Future ICT challenges at CERN
some examples

MARILENA BANDIERAMONTE, FEDERICO CARMINATI, **SOFIA VALLECORSA**

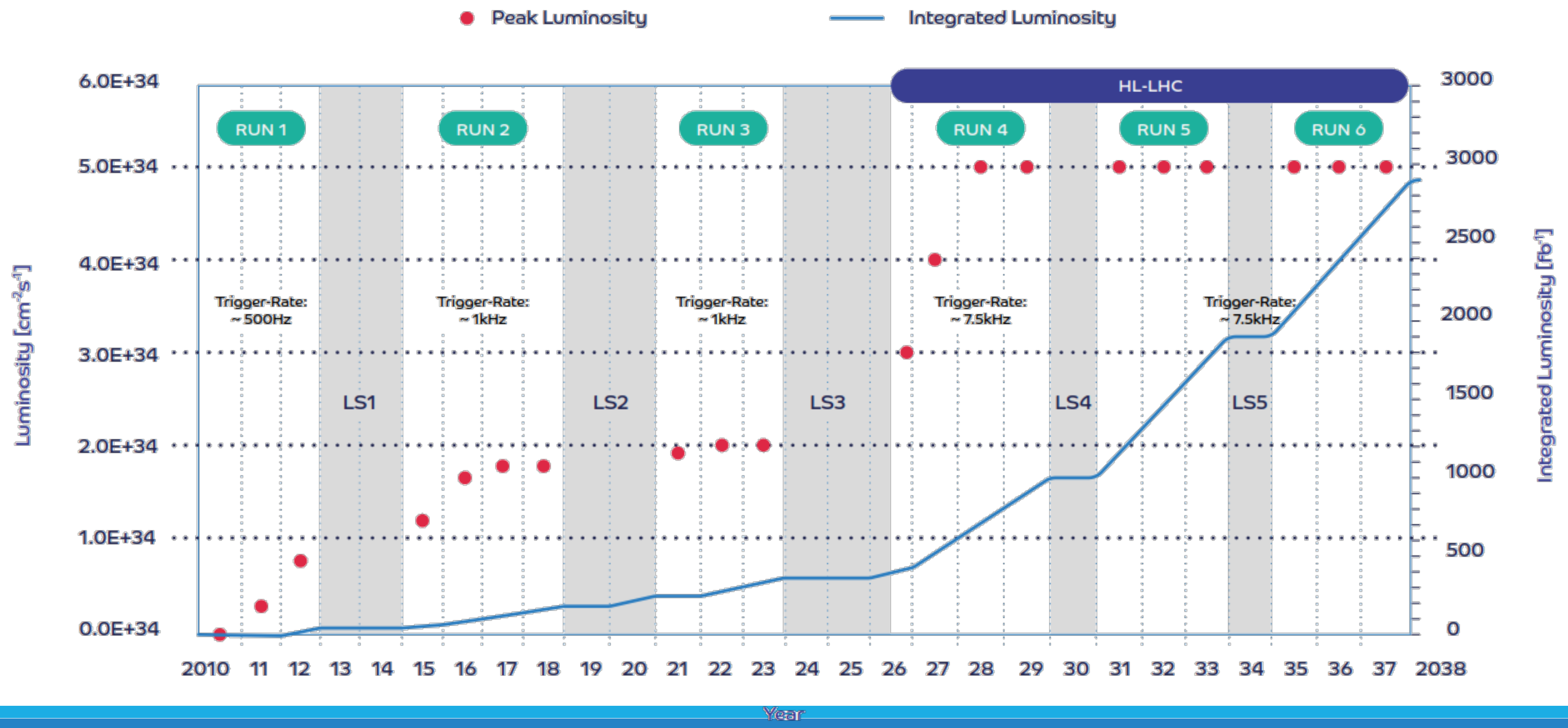
Outline

- Introduction
- CERN openlab and research plan for Phase VI
- A few examples
  - Code modernisation and simulation: GeantV
    (MoU INAF – CERN openlab)
  - Machine Learning
  - Data analysis and Data analytics
- Summary

Introduction

HEP community is spending significant efforts to get the most out of the LHC

Maximise potential for discovery & optimise resource usage

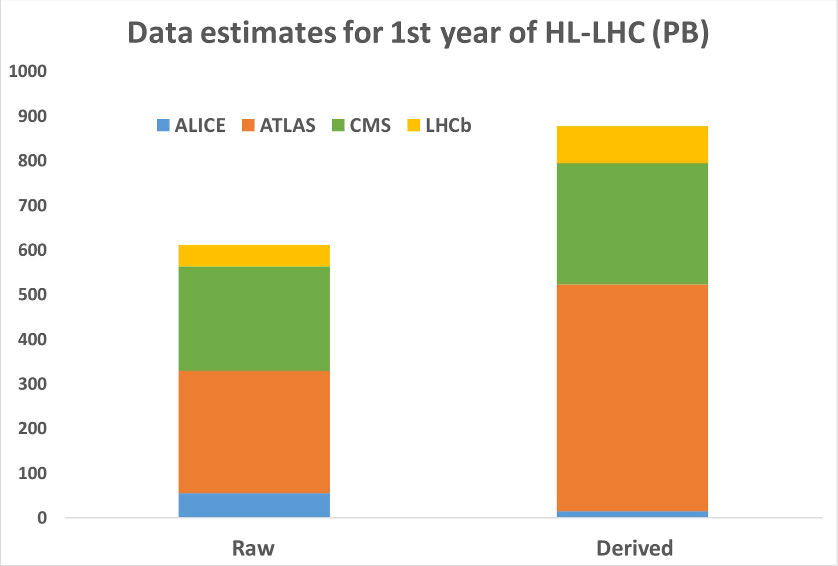


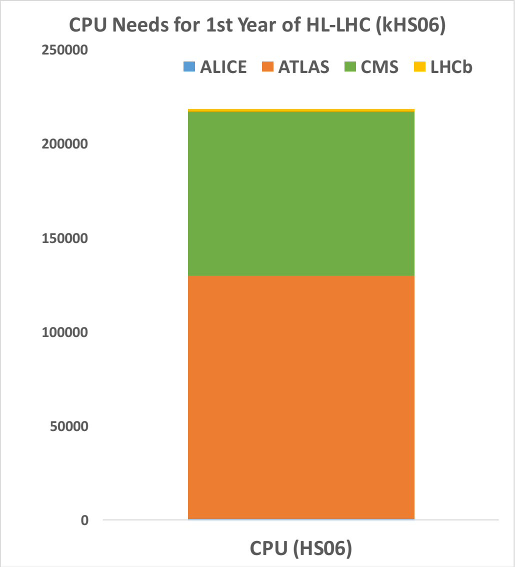
By 2025 HL-LHC data will have increased by between one /two orders of magnitude.

More collisions (x3 higher than today) and more complex events

I. Bird
2016 WLCG workshop

Estimates of resource needs for HL-LHC





Data:

- Raw 2016: 50 PB → 2027: 600 PB
- Derived (1 copy): 2016: 80 PB → 2027: 900 PB

CPU:

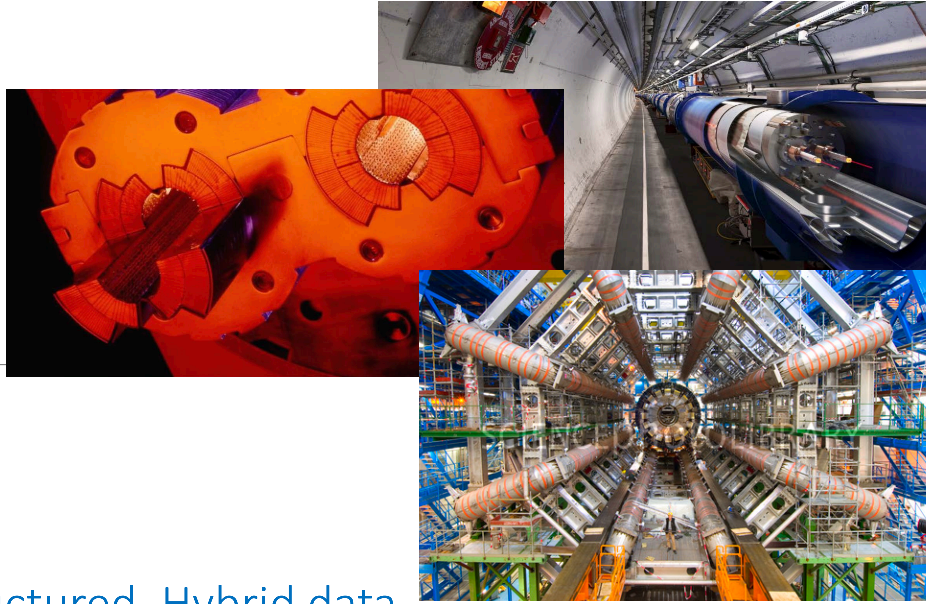
- x60 from 2016

Technology at ~20%/year will bring x6-10 in 10-11 years

Simple model based on today's computing models, with expected HL-LHC operating parameters (pile-up, trigger rates, etc.)

At least x10 above what is realistic to expect from technology with reasonably constant cost

Big Data at the LHC



From:

Accelerators infrastructure (control systems, monitoring)

Experiments (detectors & physics data)

Computing infrastructure:

- Large sets of metrics collected from system components (CPU and batch, disk and archive storage, network topology and flows, and application throughput)

- Multi-structured, Hybrid data
  - Metadata
  - Aggregation of databases
  - Human driven structured data
  - Frontier experiments
- Data model evolving with data
- Variable parametric space
- Data driven induction and deduction

CERN openlab

CERN openlab

A public-private partnership between the research community and industry

- Evaluate state-of-the-art technologies in a challenging environment and improve them
- Test in a research environment today technologies that will be used in many business sectors tomorrow.
- Train the next generation of engineers/researchers.
- Promote education and cultural exchanges.
- Communicate results and reach new audiences.
- Collaborate and exchange ideas to create knowledge and innovation.

PARTNERS	CONTRIBUTORS	ASSOCIATES	RESEARCH
HUAWEI	rackspace	COMTRADE	INFN, Innopolis University
intel	SEAGATE	Yandex	Kazan Federal University, King's College London
ORACLE	BROCADE		Newcastle University, Scimpulse foundation
SIEMENS	IDT		GSI, EMBL-EBI

F. Rademaker, openlab open day

openlab phase VI

CERN openlab

Defined research strategy for 2018-2020 phase VI in [whitepaper]

HL-LHC runs will be challenging environment

- Tools image recognition or machine learning for classification may have a big impact
- Infrastructure needs to evolve to handle the much higher data rates
  - New architectures, co-processors, FPGAs, GPUs are all candidates
- Software performance will be the key
  - Modern coding, parallelization and vectorization, portability

Phase VI main R&D areas:

- DATA-CENTRE TECHNOLOGIES AND INFRASTRUCTURES
- COMPUTING PERFORMANCE AND SOFTWARE
- MACHINE LEARNING AND DATA ANALYTICS
- APPLICATIONS TO OTHER DISCIPLINES:
  - Astrophysics, Medical Sciences

  "Exascale data processing at future astrophysics infrastructures"

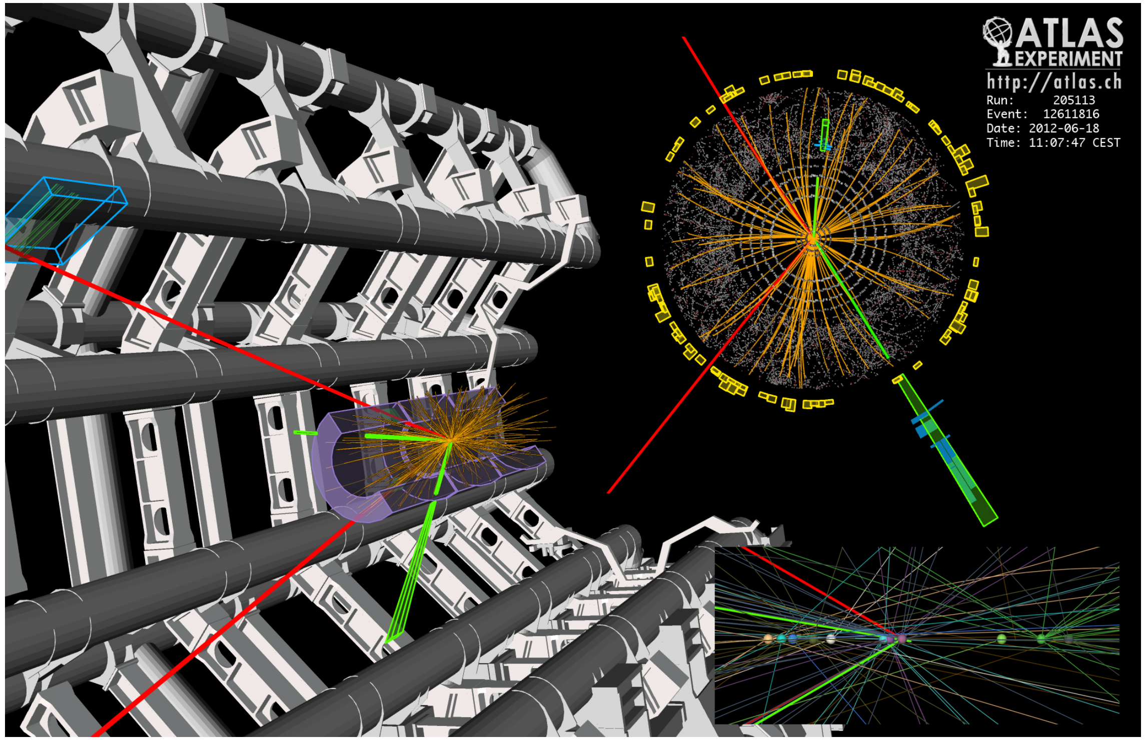
M. Girone, openlab open day

Code modernization: simulation

Motivation

Detailed simulation of particle transport and in detector geometries

- State of the art physics models, propagation in fields in geometries having complexities of millions of parts



LHC uses > 50% of WLCG power for detector simulations
(200 Computing centers in 20 countries: > 600k cores)

GeantV: Better exploiting features of modern architectures

- SIMD and NUMA topology aware
- Multi-threaded, Task-based approach
- Portable across different architectures, GPUs, HPC friendly
- Generic fast simulation integrated with full simulation

Portability

Increasing number in HPC systems use a mix of multi-cores CPUs and special purpose accelerators

Producers move towards hybrid systems (AMD Accelerated Processing Units, Intel DL-100, Arria 100, …)

Accelerators: exceptional raw power wrt simple CPU

- High energy efficiency
- Massively parallel architecture -> Substantial performance challenges for developers

GeantV insures portability via C++ templating, backends and low level optimised libraries

- GPU (offload)
- Intel Xeon Phi (AVX512)
- **IBM Power 8+ and 9 thanks to MoU between INAF and CERN openlab**

www.top500.org/

Parallel Computing Center 2015-2016

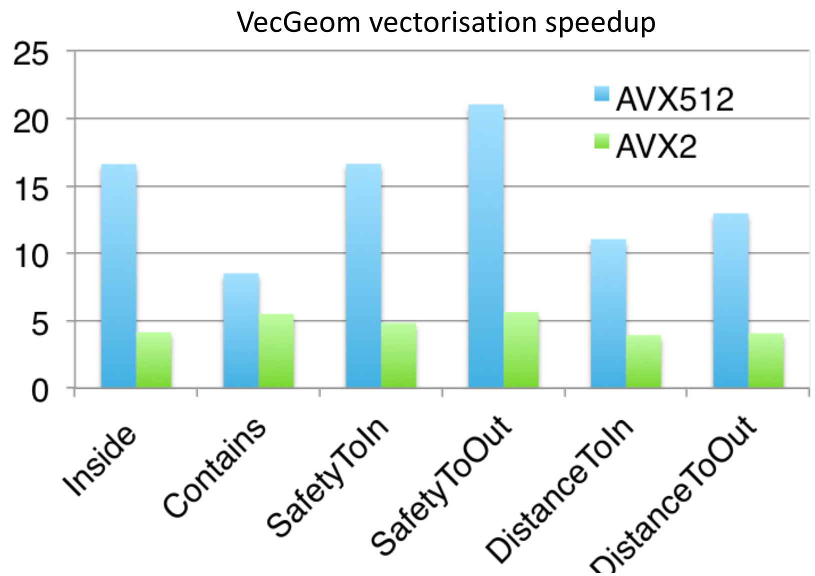
Vectorised performance

Type-based explicit vectorization: geometry and magnetic field RK propagator, working on physics

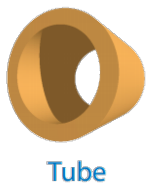
VecGeom: a library of vectorized geometry algorithms to leverage on SIMD architectures

- Substantial performance gains also in scalar mode
- Testing performance on GPU

	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

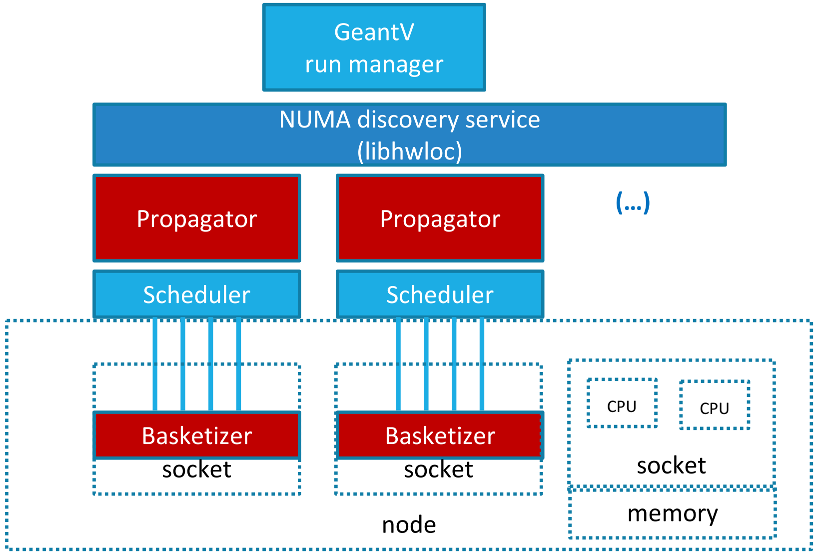


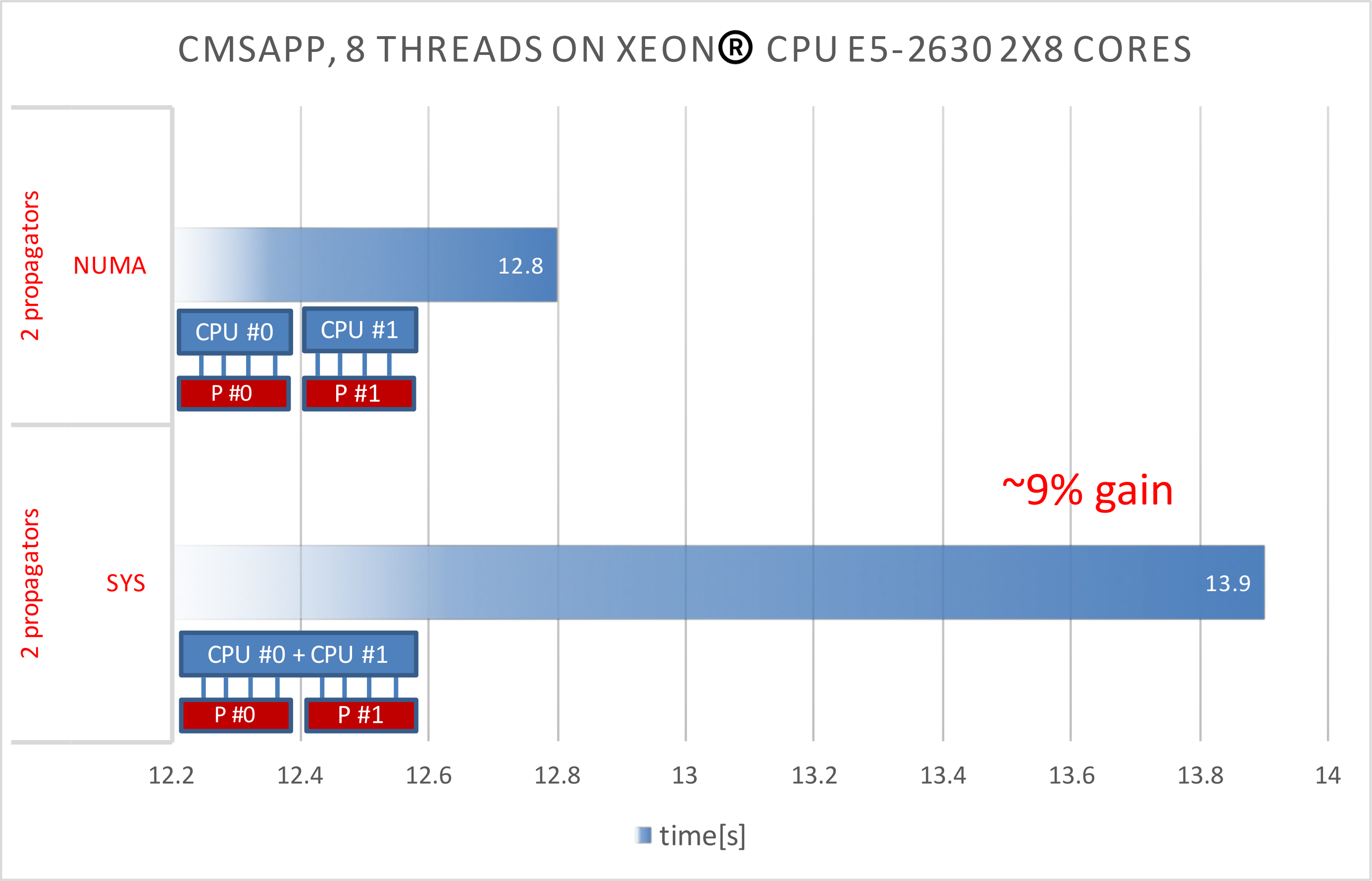
Intel Xeon Phi 7210 @1.30 Hz – 64 cores



S.V., PASC17

Topology awareness





Hardware is topology organized (node –> socket –> CPU –> caches –> memory)

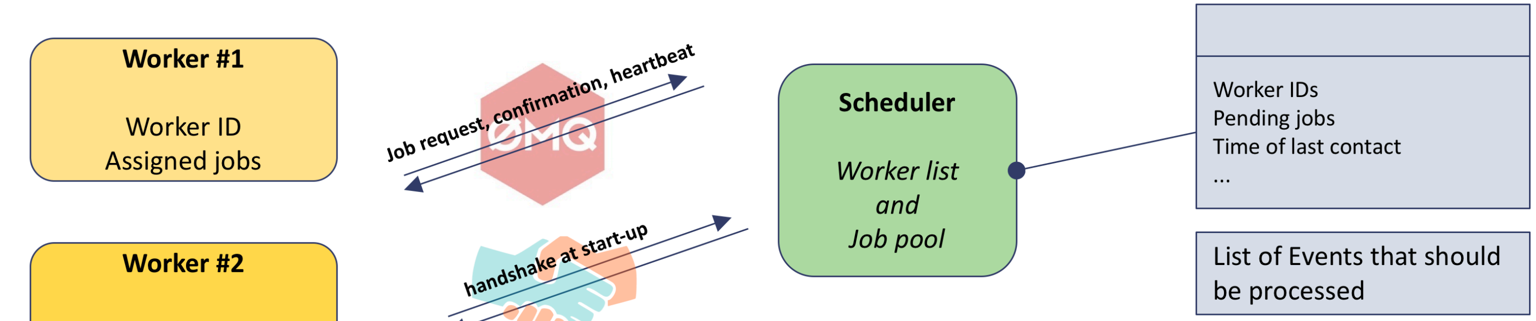
Binding together resources which are nearby can bring important benefits

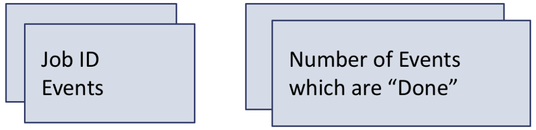
- Take into account Non-Uniform Memory Access

GeantV for HPC environments

Multi-tier mode (event servers)

- Useful to work with events from file, to handle merging and workload balancing
- Communication with event servers via MPI to get event id's in common files
- Started testing with HTCondor

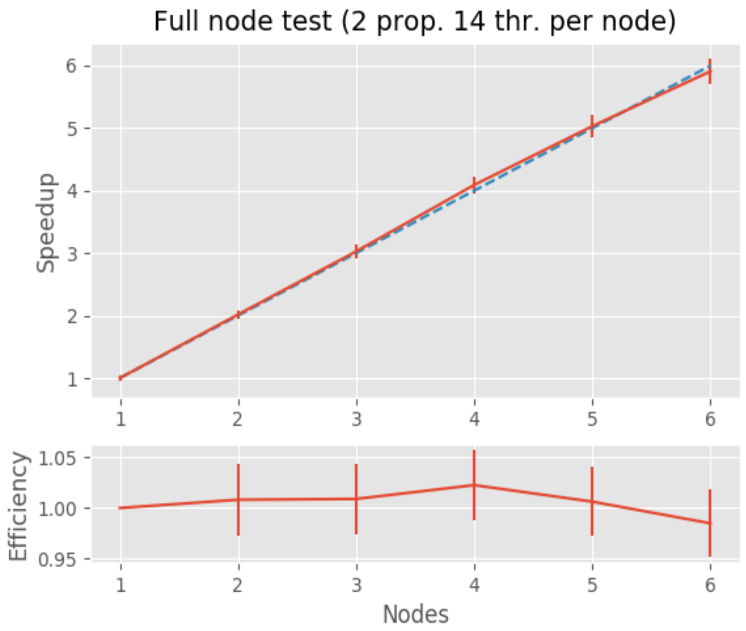




***Going beyond data parallelism*:**

- Scheduling the work from the application
- Optimizing work per machine using topology discovery;
- Handle un-even load, policies for work (re)distribution
- Resilience to hanging/dead workers

Applicable to other CPU-bound applications than GeantV



A. Gheata, ACAT 2017

Development plan

GeantV is an R&D project

- Valuable components already delivered to the community via Geant4/ROOT: VecGeom/VecCore, improved EM physics models
- Aiming at complete EM shower simulation in a vector flow

Available at gitlab.cern.ch/GeantV/geant.git

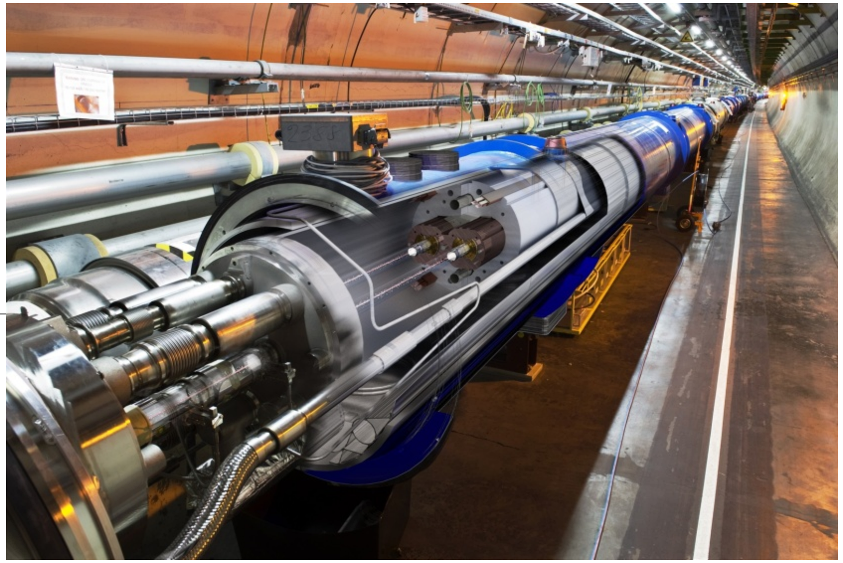
- Upcoming alpha release: full EM transport, vectorized geometry/ magnetic field, scalar physics, user interfaces, examples of different complexity

Full examples including MC truth processing and I/O ready in 2018:

- Beta release: EM transport fully vectorized, some hadronic components, more examples, fast simulation
- Extend benchmarking and optimisation to Power architectures (INAF)

Machine Learning

ML at the LHC

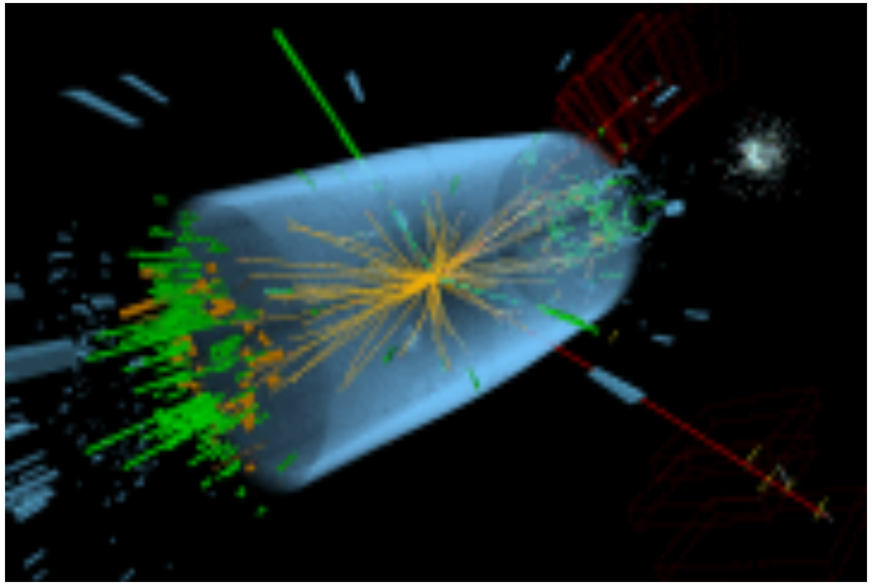


Infrastructure and accelerators:

- Integration of analysis tools with industrial control systems
- Online monitoring and fault detection
- Fault diagnosis and engineering design

Experiments and physics:

- Detector controls
- **Data quality monitoring and anomaly detection**
- Real time processing and selection
- Analysis and event identification
- **Simulation**



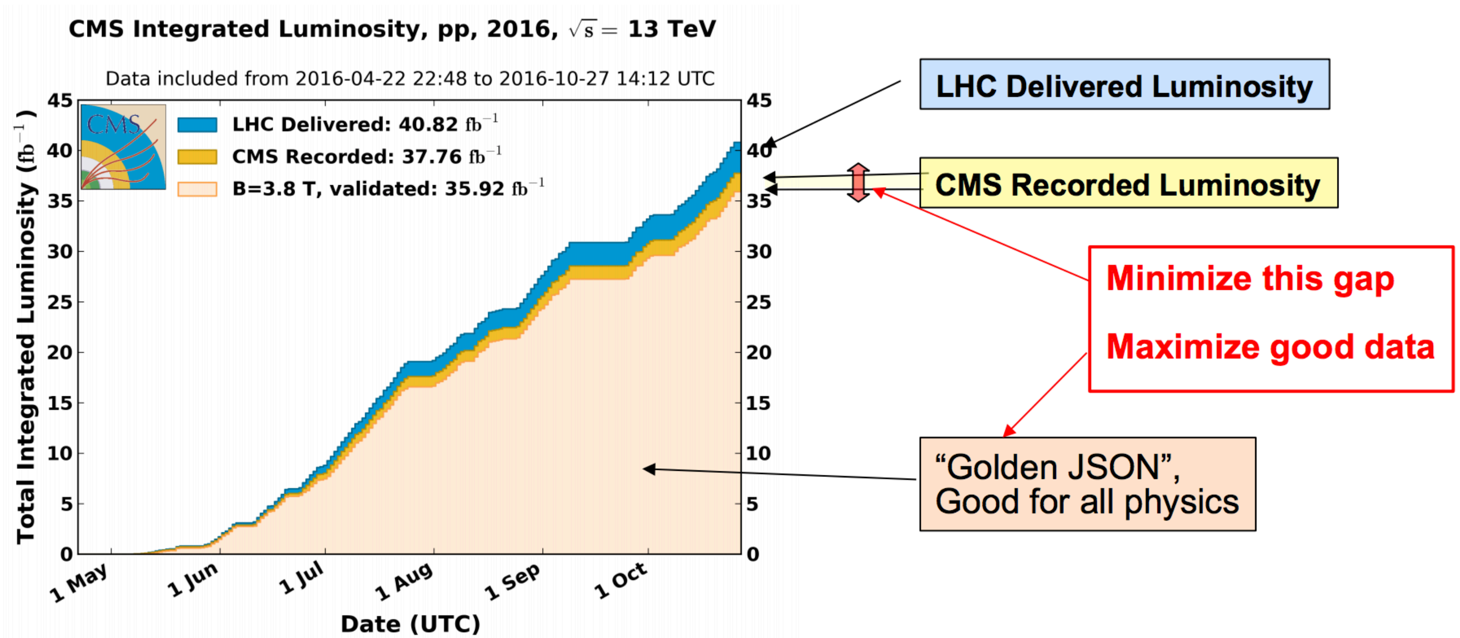
Data quality monitoring

Near-real-time applications: identify problems in the detector and data acquisition system

Currently automatic tests validated via visual inspection

Fast (~1h) reconstruction on a part of data

Full reconstructed data set monitored within ~48 h



Anomaly detection techniques to predict failures

Effective with minimal human guidance

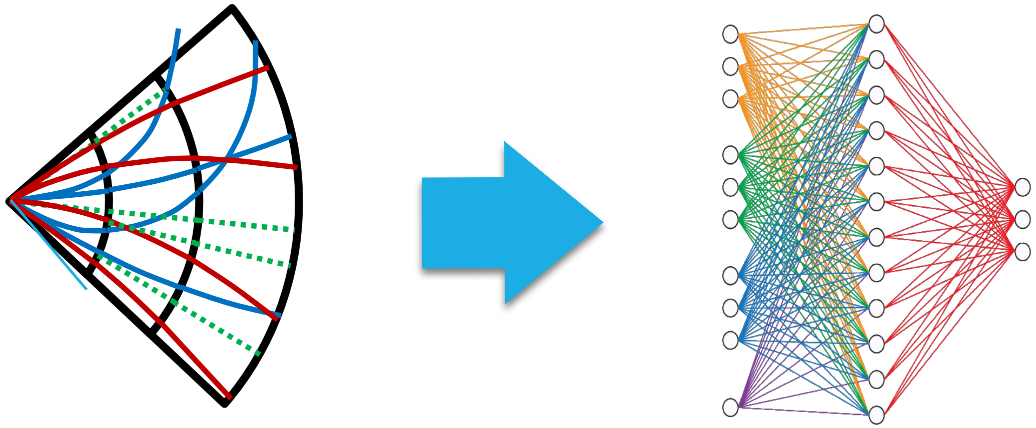
Adaptative behavior (data content, user needs, available resources , ..)

Improve the accuracy of data quality

Fast and efficient operation

V: Azzolini, CERN Openlab ML workshop 2017

Deep Learning for fast simulation



Generic approach

Can encapsulate expensive computations

DNN inference step is faster than algorithmic approach

Already parallelized and optimized for GPUs/HPCs.

Industry building highly optimized software, hardware, and cloud services.

Can we keep accuracy while doing things faster?
Can we sustain the increase in detector complexity (future highly-granular calorimeters are more demanding)?
What resources are needed?

How generic the network can be?
Can we "adjust" architecture to fit a large class of detectors?

S.V. ACAT2017

DL engine for fast simulation

First proof of concept developed within GeantV for a generic, configurable tool

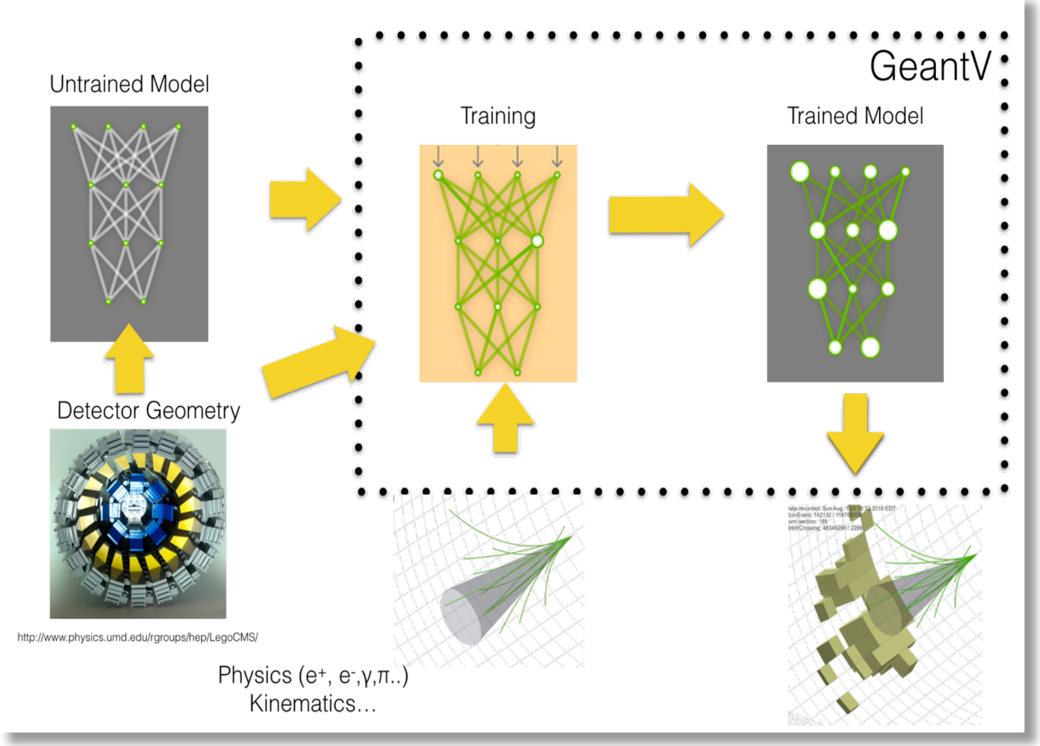
- Initially embed inference step
- Automate training according to use case

Available as standalone tool, include also in Geant4 as soon as possible

Test Generative Adversarial Networks (*)

- Realistic generation of samples

Keep training time under control

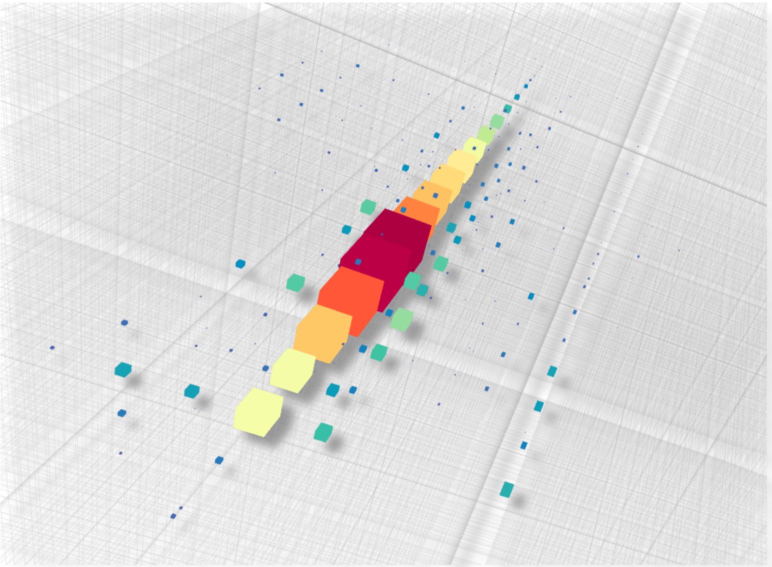


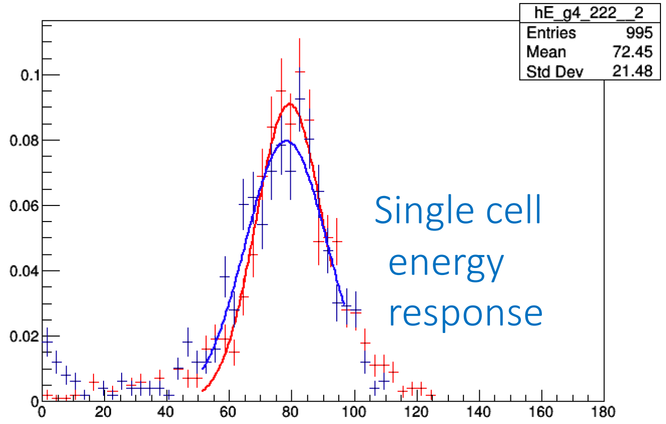
(*) Goodfellow et al. 2014

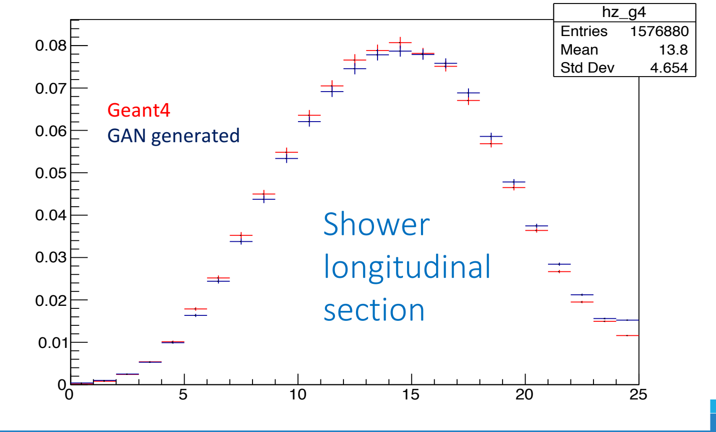
Calorimeter fast simulation

Generative Adversarial Networks based on 3D convolutions

- High granularity EM calorimeter example(*)
- Train on Geant4 simulation
- Detailed validation (single cell response, shower shapes, particle energy)







Physics results are very promising

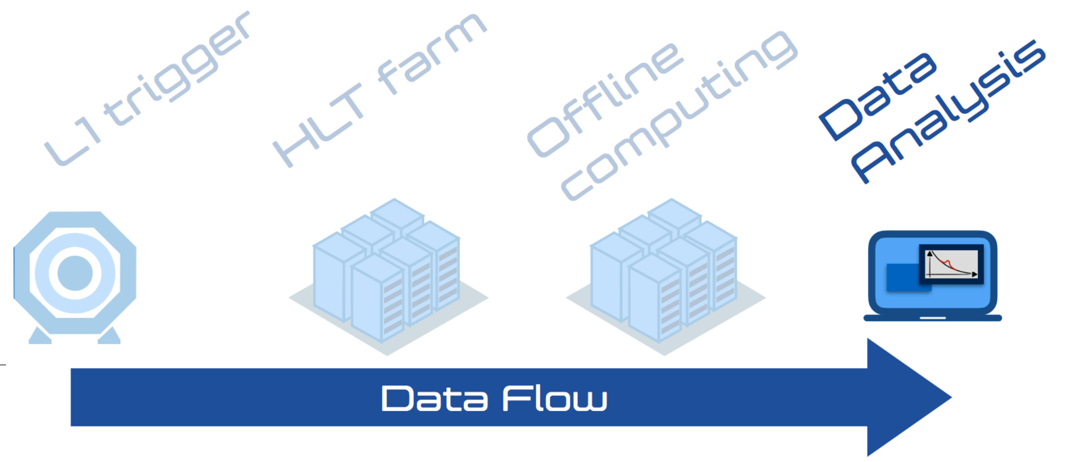
Computing time speedup is huge

		Time/Shower (msec)
Detailed Simulation	Intel Xeon E5-2683	56000
GeantV GAN (batchsize 128)	Intel i7 (my laptop!)	66
GeantV. GAN (batchsize 128)	GeForce GTX 1080	0.04

Analytics

Data Analysis

Data Analysis



Small groups, individually implemented analysis code

- Processing is analysis dependent:
  - Slimming (filter specific collisions) & Skimming (reduce content per collision)
  - Calculation of new quantities
- Multi-step workflow (no interactive analysis)
- Rerun framework code
  - correct problems/ mistakes
- Can take weeks on GRID and local batch systems
  - Experiments now centralize first step
- Not all is actual CPU time
  - also bookkeeping, resubmission of failed jobs, etc.

- Up to ~ 500 Hz In / 100-1000 events out
- <30 KB per event
- Processing time irrelevant

Currently based on ROOT

ROOT – development plan

PCC to modernize ROOT Math and I/O libraries for multicore and many-core architectures

- Parallel, declarative analysis (TDataFrame)
- More internal use of parallelism and vectorization
- Better machine learning integration (keras, tensorflow etc) and features (e.g. convolutional neural net);
- Web-based (HTML, CSS, JavaScript) graphics, GUI, event display
- Robust, light-weight and fast histograms
- Improved PyROOT, better C++ integration, high-bandwidth connection to numpy
- On-demand build of parts of ROOT; reduced dictionary cost
- New platforms, new C++

https://root-forum.cern.ch

Data Analytics for Big Data

New toolkits and platforms have emerged to support the analysis of PB and EB datasets in industry.

Applying these technologies to HEP could reduce time-to-physics

Data analysis needs fast turn-around

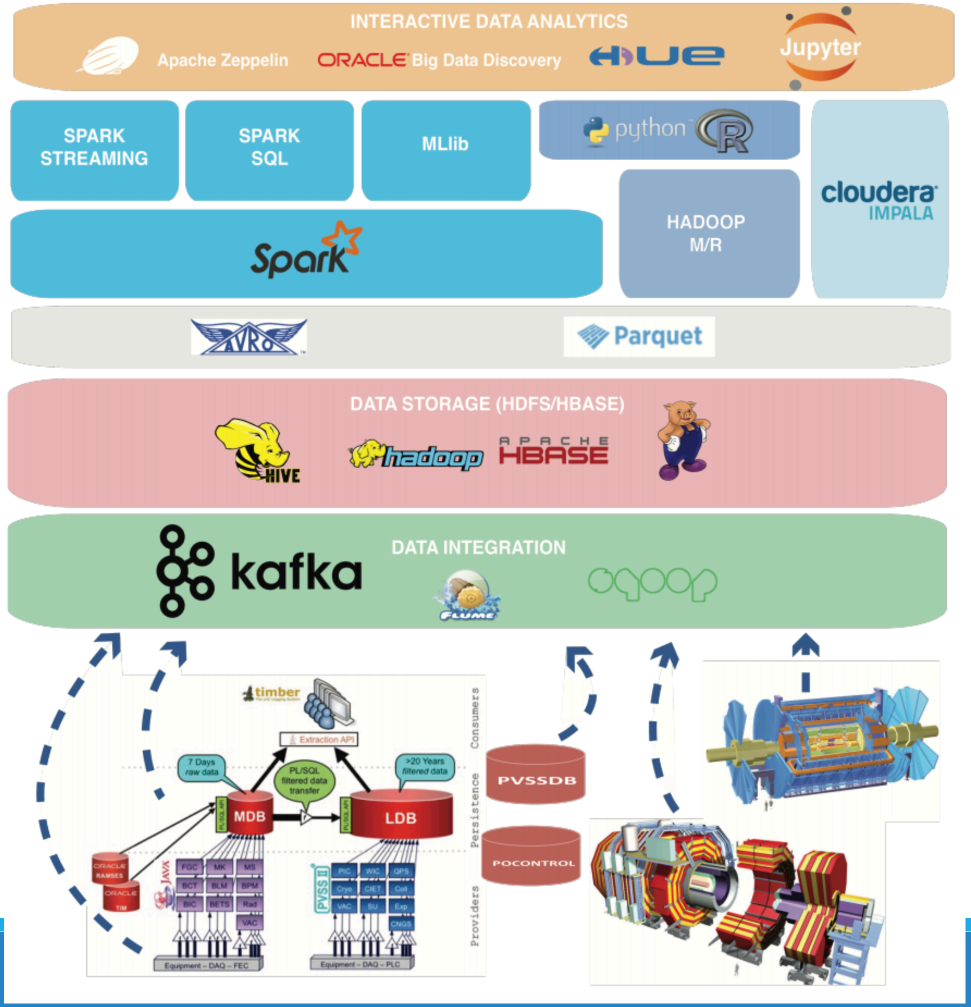
- “Interactivity” is a big need for efficient data exploration

Data volumes will soon reach multi-PB sizes

- input data composition different for every analysis

New scalable data services being tested

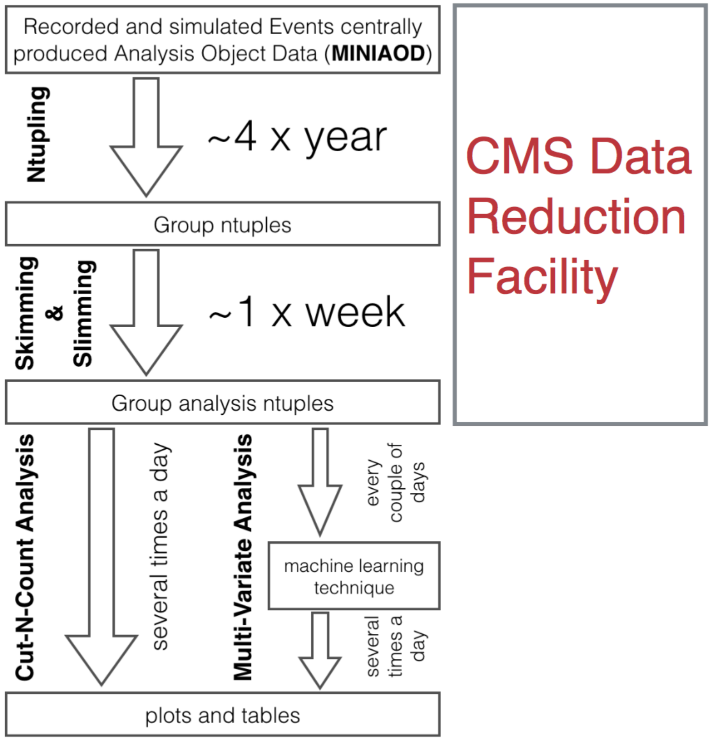
- Scalable & Time Series databases
- Hadoop ecosystem
- Interactive data analytics (Jupyter..)



CMS Data reduction facility

based on

CMS Data Reduction Facility



- CERN Openlab project with Intel (2 years)
- Demonstration facility optimized to read through petabyte sized storage volumes
  - Produce sample of reduced data based on potentially complicated user queries
  - Time scale of hours and not weeks as it currently requires.
- If successful, this type of facility could be a big shift in how effort and time is used in physics analysis
  - Same infrastructure and techniques should be applicable to many sciences

O. Gutche, openlab open day

LHC Software on HPC

Improving utilization of “supercomputers” by running applications on idle cores

- LHC experiment applications are tailored to run on *high-throughput computing* resources
- A core framework that allows hundreds of researchers to plug in specific algorithms
- many GB for a single release - New releases on a daily basis

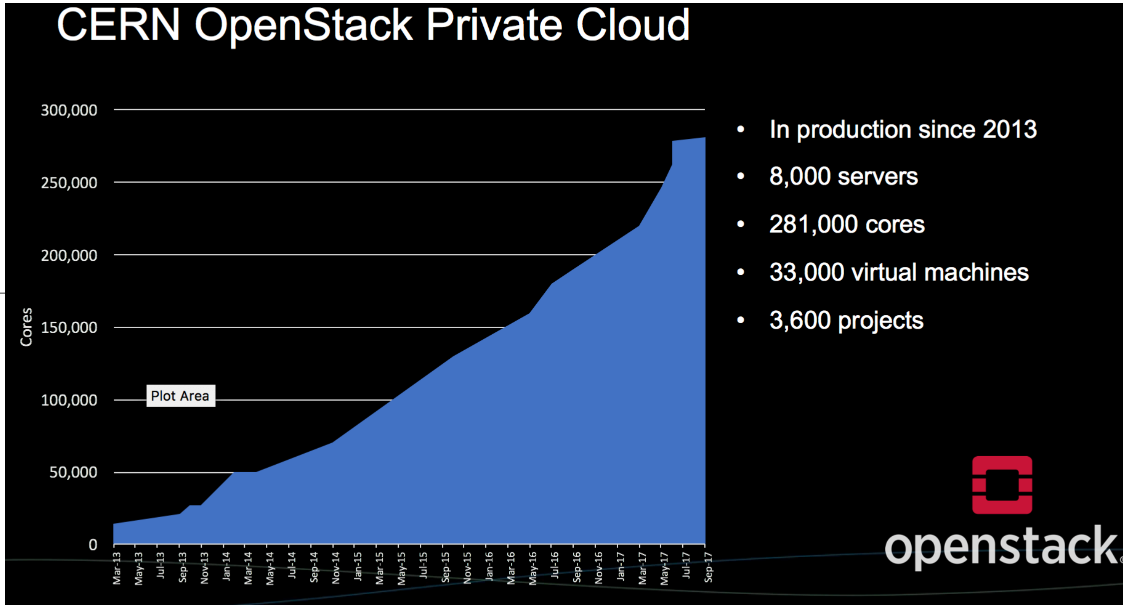
Distribute software stacks to world-wide distributed computing resources via CernVM-FS: purpose-built, global, POSIX file system

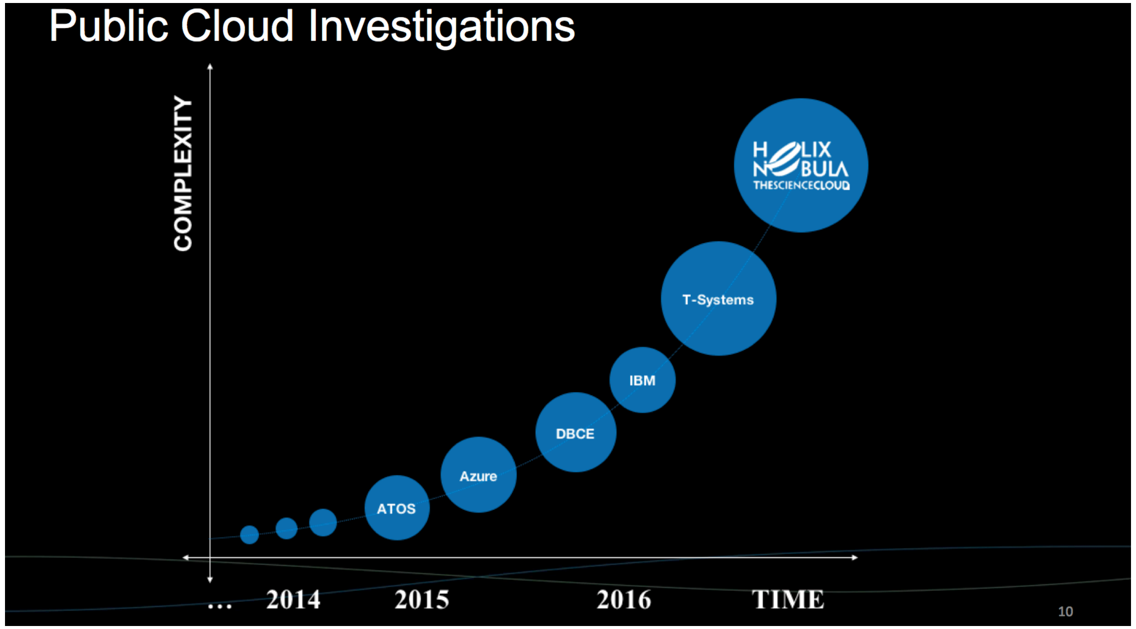
- Tests at NERSC, Berkeley, LRZ, Munich, and CSCS in Lugano.

J. Blomer, ISC 2017

Cloud computing

- CERN has its own private cloud
- Investigate scale-out with public providers without impact on users
- **Helix Nebula** –a Pre-Commercial Procurement tender for a European hybrid cloud
    - support deployment of high-performance computing and big-data capabilities for scientific research
    - Available to multiple user groups in HEP, astronomy, life sciences, …





T. Bell, "Accelerating Cloud through science"

Summary

Activities and initiatives are ongoing within the HEP to prepare for future HL LHC runs

- CERN openlab represents an effective framework for collaboration with industry partners and other scientific communities

Important IT challenges trigger sustained efforts to

- Modernize HEP code and benefit from new architectures
- Improve/optimise usage of HPC and distributed environments
- Introduce strategies to deal with Big Data (Machine Learning, Big data Analytics)

Share experience and techniques developed in HEP with other sciences facing similar challenges

- Exascale data processing at future astrophysics infrastructures, such as SKA

MoU INAF - CERN openlab

INAF is involved in several international projects and has significant experience in ICT field

Collaboration INAF-CERN openlab is highly profitable to both communities

Some interesting points

- GeantV porting and optimisation on IBM Power8+ and Power9 platforms
- Test and optimisation for HPC environments
- Explore possible cooperation on machine learning applications to simulation and other use-cases
- "Exa"scale data processing

Thanks

GeantV approach

Classical simulation

One track at a time through all stepping stages

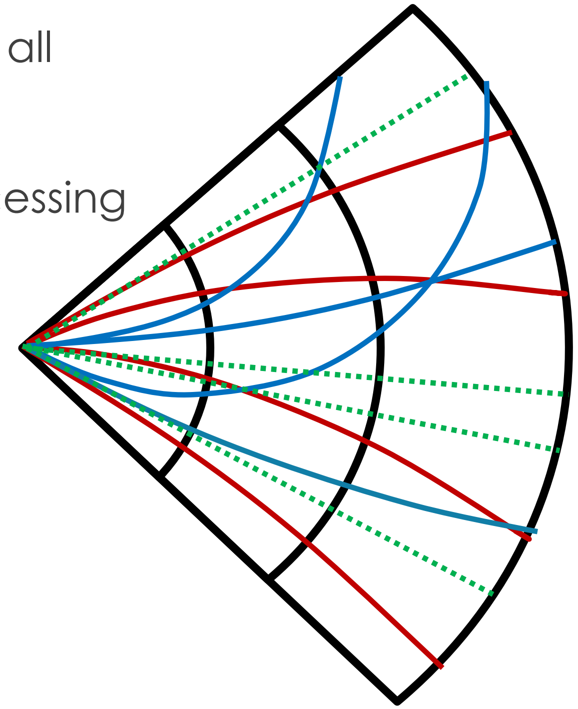
Sequential stack-driven processing

Single event transport

Event-level embarrassing parallelism

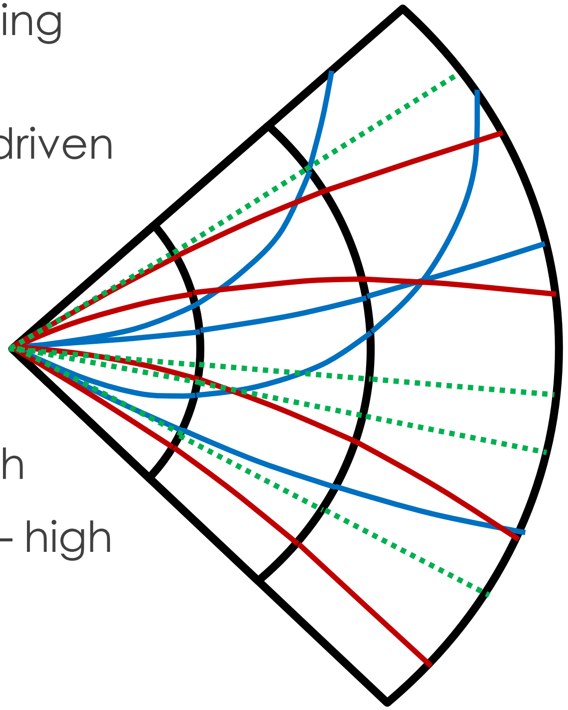
Cache coherency – low

Vectorization potential – low (scalar auto-vectorization)

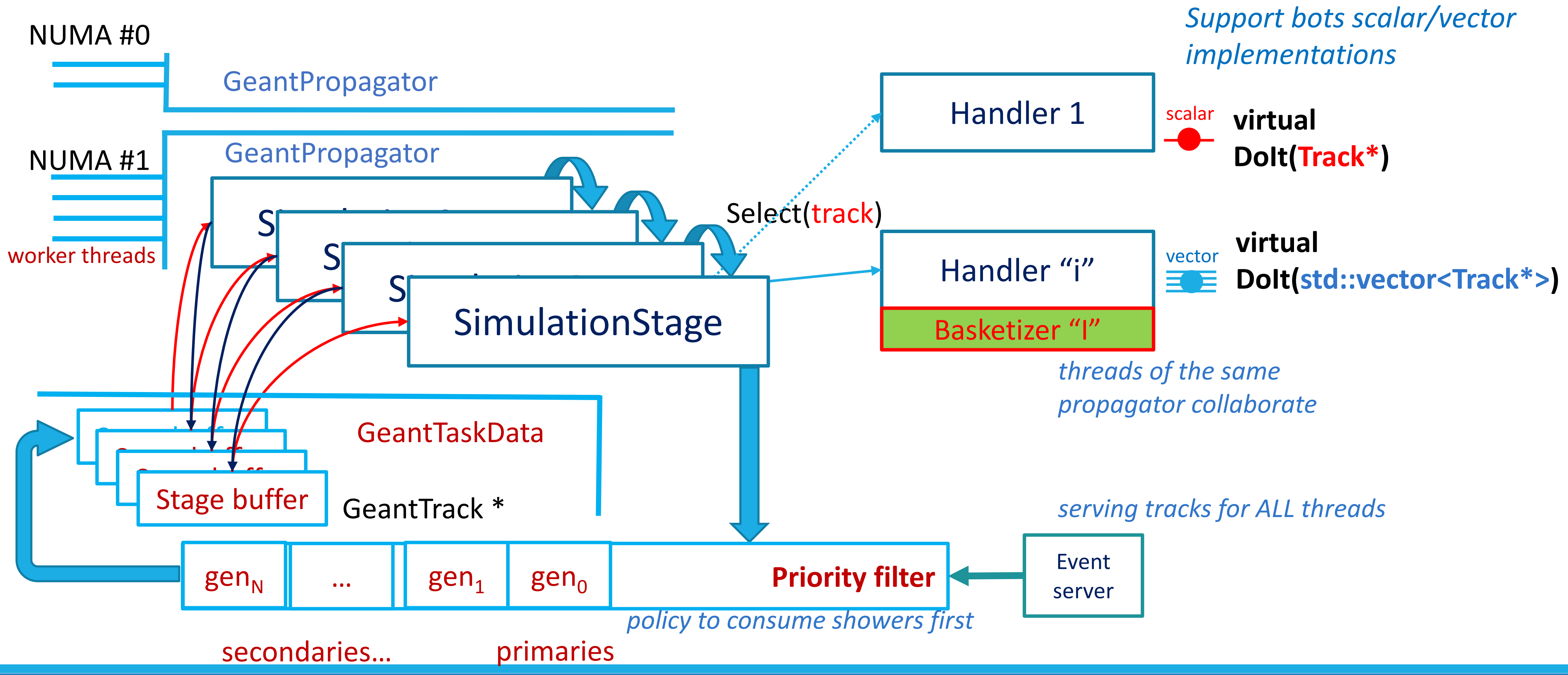


GeantV simulation

- Groups of tracks executing together each stage
- Non-sequential basket-driven processing
- Multi event transport
- Track-level fine-grain parallelism
- Cache coherency – high
- Vectorization potential – high (explicit multi-particle interfaces)



A generic vector flow approach



GeantV portable performance

Long-term maintainability of the code

- One single version of each algorithm

Platform specialization via C++ templates and low level optimised libraries

1 particle API

Many particle API (SIMD)

Common C++ template functions

```
double distance( double );
```

```
UME::SIMD::double_v distance( UME::SIMD::double_v );
```

```
template<class Backend>
Backend::double_t
common_distance_function(
Backend::double_t input )
{
   // Algorithm using Backend types
}
```

```
struct ScalarBackend
{
   typedef double double_t;
   typedef bool   bool_t;
   static const bool IsScalar=true;
   static const bool IsSIMD=false;
};
```

```
struct VectorBackend
{
   typedef UME::SIMD::double_v double_t;
   typedef UME::SIMD::bool_v bool_t;
   static const bool IsScalar=false;
   static const bool IsSIMD=true;
};
```

Backend: (trait) struct encapsulating standard types/properties

Supported SIMD backends: Vc, UME::SIMD

Vectorization tools: VecCore

Type-based explicit vectorization

- There are few libraries providing this: Vc, UME::SIMD

Geometry code & magnetic field RK propagator written in this way

- EM physics models coming next

Can profit also expressing internal vectorization when loop auto-vectorization is not possible

```
distance( double &);
```

Scalar interface

```
distance( vector_type  &);
```

Vector interface

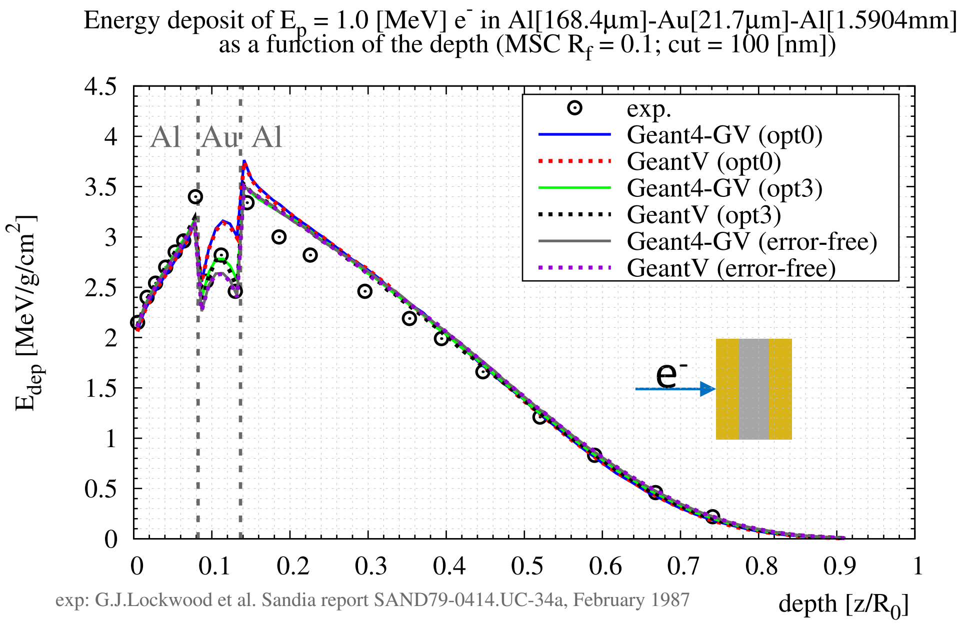
```
template<typename Real_v>
void DistanceImplementation( const Real_v &input,  Real_v & distance)
{
   Distance computation algorithm here  in terms of VecCore
   // Real_v is an arbitrary scalar/SIMD type. VecCore library wraps
   // math/vector operations expressed for different vector backends (Vc,
   // UME::SIMD)
}
```

```
template <typename T = Real_s>
class ScalarT {
public:
  using Real_v   = T;
  using Float_v  = Float_s;
  using Double_v = Double_s;
// Functions operating with scalar types
```

```
template <typename T = Real_s>
class VcVectorT {
public:
  using Real_v   = Vc::Vector<T>;
  using Float_v  = Vc::Vector<Float_s>;
  using Double_v = Vc::Vector<Double_s>;
// Functions operating with vector types
```

https://github.com/root-project/veccore

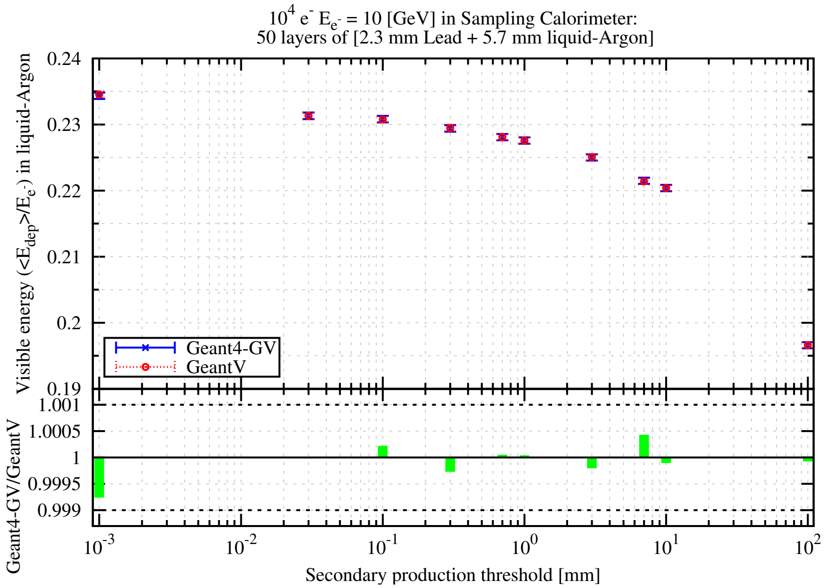
EM Physics models in GeantV



Multi-layered target

10^5 1 [GeV] e- in ATLAS bar. simpl. cal. : 50 layers of [2.3 mm Pb + 5.7 mm lAr]; p.cut = 0.7 [mm]

	e^-/e^+: ionisation, bremsstrahlung, msc; γ: Compton, conversion							
	GeantV				Geant4			
material	E_d[GeV]	rms [MeV]	tr.l. [m]	rms [cm]	E_d[GeV]	rms [MeV]	tr.l. [m]	rms [cm]
Pb	0.69450	15.198	51.015	1.189	0.69448	15.234	51.016	1.192
lAr	0.22792	14.675	106.11	7.592	0.22796	14.656	106.13	7.582



Mean number of :

gamma	405.87	406.15
electron	9411.49	9419.44
positron	53.77	53.71
charged steps	11470	11476
neutral steps	49177	49222

ATLAS simplified sampling calorimeter

Scalar EM models revisited in a vectorization friendly way (e.g. vectorizable sampling) and validated against Geant4 version. Vectorization work planned for 2018 to have vectorized shower simulation.

Generative models for simulation

Many models: Generative Stochastic Networks, Variational Auto-Econders, Generative Adversarial Networks ..

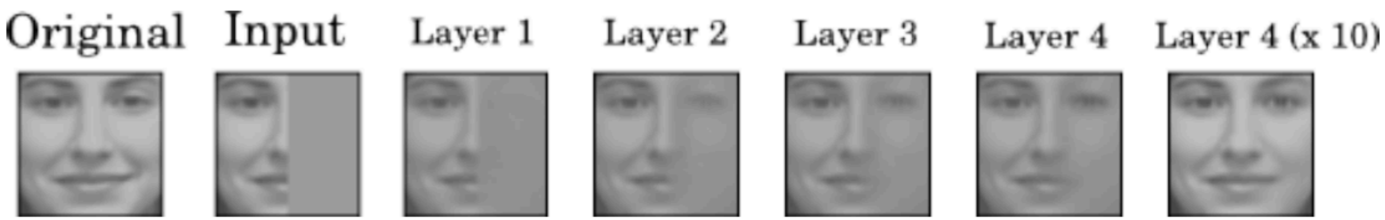
- Realistic generation of samples
- Use complicated probability distributions
- Optimise multiple output for a single input
- Can do interpolation
- Work well with missing data

Samples of images of bedrooms generated by a DCGAN trained on the LSUN dataset.

https://arxiv.org/pdf/1701.00160v1.pdf

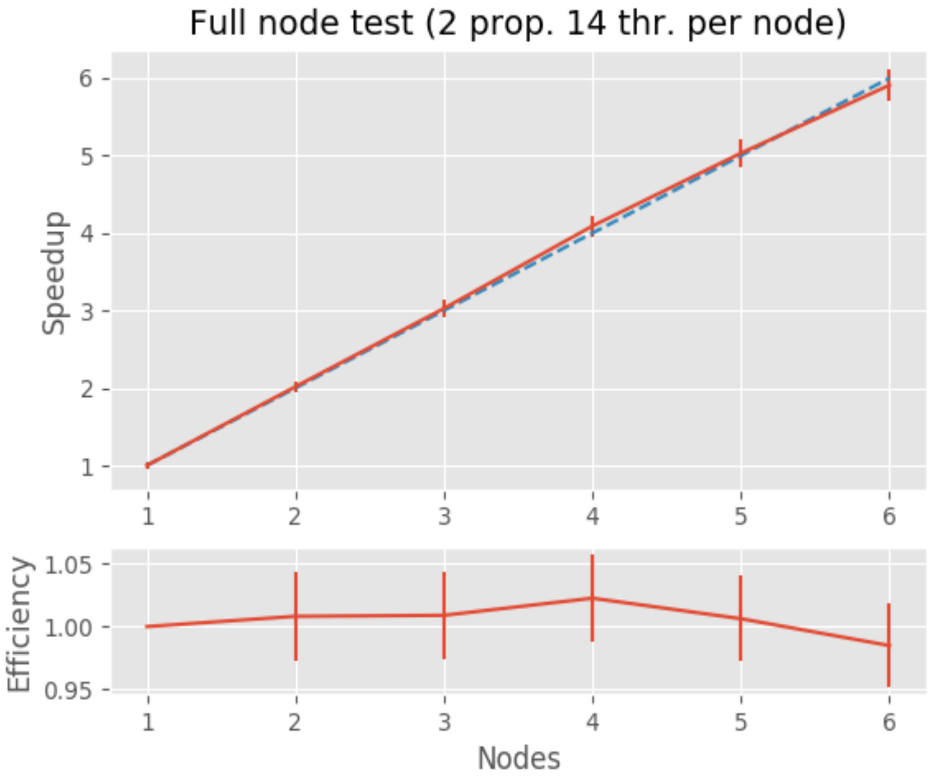
‘Small blue bird with black wings’ → ‘Small yellow bird with black wings’

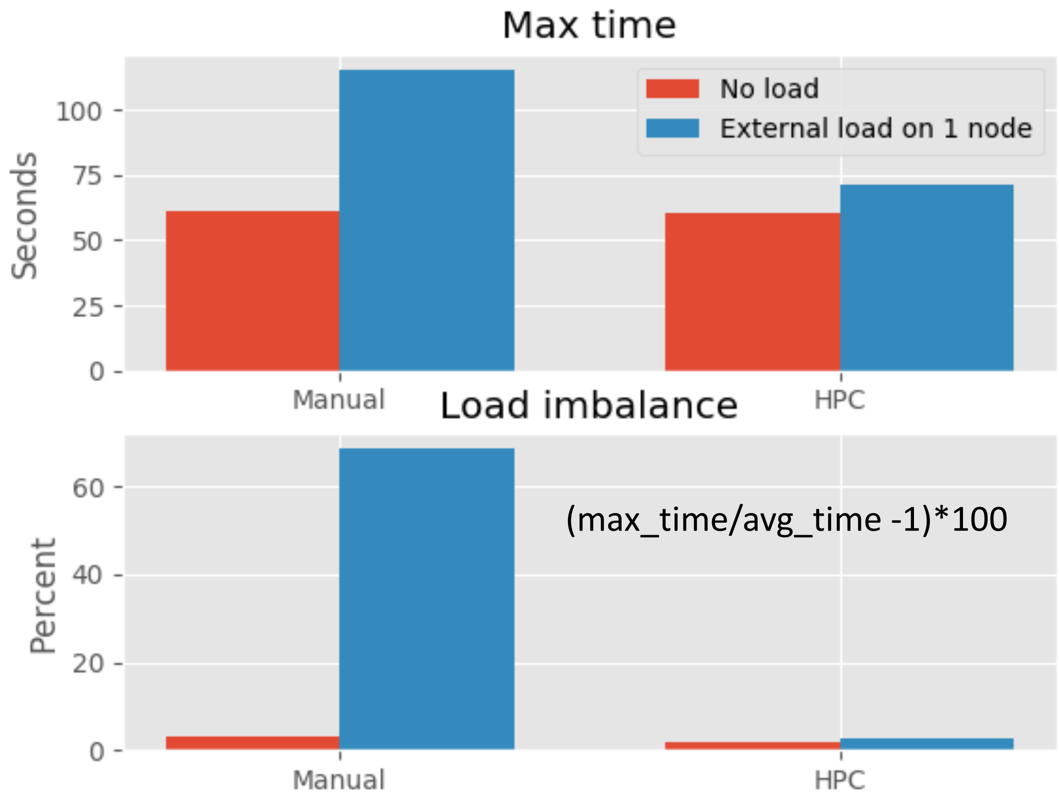
https://arxiv.org/pdf/1605.05396.pdf



Ranzato, Susskind, Mnih, Hinton, IEEE CVPR 2011

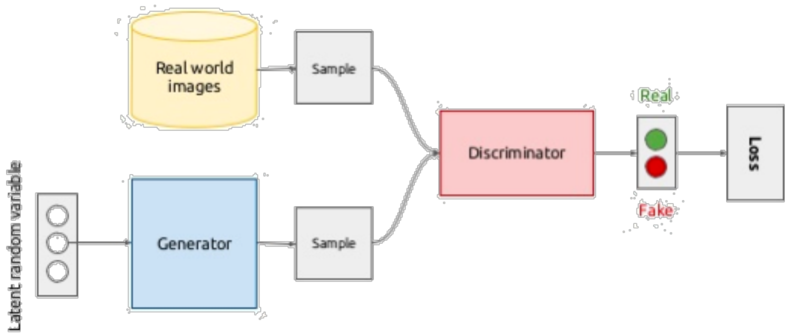
GeantV HPC mode: preliminary results





GAN Training time

- Using DL techniques for fast simulation is profitable if training time is not a bottleneck
  - Depending on the use case retraining might be necessary
  - Hyper-parameter scan and meta-optimisation
  - 3D generative adversarial networks are not "out-of-the-box"
    - Complex training process
    - Training on 150k events for 30 epochs takes ~24h on NVIDIA GTX-1080

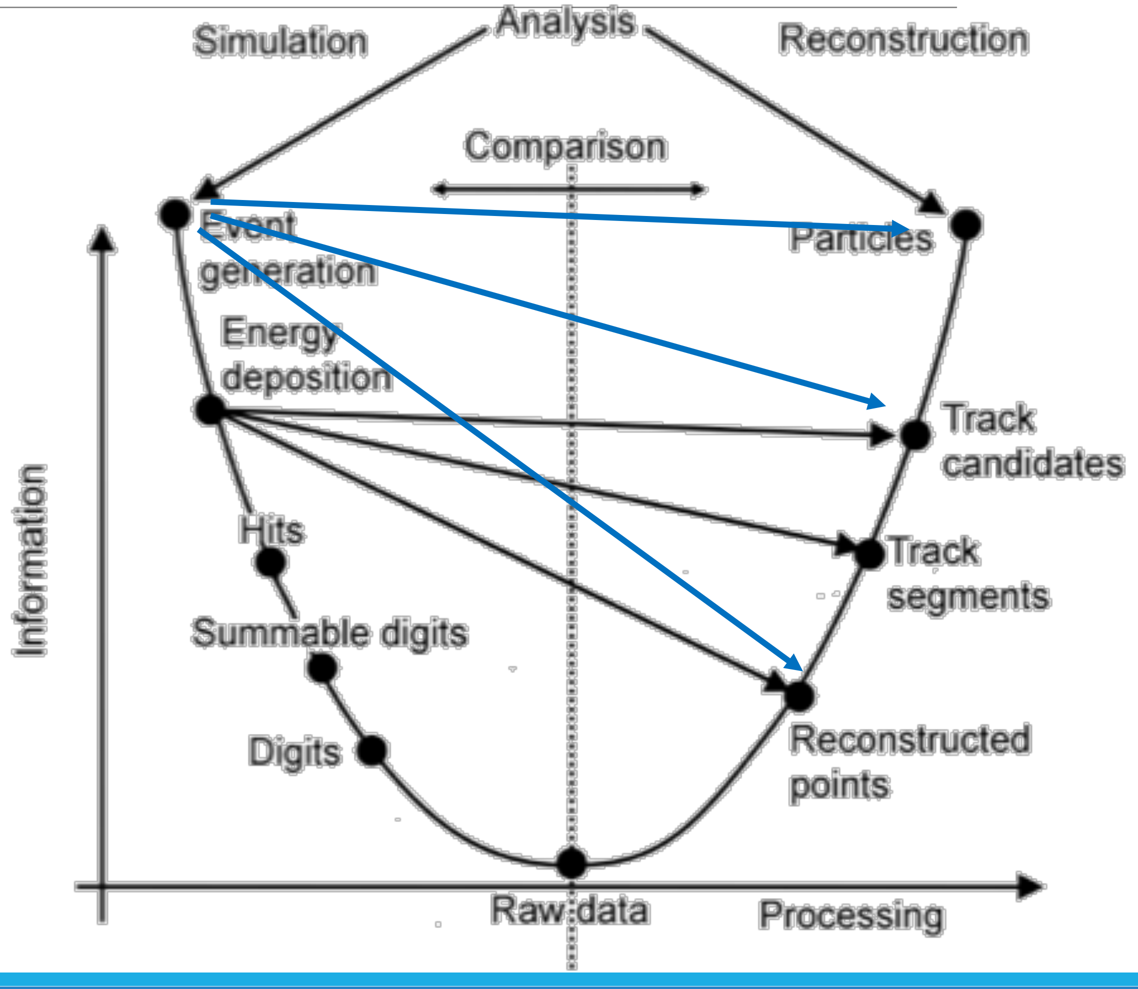


Fast simulation

...Digitisation
Reconstruction ...

Even larger speedup gained by replacing digitization and reconstruction steps

ML/DL tools are capable of "learning" extremely complicated feature spaces



Vectorised physics

Physics: Work mostly focused on building a complete set of vectorizable EM models

Compton scattering model:
comparing scalar with vector performance