOSR, FY18 Q4) Prototype usage of containers and related technologies to support ATDM developer workflows**

- On track. NDA now in place with Sylabs, the Singularity container technology company, which has allowed regular discussions with them about needs and capabilities using their container deployment solutions.
- **(AOSR, FY18 Q4) Resource manager applied to Darma+Kokkos use case scenarios**
 - On track. Note that the OpenMP affinity layer is being used to manage coordination of thread placement and usage, rather than a custom resource manager. This arrangement is advantageous in that it is standards-based for sustainability.
- **(AOSR, FY18 Q4) Prototype of message-based, open-source simulation framework capable of quantifying MPI resource usage for MPI-based ATDM workloads**
 - On track for at least internal release of the tool (and external release contingent on SNL software copyright process).

Additional ECP Milestones: *(If you have ECP milestones that are not aligned with PLAR deliverables)*

- **(AOSR, FY18 Q1) Coordinate with ATDM DevOps to make plan for utilizing containers for automated build and testing of Trilinos**
 - Completed in Q2.
- **AOSR, FY18 Q2) Characterization of MPI resource usage for ATDM workloads and its impacts on performance**
 - Completed in Q2.
- **(AOSR, FY18 Q4) Development and evaluation of system software implementations and interfaces for operating systems, multithreading, and communication technologies**
 - On track. Note that this milestone combined separate milestones for OS, runtimes, and standards body work. Runtime work this quarter included implementing a Qthreads version of NimbleSM (formerly miniMultiscale). Stephen Olivier and Ryan Grant continue their work in the OpenMP language committee and MPI forum, respectively.

Activities:

- **Task parallel over-decomposition (TPOD)**
 - We continue to collaborate with Richard Barrett (5800) on the use of task-parallel decomposition within MPI processes.
- **Q2: Support stk::simd use in ATDM products. An open source copyright will be obtained for the stk::simd library, and the library will be made available through an independent, externally hosted repository. The Sierra/SM team will serve as a point of contact for questions about the library.**

- ATDM is now planning to use SIMD through a different SIMD library implementation. The role of STK::SIMD is still unclear. The Sierra/SM team will continue to support SIMD in ATDM and will schedule work as necessary.
- **Q2: Integrate a light weight version of the Sierra/SM explicit contact algorithm into the existing technology demonstrator code. The technology demonstrator will allow experimentation on NGP technologies that will be beneficial to Sierra, and in particular Sierra/SM contact.**
 - Sierra/SM teamed with Technology Demonstrator (TD) developers to integrate a GPU enabled bounding box search and a GPU enabled closest point projection into the TD application. These are the core components needed for a GPU contact algorithm and are available thanks to collaboration in previous years between ATDM and Sierra/SM. The TD application is linking these in, exercising both CPU and GPU (through a unit test) execution, and combining them with an enforcement algorithm to model contact.
- **Q4: Using the technology demonstrator mini app, design a pattern for NGP contact using algorithms that are notionally like those in Sierra/SM. This includes consideration of Kokkos-GPU, Kokkos-Threading, SIMD, and asynchronous multitask concepts. Follow on work in FY19 and FY20 will integrate the successful patterns and concepts found in the mini-app back into Sierra.**
 - A primary goal of this activity is correct and performant execution of TD contact on GPUs. In conjunction with TD developers, the Sierra/SM team made progress on an issue that blocked this goal, namely successful compilation of the TD application (including contact) with Relocatable Device Code (RDC) turned on. RDC allows for modern code architecture in algorithms targeted at GPUs and is needed to integrate the desired Sierra products (e.g. Lame, Math Toolkit, Geometry Toolkit) into the TD application. Continuing work will finish this compilation and evaluate GPU execution performance.