High Energy Physics
Center for Computational Excellence
Detector Simulation

Tom Evans
for Celeritas, HEP-CCE

DOE HEP CCE All Hands
Dec 18 2023

HEP-CCE Phase 2 – Detector Simulation

- Platform portable geometric tracking
- Optical photons

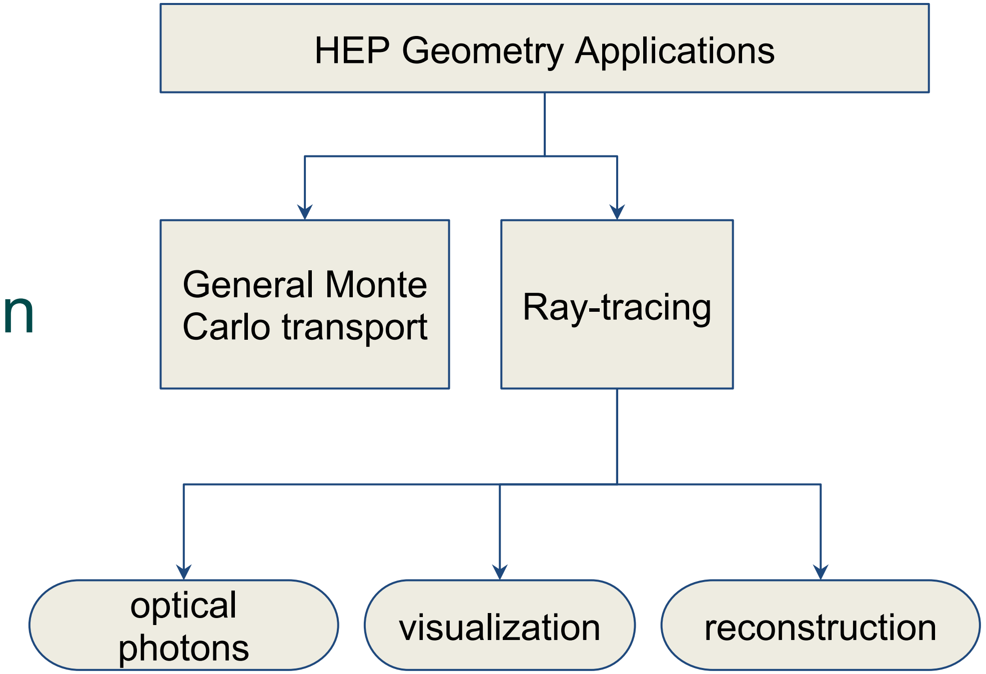
Specific objectives and deliverables discussed this afternoon

Geometric capabilities - current status

- Monte Carlo general particle simulations use **VecGeom** as the standard package
  - Adapted from CPU vectorization prototype
  - Extended to work for Nvidia CUDA
  - Supports Geant4 model construction and tracking
- Optical photons use several approaches
  - Native Geant4 navigation in full event-stepping loop
  - Opticks HEP library (uses Nvidia OptiX™ ray-tracing engine)
- Drawbacks:
  - No platform portability - existing libraries are Nvidia-centric
  - Distinct code bases reduce opportunities to amortize common capabilities
- Objective:
  - Support platform portability across DOE LCFs
  - Unified geometry components that can support multiple applications

Geometric modeling in HEP - application objectives

- General Monte Carlo navigation
    - Robust boundary crossings (double precision)
    - Error checking
    - Graph-based search and subcell optimations
- Ray-tracing for optical photons and visualization
    - Graph-based multi-surface tracking
    - Track batching
    - Variable-precision
- Commonalities:
    - Portability: (Intel, AMD, Nvidia GPUs, multicore)
    - Physics-based simulations require high-precision tracking
    - Bounding volume hierarchy (BVH) graph-based optimizations
    - Many low-level components can be shared
        - High-level APIs will be different depending on application



Geometry packages must support existing HEP model formats and descriptions

Geometry - benefits to HEP community

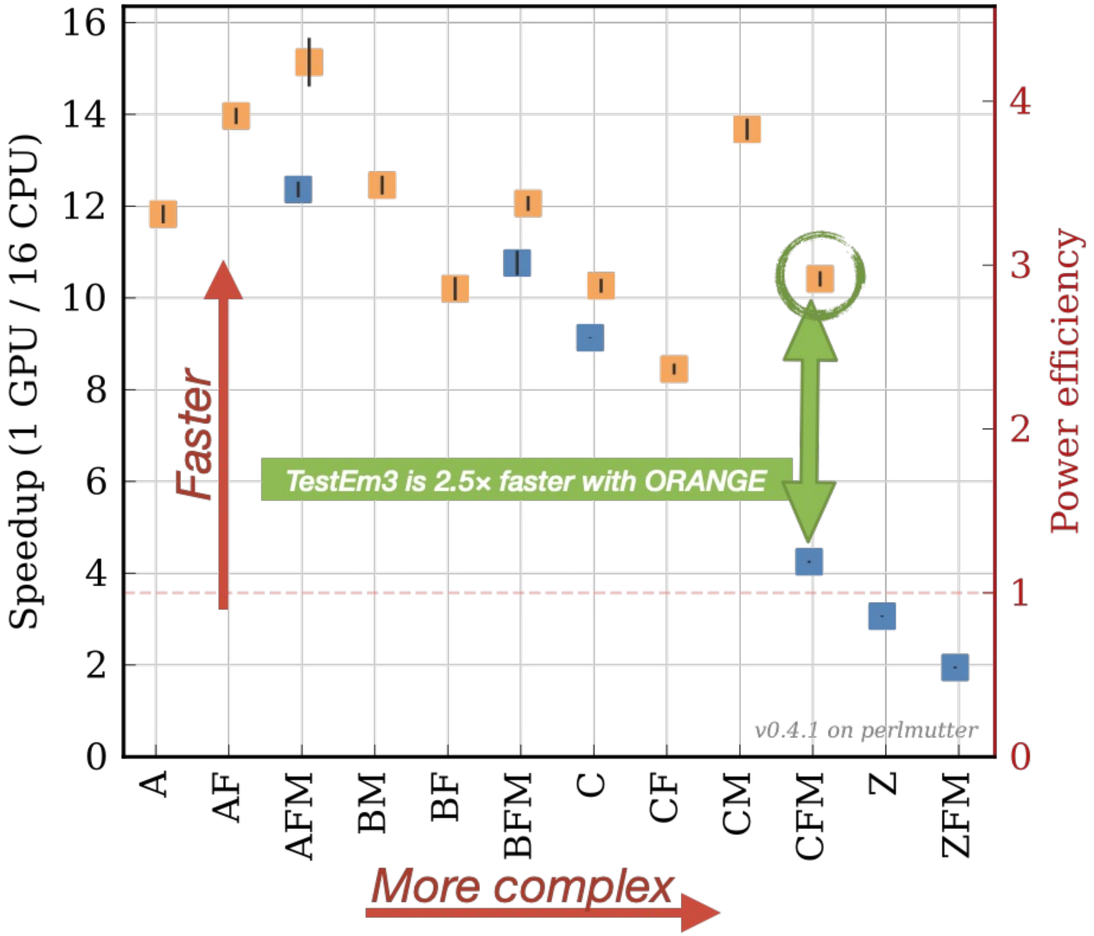
- Provide geometric capabilities for spectrum of applications
  - Monte Carlo detector simulation
  - Optical photon ray-tracing
  - Reconstruction tracking
  - Visualization
- Platform portability
  - Support current and future architectures in DOE LCFs
- Community sustainability
  - Interdisciplinary applications beyond HEP
- ORANGE geometry engine
  - New engine developed in Celeritas
  - Tracking based on CSG tree of surfaces **constructed from volumes**
  - Collaboration to implement surface-based tracking in **VecGeom**

Problem definition

A	"infinite" medium
B	simple-cms
C	idealized calorimeter
Z	cms2018

Modifier

F	+field
M	+msc



Optical photon transport - current status

- Critical application spaces:
  - DUNE (art)
  - LZ (BACCARAT)
  - CalVision
- Current approaches:
  - Direct transport using Geant4 generalized navigation engine
    - e.g., BACCARAT spends >95% of CPU time on optical photon transport when using Geant4
  - Full transport replaced with *fast* approximation methods
  - Optimized optical photon transport using Opticks
- Drawbacks:
  - Portability
    - Geant4 capabilities CPU-only
    - Opticks only available on Nvidia architectures
  - Performance for direct simulation

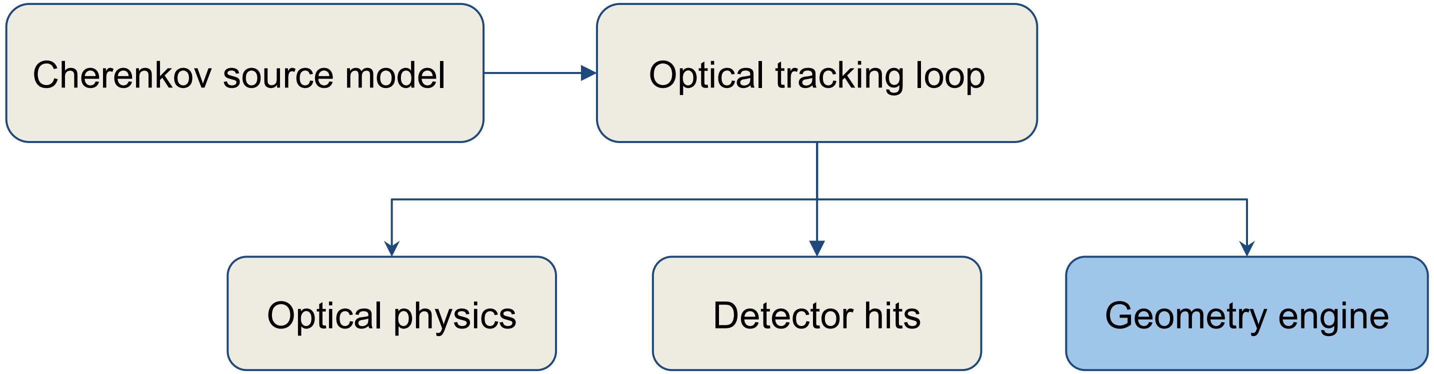
Optical photon transport - application objectives

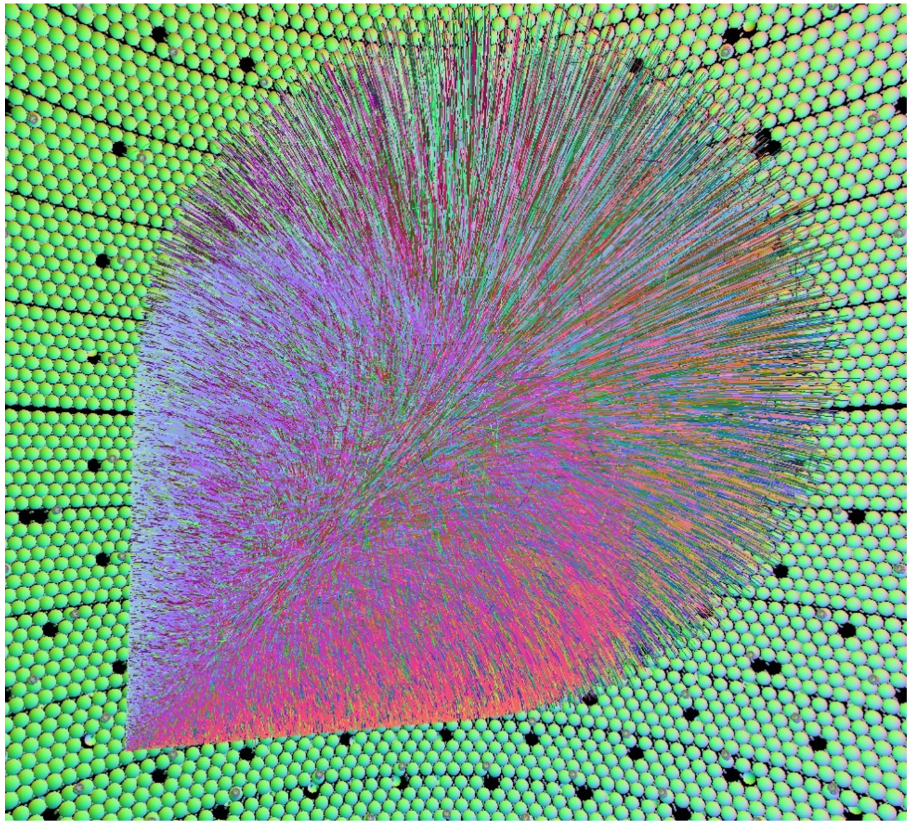
- Fast throughput for direct transport
  - Avoid or accelerate fast approximations (e.g. optical maps in BACCARAT)
- Platform portability
- Ease of integration into existing application frameworks
  - art
  - BACCARAT
  - future luminescent detector applications
- Compatibility with Geant4 geometry models
  - Large detector regions
  - Large detector arrays

Optical photon transport efficiency is closely tied to geometric modeling capabilities

Optical photon transport approaches

- Opticks demonstrated that very high efficiencies can be obtained using GPU ray-tracing
  - >1000x Geant4 direct simulation
- Efficiency gains result from:
  - Specialized optical photon ray-tracing loop
  - Hardware acceleration (BVH traversal)





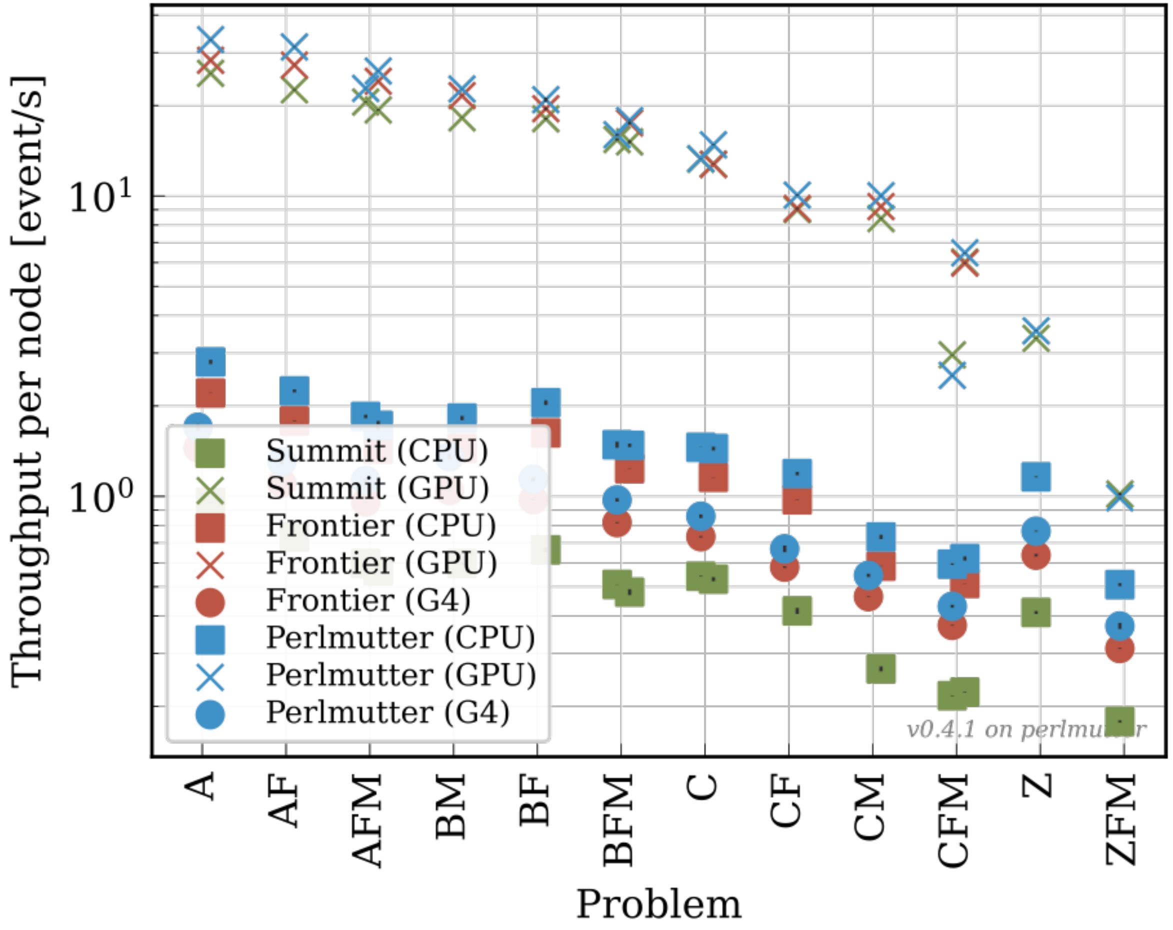
Optical photon shower (JUNO neutrino detector); S.C. Blyth, *Opticks: GPU Optical Photon Simulation for Particle Physics with Nvidia OptiX™*, CHEP 2018.

Detector simulation throughput

- GPUs *cannot* be ignored if present
- AI/ML "revolution" guarantees more coprocessors at all scales

Per-node stats for DOE supercomputers

Machine	Arch	Card	TDP (W)	Cores*	Cards
Summit	CPU	IBM Power9	190	‡22	2
	GPU	Nvidia V100	250	80	6
Perlmutter	CPU	AMD EPYC 7763	280	64	1
	GPU	Nvidia A100	250	108	4
Frontier	CPU	AMD EPYC 7453	225	‡64	1
	GPU	AMD MI250x	500	220	†4



EM only, no SDs

**or SMs;*
†Each card has 2 GPUs
‡One core reserved per GPU