GeantV - Next generation simulation prototype

Sofia Vallecorsa[1] for the GeantV Project

[1]CERN, Sofia.Vallecorsa@cern.ch

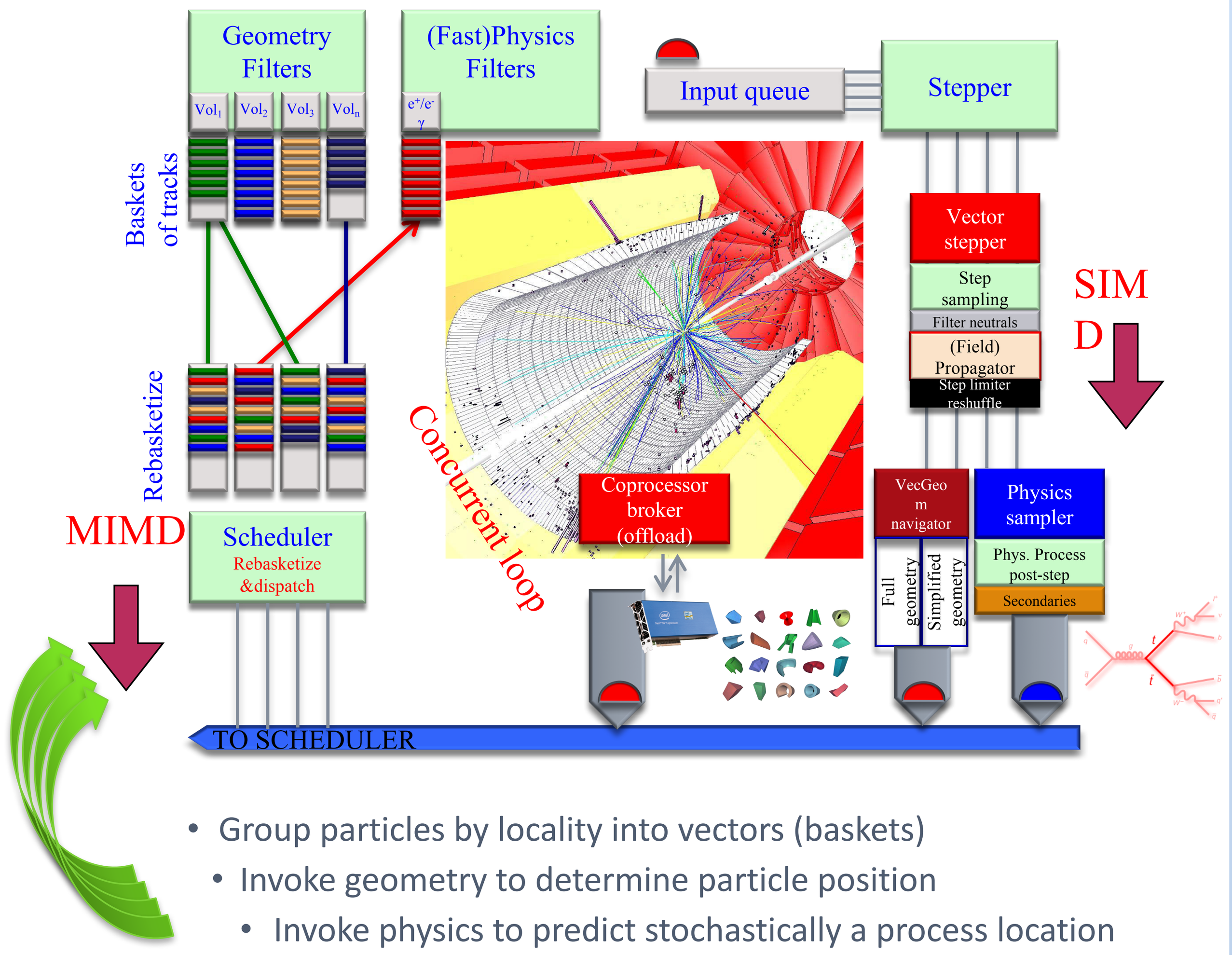
The project

Introduction and Motivation

Simulation of particle transport through matter is fundamental for understanding the physics of **High Energy Physics** (HEP) experiments, as the ones at the Large Hadron Collider (**LHC**) at CERN. Such experiments have dedicated so far most of their worldwide distributed CPU budget – in the range of half a million CPU-years equivalent – to simulation. In particular, the most computing-intensive components are **geometry** modeling, **navigation** through millions of objects and **physics** models.

Parallel particle transport

The **scope** of the project is the development of a community supported, open-source, next generation particle transport code for HEP integrating both detailed and fast simulation algorithms, optimized for the emerging **parallel and vector architectures**. **Beta release** by end of 2018.

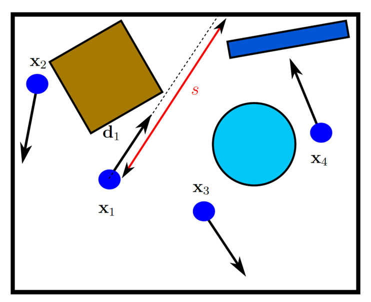


- Group particles by locality into vectors (baskets)
  - Invoke geometry to determine particle position
    - Invoke physics to predict stochastically a process location
    - Validate proposed physics step against geometry
      - Propagate and regroup baskets

Geometry redesign for vectorization

VecGeom is a geometry modeler evolved from legacy geometry libraries (Geant4, Usolids, ROOT). It introduces a **many-particle API** besides the standard scalar one, and relies on **templated backend abstraction** to enable both platform/architecture specific optimizations and vector/scalar API polymorphism.

Vec(torized)Geom(etry) = Evolved Usolids
+ many-particle API
+ geometry mode/navigation

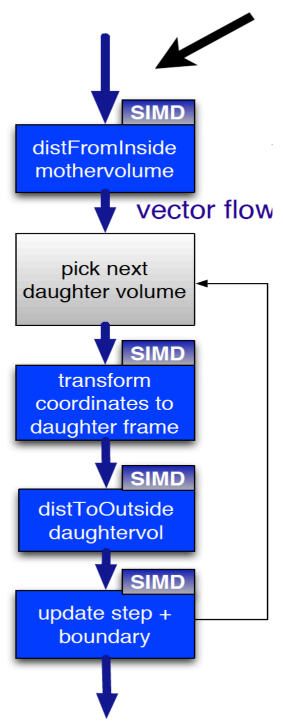


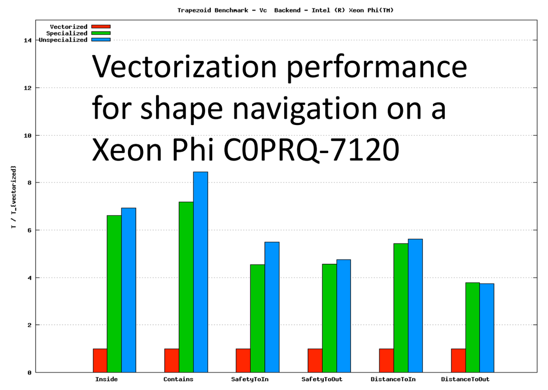
GeantV is a collaboration among several research institutes. It is also partially funded by Intel Parallel Computing Center program (geant.cern.ch)

Achievements

Geometry Performance

VecGeom can run chains of algorithms in

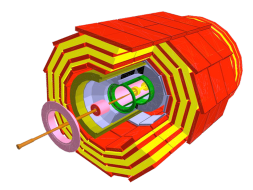




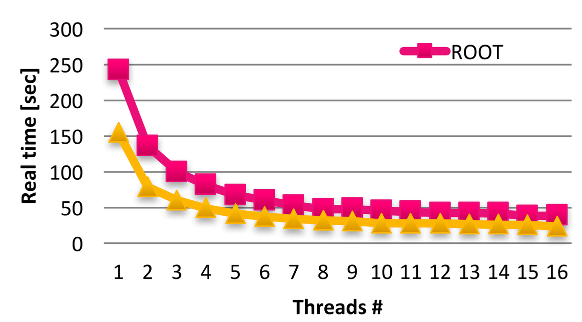
Overall performance for a toy detector (4 boxes, 3 tubes, 2 cones) vs ROOT/5.34.17 (arxiv/1312.0816)

	16 particles	1024 particles	SIMD max
Intel Ivy-Bridge (AVX)	~2.8x	~4x	4x
Intel Haswell (AVX2)	~3x	~5x	4x
Intel Xeon Phi (AVX-512)	~4.1	~4.8	8x

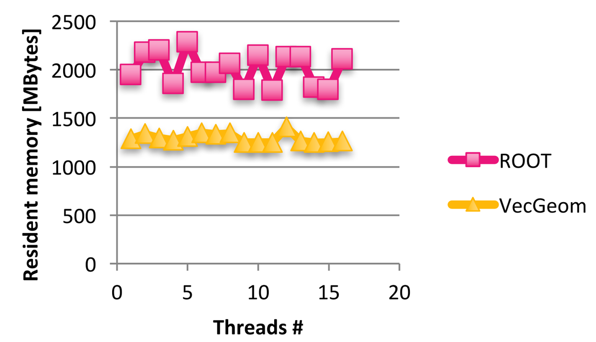
Scalar navigation performance in **full CMS** (Compact Muon Solenoid, one of the major LHC experiments) geometry. Left, real time for the simulation of 10 pp events at 7TeV using VecGeom instead of legacy existing geometry. Right, the resident memory of the full application. (Platform: Intel Haswell)



Real time VecGeom versus ROOT geometry



Resident memory VecGeom versus ROOT



Intel Xeon Phi for detector simulations

The X-Ray benchmark tests geometry navigation in a **real detector geometry**: a detector volume is scanned with virtual rays along a given direction. Each ray is propagated from boundary to boundary and the number of crossings is counted.

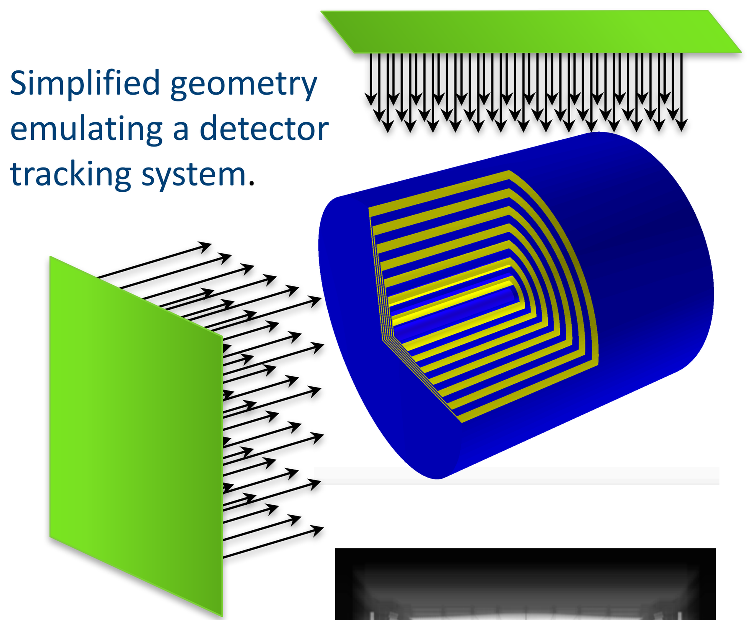
Scalar case: Simple loop over pixels, generating a ray

Ideal vectorization case: Fill vectors with N times the same X-Ray, using this as reference for the maximum achievable vectorization

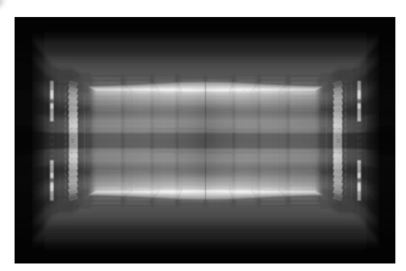
Realistic (basket) case: Fill baskets per geometry volume as particles are entering (as in GeantV)

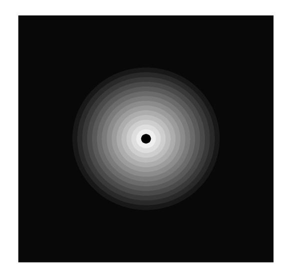
X2 gain from vectorization when filling all vector pipelines for AVX512 wrt AVX2

Simplified geometry emulating a detector tracking system.

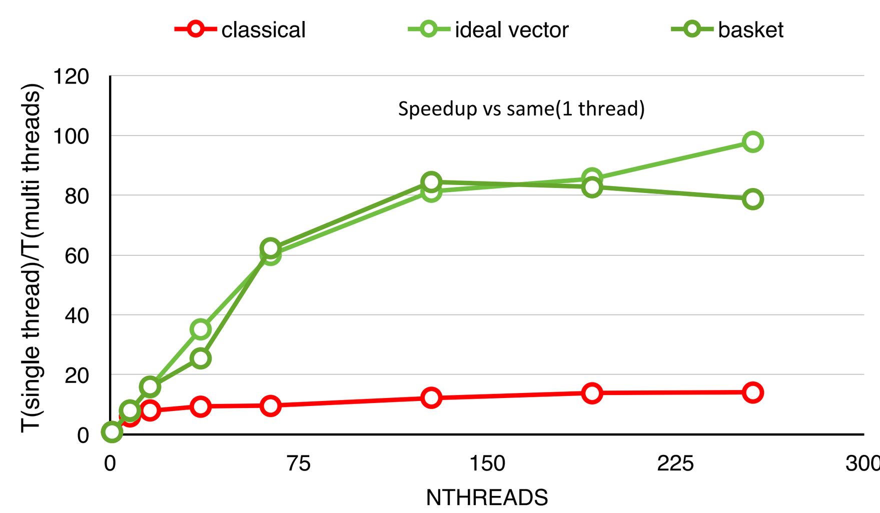


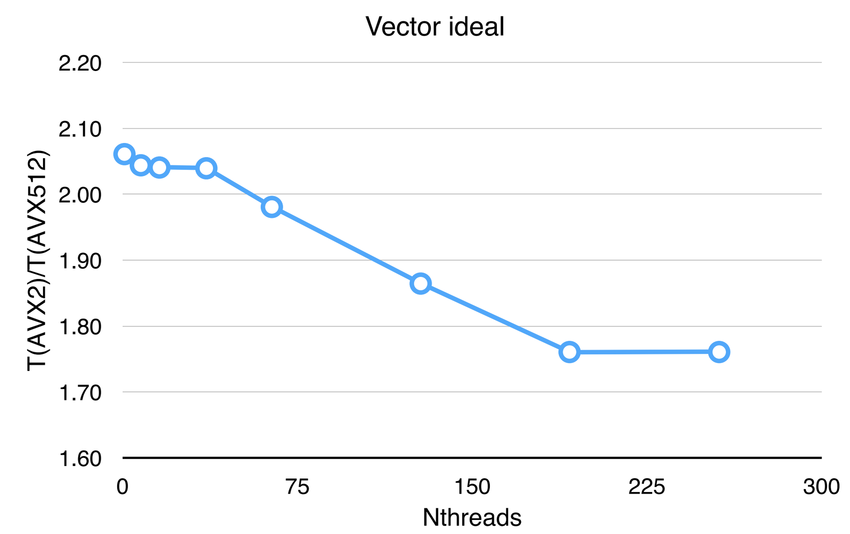
Dispatching one full scan (image) per task





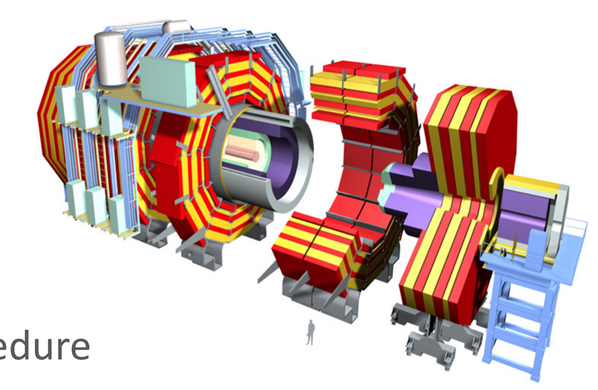
A pixel is produced for each ray having a grey value proportional to the number of crossings.

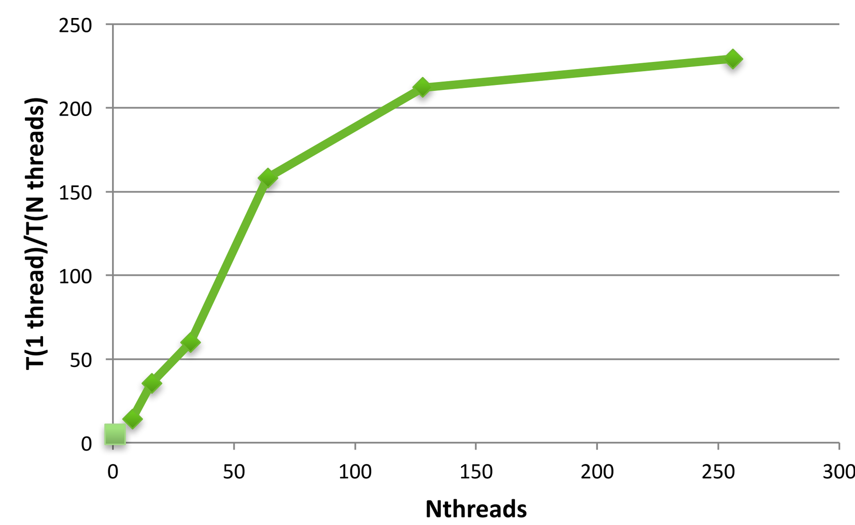




The full prototype

- Exercise at the scale of LHC experiments
- **Full geometry** + uniform magnetic field
- **Tabulated physics**, 1MeV energy threshold
- **Full track transport** and basketization procedure
- **Compare scalability to classical approach single-thread**

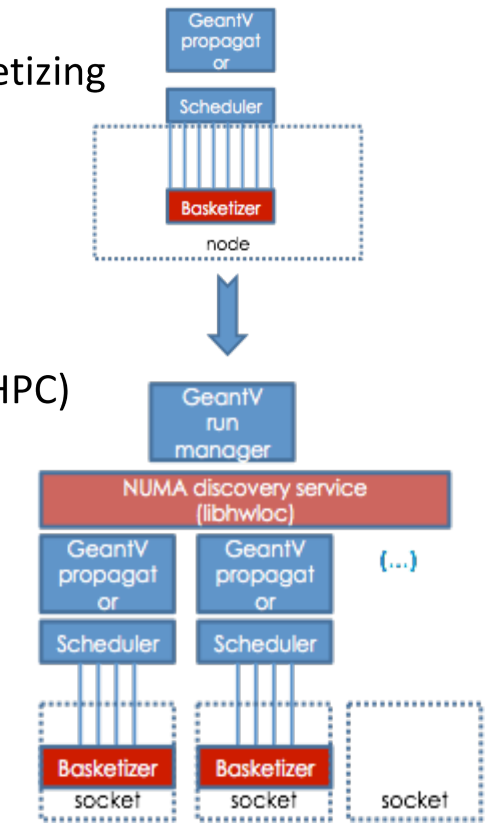


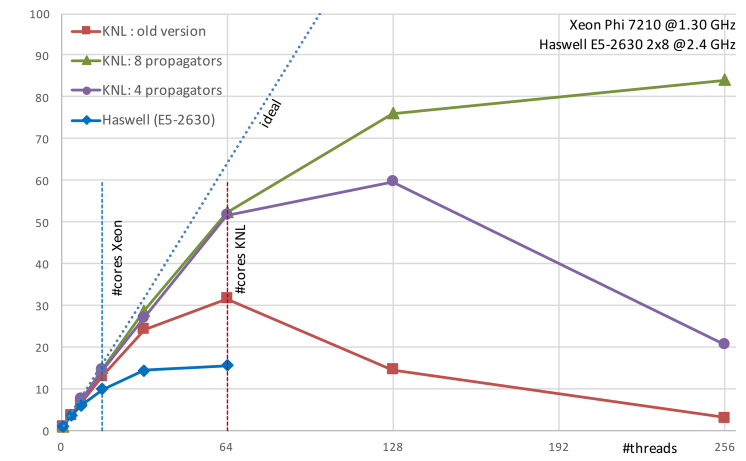


Development

Sub-node clustering

- Known scalability issues due to re-basketizing synchronization
- Deploy several propagators to cluster resources at sub-node level
- **Improve scalability**
- Address many-node and multi-socket (HPC) modes + non-homogenous resources



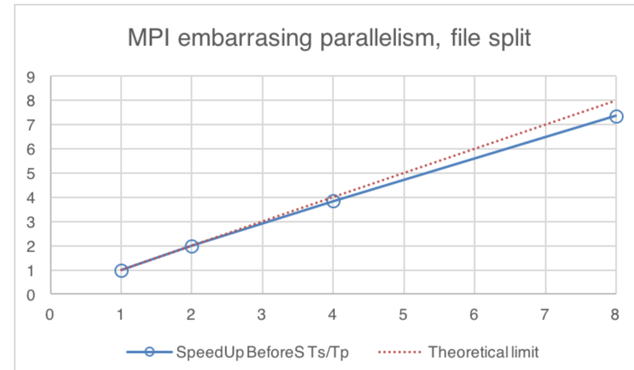


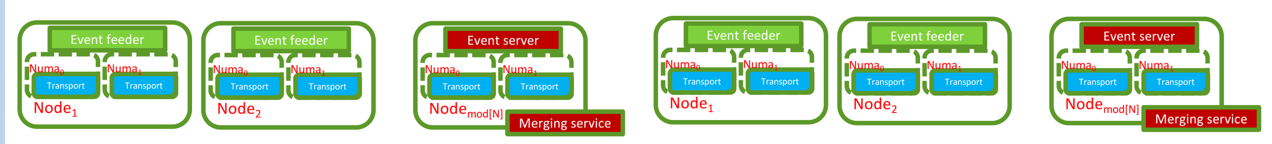
First tests on KNL

GeantV for HPC environment

Standard mode (1 process per node)
- Need work balancing
- Check output granularity (merging may be required)

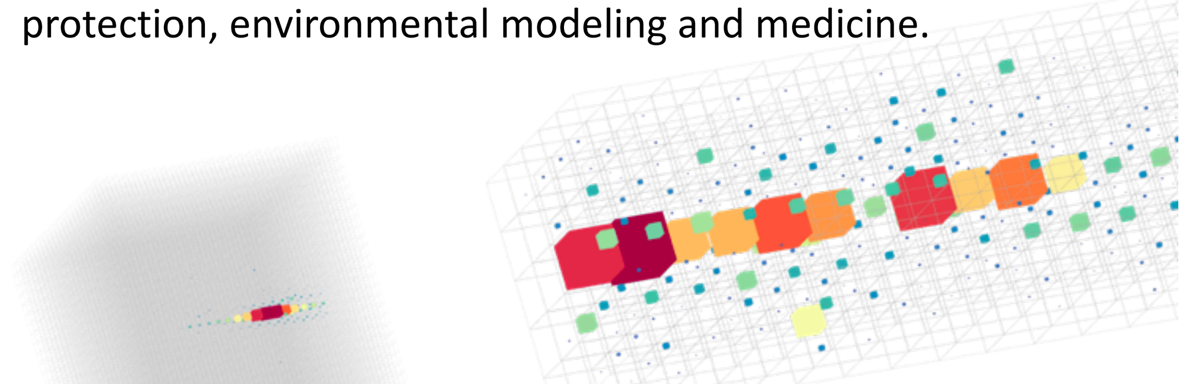
Multi-tier mode (event servers)
- Gets events from file, handles merging and workload balancing
- Communication with event servers via MPI





Machine Learning for fast simulation

A faster approach is to treat traditional simulation as a black-box and replace it by a **deep learning** algorithm trained on different particle quantities. We are testing several techniques such as generative adversarial networks (GANs) to replace the Monte Carlo approach. We expect to achieve a **significant speedup** (x25) with respect to GeantV full simulation approach. Development of such tool can further benefit other fields, such as radioactivity protection, environmental modeling and medicine.



Single particle energy deposits in the Linear Collider Detector calorimeters generated by a GAN