Sandia National Laboratories

Exceptional service in the national interest

HPSF — HIGH PERFORMANCE SOFTWARE FOUNDATION

THE LINUX FOUNDATION

AN UPDATE ON TRILINOS DISCRETIZATION AND ANALYSIS CAPABILITIES

Mauro Perego
Sandia National Laboratories

Contributors:
P. Kuberry, D. Noble, R. Pawlowski, E. Phipps,
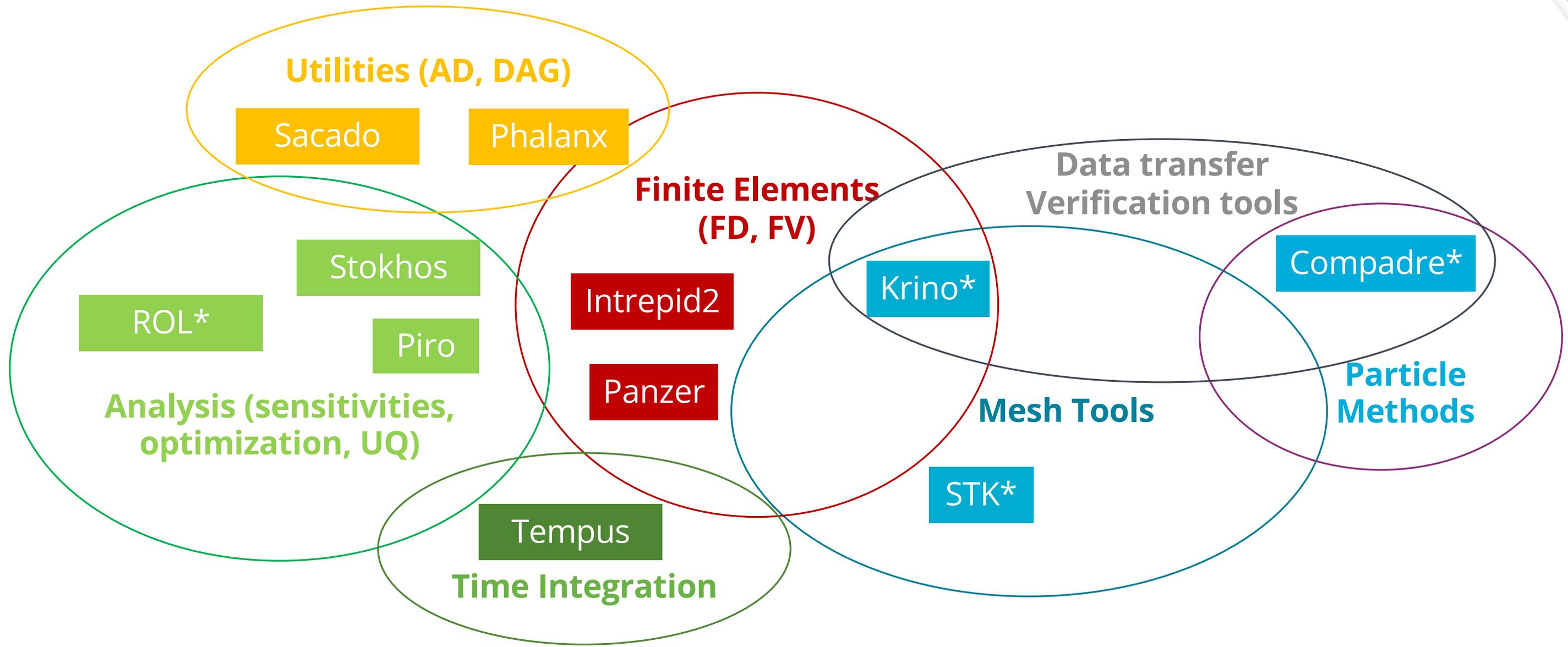C. Ober, D. Ridzal, N. Roberts, A. Williams

HPSF conference 2026,
March 16-20, Chicago, IL

SANDXXX

DISCRETIZATION AND ANALYSIS AREA: CAPABILITIES/PACKAGES

Utilities (AD, DAG)

Sacado

Phalanx

Finite Elements (FD, FV)

Data transfer Verification tools

Stokhos

ROL*

Piro

Intrepid2

Krino*

Compadre*

Panzer

Analysis (sensitivities, optimization, UQ)

Mesh Tools

Particle Methods

STK*

Tempus

Time Integration

*Packages snapshotted into Trilinos

TRILINOS

DISCRETIZATION AND ANALYSIS AREA: UPDATES & PLANS

Epetra/Intrepid archival
- Finalized work on test coverage without Epetra (Panzer, Piro, ROL)
- ROL tests transitioning to use Intrepid2

Other broad efforts
- Refactor of views of AD types to support *mdspan*
- Porting to SYCL - ongoing
- Trilinos update paper:
  Mayr, Heinlein, Glusa et al: *"Trilinos: enabling scientific computing across diverse hardware architectures at scale",* to appear in ACM Transactions on Mathematical Software

DISCRETIZATION AND ANALYSIS AREA: UPDATES & PLANS

Intrepid2 (POCs: M. Perego and N. Roberts)

Completed:
- Performance improvements/code refactoring to address changes in Kokkos 5.0
- Optimizations leveraging structure in FE discretizations:
  - Roberts, Perego, Pawlowski, "Intrepid2: Structure preserving optimization for high-order discretizations", to appear in ICIAM proceedings book by Springer.

Ongoing:
- Work for fast high-order assembly to enable matrix-free operations
- Support of shell elements

INTREPID2 DEEP DIVE: STRUCTURE-AWARE OPTIMIZATIONS
(WORK BY N. ROBERTS)

Typical high-level FEM codes ignore or discard structure in order to maintain generality.
Example: Jacobians on affine meshes are constant on each element, however standard classes compute and store them on each quadrature point!

New structures and algorithms allow for structure-aware optimizations, while maintaining same mathematical workflow.

Example: Quadrature rule definition.

Classic implementation

```
using DRV = Kokkos::DynRankView<double,DeviceType>;
DRV points(P,D);
DRV weights(P);
cubature->getCubature(points, weights);
```

New structure-aware implementation

```
auto points  = cubature->allocateCubaturePoints();  // TensorPoints<double,DeviceType>
auto weights = cubature->allocateCubatureWeights(); // TensorData<double,DeviceType>
cubature->getCubature(points, weights);
```

INTREPID2 DEEP DIVE: STRUCTURE-AWARE OPTIMIZATIONS
(WORK BY N. ROBERTS)

Theoretical advantage of a structure-aware implementation for FE of order p and dimension d.

	Storage	Assembly	Evaluation
Full Assembly + matvec	$O(p^{2d})$	$O(p^{3d})$	$O(p^{2d})$
Sum-Factorized Full Assemby + matvec	$O(p^{2d})$	$O(p^{2d+1})$	$O(p^{2d})$
Partial Assembly + matrix-free action	$O(p^{d})$	$O(p^{d})$	$O(p^{d+1})$

Anderson et al, MFEM: A modular finite element methods library. doi: 10.1016/j.camwa.2020.06.009.

New implementations in Intrepid2 preserve the structure and perform sum-factorization.
As an example, for hexa or wedge elements
- Basis functions are tensor-product lower-dimensional basis functions
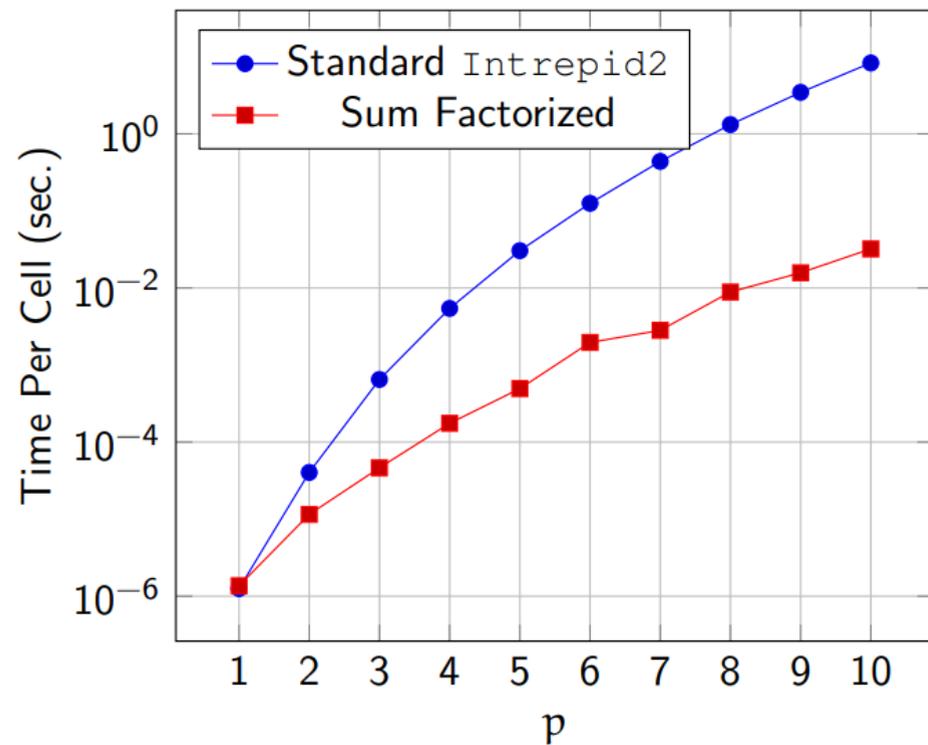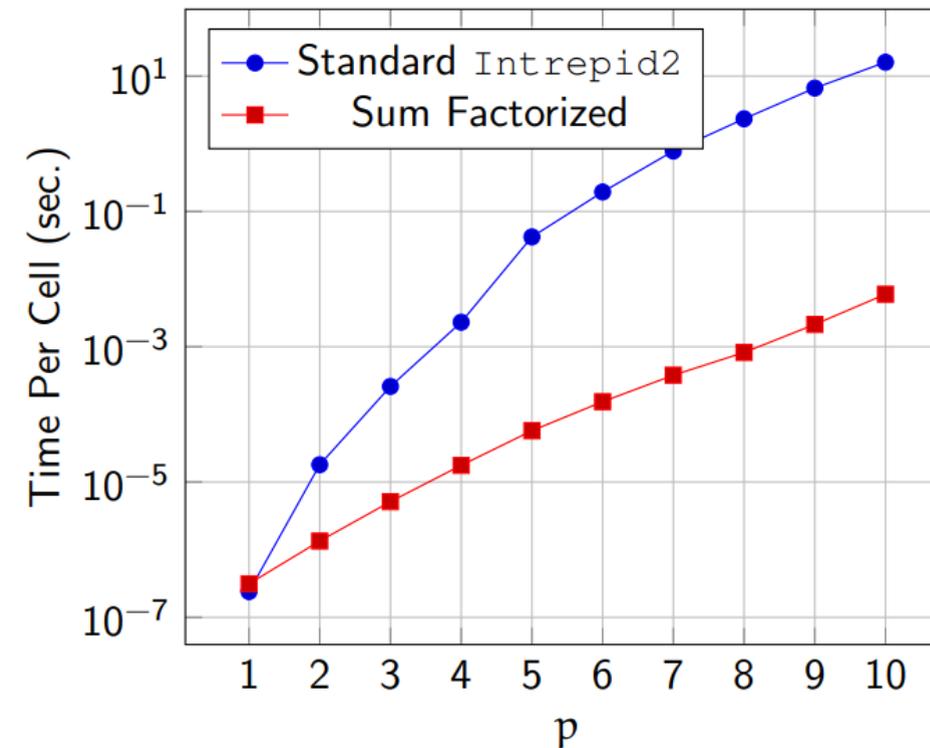- Quadrature rules are tensor products of lower-dimensional rules

Local Assembly, integration of *(v,v) + (curl v, curl v)* term on hexahedra

OpenMP (16 threads, Apple M2 Ultra)

CUDA (H100)

Ongoing work on partial assembly with matrix-free mat-vec operators, also exploring use of matrix-free, p-multigrid MueLu preconditioner (w/ N. Roberts, C. Glusa)

DISCRETIZATION AND ANALYSIS AREA: UPDATES & PLANS

Sacado (POC: E. Phipps)

- **Completed:**
- Kokkos integration **refactored to support Kokkos mdspan** transition (thanks to Christian Trott)
- **Ongoing:**
- **SYCL support** (thanks to Roger Pawlowski)

Phalanx (POC: R. Pawlowski)

- **Completed:**
- Cleanup for new kokkos releases and supporting hip compiler
- **Planned/Ongoing:**
- **Support for Shared Space**, replacing deprecated UVM default in ExecSpace
  - Short term: potential hack to MDField/interfaces, depending on kokkos removal of UVM
  - Long term: prefer a backwards incompatible change to PHX::Device definition

DISCRETIZATION AND ANALYSIS AREA: UPDATES & PLANS

Panzer (POC: R. Pawlowski)

Completed:
- Caching option in connection manager
- Improved initialization performance for mesh objects
- Recover ROL transient optimization support (ongoing performance optimizations)
- Evaluator for loading cell data from Exodus

Planned/ongoing:
- Continue adding capability for ROL transient optimization
- Performance improvements for ATS-4

Piro (POC: M. Perego)

Completed:
- Added Tpetra tests to replace deprecated Epetra-based capability tests
- Pruned some capabilities that were not supported

Planned:
- Improve support for transient simulation-constrained optimization using Tempus and ROL

DISCRETIZATION AND ANALYSIS AREA: UPDATES & PLANS

Stokhos (POC: E. Phipps)

Completed:
- Refactoring Kokkos integration of ensemble scalar type for mdspan (thanks to Christian Trott)
- Removed Kokkos-based PCE scalar type in Trilinos 17
  - Including all Tpetra-stack instantiations/interfaces
  - PCE scalar type already disabled by default and no longer tested
  - Non-kokkos-based PCE scalar type will NOT be removed
- Removed Epetra-based stochastic Galerkin support in accordance - No current plans to port to Tpetra

Compadre (POC: P. Kuberry)

Planned/Ongoing:
- Implementation of **RKPM**.
- Add weight term for each source site that passes into the weighting function for multi-material applications
- Performance improvements on GPUs (w/ Jake Koester)

DISCRETIZATION AND ANALYSIS AREA: UPDATES & PLANS

ROL (POC: D. Ridzal)

Completed:
- ROL is now a **standalone project**: github.com/sandialabs/rol
  - Following the Kokkos example, with snapshotting.
- **Epetra removal:**
  - All Epetra functionality is now covered with Tpetra.
  - ROL/Epetra adapters will be removed in 2027.
- **Intrepid removal:**
  - Intrepid - Intrepid2 conversion for PDE-OPT tests (50k+ lines).
  - No conversion of **Intrelab**, the Matlab interface for Intrepid, is planned.
  - Maintain Intrepid examples in FY26, then remove the code.

Ongoing:
- First comprehensive **ROL paper**;
  to be submitted to ACM TOMS,
  draft soon on arXiv.

DISCRETIZATION AND ANALYSIS AREA: UPDATES & PLANS

STK (POC: A. Williams)

    Completed:
- Improved AMD GPU support
- Cuda-aware communication support for parallel-assembly operations for mesh Fields
- New Field-data access API improves GPU workflow and enables unified-memory usage on ATS-4

    Ongoing:
- User facing stk_transfer executable for file-based exodus handoffs
    - Translating element fields to nodal fields, gauss-point fields, etc.
    - Translating fields from M-proc mesh decompositions to N-proc, etc.

Kryno (POCs: D. Noble)

    Completed:
- Adaptive refinement for Tet10 meshes
- Uniform refinement for Quad and Bar elements
- A 2d, adaptive, semi-Lagrangian capability for interface advection
- Support for level set conforming interfaces with full sensitivities for level set topology optimization

    Planned/Ongoing:
- An ability to snap to geometry during adaptive refinement
- A 3d, adaptive, semi-Lagrangian capability for interface advection
- Support for initializing implicit neural representations from STL geometry
- Support for creating interface conforming meshes from implicit neural representations

INCOMPLETE LIST OF CONTRIBUTORS AND SPONSOR

Contributors:

Ross Bartlett,
Pavel, Bochev,
Brian Carnes,
Eric Cyr,
Carter Edward,
Nathan Ellingwood,
Christian Glusa,
Aurya Javeed,
Kyungjoo Kim,
Drew Kouri,
Paul Kuberry
Kim Liegeois,
David Noble,
Roger Pawlowski,
Mauro Perego,
Kara Peterson,
Eric Phipps,
Curt Ober,
Bryan Reuter,
Denis Ridzal,

Nathan Roberts,
Andy Salinger,
Chris Siefert,
Irina Tezaur,
Greg Von Winckel,
Alan Williams

Sponsors: