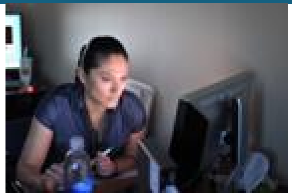
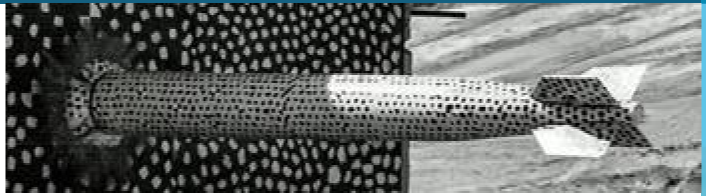
SAND2020-2204PE

# CHEETAH-MC

## Next-Generation Monte Carlo Transport





*PRESENTED BY*

Kerry Bossler

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.
SAND2020-2204 PE

# What is CHEETAH-MC?

- New Monte Carlo coupled electron-photon transport code under active development
  - Built on the foundation of the Integrated Tiger Series (ITS)
  - Simulates behavior of electrons and photons in one or more fixed background materials
  - Designed to be user-friendly and run on next-generation heterogeneous architectures
- Presentation outline
  - Where are we today?
  - Where are we going?

# Where are we today?

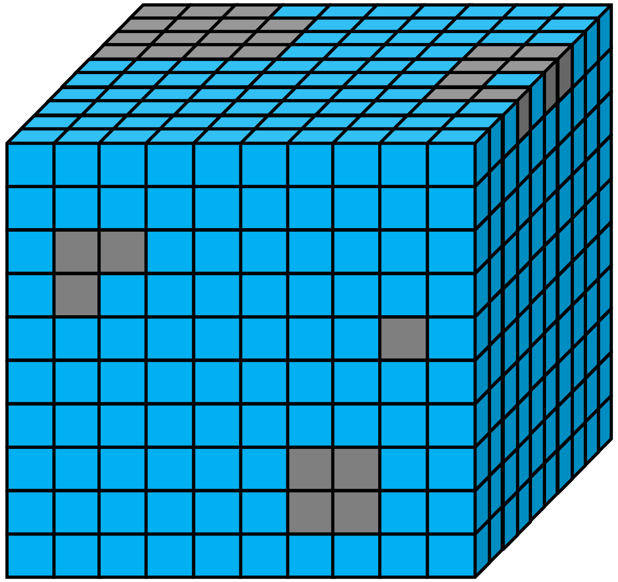
Features that exist or are under development in FY20

# Build System

- Based on modern CMake and compatible with Spack
- Build Options:
  - Serial CPU – without Kokkos
  - Serial CPU, Pthreads, OpenMP, and CUDA – with Kokkos
  - Adding MPI support in FY20
- TPLs can be included either as a subdirectory or pre-installed
  - **SKEPTXS** for reading, parsing, and interpolating cross sections
  - **Albacore** for a set of useful libraries and tools shared between multiple codes
  - **YAML/JSON** for input parsing and validation
  - **Googletest** for unit testing
  - **Kokkos** for running on different architectures

# Geometry Options

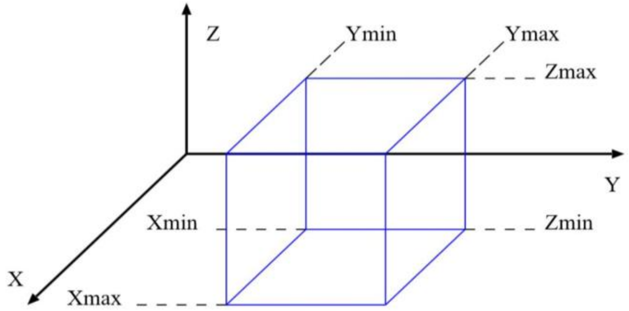
## Voxel Geometry

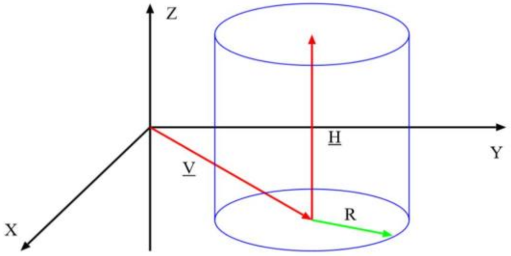


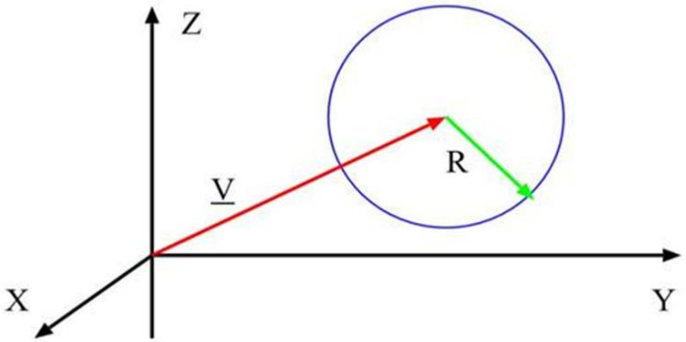
Material 1

Material 2

## Combinatorial Geometry (CG)




**Adding Boolean Combinations FY20**

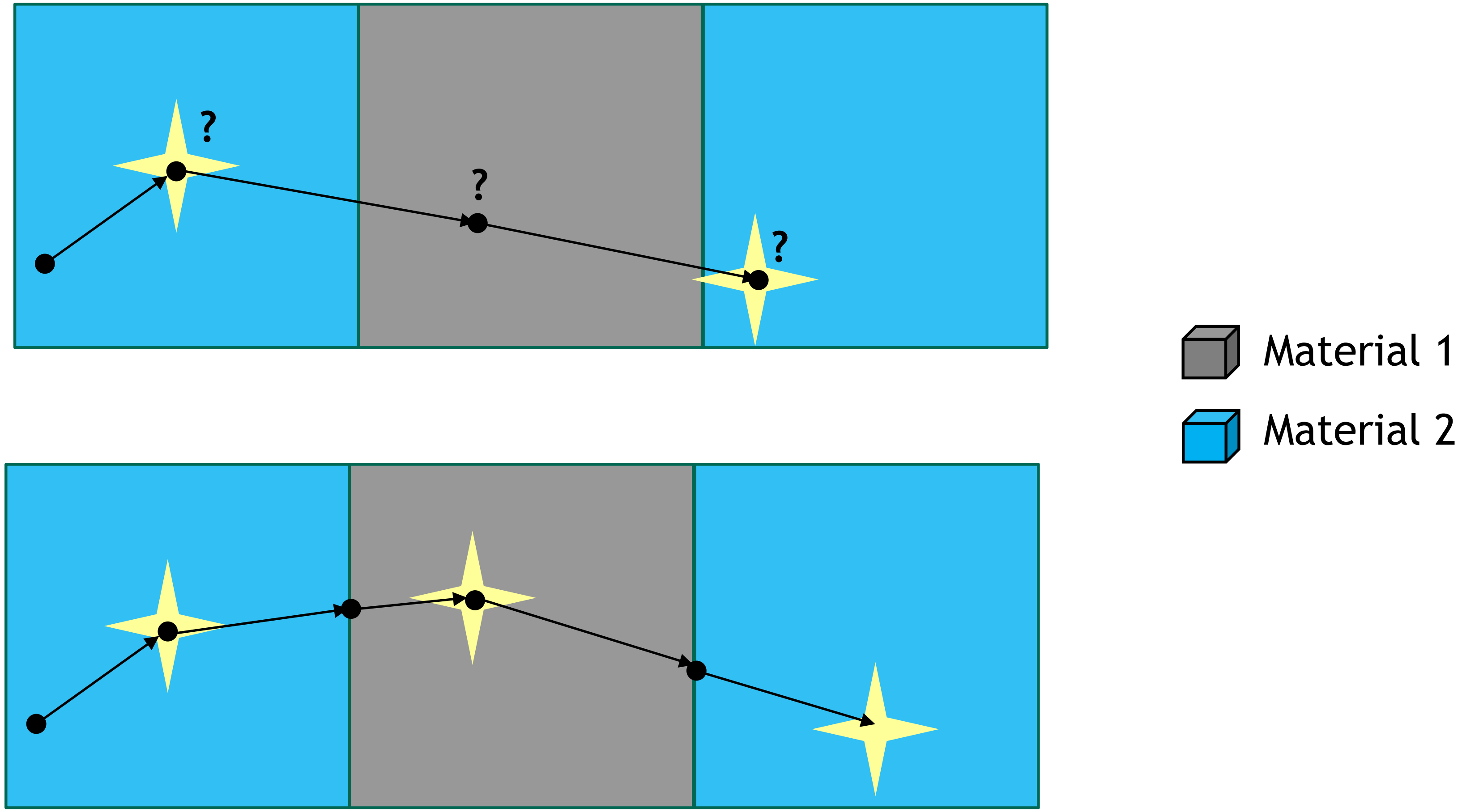
# Particle Tracking Options

WOODCOCK TRACKING

- Good for GPU
- Ignores surfaces
- Pseudo-collisions

SURFACE TRACKING

- Traditional method
- Tracks surfaces
- Real collisions



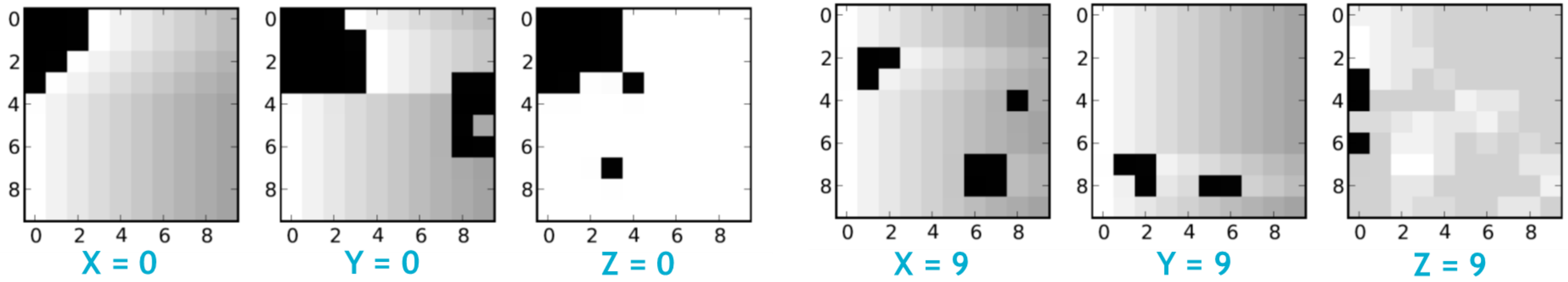
# Other Input Options

- Particle Source
  - Mono-energetic and mono-directional
  - Starting location can be uniformly sampled as either a point source, line source, surface source, or volume source
- Physics interactions
  - Only photons interactions are being implemented in FY20 (i.e., coherent scattering, incoherent scattering, photoelectric effect, pair production)
  - SKEPTXS input can be added directly to CHEETAH input
  - Support multiple materials but each "material" can only include one element

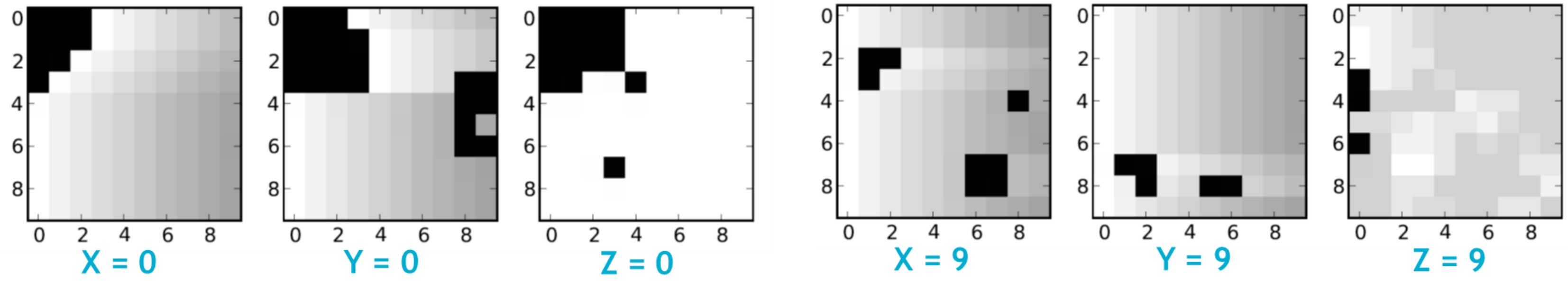
# Code Outputs

- Particle Counts
  - Total number that escaped or were absorbed
  - Fraction of particles that were absorbed per voxel element or CG primitive

## Analytic Solution for Direction +Z




## CHEETAH-MC Results for Direction +Z



# Where are we going?

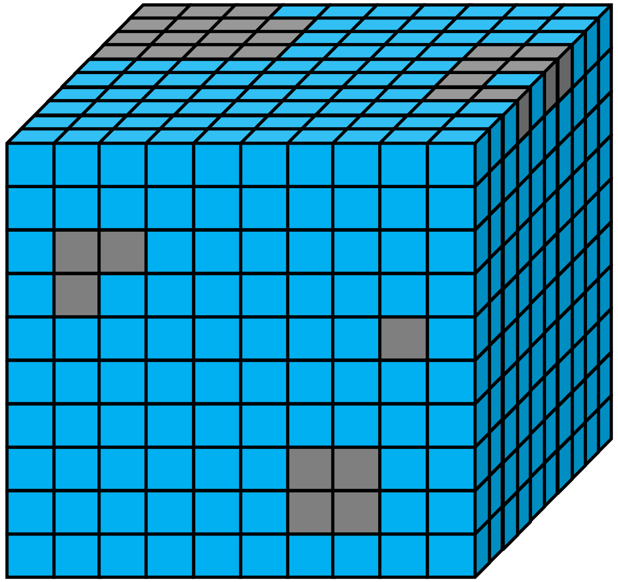
Future Plans for CHEETAH-MC beyond FY20

# Initial Software Architecture Design

| Layer | Problem Specification | Solution Specification | Execution Environment | Post-Processing Environment | Notes |
|---|---|---|---|---|---|
| User Environment | MC Framework Spec.<br>Geometry Editors<br>Source/I.C. Editors | Solution Spec.<br>• Particle Type (photon, electron)<br>• Woodcock Tracking, CH, Single Scattering/GBFP…<br>Output Spec. | Execution Tool<br>• Machine Configuration<br>• Checkpoint/Restart<br>• Data Streaming | Visualization Tools<br>Performance Analysis<br>Data Analysis Tools | • UNIX<br>• WIN64<br>• Mac OS |
| Language Infrastructure (Portability/Performance) | Very High Level NGP Language Layer (e.g., Kokkos)<br>HPC Language Layer (e.g., OpenMP, CUDA, OpenCL, MPI)<br>Procedural Language Layer (e.g., C++, Python, OpenGL (?)) | | | | |
| Core Infrastructure | Geometry Decomposition Libs<br>Domain Discretization Libs<br>SKEPTXS — Foreign Library | Knowledge Bases — V&V<br>Machine/Deep Learning | CHEETAH Solver Libs<br>Tally Server<br>"Foreign" Interface Libs | Tally Server Analysis<br>Kernel Density Estimator | • Physics<br>• Transport Methods (Event/History-based)<br>• Geometry Transport<br>  • CG<br>  • Mesh<br>  • Voxel<br>• SPIN<br>• SKEPTXS |
| System Infrastructure | PRNG<br>Geometry Modeling Libs<br>• CG<br>• CAD<br>• Facet<br>• Mesh<br>• Voxel | | "Foreign" System Libs<br>Parallel/HPC Communication Libs (?) | Data Visualization Libs | |
| | OpenGL/Mesa, Boost, Deep Learning -> Computational Steering<br>• In-Situ Visualization<br>• In-Situ AI | Embedded UQ/Stochastic Media | | | |

# Geometry Options

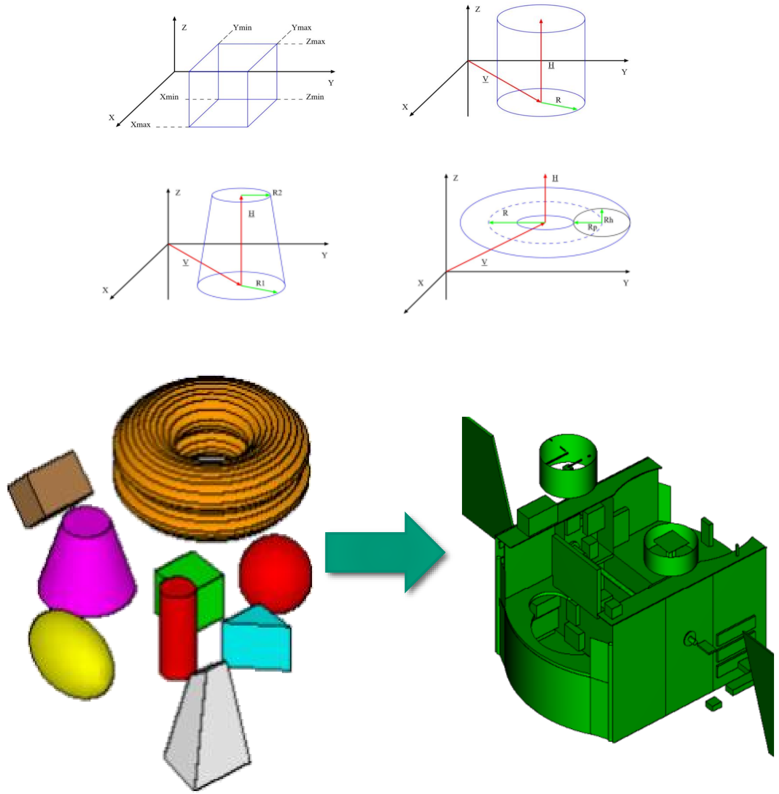
## Voxel Geometry



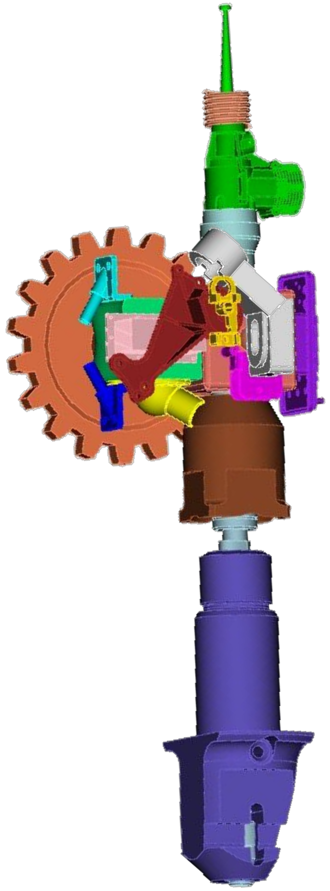
Material 1

Material 2

## Combinatorial Geometry (CG)

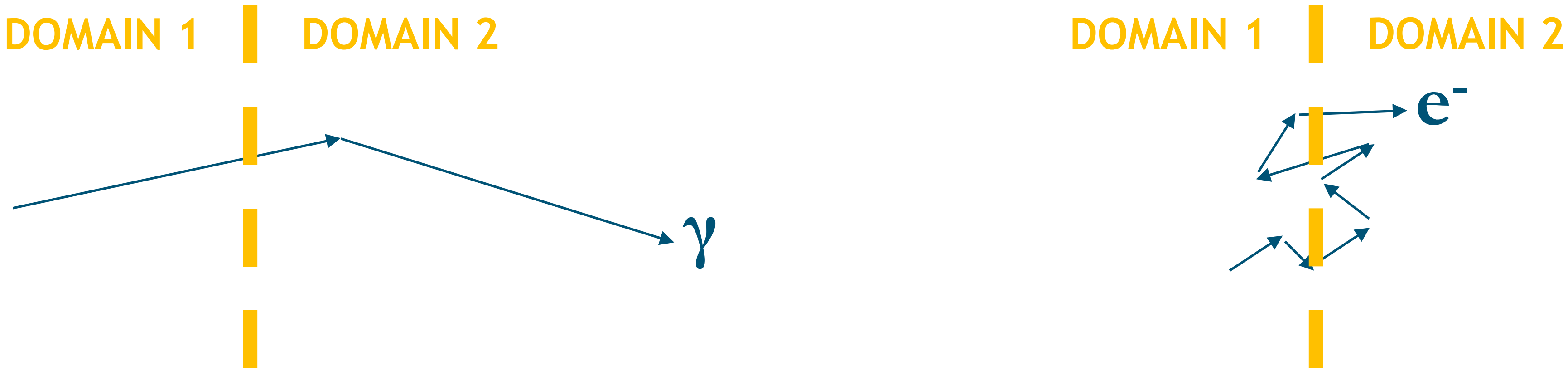


## CAD Geometry



# Adding Electron Physics

- CHEETAH-MC will eventually be a **coupled** electron-photon transport code like its predecessor ITS
- Adding support for electrons is going to be challenging on the GPU
  - Photons produce electrons, and electrons produce photons – more divergence!
  - Likely need to store electron data – more memory!
  - Can we use domain decomposition for electrons that like to scatter a lot, but not travel very far?



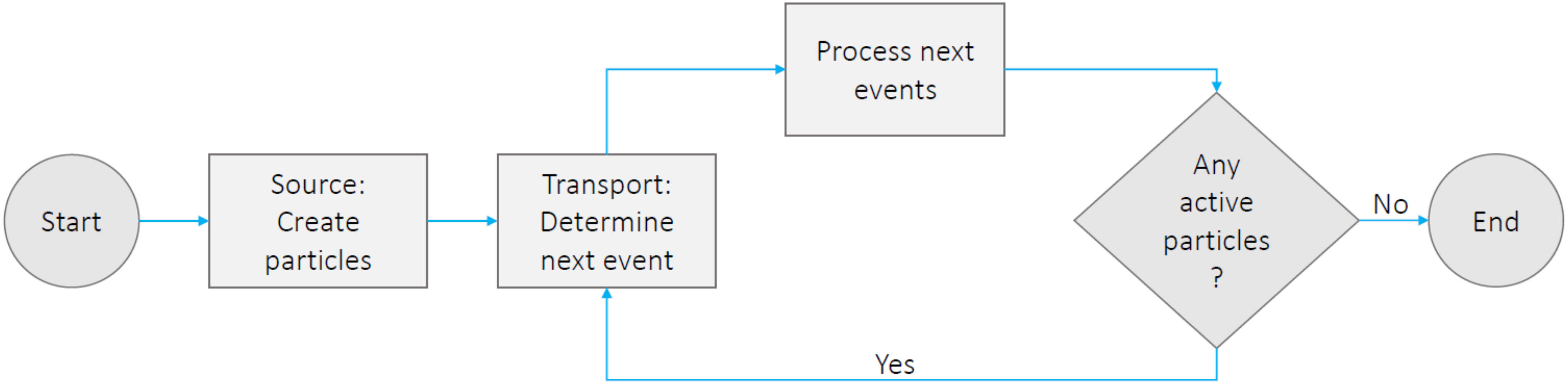
# Event-Based Monte Carlo Particle Transport Algorithm

## Types of Events

### Source
- Create Electron
- Create Photon

### Geometry
- Internal Surface
- Reflection
- Escape

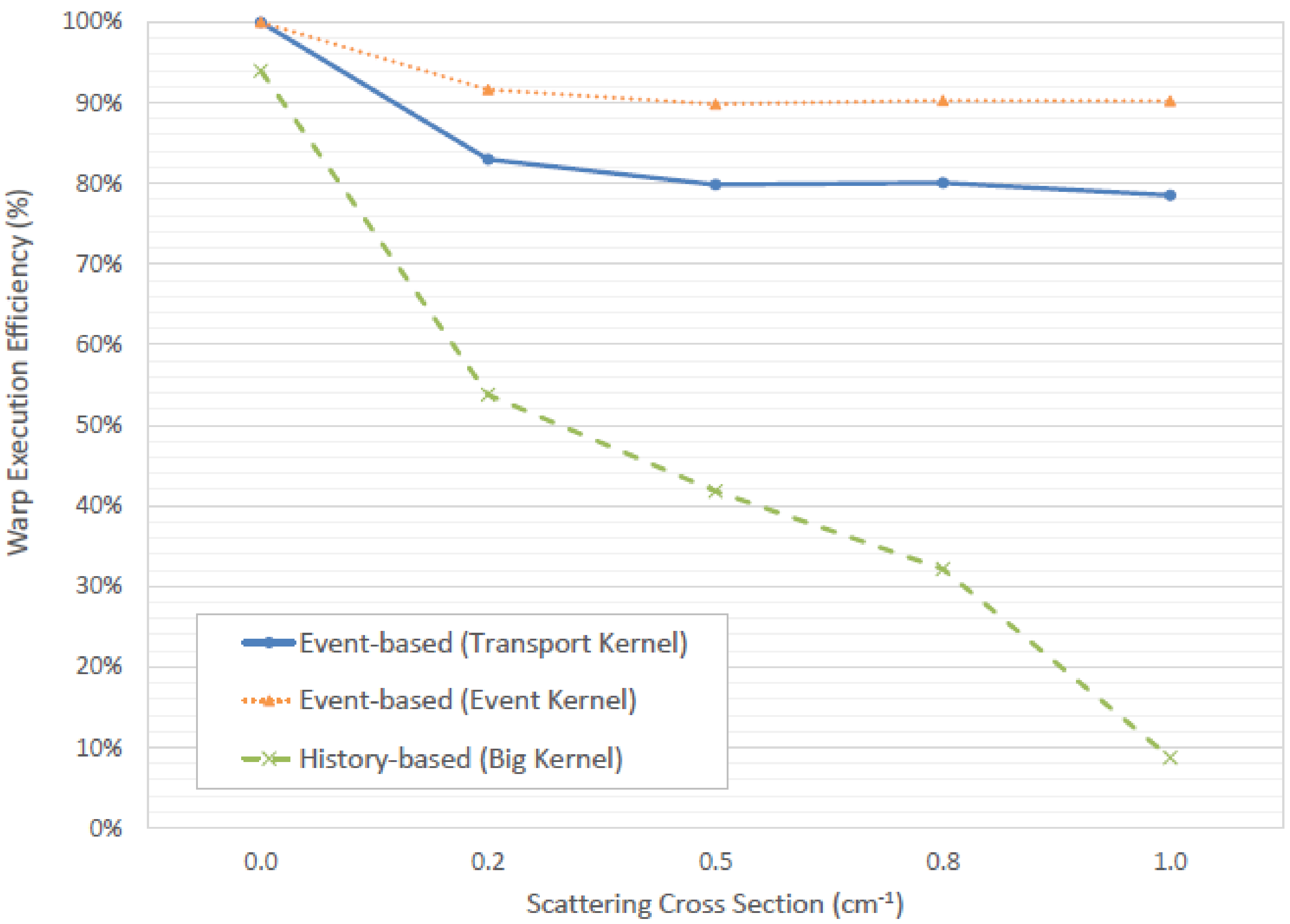
### Physics
- Absorb
- Scatter



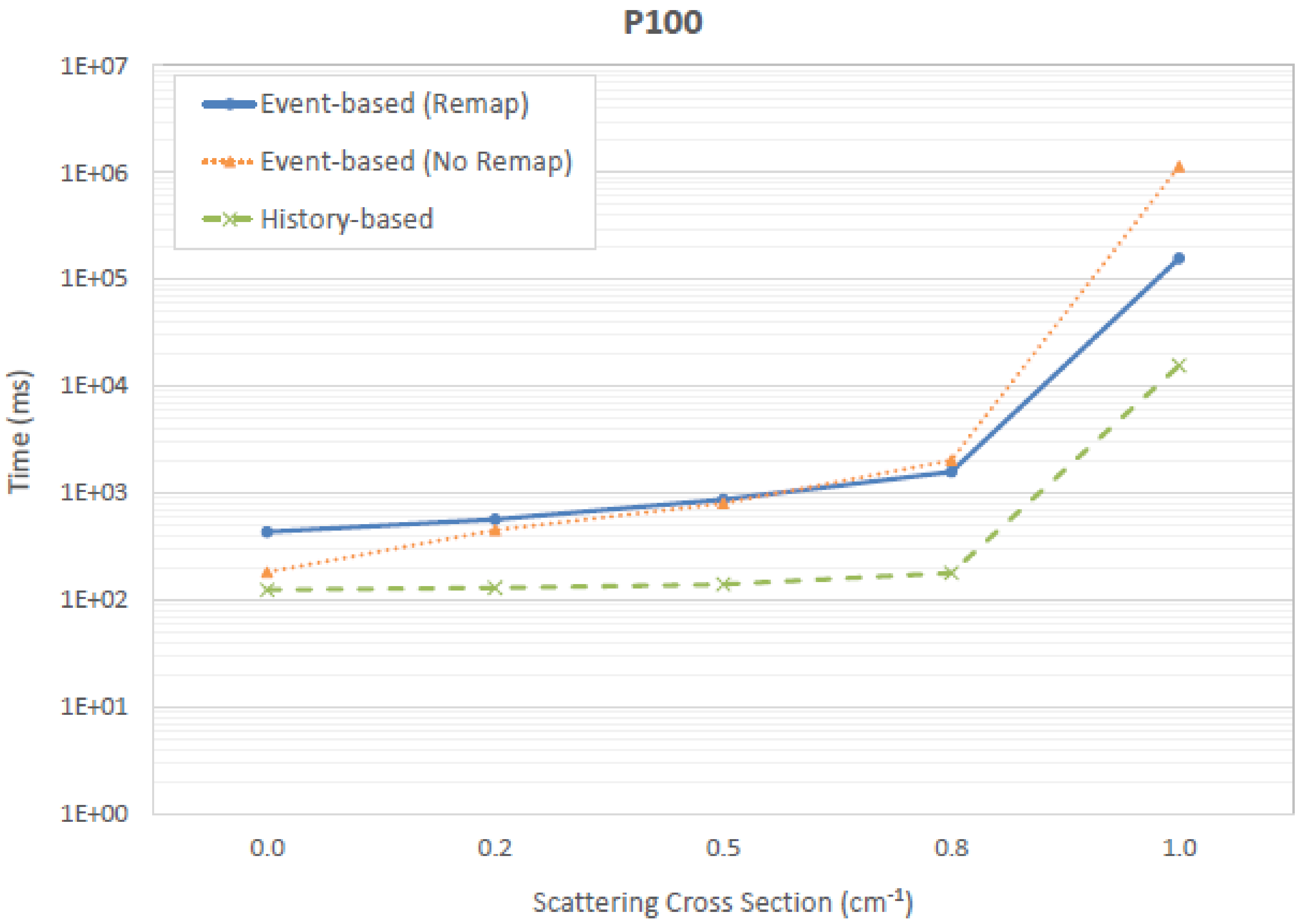
# Reducing Divergence

- Warp execution efficiency is a measure of branch divergence in the code
- Divergence for the Big Kernel increases dramatically as more scattering is added
- Transport and Event Kernel approach a fixed amount of divergence

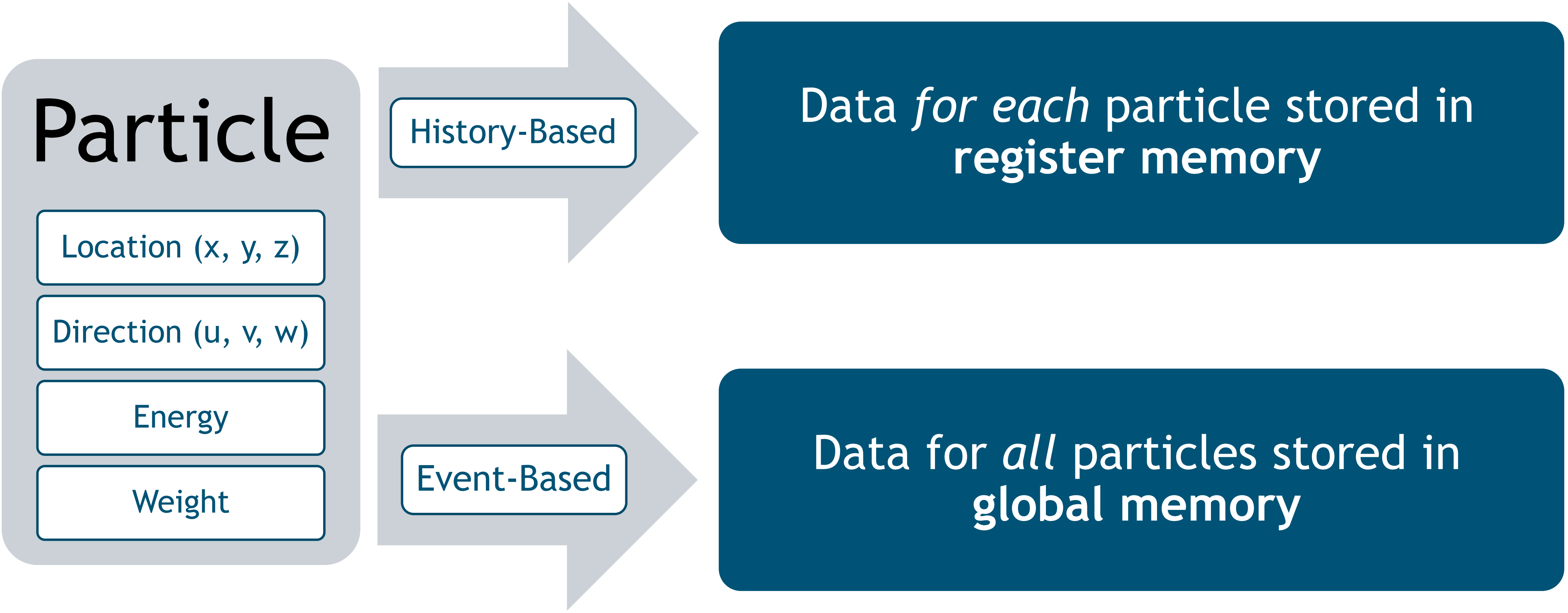
**Event-based algorithm significantly reduces divergence!**



# History-Based versus Event-Based Performance

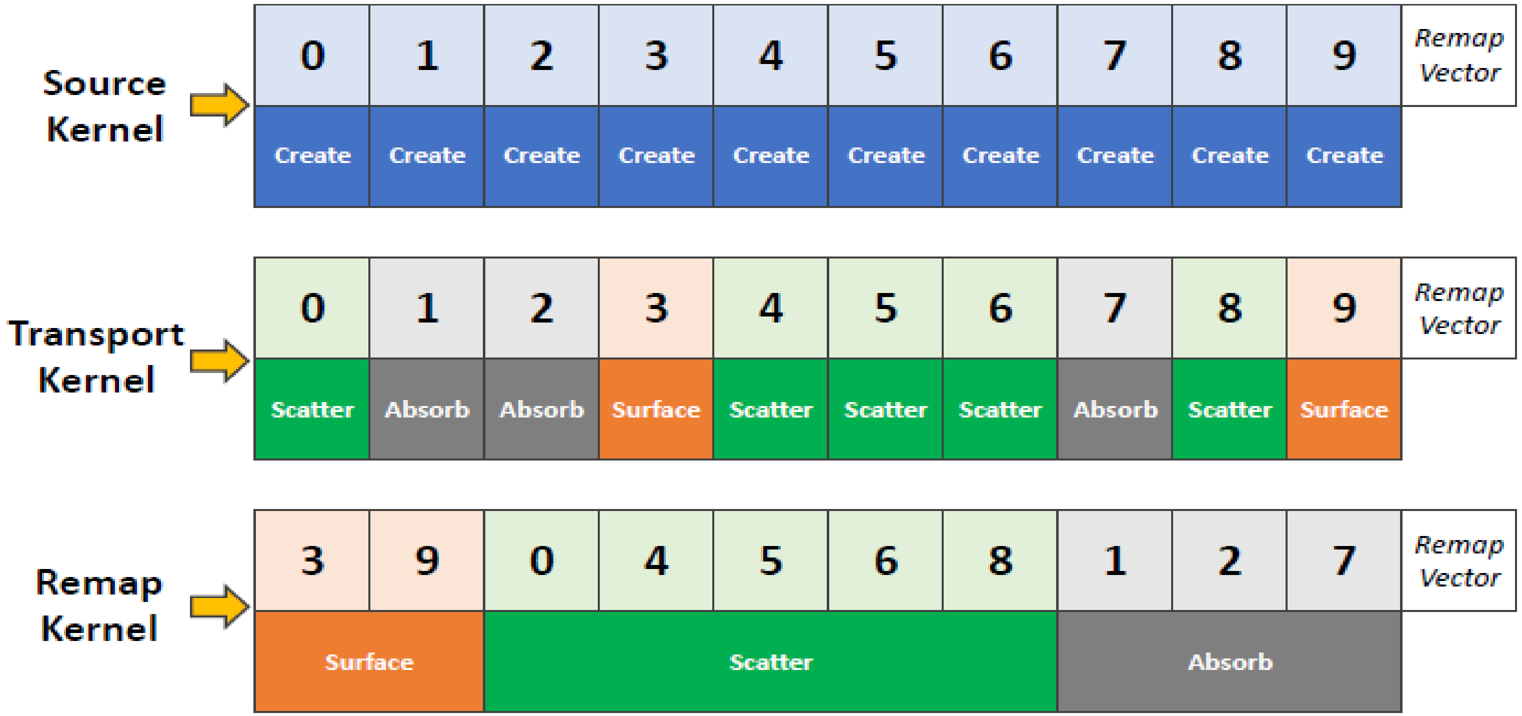


# Impact of Particle Storage Model



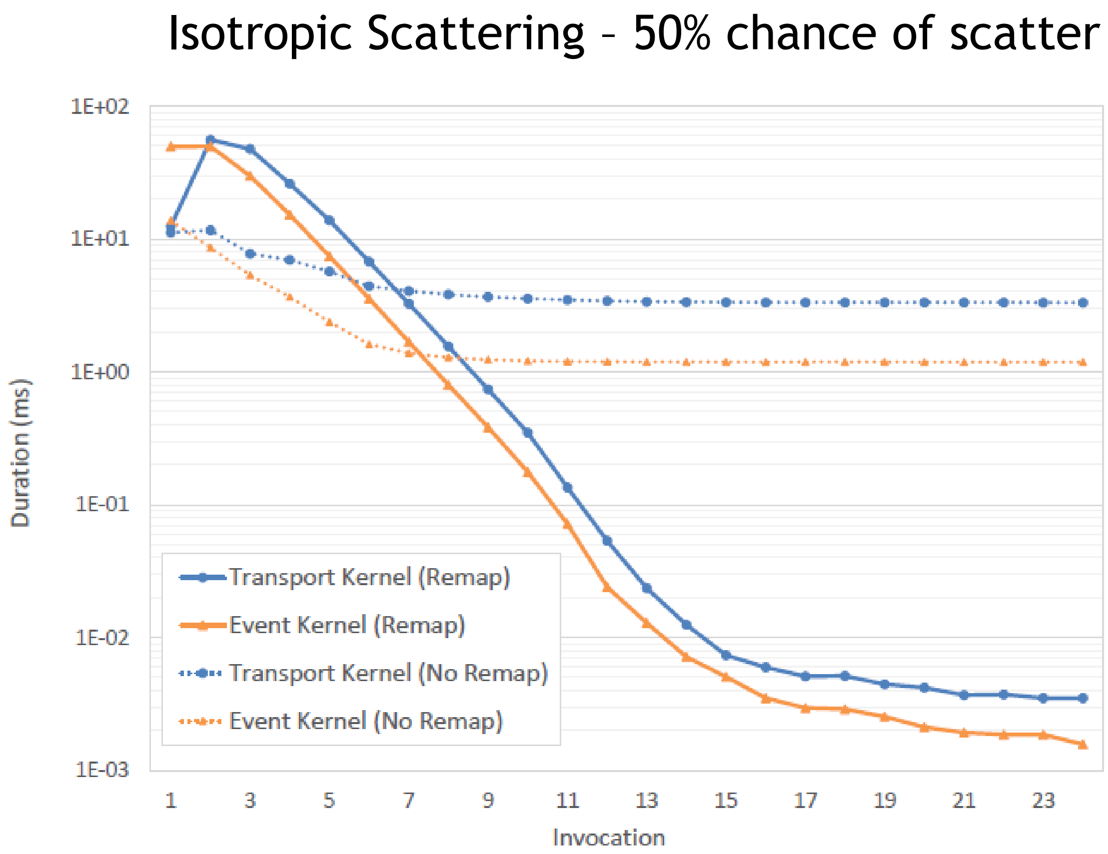
# Impact of Particle Remapping

- Particle remapping sorts events that particles will experience next into groups
- Avoids moving all the particle data with every step in the random walk



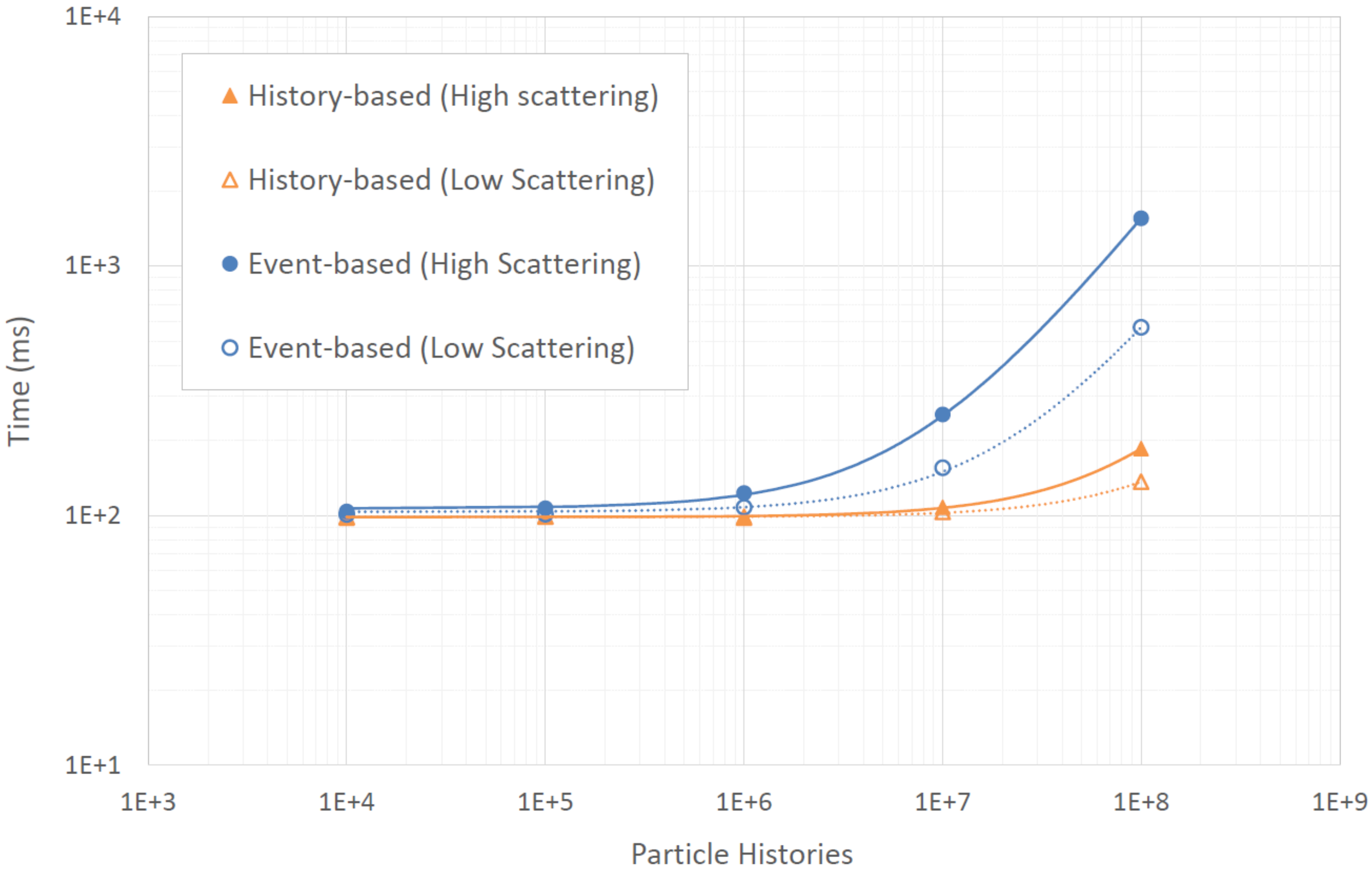
# Impact of Particle Remapping

- Greater cost initially due to increased random memory access pattern
- Remapping gets significantly more efficient as more particles are terminated
- Cannot ignore terminated particles without remapping as they are scattered throughout the particle bank



# Cost of Transport Algorithm Per Particle

- Event-based algorithm costs increasingly more per particle
- Adding more particles means
  - Need more global memory to store particle data
  - Increases number of global memory access transactions
  - Takes longer to remap particles



# Summary

- CHEETAH-MC is a new Monte Carlo transport code that is intended to be used for application work in place of ITS on next-generation architectures
- Current capabilities include
  - Woodcock or surface tracking in voxel or CG geometry (RPP, RCC, SPH primitives only)
  - Coherent scattering or photon attenuation
  - Mono-directional and mono-energetic point, line, surface, or volume source
  - Particle count tallies
- Planning to extend physics options to include full coupled electron-photon interactions over the next couple years
  - Will consider event-based transport algorithm to limit divergence