MTV Student Virtual Research Symposium

Hammer: a flexible research framework for radiation transport solvers

June 11 2020

Kyle Beyer
University of Michigan
Department of Nuclear Engineering and Radiological Sciences
PhD, Year 1
Prof. Brian Kiedrowski

Evan Gonzalez, Noah Kleedtke, Aaron Tumulak, Emily Vu, Avery Grieve, Thomas Huster, Eric Pearson, Matan Gottesman, Lincoln Johnston, James Pelkey, Joseph Donato, Julien Faro, Kayce Duggan, Jeremy Eshbaugh

beykyle@umich.edu

Motivation and Mission Relevance

Computational radiation transport tools

- model complex radiation transport scenarios to support numerous applications including the design and analysis of radiation detectors, nuclear reactors, shielding and criticality safety, and numerous others
- Typically grouped into two classes: deterministic and Monte Carlo
    - Deterministic transport is fast, but requires discretization in space, direction, and energy
    - Monte Carlo does not require discretization and is very accurate, but is usually slower

Our goal: Develop an open source framework to design novel radiation transport techniques encompassing both deterministic and Monte Carlo transport to provide a flexible tool for researchers to rapidly prototype novel ideas

Hammer is developed, maintained, and managed by students!

Why does Hammer exist?

Production codes

- Fast and well validated, and ideal for production analyses
- Steep learning curve to test new ideas that require modifying the source
- Long history of these products leads to managing around legacy coding practices

Hammer

- Captures complexity of real-world nonproliferation problems
- Easy to add new features
- Modern software engineering best practices
- Educational platform for modern SQA

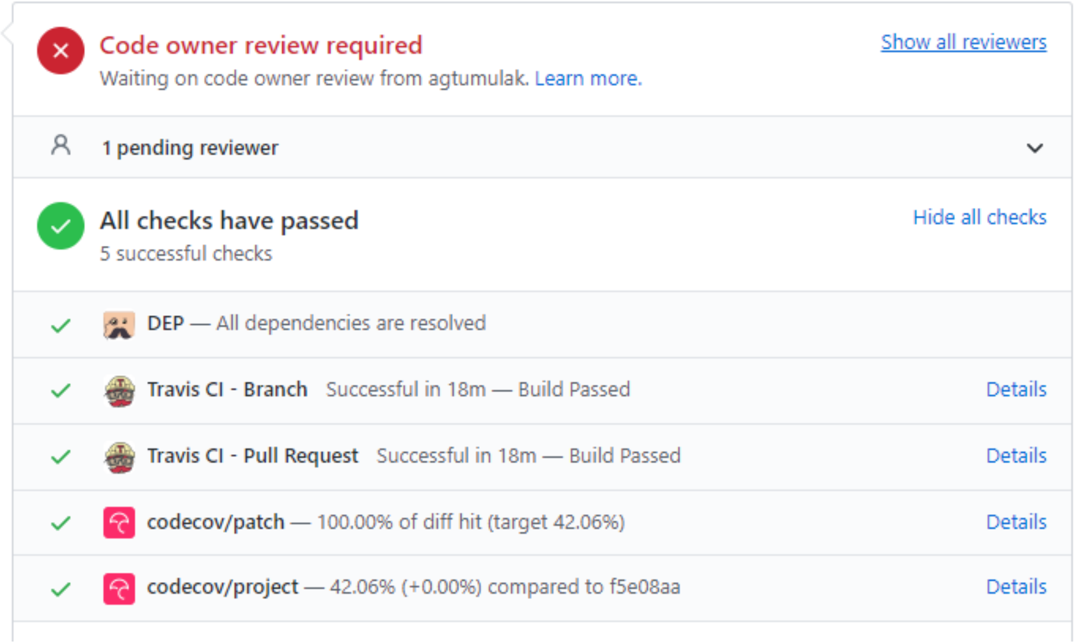
Research/test codes

- Very simple problems
- Disposable; only used to test one new algorithm/idea without broader context
- Not fully featured

What technologies does Hammer use?

We use:

- object-oriented c++17
- cross-platform CMake build system
  - Windows, MacOS, Linux, CAEN Linux server
- git version control; currently hosted on Github
- unit testing & continuous integration
  - Travis CI, CodeCov, Catch2
- supports threading (MPI support in future)
- Doxygen documentation
- Paraview output support
- human-readable xml interface



Human-readable interface

- human-readable xml interface, parsed with pugixml

```xml
<materials>
  <material name="test_mat" density="1.0">
    <nuclide name="h1" frac="0.5"/>
  </material>
</materials>

<estimators>
  <estimator name="total_flux_tl" type="flux" method="track length" apply="my_rectangular_mesh" particle="neutron" output_format="vtk">
  </estimator>
  <estimator name="total_flux_col" type="flux" method="collision" apply="my_rectangular_mesh" particle="neutron" output_format="vtk">
  </estimator>
</estimators>

<surfaces>
  <sphere name="test_sphere_1" x0="0.0" y0="0.0" z0="0.0" rad="1.0"/>
  <sphere name="test_sphere_2" x0="0.0" y0="0.0" z0="0.0" rad="2.0"/>
</surfaces>

<output path="../output/" />

<cells>
  <cell name="test_cell_1" material="test_mat">
    <surface name="test_sphere_1" sense="-1"/>
    <variance_reduction type="forced collisions" roulette="1"/>
  </cell>
  <cell name="test_cell_2" material="test_mat">
    <surface name="test_sphere_1" sense="+1"/>
    <surface name="test_sphere_2" sense="-1"/>
    <variance_reduction type="forced collisions" roulette="1"/>
  </cell>
</cells>

<meshes>
    <cartesian_mesh name="my_rectangular_mesh" bins='1000000'>
      <x min="0" max="0.05" bins="100" />
      <y min="0" max="0.05" bins="100" />
      <z min="0" max="0.1" bins="100"/>
    </cartesian_mesh>
</meshes>
```

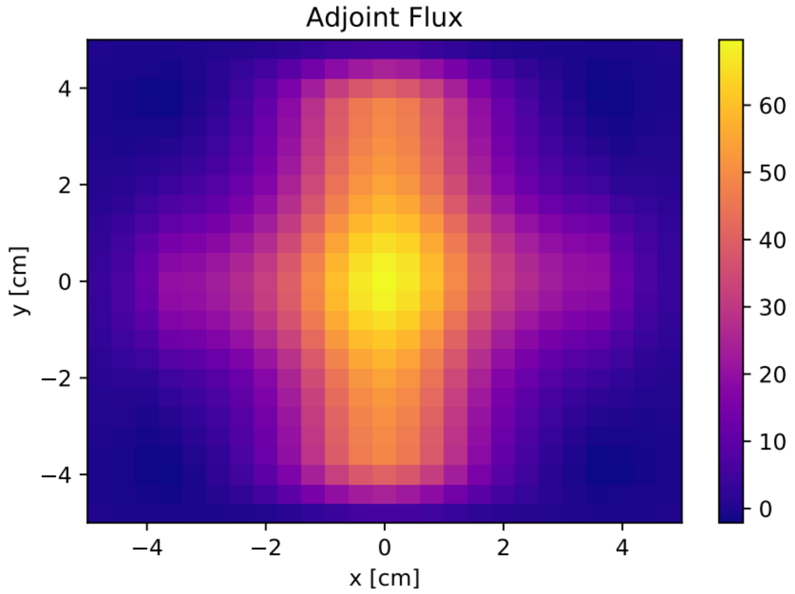
What can Hammer do?

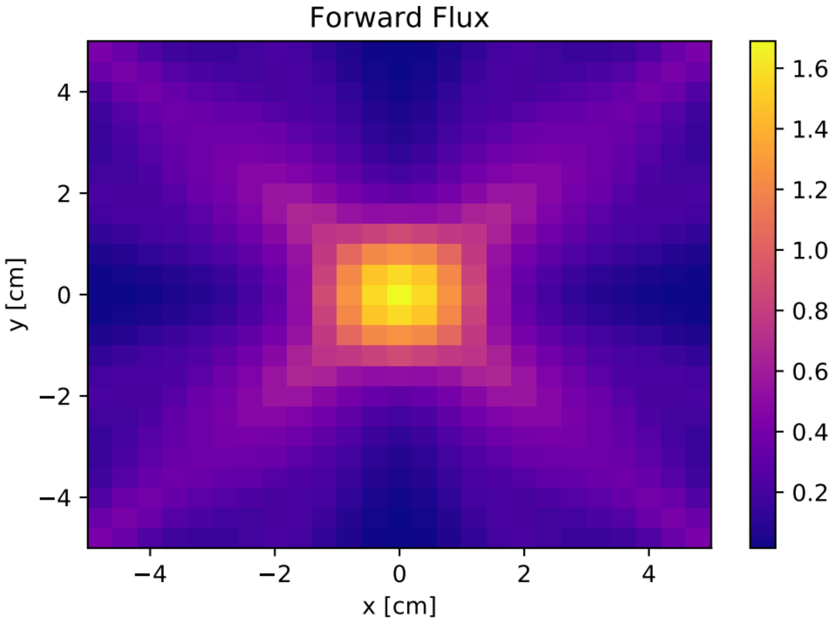
Fixed-source calculations:

- 3D Monte Carlo; multigroup and continuous energy
- 2D SN; forward and adjoint
- neutrons and photons

Mesh based calculations:

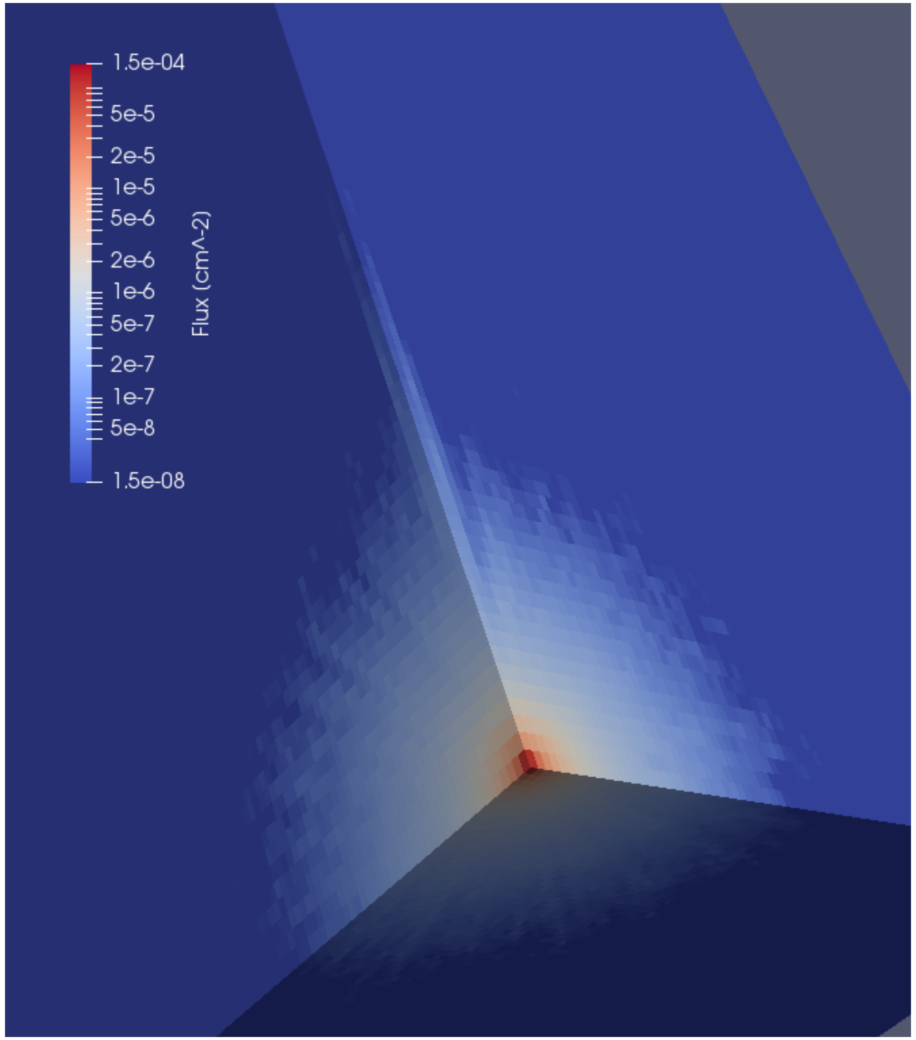
- structured rectangular, cylindrical and spherical
- unstructured tetrahedral
- 1D/2D/3D with rotations
- used for estimators, variance reduction, SN





What else can Hammer do?

- **Full-featured Monte Carlo variance reduction:**
  - importance splitting, rouletting
  - weight windows
  - exponential transform
  - forced collisions
  - forced flight (DXTRAN)
- **Monte Carlo estimators:**
  - track length/collision flux in volume
  - flux/current on surface
  - Flux at a point
  - bin over energy, group, collision order, time, angle, spatial mesh, …
  - flux tally modifier for reaction rate



What will Hammer be able to do?

Ongoing feature projects:

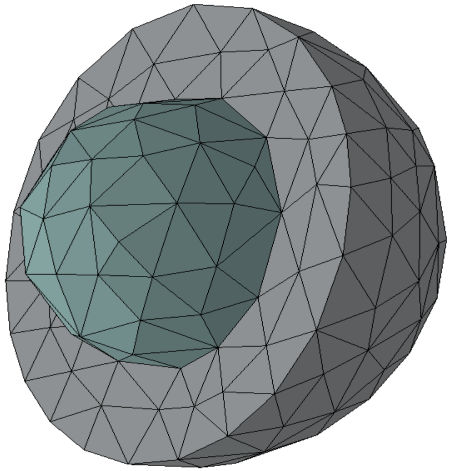
- fully featured CSG: lattices, regions, complements, …
- transport in stochastic media
- Hybrid MC/deterministic methods, including FW-CADIS
- in-house physics event generator for fission, radioactive decay, …

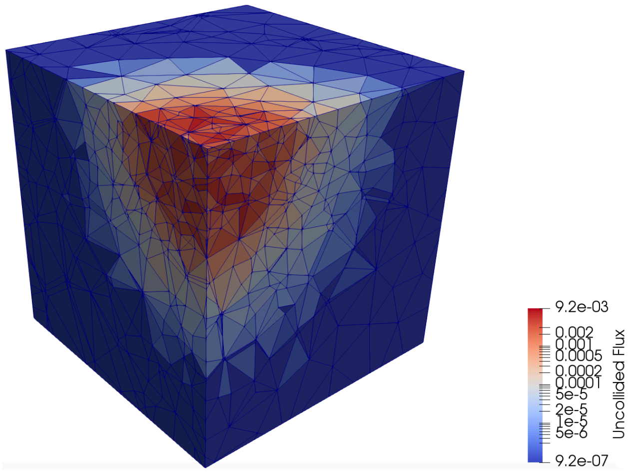
Up next:

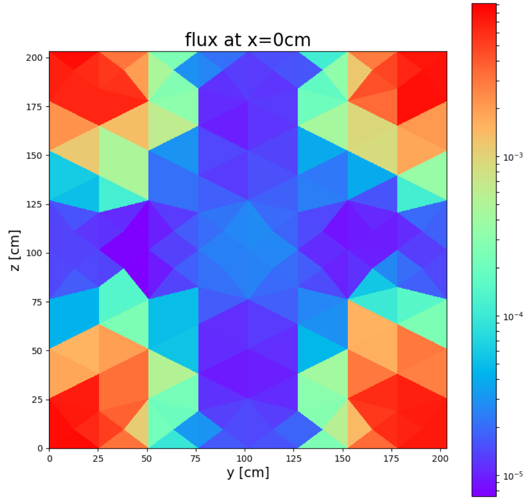
- k/α-eigenvalue
- differential operator sampling for sensitivity analysis
- charged particle transport
- neutrino transport
- depletion
- MPI
- MCNPX-PoliMi style particle tracking for custom analysis of detector problems

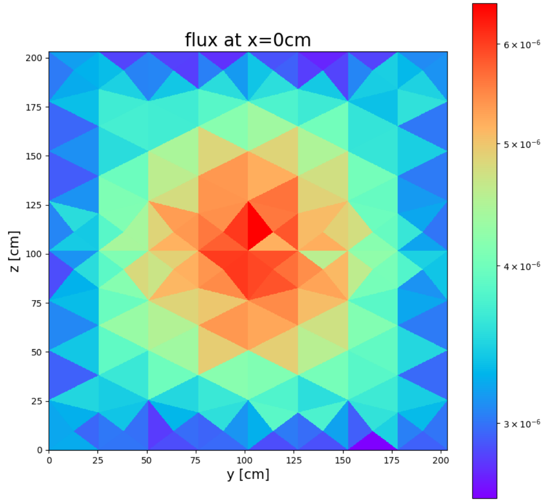
Hybrid Methods: Direct Coupling

- Collaboration with deterministic solver, NC State code, THOR
  - Highly parallel S_N transport solver on an unstructured tetrahedral mesh
  - suffers from numerical artifacts called "**ray effects**"
- Hammer provides n^{th}-order collided source to the THOR
- Coupling mitigates ray effects

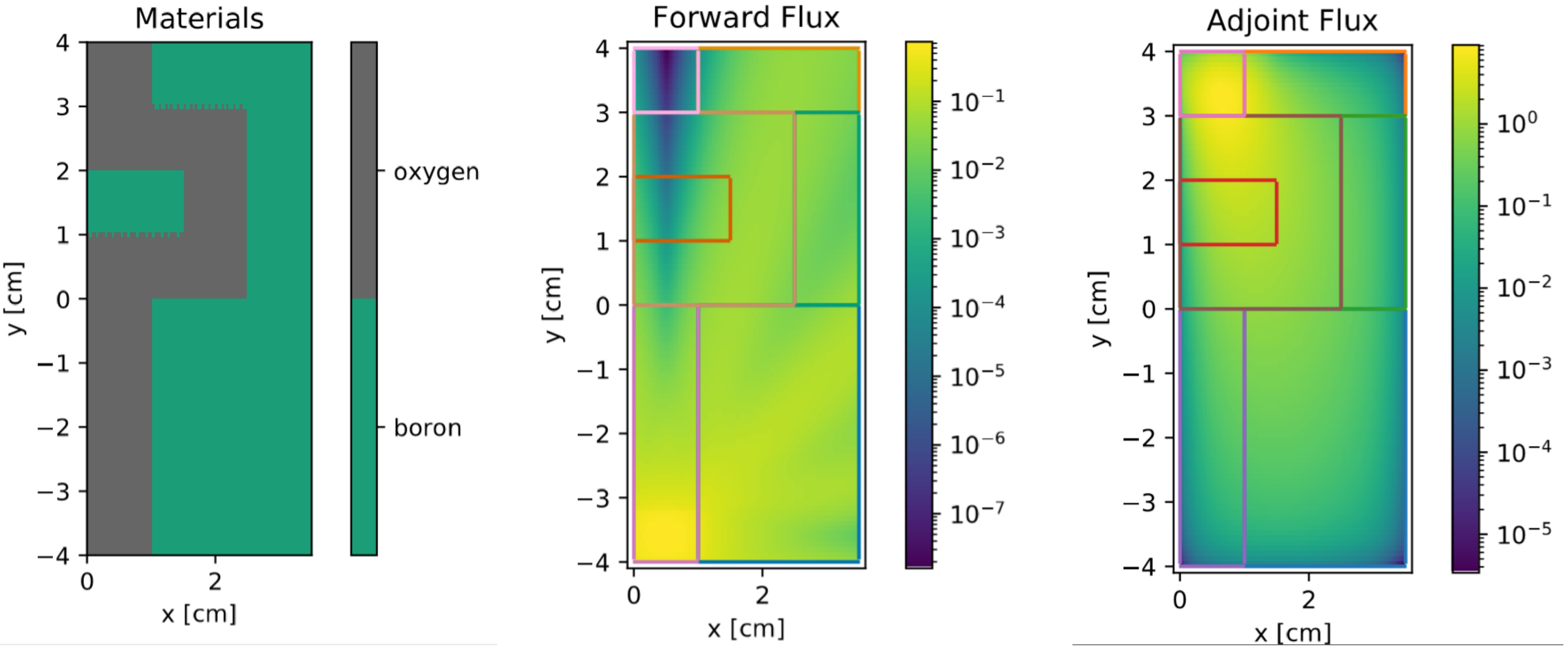








Hybrid Methods: FW-CADIS



MTV and National Lab Impact

We have summer 2020 internships/ongoing collaborations with:

- LANL (x2)
  - advanced forced flight variance reduction, hybrid methods, unstructured mesh, critical experiment design
- ORNL (x2)
  - expand variance reduction and sensitivity analysis capabilities of Shift
- LLNL (x2)
  - transport in stochastic media
- SNL (x1)
  - transport in stochastic media

beykyle@umich.edu

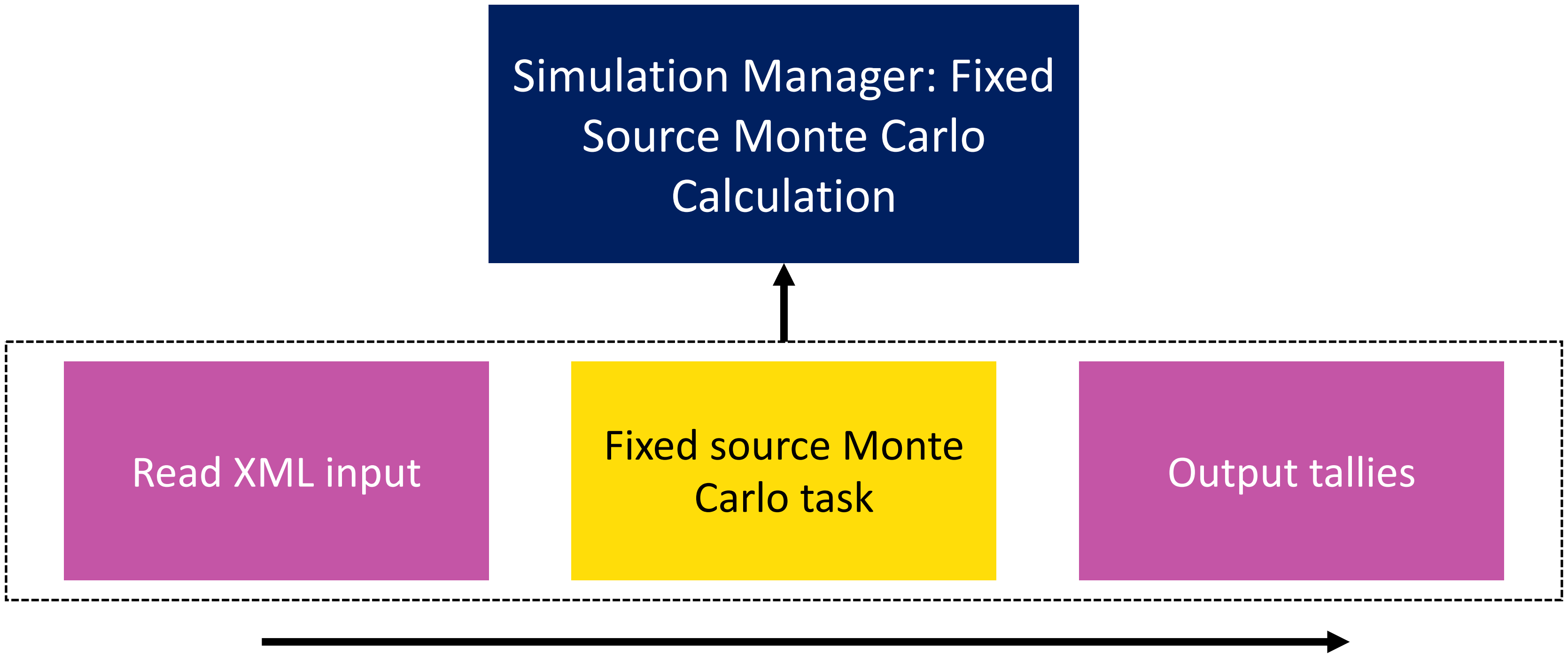
Conclusions and Impact

Hammer will:

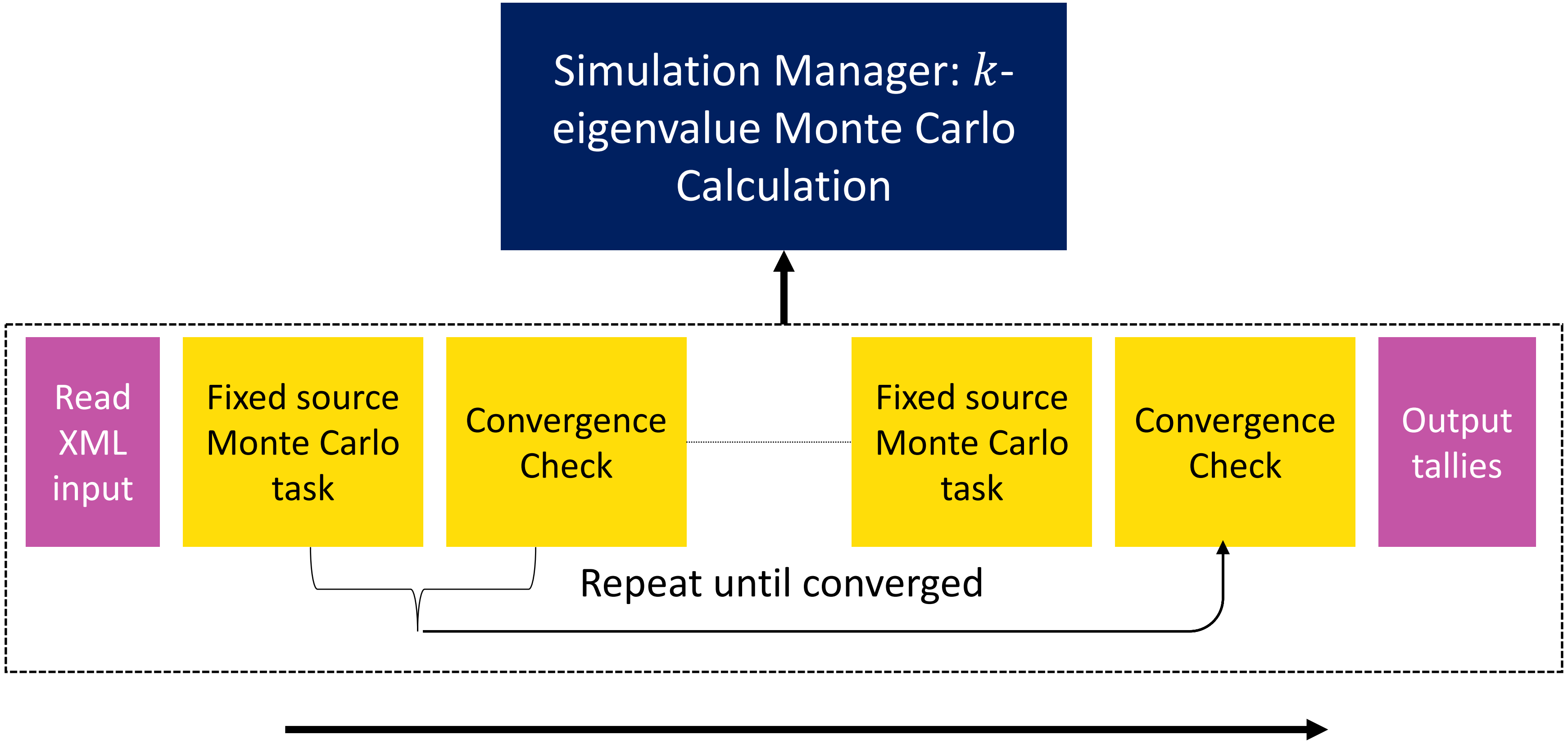
- support the development of next-generation monitoring technologies by providing the tools to design and model them
- make it easier to tailor the solver to a specific application/experiment
- provide hands on training for the next generation of computational physicists in in-demand software engineering and project management skills

Contact: Prof. Brian Kiedrowski (bckiedro@umich.edu)

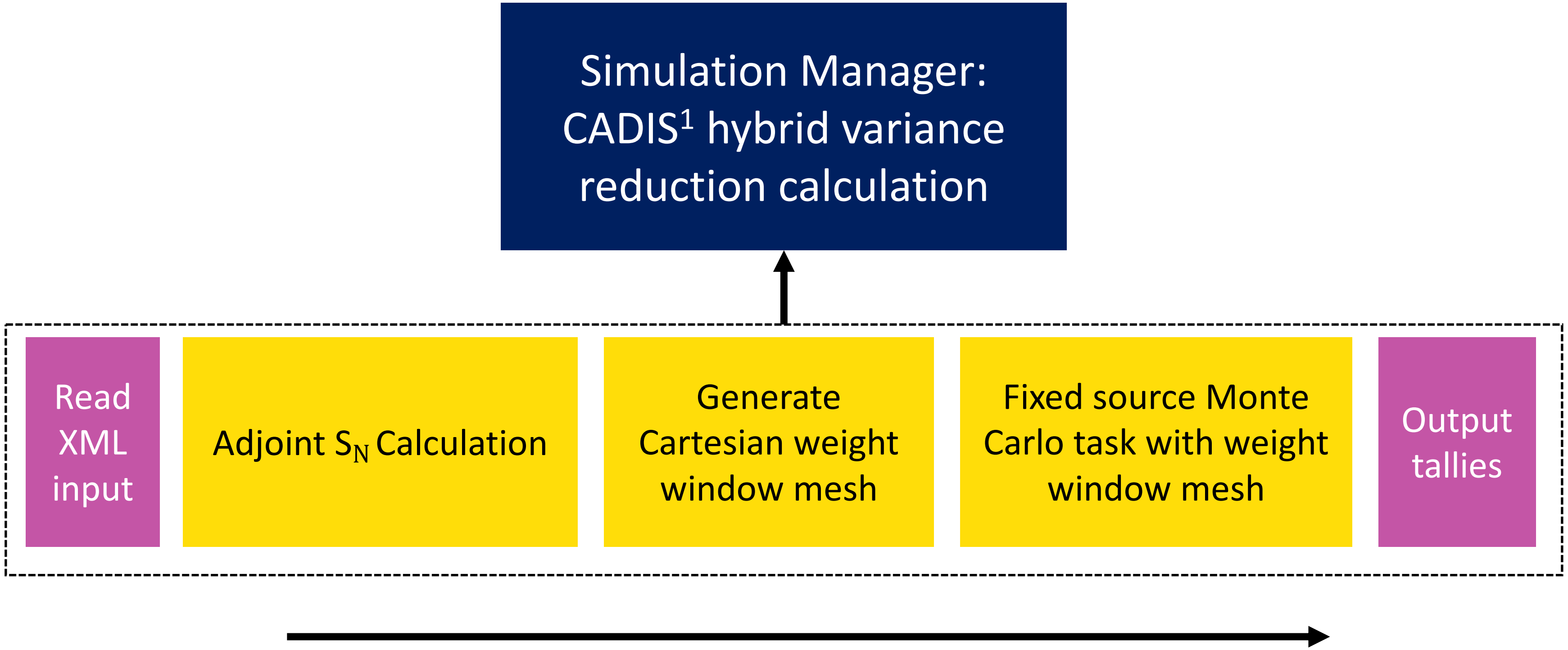
Hammer: a modular design



Hammer: a modular design



Hammer: a modular design



1. J. C. Wagner and A. Haghighat. "Automated Variance Reduction of Monte Carlo Shielding Calculations Using the Discrete Ordinates Adjoint Function," *Nuclear Science and Engineering*, 128, pp. 186–208 (1998).

Acknowledgements

The Consortium for Monitoring, Technology, and Verification would like to thank the NNSA and DOE for the continued support of these research activities.

This work was funded by the Consortium for Monitoring, Technology, and Verification under Department of Energy National Nuclear Security Administration award number DE-NA0003920

beykyle@umich.edu