Sandia National Laboratories

Exceptional service in the national interest

TRILINOS INTRODUCTION AND OVERVIEW

Curtis Ober

Trilinos Product Owner

May 7th, 2025

HPSF Conference, Chicago, IL

U.S. DEPARTMENT of ENERGY | National Nuclear Security Administration

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND2025-05380C

Welcome to the TUG Meeting at the HPSF Conference!

- Shifting to meet regularly at the HPSF Conference
  - Participant in HPSF opportunities/goals
    - *"High-Performance Software Foundation (**HPSF**) aims to build, promote, and advance a portable core software stack for HPC by increasing adoption, lowering barriers to contribution, and supporting development efforts."*
  - Meet with other HPSF projects to broaden collaboration
  - Introduce Trilinos to new developers and users
- Coordinating with EuroTUG
  - EuroTUG is moving to a bi-annual schedule
  - Next meeting is planned for Spring/Summer of 2026
- Focused "Sandia Trilinos" meeting will occur in October

TRILINOS BACKGROUND

"The Trilinos Project is an effort to *facilitate the design, development*, integration and ongoing support of *mathematical software libraries* within an *object-oriented framework* for the solution of *large-scale, complex multi-physics* engineering and scientific problems. Trilinos addresses two fundamental issues of developing software for these problems: (i) Providing a *streamlined process and set of tools* for development of new algorithmic implementations and (ii) promoting *interoperability of independently developed software*."

Heroux, et al, "An Overview of the Trilinos Project", ACM Transactions on Mathematical Software, Vol. V, No. N, December 2004, Pages 1–27.

"The Trilinos project was established to address two important needs: (1) bringing teams of library developers together in order to *leverage commonalities and produce compatible software components*, formally called packages and (2) to amortize the cost and efforts associated with more formal *software engineering requirements*. With a modest level of coordination and without unduly compromising package team autonomy, Trilinos project members could leverage each other's efforts, consolidate commonly needed tools, make packages compatible, and define a *common set of software engineering tools and processes*."

Heroux and Willenbring, "A new overview of the Trilinos project", Scientific Programming 20 (2012) 83–88

TRILINOS

- … is a **community** of developers, users and user-developers
- … is open-source software with a modular design
- … promotes **collaborative** algorithm and technology development
- … provides a comprehensive suite of mathematical libraries
- … provides **interoperable**, scalable, and high-performance software packages
- … enables rapid development of robust, efficient algorithms
- … targets new and emerging **HPC** architectures
- … facilitates development of large-scale, **complex multi-physics** problems
- … is used in computational fluid dynamics, structural mechanics, electromagnetics, and many more

TRILINOS LEADERSHIP

- **Technical Steering Committee**
  - Responsible for Trilinos-wide decisions
  - Determine project wide rules
  - Decide on package inclusion
  - Form committees to address tasks
- **Previous Members**
  - Thank you for serving!


Sam Browne


Christian Glusa


Curtis Ober


Roger Pawlowsk


Mauro Perego


Eric Phipps


Siva Rajamanickam


Heidi Thornquist


Jim Willenbring


Michael Wolf


Mike Heroux

- **New Members**
  - Curtis Ober
    - Trilinos Representative to TAC
    - Trilinos Product Owner
    - Trilinos Developer (Tempus)
  - Christian Glusa
    - Sandia Trilinos Strategic Lead
    - Trilinos Developer (Teko, MueLu, …)
  - Jim Willenbring
    - Sandia Representative to TAC
    - Trilinos Developer
    - Former Trilinos Framework Lead

TRILINOS AREAS

DevSecOps:


Sam Browne


Mauro Perego

Discretization and Analysis Area:

- Finite Element Assembly (Intrepid2, Panzer, Shards)
- DAG-based Field Eval. (Phalanx)
- Mesh Methods & Structure (Krino, Percept, STK)
- Automatic Differentiation (Sacado)
- Optimization Methods (ROL)
- Stochastic Galerkin UQ (Stokhos)
- Time Integration (Tempus)
- Nonlinear Anal. Drivers (Piro)
- Small Tensor Alegra (MiniTensor)
- Meshless Discretization (Compadre)
- Coupled Physics Examples (TrilinosCouplings)


Jonathan Hu

Solvers Area:

- Solver Interface (Stratimikos)
- Nonlinear Solvers (NOX)
- Krylov Solvers (Belos)
- Eigen Solvers (Anasazi)
- Direct Sparse & Dense Solvers (Amesos2, Adelus)
- Block Solvers (Teko)
- Multigrid Methods (MueLu)
- Multilevel Domain Decomposition Methods (ShyLU)
  - Iterative Substructuring (ShyLU)
  - Overlapping Schwarz (ShyLU/FROSch)
- Polynomial Smoothers, SOR, ILU (Ifpack2)
- Linear Factorization (KLU, ShyLU/Basker)
- Cholesky and LDL Factorizations (ShyLU/Tacho)


Roger Pawlowski

Core Area:

- Python Interface (PyTrilinos2)
- Abstract Numerical Algorithms & Interfaces (Thyra)
- Matrix Generation Tools (Galeri)
- Vector Reduction/Transformation Operators (RTOp)
- Distributed Linear Algebra (Tpetra)
- Common SE Utilities (Teuchos)
- Optimized Math Kernels (Kokkos Kernels)
- Performance Portability (Kokkos)
- Finite Element Utilities (SEACAS)
- Load balancing and partitioning (Zoltan/Zoltan2)
- Mesh Generation (Simple Shapes) (Pamgen)

CORE AREA - PROVIDES BASIC CAPABILITIES

Roger Pawlowski

Algebra/Numerics	Description	Package Lead
Kokkos (@kokkos)	Performance portability library	Christian Trott (@crtrott)
Kokkos Kernels (@kokkos-kernels)	Optimized math kernels for Kokkos library	Luc Berger-Vergiat (@lucbv)
RTOp (@rtop)	Vector Reduction/Transformation Operators	Ross Bartlett (@bartlettroscoe)
Thyra (@thyra)	Abstract numerical algorithms and interfaces	Ross Bartlett (@bartlettroscoe)
Tpetra (@tpetra)	Distributed linear algebra objects	Chris Siefert (@csiefert2)
Utilities		
Galeri (@galeri)	Matrix generation tools	Jonathan Hu (@jhux2)
PyTrilinos2 (@pytrilinos2)	Python bindings for Trilinos packages	Christian Glusa (@cgcgcg)
SEACAS (@SEACAS)	Utilities supporting finite element analysis	Greg Sjaardema (@gsjaardema)
Teuchos (@teuchos)	Common tools *(smart pointers, parameter lists, ...)*	Roger Pawlowski (@rppawlo)
Meshing / Domain Decomposition		
Pamgen (@pamgen)	Mesh generation tools for simple shapes	Roger Pawlowski (@rppawlo)
Zoltan/Zoltan2 (@zoltan, @zoltan2)	Load balancing and partitioning	Erik Boman (@egboman)

SOLVERS AREA - PROVIDES LINEAR AND NONLINEAR SOLVERS

Jonathan Hu

Linear / Nonlinear Solvers	Description	Package Lead
Adelus (@adelus)	Scalable dense linear solvers	Vinh Dang (@vqd8a)
Amesos2 (@Amesos2)	Direct sparse solvers	Siva Rajamanickam (@srajama1)
Belos (@belos)	Iterative linear solvers	Heidi Thornquist (@hkthorn)
NOX (@nox)	Nonlinear solvers	Roger Pawlowski (@rppawlo)
ShyLU (@shylu)	Hybrid linear solvers	Siva Rajamanickam (@srajama1)
Preconditioners		
Ifpack2 (@ifpack2)	Preconditioners	Jonathan Hu (@jhux2)
MueLu (@muelu)	Multigrid solvers	Jonathan Hu (@jhux2)
Teko (@teko)	Block preconditioners	Malachi Phillips (@MalachiTimothyPhillips)
Others		
Anasazi (@anasazi)	Eigenvalue solvers	Heidi Thornquist (@hkthorn)
Stratimikos (@stratimikos)	Solver and preconditioner wrappers	Ross Bartlett (@bartlettroscoe)
Xpetra (@xpetra)	Linear algebra interfaces	Jonathan Hu (@jhux2)

DISCRETIZATION AND ANALYSIS - MODULAR, INTEROPERABLE AND EXTENSIBLE TOOLS


Mauro Perego

Compadre (@Compadre)	Toolkit for meshless discretizations	Paul Kuberry (@Paul Kuberry)
Intrepid2 (@Intrepid2)	High-order compatible finite elements	Mauro Perego (@mperego)
Krino (@krino)	Meshing tool with level-set methods	David Noble (@drnobleabq)
MiniTensor	Algebra for small vectors/tensors	Alejandro Mota (@lxmota)
Panzer (@panzer)	Finite element analysis and assembly	Roger Pawlowski (@rppawlo)
Percept (@percept)	Mesh adaptation and data transfer	Brian Carnes (@bricar)
Phalanx (@phalanx)	DAG-based local field evaluation kernel	Roger Pawlowski (@rppawlo)
Piro (@piro)	Driver classes for common nonlinear analysis	Mauro Perego (@mperego)
ROL (@rol)	Large-scale optimization algorithms	Denis Ridzal (@Denis Ridzal)
Sacado (@Sacado)	Automatic differentiation	Eric Phipps (@Eric Phipps)
Shards (@shards)	Finite element topologies	Mauro Perego (@mperego)
STK (@STK)	Scalable mesh data structures	Alan Williams (@alanw0)
Stokhos (@Stokhos)	Stochastic Galerkin UQ methods	Eric Phipps (@Eric Phipps)
Tempus (@tempus)	Time Integration	Curtis Ober (@ccober6)
TrilinosCouplings (@trilinoscouplings)	Coupled physics examples	Mauro Perego (@mperego)

DEV SEC OPS – SUPPORTS DAILY OPERATIONS AND CI/CD

Sam Browne

- Provides streamlined processes and set of tools for development of Trilinos packages
  - Implements various development, testing, automation tools, and IT infrastructure
  - Defines and sets development, test, release, update, and support processes
- Organizes and maintains Trilinos release process
- Leads DevSecOps team in performing associated tasks

Anderson Chauphan

Justin LaPre

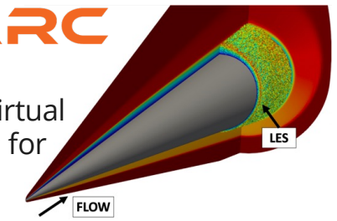
Joe Frye

COLLABORATIONS WITH USERS & STAKEHOLDERS

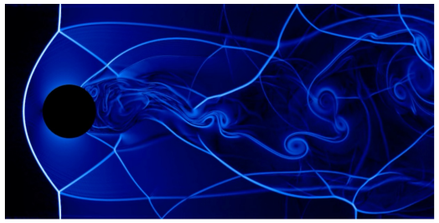
Trilinos currently has collaborations with ~25 applications …

SPARC

Credible full-system virtual flight-testing platform for hypersonic vehicles

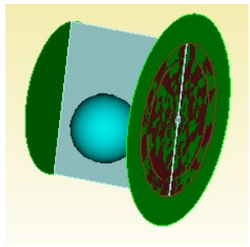


2D-3D Compressible CFD



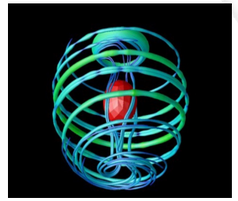
Gemma

High-fidelity, robust, solutions for Maxwell's Equations

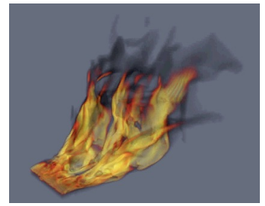


deal.II

Rapid development of modern finite element codes



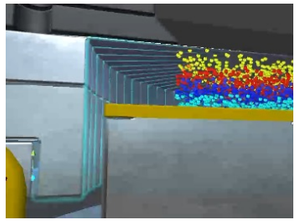
COMPSIM
THERMAL FLUIDS



Heat Transfer, Thermal Radiation, Chemistry, Multiphase flow, Fire/Combustion

EMPIRE

Plasma simulations over a broad density ranges

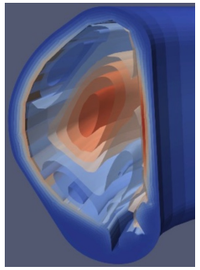


Xyce™ PARALLEL ELECTRONIC SIMULATOR

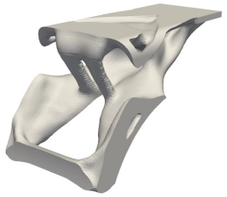
High-performance analog circuit simulator

Drekar

Plasma Confinement, Tokamak Instabilities / Disruptions



plato
OPTIMIZATION-BASED DESIGN



CHARON

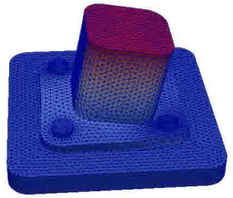
Semiconductor device modeling

COMPSIM
STRUCTURAL DYNAMICS

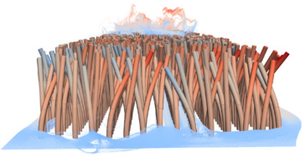
Vibration Calculations for Structural Dynamics

Albany

Finite element code for the solution and analysis of multi-physics problems



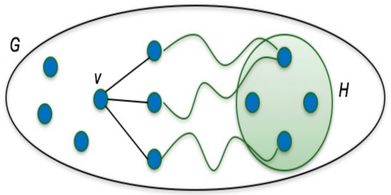
4C
Comprehensive Computational Community Code



Multi-physics research code for a plethora of computational mechanics

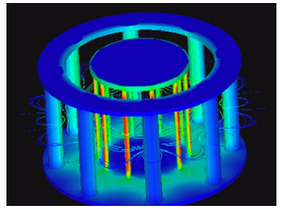
Grafiki

High-Performance Distributed Graph-based Algorithms

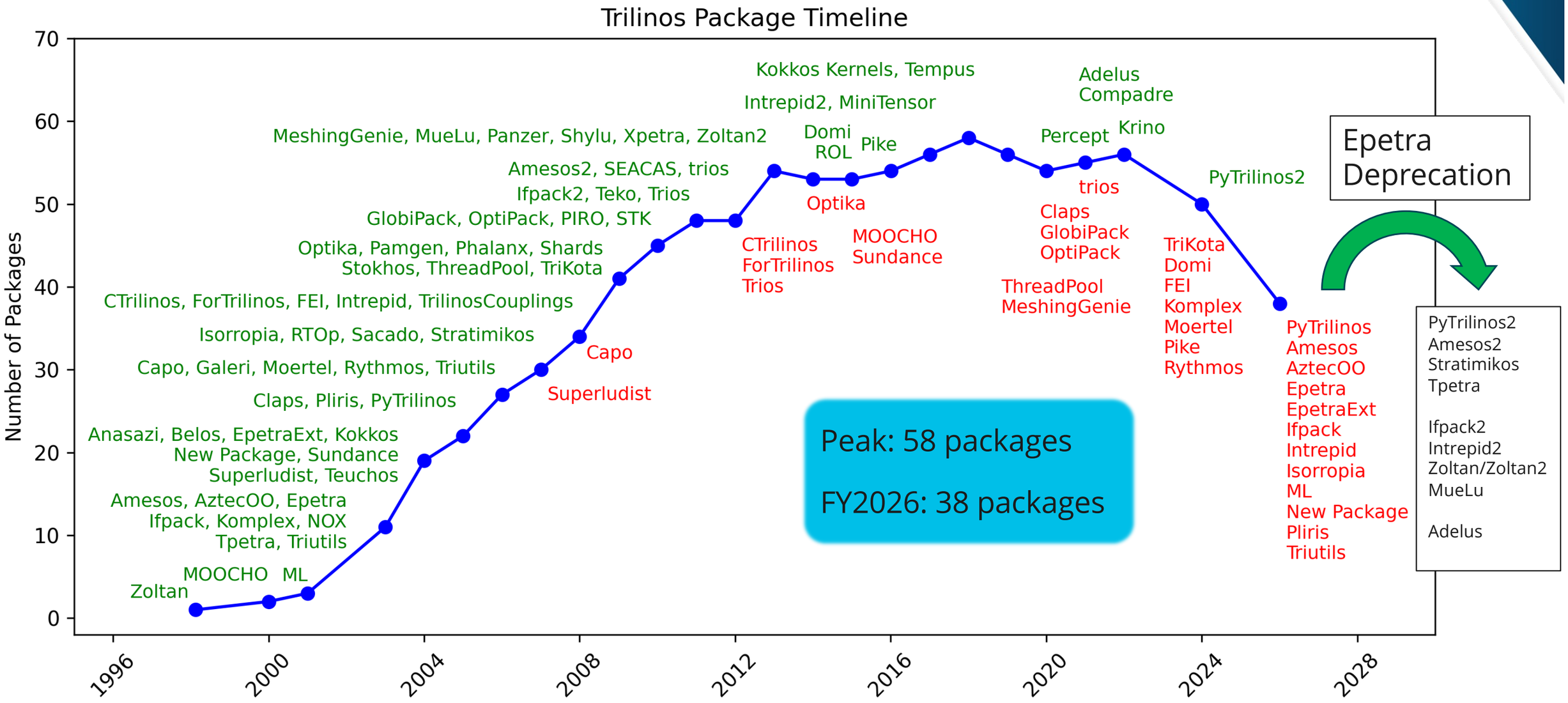


Alegra

Shock hydrodynamics and multi-physics



TRILINOS PACKAGE HISTORY



EPETRA REMOVAL – SEPT. 2025!

- Epetra/Tpetra has been around for 20+ years
  - Many applications have been dependent on Epetra
- Driver for deprecation
  - Reduce costs of duplicative code
  - Reduce complexity of Trilinos DAG
- **Deprecated packages** will be moved from the Trilinos repo to individual package repos
- If you are still using
  - **Deprecated packages**, transition to Tpetra packages **ASAP**.
  - **Dependent packages**,
    - Build and test without Epetra
    - Test capabilities for functionality/performance

Deprecated Epetra Stack

- Amesos
- AztecOO
- Epetra
- EpetraExt
- Ifpack
- Intrepid
- Isorropia
- ML
- New Package
- Pliris
- PyTrilinos
- ThyraEpetraAdapters
- ThyraEpetraExtAdapters
- Triutils
- ShyLU_DDCore

Dependent on Epetra

- Amesos2
- Anasazi
- Belos
- Galeri
- MueLu
- NOX
- PanzerDiscFE
- PanzerDofMgr
- Piro
- ROL
- ShyLU_DDFROSch,
- Teko
- TpetraCore
- TrilinosCouplings
- Stokhos
- Stratimikos
- Xpetra
- Zoltan2Core

TRILINOS USERS-DEVELOPERS GROUP (TUG) MEETING

Wednesday, May 7

Introduction and Area Leads	1:35pm CDT	**Trilinos Introduction and Overview - Curtis Ober, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2
	1:55pm CDT	**An Overview of the Trilinos Core Products - Roger Pawlowski, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2
	2:15pm CDT	**Why Your Science Application Should Be Using Trilinos Linear Solvers - Jonathan Hu, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2
	2:35pm CDT	**Introduction to Trilinos Discretization and Analysis Capabilities - Mauro Perego, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2
	2:55pm CDT	**Trilinos CI Testing/Contribution Overview - Samuel E. Browne, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2
User Talks	3:40pm CDT	**The Role of Trilinos in 4C: Advancing Coupled Multiphysics Simulations - Matthias Mayr, University of the Bundeswehr Munich** MISSISSIPPI RIVER 1 & 2
	4:00pm CDT	**Automated Preconditioner Design in the Trilinos/Teko Package - Malachi Phillips, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2
Performance	4:20pm CDT	**Keeping Trilinos Running Performantly Everywhere Every Night - Chris Siefert, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2
Interface Demo	4:40pm CDT	**PyTrilinos2: Using Trilinos from Python - Christian Glusa, Sandia National Laboratories** MISSISSIPPI RIVER 1 & 2

THANKS!

ANY QUESTIONS?