LSSw Meeting 10

3 – 4:30 pm ET, July 28, 2022

Last LSSw meeting of this series

- This is the final LSSw meeting of this series
- Planning to start a new series again in Fall 2022
- Virtual via Zoom registration
- Announcement of dates and times will be via
  - announce@lssw.io – email list
  - LSSw Slack workspace
  - https://lssw.io – Website – Sign up for email list and Slack on this site

LSSw Meeting 10: July 21, 2022

Expanding Laboratory, University, and Industry Collaborations: An Industry Panel Discussion

This month our panelists are:

- John Cary, Tech-X Corp
- Sarah Knepper, Intel Corp
- Pete Mendygral, HPE, Inc
- Jeff Larkin, NVIDIA Corp
- Bob Lucas, ANSYS, Inc

Opening remarks prompts:

- What are some existing examples of scientific software collaboration between federal agency-sponsored programs (at labs and universities) and software vendor product development?
- What has worked and not worked well with past leverage and complementarity efforts?
- What are some near-term opportunities to improve leverage and complementarity?
- What are some long-term opportunities and constraints on leverage and complementarity?

Expanding Laboratory, University, and Industry Collaborations

(July 28, 2022)

John R. Cary, CTO, Tech-X Corporation, www.txcorp.com

CAE for plasma, electromagnetics

Specializing in HPC (MPI, GPU) and cross platform

Unique in combining commercial and non-commercial efforts

Mesozoic:

- 🐀 John Cary, Tech-X Corp
- 🦖 Sarah Knepper, Intel Corp
- 🦖 Pete Mendrygal, HPE, Inc
- 🦖 Jeff Larkin, NVIDIA Corp
- 🦖 Bob Lucas, ANSYS, Inc

Tech-X: working with DoE for 28 years

- Develop commercial codes
  - HPC Photonics (SBIR)
  - Low temperature plasma (SBIR)
  - Development of code for RF propagation in plasma (FES SciDAC), unique in algorithm, implementation, using expertise in conformal boundaries and CAD with finite difference.
  - Cloud available HPC

Making use of DOE software

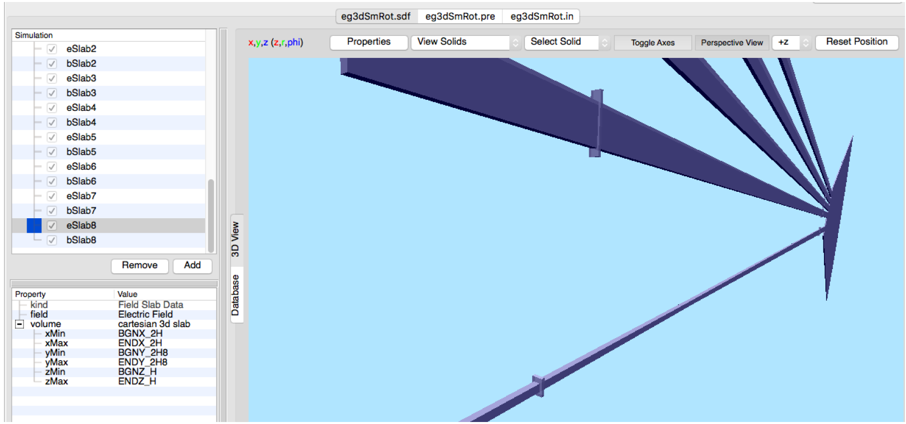
- Develop open-source codes and tools
  - Trilinos (unfunded) for, e.g., porting to Windows
  - VisIt (unfunded)
  - PETSc (funded)
  - NIMROD (funded) consulting - commercial fusion
  - SPACK (funded)

DOE software **in** commercial products

DOE software **to create** commercial products

Successful projects are characterized by

- Crossing the academic/industrial cultural barrier
  - CAD
  - Windows
  - GUI
  - Installers
  - Fat binaries (esp CUDA)
- Tasks needing Tech-X's unique skills
  - Linear algebra
  - Conformal FDTD
  - Particle in cell
  - User interfaces for scientific workflows

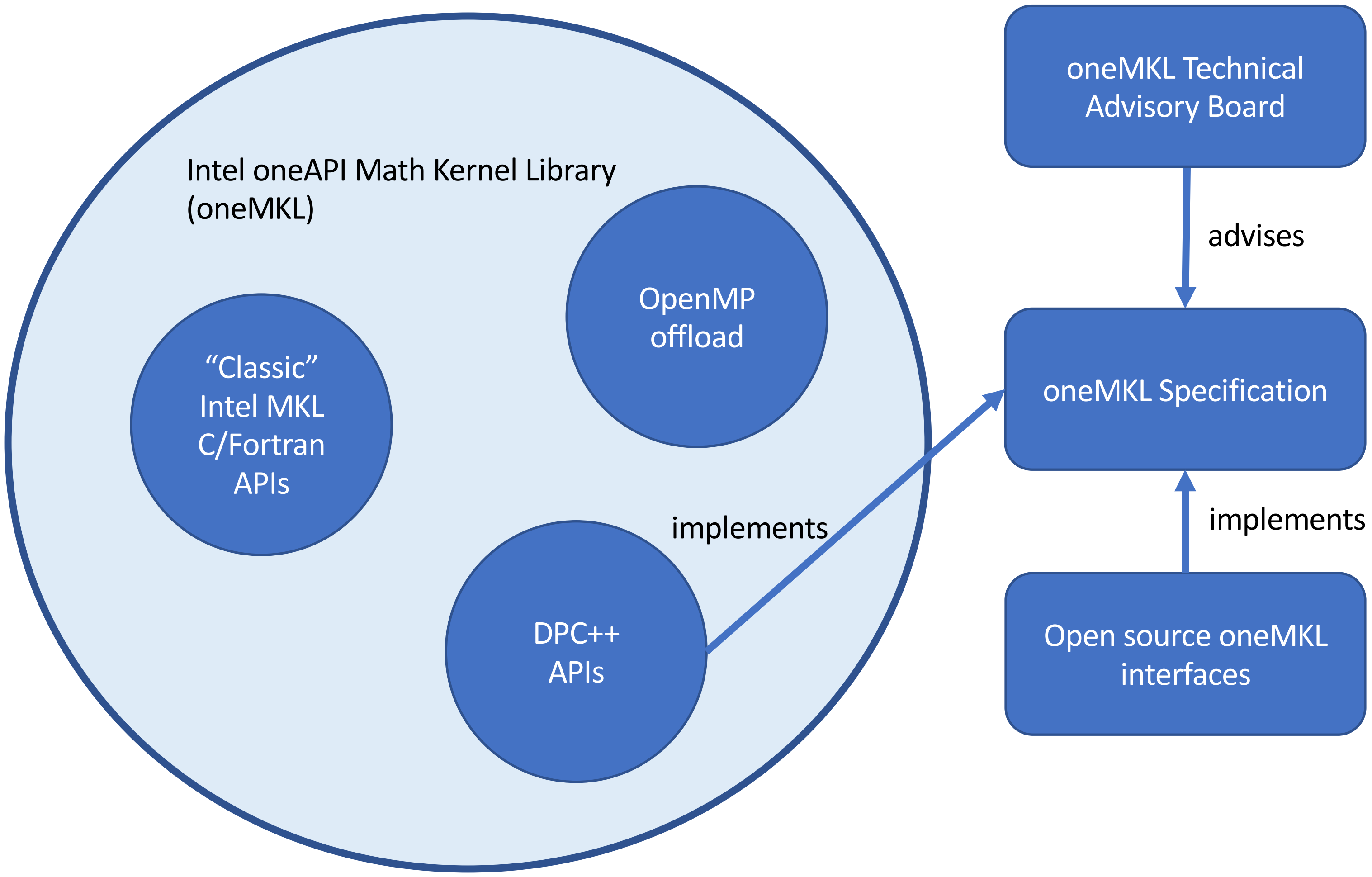


Antipatterns and opportunities

- More work is needed when the DOE workflow does not match up to industrial workflow
  - Anything with CAD
  - Windows
  - DOE computing centers versus commercial clouds
- Opportunities arise in standards and to align DOE with industrial directions
  - Democratizing HPC usage
  - Demonstrate commercial GPU success on other than CUDA
  - Single instance requisitioning
  - The front end/back end problem

Sarah Knepper

- Software Engineering Manager, Intel oneAPI Math Kernel Library (oneMKL)
- Over a decade with the Intel oneMKL team
- Leads oneMKL Technical Advisory Board
- Leads internal sync meetings between application and oneMKL teams



Intel oneAPI Math Kernel Library
(oneMKL)
"Classic"
Intel MKL
C/Fortran
APIs
OpenMP
offload
DPC++
APIs
implements
oneMKL Technical
Advisory Board
advises
oneMKL Specification
implements
Open source oneMKL
interfaces

Prompts

- What are some existing examples of scientific software collaboration between federal agency sponsored programs (at labs and universities) and software vendor product development?
  - oneAPI Centers of Excellence
  - oneAPI Technical Advisory Boards
  - Contributions to open source oneAPI libraries and oneAPI specification
- What has worked and not worked well with past leverage and complementarity efforts?
  - Good: Analyzing key apps for opportunities for functionality and performance; active and timely feedback
  - Challenges: Coordination among a changing software stack; prioritization among multiple activities
- What are some near-term opportunities to improve leverage and complementarity?
  - Continued engagement on oneAPI initiatives, trainings/hackathons, input on spec
- What are some long-term opportunities and constraints on leverage and complementarity?
  - Broad adoption of oneAPI

Jeff Larkin

- Principal HPC Application Architect
- 7+ Years Cray Supercomputing Center of Excellence, Profiling, Optimizing, & Porting applications
- 9+ Years NVIDIA, including Center of Excellence for Summit at ORNL

Bob Lucas

- Distinguished Engineer, Livermore Software Technology, Ansys
- I’ve been developing parallel linear and eigenvalue solvers for over three decades. My interests extend to accelerators, programming models, and most other aspects of HPC.

What are some existing examples of scientific software collaboration between federal agency sponsored programs (at labs and universities) and software vendor product development?

Often the government, or its contractors, pioneer new technology, which then transitions to first commercial use, then maintenance and further development. NASTRAN and LS-DYNA are important examples in Mechanical Computer Aided Engineering.

Open source software components can also enable rapid development of new commercial codes. I recently saw a briefing by a small company that extensively leveraged PETSc.

Conversely, the government and its contractors often use commercial software. Fortran and C are great examples.

What has worked and not worked well with past leverage and complementarity efforts?

One thing that has been very helpful is access to new technology, at scales beyond that available commercially. I was twice able to run LS-DYNA on all of Titan's production nodes.

From the perspective of somebody working on codes that are half a Century old, its not always easy to adopt new research components. I'm a big fan of PGAS languages, but I can't use them in practice.

Also, cutting edge technology doesn't always survive in the marketplace. We invested a lot of time into using Intel MIC chips as linear algebra accelerators.

What are some near-term opportunities to improve leverage and complementarity?

With the end of Dennard scaling, and soon Moore’s Law, specialization is becoming increasingly important for maximizing performance. Access to novel systems and the tools developed to use them can help us navigate this space.

Joint projects to develop new tools and other software components to solve common problems. The MPI forum is a good example.

What are some long-term opportunities and constraints on leverage and complementarity?

Increasingly heterogeneous systems will require tools like Kokkos to enable code portability. The government and its contractors have the same issues with evolving large bodies of code that we have.

A constraint is access to critical source code. It might be proprietary or classified.

Another constraint is our reluctance to adopt new technology before we're confident it will survive. It's easier to explore that space in academe or Labs (govt. or corporate).

Pete Mendygral

- Distinguished Technologist in HPE Cray Programming Environment
  - Technical lead for a new development project
  - Past work in the areas of application optimization, communication libraries, compilers, distributed DL, networks
- Trained computational astrophysicist
  - Developed large applications for studying astrophysical jets from supermassive blackholes (including processes for thermal and non-thermal emission)
  - Published techniques for efficient MPI one-sided + OpenMP
- Highlighted publications/SW development with DOE labs
  - Scalable DL training (CosmoFlow)
  - Tools for studying network performance under load (GPCNeT)
- Representing an industry perspective

Hewlett Packard Enterprise

Prompts

- What are some existing examples of scientific software collaboration between federal agency sponsored programs (at labs and universities) and software vendor product development?

- In CPE, we use (variety of collaboration depths)
    - MPICH
    - LLVM
    - PETSc
    - Many other projects less core to our products

Prompts

- What has worked and not worked well with past leverage and complementarity efforts?

- Worked
  - Efforts building features and APIs portable across systems and time
  - Projects with ~modular architectures that vendors can specialize from
  - Projects that feed into community building

- Not worked
  - Efforts targeting specific HW features not carried forward
  - Large monolithic projects where vendors spend most of the time building/packaging (ie little value to add)
  - Projects with a very narrow user base

Prompts

- What are some near-term opportunities to improve leverage and complementarity?
- Building the right solutions for labs/users so CPE can be used in CICD pipelines and deliver SW built with CPE
- Stronger partnerships centered on community building and engineering practices

Prompts

- What are some long-term opportunities and constraints on leverage and complementarity?

- Opportunities for better overall community architecture
  - Community-driven building-block abstractions that are interoperable
  - Composable to enable vendor-specific implementations

- Supporting use-cases that stretch beyond core HPC systems should force us to envision an ecosystem that is easier to use and broadly adopted
  - Vendors are constrained by addressable market