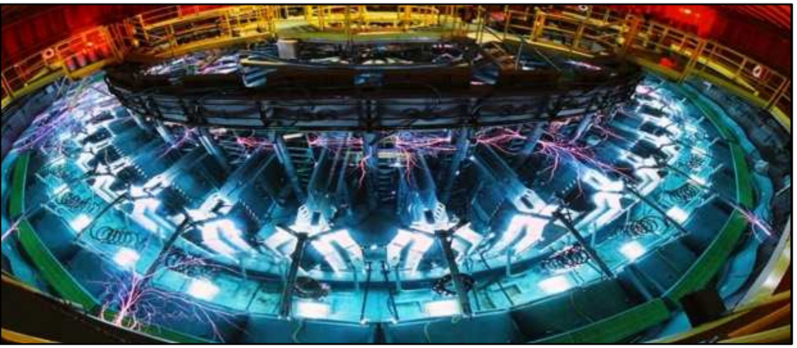
SAND2015-10210C

Exceptional service in the national interest

Sandia National Laboratories



Programming Models for Parallel Architectures and Requirements for Pre-Exascale

S.D. Hammond, Center for Computing Research
Scalable Computer Architectures
Sandia National Laboratories, NM
sdhammo@sandia.gov

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Disclaimer

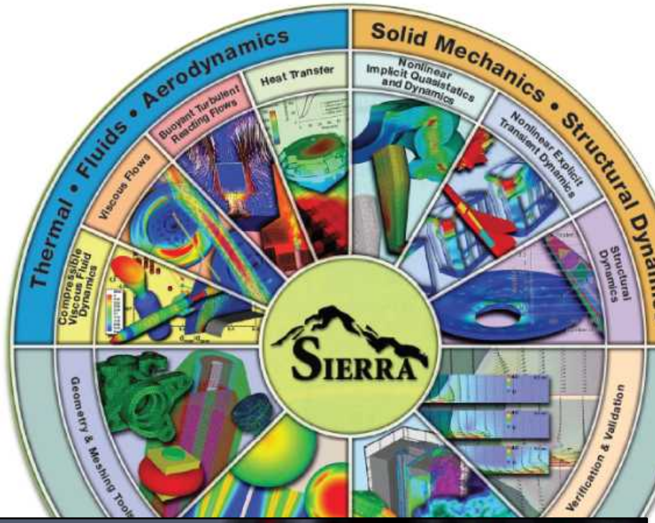
- This is just *an* opinion from Sandia
- Does **not** represent any official position from Sandia or our research group
- But .. we do need a serious conversation about how to make the ecosystem get to where we need it to be
- Part of longer term conversation with/in the community
    - Help be part of it ..

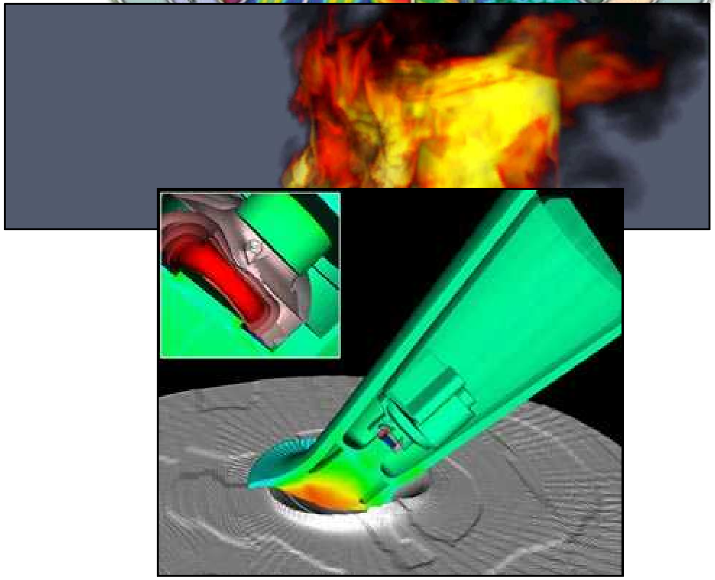
Mini-Overview of Sandia

- National Laboratory with sites in across the country (DOE, DoD, Industry etc)

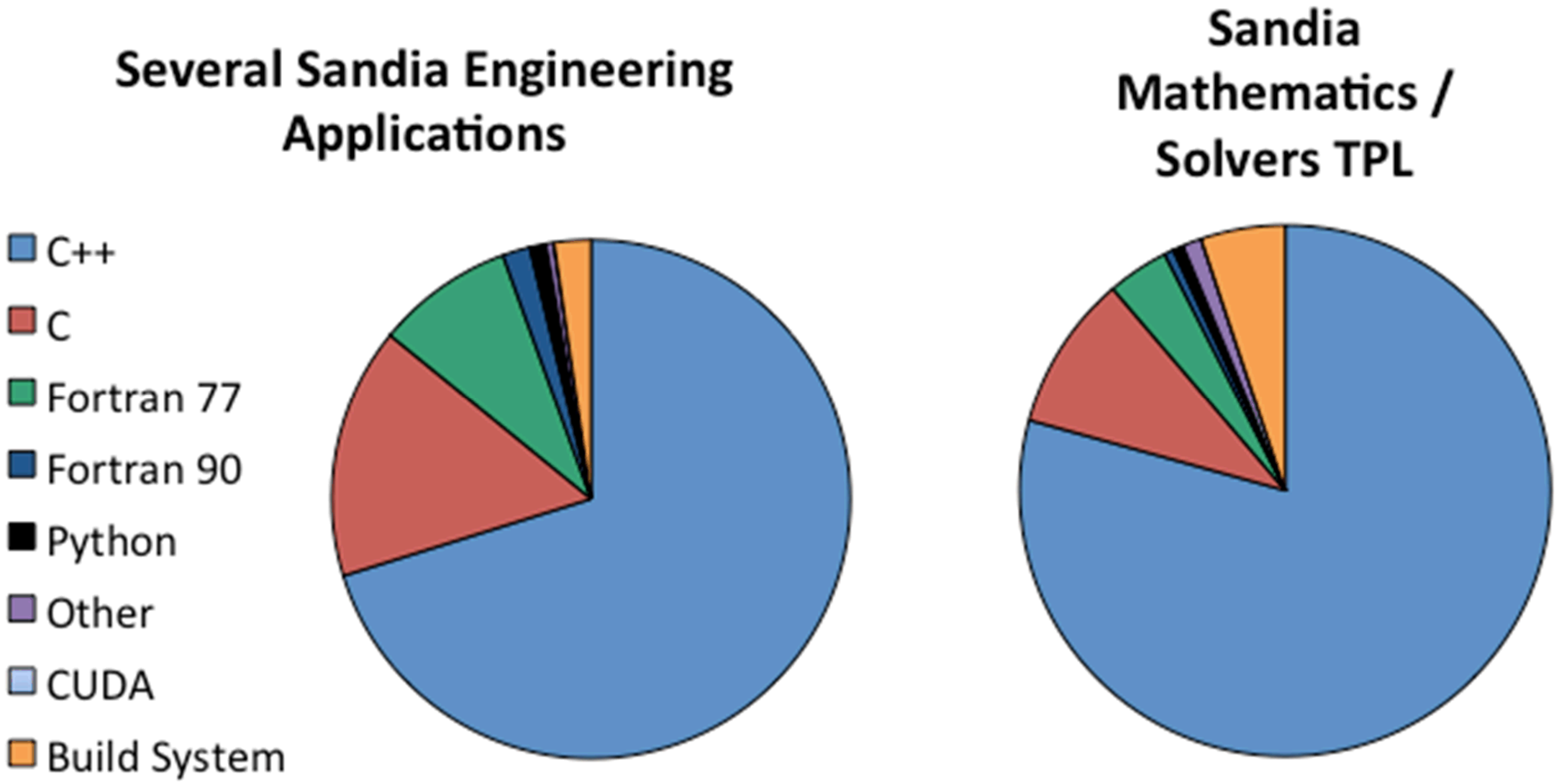
- Part of the NNSA Trilab complex associated with ensuring safety of the nations nuclear arsenal (Sandia focused on engineering)

- We do *much* more
  - Leadership in wide range of engineering
  - Supports complex data analytics research
  - Renewable energy
  - Partnerships with industry
  - Systems for space/satellites/hostiles
  - Strong mathematics research
  - Quantum computing and novel devices
- **All supported by broad HPC requirements**


Thermal • Fluids • Aerodynamics
Solid Mechanics • Structural Dynamics
Heat Transfer
Buoyant Turbulent Reacting Flows
Viscous Flows
Compressible Viscous Fluid Dynamics
Nonlinear Implicit Quasistatics and Dynamics
Nonlinear Explicit Transient Dynamics
Structural Dynamics
Verification & Validation
Geometry & Meshing Tools
SIERRA



What is the Problem?



Several Sandia Engineering Applications

Sandia Mathematics / Solvers TPL

~11.6M **Application** Lines of Code
(Several Applications, Much Shared)
>50 Third Party Libraries

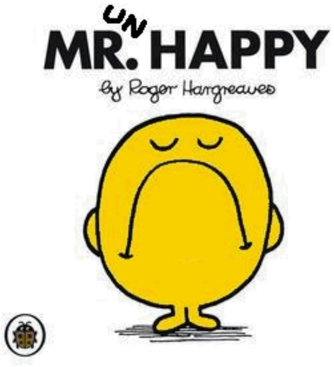
~4.2M Lines of Code
(Very Large Proportion Shared)

This is just a **small** part of our application portfolio

https://github.com/trilinos/trilinos

This is lines of code, does not include comments, white space, documentation etc, no meshing, viz, analysis *etc*

So What Really is the Problem?



- Lots of codes and applications
  - Think meshing, viz, analysis, simulation etc etc
- These codes are **big**
  - Lots of third party libraries – these are great until just one doesn’t run on your platform and then you’re in trouble with the users
- These codes are **trusted** by our user community, by our sponsors, our customers (the people who *actually* pay the bills)
- Scale and age are a big problem
  - Legacy code sometimes means legacy algorithms and legacy language standards

So What are We Doing?

- Searching for the Holy Grail

✓ Portable

✓ Performant

✓ Scalable

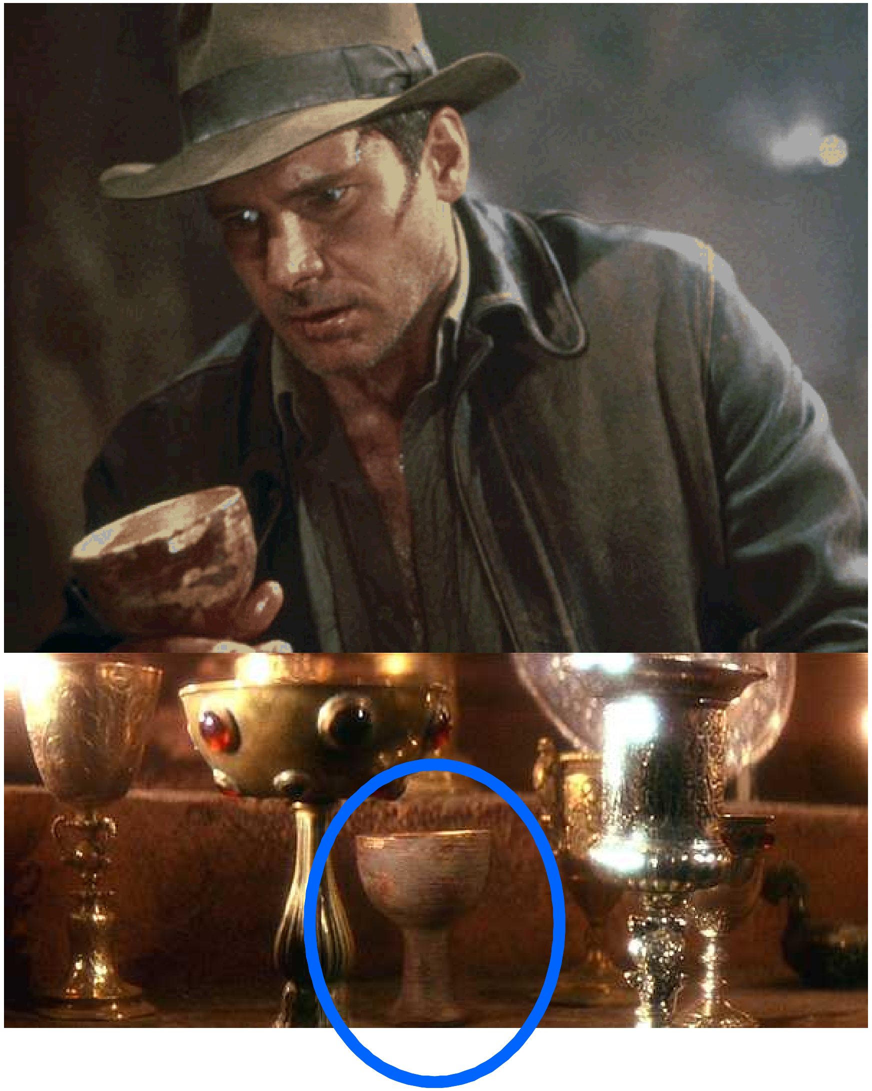
✓ Open Standard

✓ Easy to use

✓ Even easier to Debug

✓ Easier to Profile

- Many things are promised but true success is the hearts and minds of programmers



What Are We Doing?

- **Our main answer is to use Kokkos**

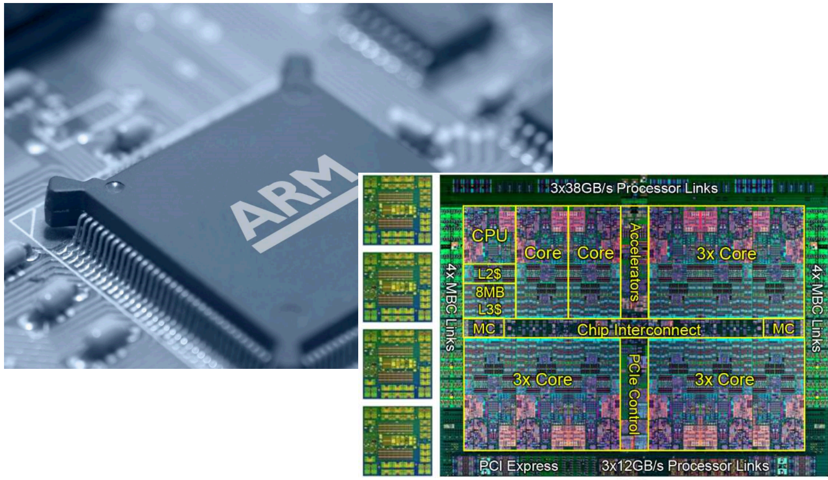
- C++ Meta Templated Library
- Prototype for how we thing future C++ may look
- Incredibly powerful
- Abstracts out backends (CPU, Phi, GPU)
- Abstracts out multiple memory spaces

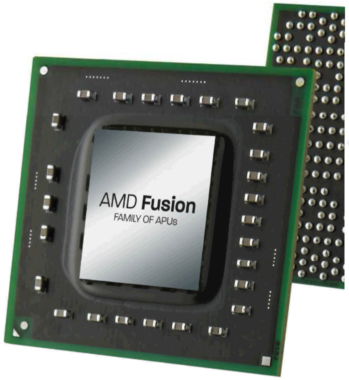
- http://www.github.com/kokkos
- Carter Edwards and Christian Trott at Sandia

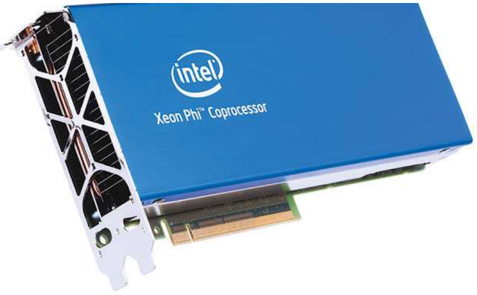
- RAJA from LLNL (Jeff Keasler and Rich Hornung)

SC Poster: **C++ ABSTRACTION LAYERS – PERFORMANCE, PORTABILITY AND PRODUCTIVITY**

http://prod.sandia.gov/techlib/access-control.cgi/2015/157886.pdf







What Else Are You Thinking?

- **Directives – so annoying to be stuck in the middle of multiple standards**
  - Where is the multi-level memory support?
  - Multiple devices?
  - Why am I writing more directives than code!?
  - Still .. I think the route for many in the community (Fortran for sure)

- **DSLs – just don't seem to be broad enough for what we need**
  - Multi-physics demands that all this fits together
  - How do I debug multiple DSLs
  - Why don't you give up *your* DSL and use mine?

- **Asynchronous Many Tasks – just starting to become serious**
  - Problem is no standard runtime, not even an agreement on what is needed? Same thing as directives all over again
  - Often a "one-application" programming model

If I could have just one thing….

- I'd wish for what we all want ..

- World Peace
- Harmony
- And .. a Long Life

If I could have just one thing…

- **World Peace – Standardization (Abstraction?)**
  - To stop being stuck in the middle of wars over standards where I am forced to pick a side by tools, compilers, *etc*
  - What I *really* want is one code and to select the best hardware
  - Yes, we need some of this but we also need to understand that this *limits* what we can choose to do and I think it limits vendors ability to sell

- **Harmony - Interoperability**
  - To have programming models work together on the same hardware
  - No .. You don't own *all* the hardware, you just own this bit *now*
  - Critical for large applications which needs libraries and interoperability

- **A Long Life – Longevity (Standards? Support? Future Proof?)**
  - To have standards which are going to work going forward
  - To be truly be worth the considerable investment and give me a good return on my time

Sandia National Laboratories

Exceptional service in the national interest

http://www.github.com/kokkos