ORNL is managed by UT–Battelle, LLC for the US Department of Energy

Unbounded surface models and ORANGE

Seth R Johnson

HPC Methods for Nuclear Applications

Celeritas core team:

Philippe Canal *(FNAL)*, Julien Esseiva *(LBNL)*, Tom Evans *(ORNL)*, Seth R Johnson *(ORNL)*, Soon Yung Jun *(FNAL)*, Guilherme Lima *(FNAL)*, Amanda Lund *(ANL)*, Paul Romano *(ANL)*, Stefano Tognini *(ORNL)*

SWIFT-HEP Collaboration Meeting
30 March, 2023

Nuclear engineering and surface-based geometry

- Many nuclear engineering codes use "unbounded" surfaces and constructive solid geometry
  - MCNP, KENO, earlier codes: >40 years of history
  - Quadric surface cards, CSG cell cards
  - Neutral particles or no magnetic fields
- 2017: Shift GPU code (part of ECP) uses simple but efficient surface-based reactor model (nested cylinders and arrays)
- 2020: New CPU geometry for Shift
- 2021: Initial GPU port for Celeritas

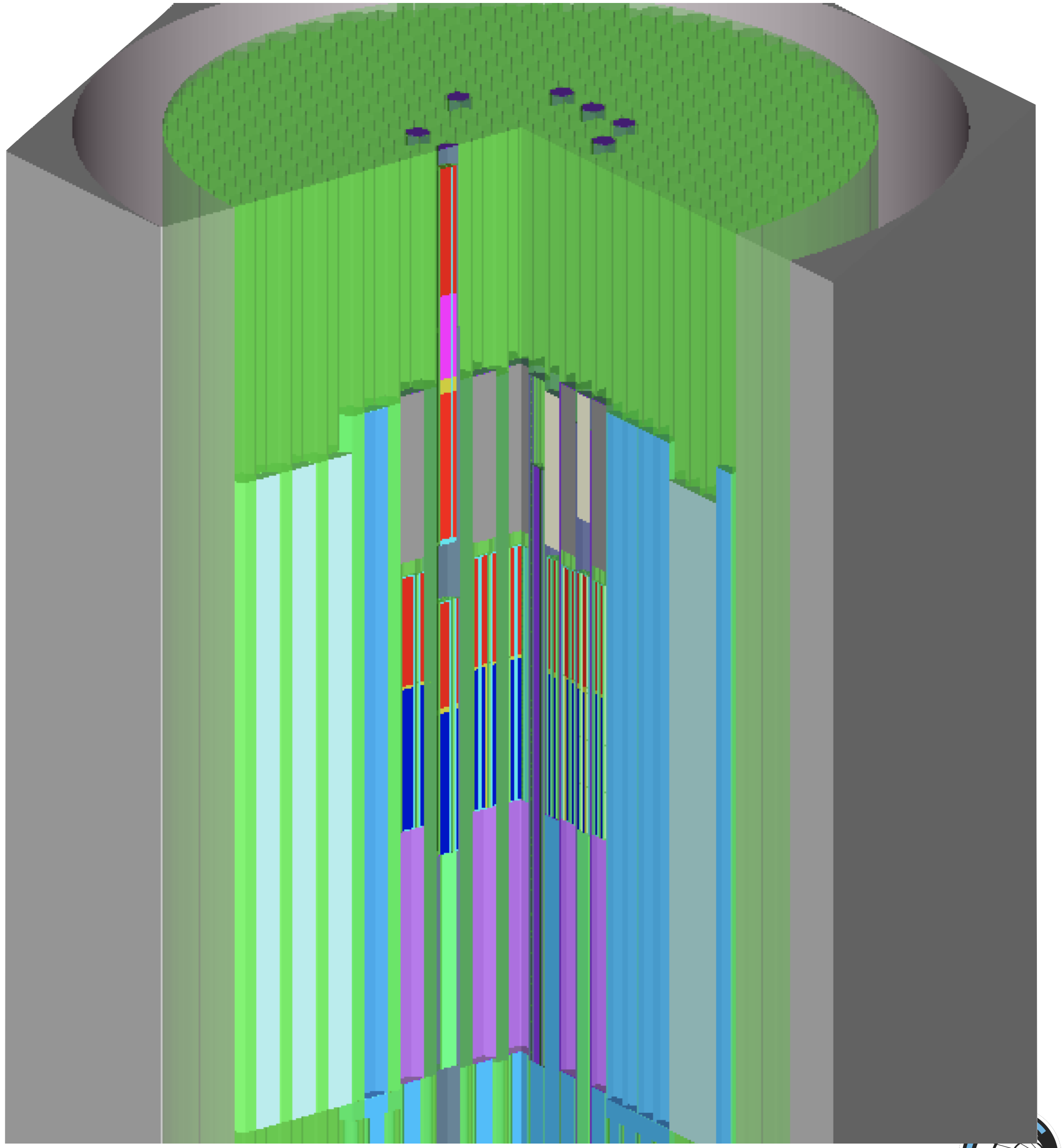


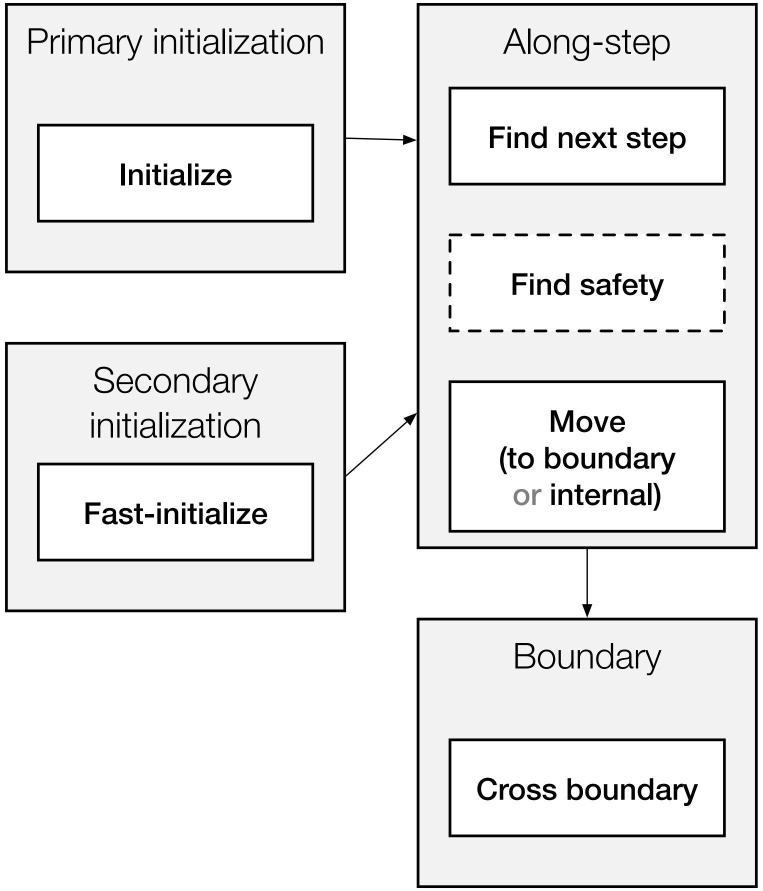
Image credit: Steve Skutnik (ORNL)

ORANGE *Oak Ridge Advanced Nested Geometry Engine*

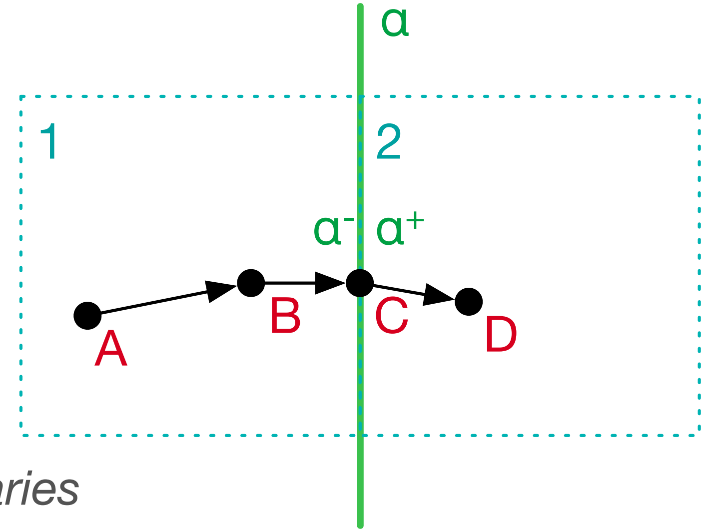
- Unbounded surface-based implementation with multiple input options
  - Designed for deeply nested reactor models
  - Tracking based on CSG tree of surfaces comprising volumes
  - Surfaces and CSG tree are constructed at setup time from user-provided solids
- Current development in Celeritas
  - Fills the need for a portable (CUDA/HIP) geometry implementation for testing in Celeritas
  - Currently being extended to full GPU capability for reactor modeling *(Elliott Biondo, ORNL)*
  - Not a primary target for detector applications due to resource constraints
  - Good playground for research in accelerating surface intersection on GPU

Surface-based tracking in Celeritas

Celeritas geometry interface

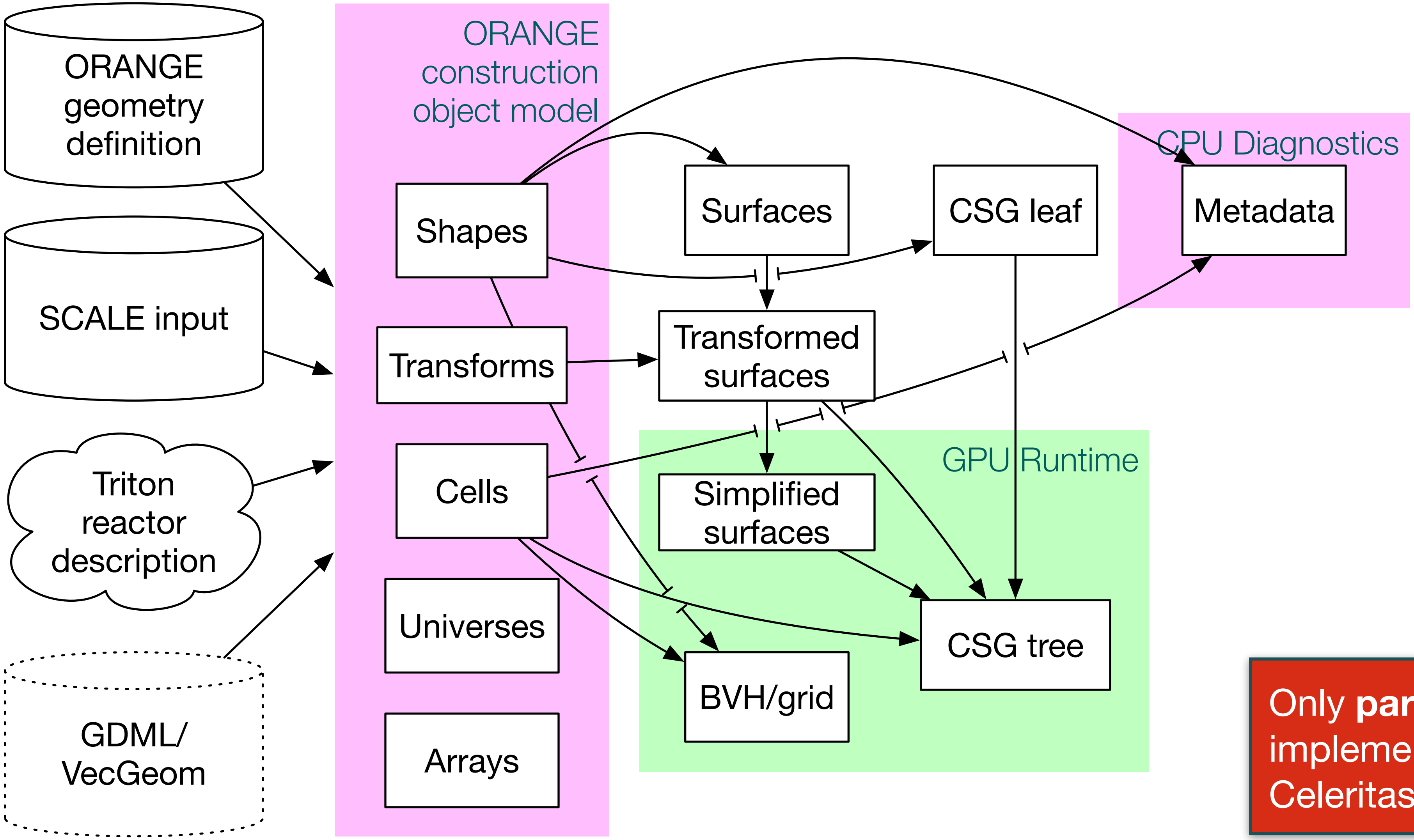


	Position	Volume	Surface+Sense
(input)	A	—	
Initialize	A	1	—
Find step	A	1	—
Move internal	B	1	—
Move to bndy	C	1	α inside ★
Cross bndy	C	2	α outside ★
Move internal	D	2	—



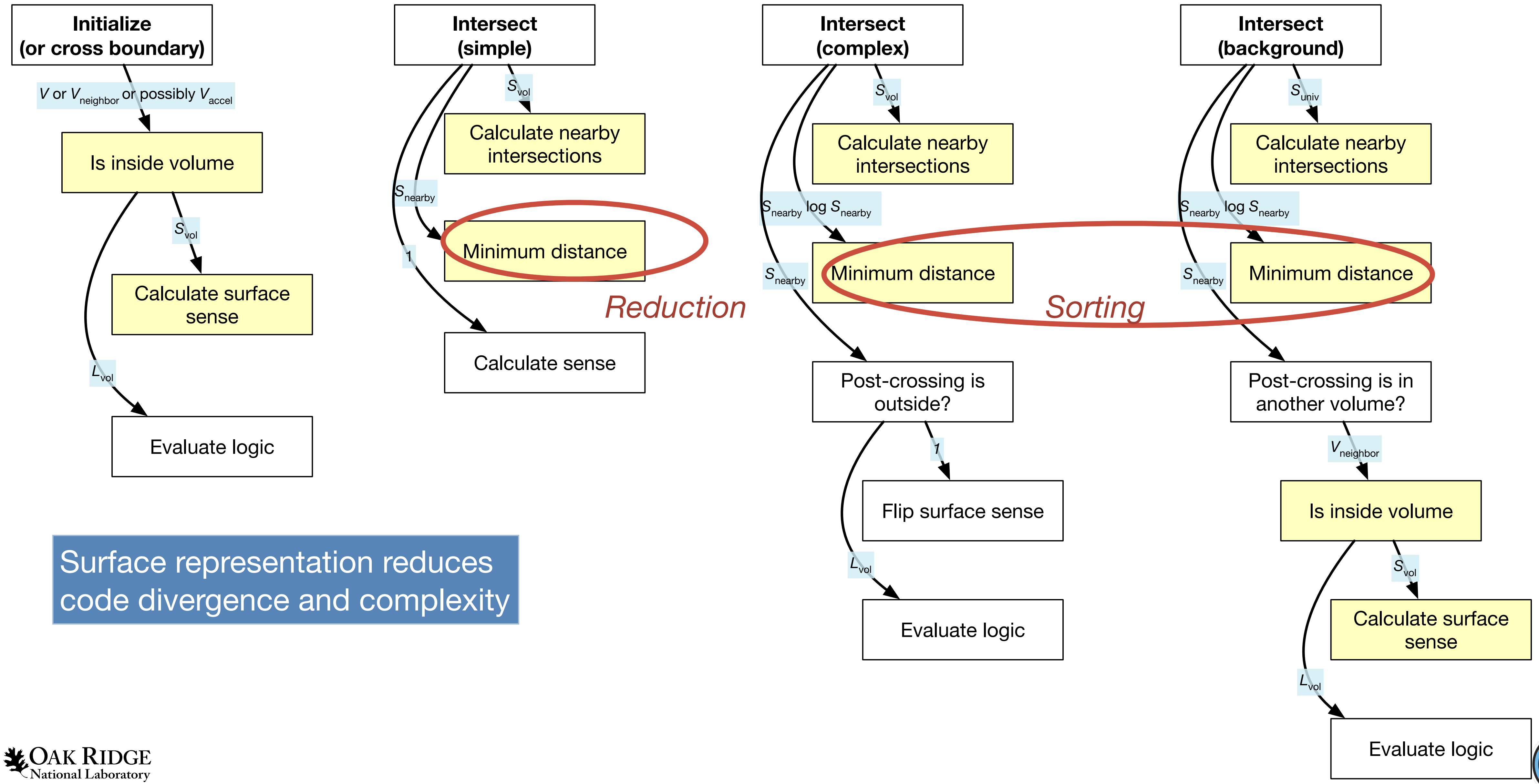
★ *exact handling of direction changes on boundaries*

ORANGE surface/volume construction



Only **partially** implemented in Celeritas ORANGE

Key tracking algorithms



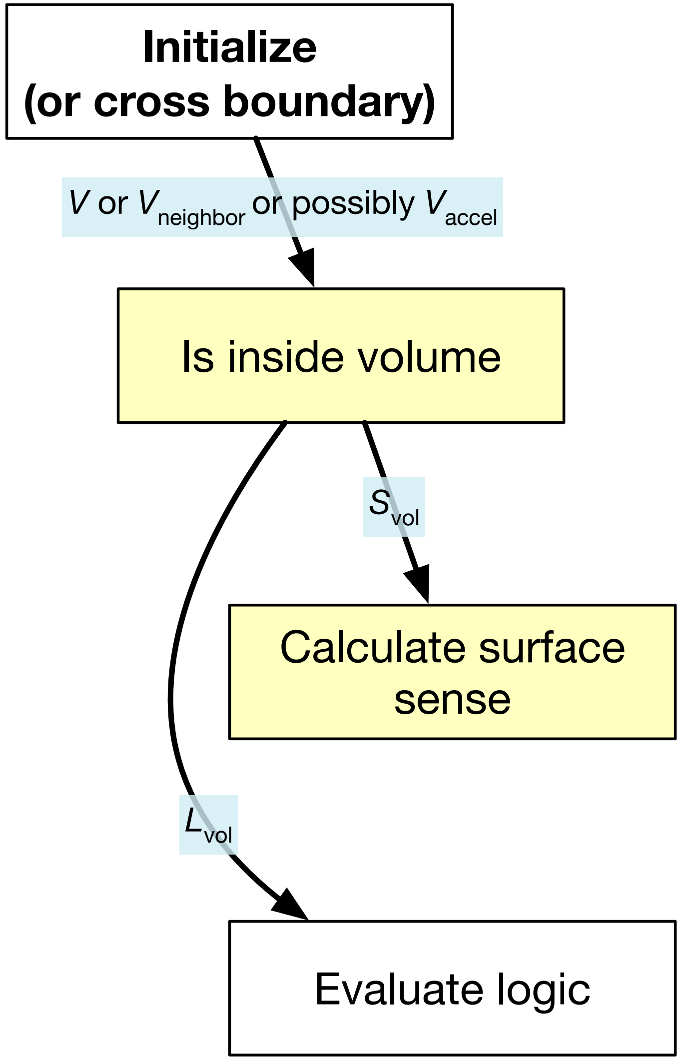
Surface representation reduces code divergence and complexity

Shortcomings of current implementation

- Reactor-targeted geometry means tracking in multiple levels simultaneously
  - Higher memory usage per track, not ideal for GPU
  - Detector geometry implementation could reuse most of ORANGE components and track on only one or two levels
- “Safety distance” is limited to simple surfaces in convex volumes
  - Plenty of research in this area: “maximally-inscribed ball” etc.
  - Precomputed “safety” data can use original information about the solids!
- Background intersection algorithm is optimized for sparse daughters

Opportunities for inter-thread cooperation

- Current Celeritas/ORANGE model is CPU-like
- Single thread corresponds to single track
- No inter-thread cooperation for geometry
- ExaSMR code has shown performance improvements by launching more threads with several threads contributing to a single track
- ORANGE can do the same for key parallelizable algorithms that are now "serial" loops



Locating the volume corresponding to a point