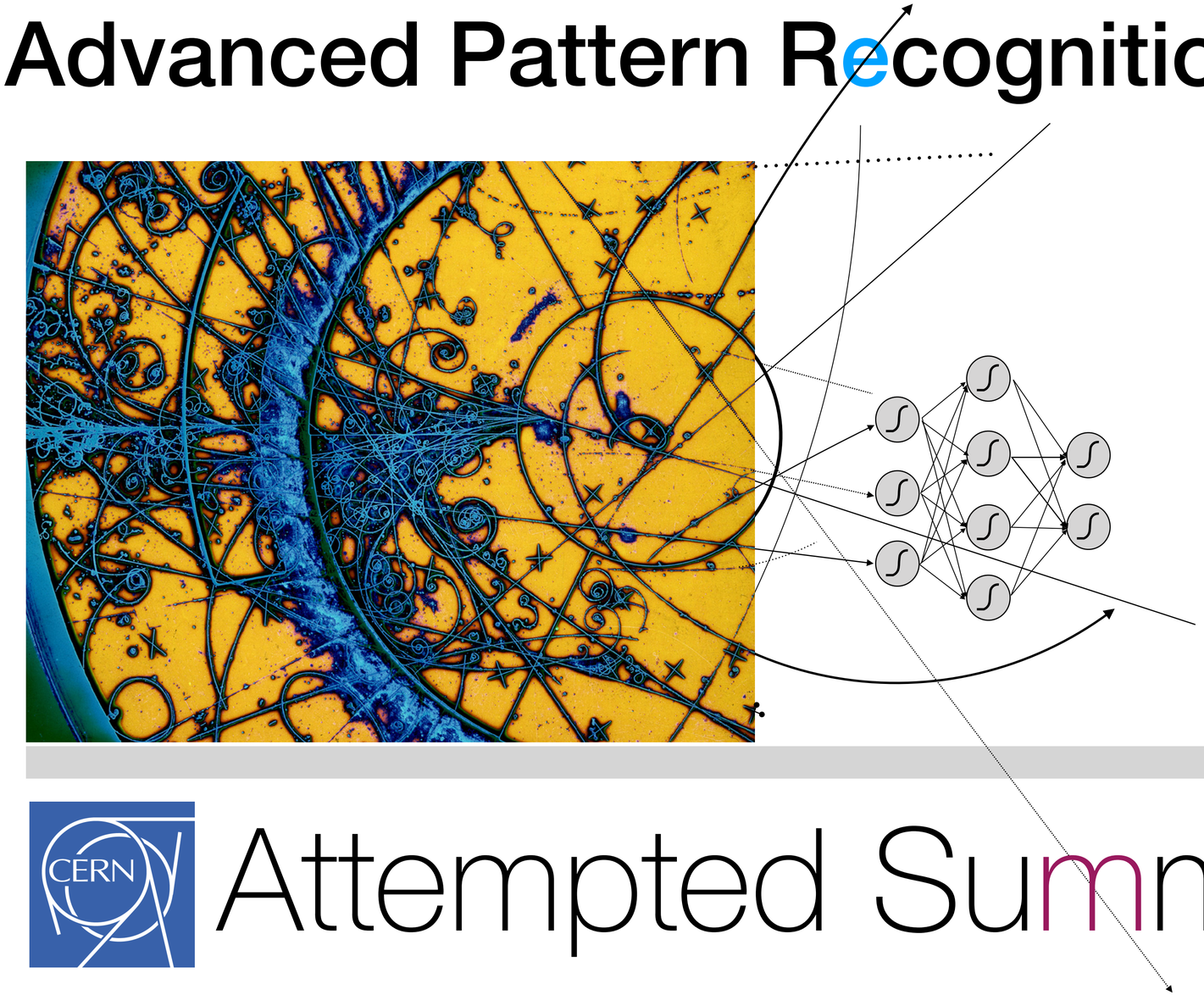
Learning to discover:
Advanced Pattern Recognition



Attempted Summary

A. Salzburger (CERN)

@SaltyBurger

Scientific contributions

- Twenty 1.5 hour contributions in our workshop
  - And we needed that time
  - And that’s a good thing (it was the intention of this format)

- Three hands-on tutorials
  - And a lot of the discussions had additional

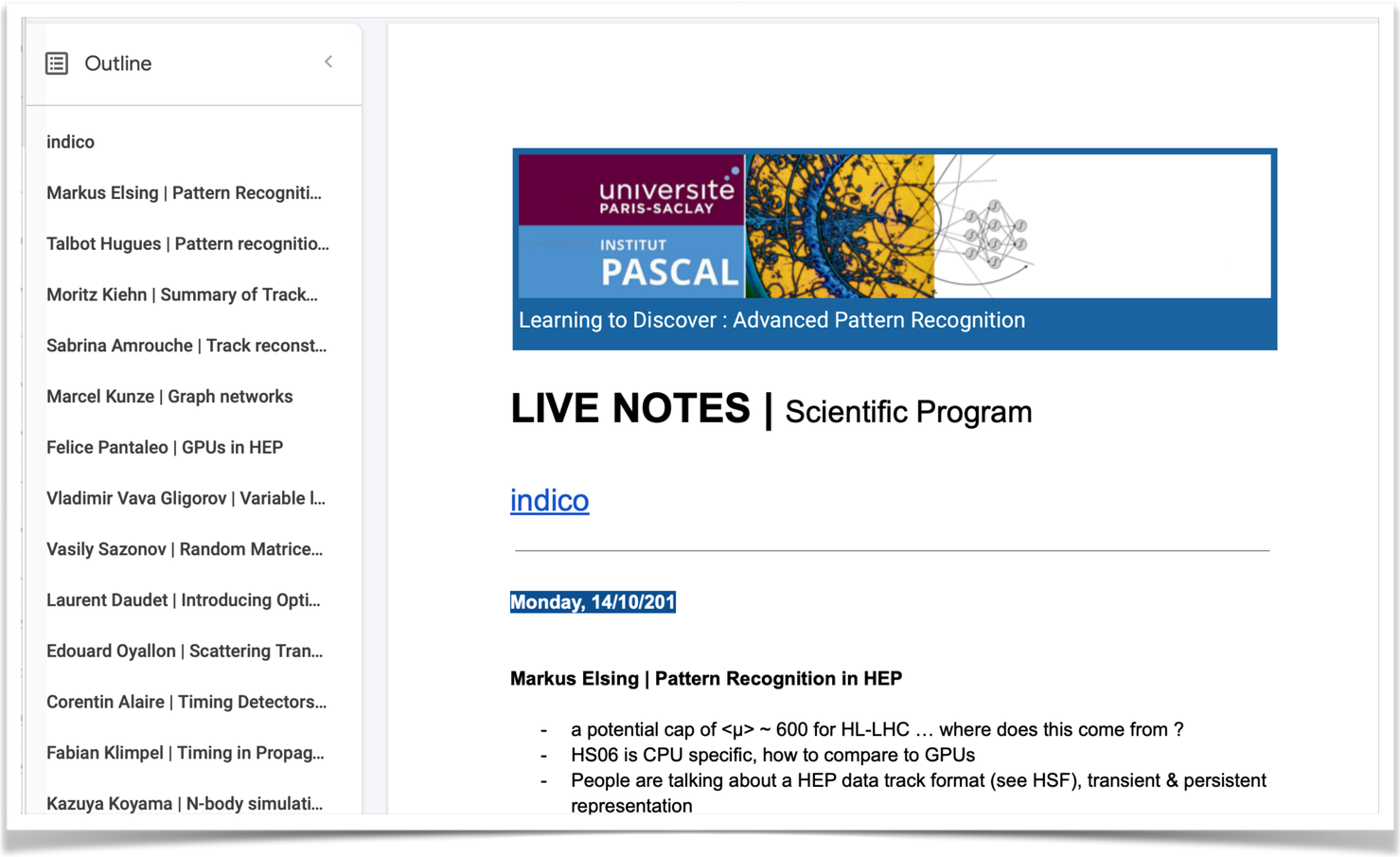
- One joint day with the AstroParticle workshop
  - And I would say this was very interesting indeed

Discussions

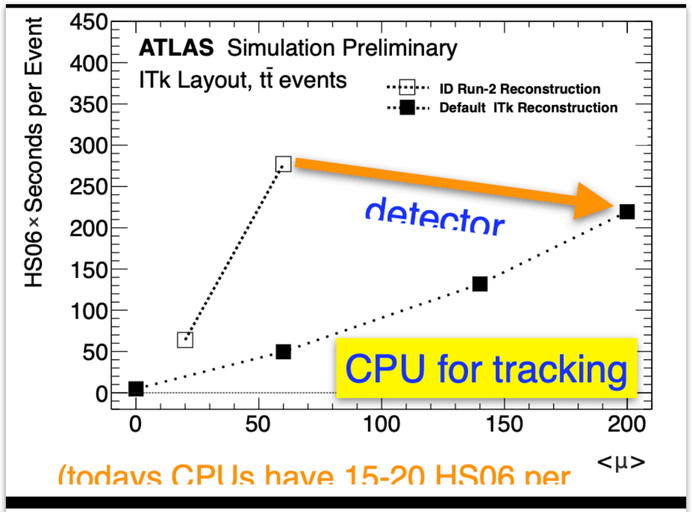
- I was super happy to see a lot of those happening in the free time
  - I hope this format pays really out

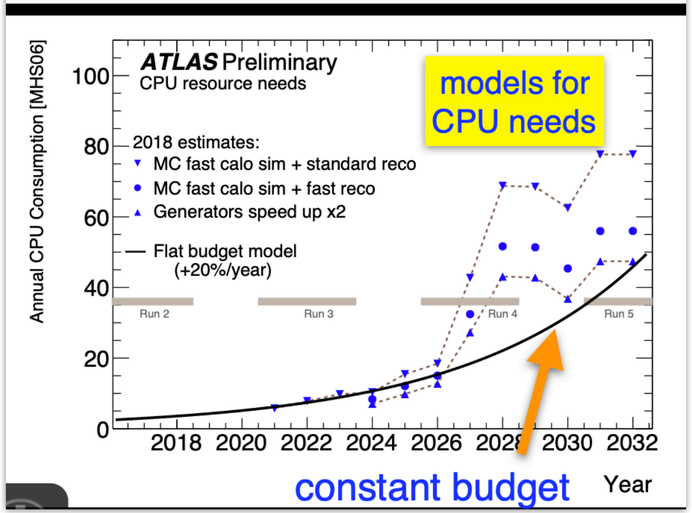
Some highlights & follow-ups

- Will take a bit of time to go through the [live notes]
  - The following is a small selections of my personal take-aways



Do we have a problem ?





- HL-LHC will certainly put our computing under pressure
  - BUT we still have margin to in our existing Tracking SW
  - It's too early to give up on that
- Interesting aspects
  - what can we do more if can do the exiting job faster
  - Which role will ML play here

Markus Elsing

A picture of me

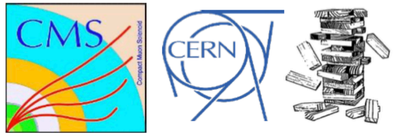
Markus Elsing | Pattern Recognition in HEP

- a potential cap of <μ> ~ 600 for HL-LHC … where does this come from ?
- HS06 is CPU specific, how to compare to GPUs
- People are talking about a HEP data track format (see HSF), transient & persistent

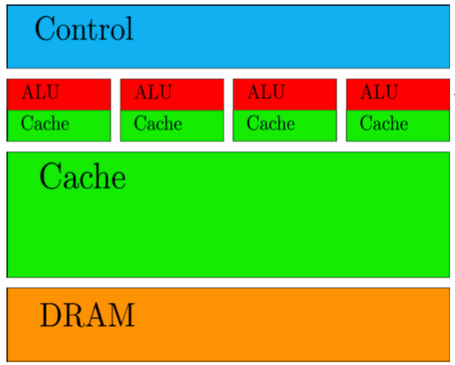
Or do we just need to accelerate?

Felice Pantaleo

Architectures

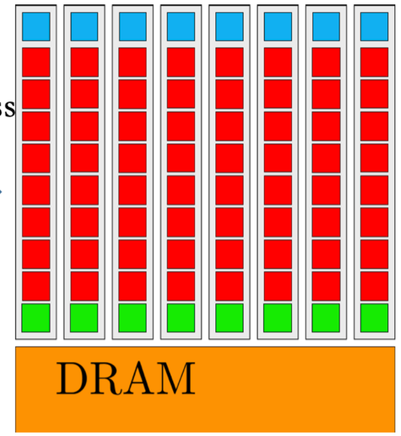


Control
ALU
Cache
DRAM

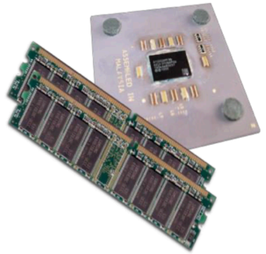


CPU

PCI Express
NVLink



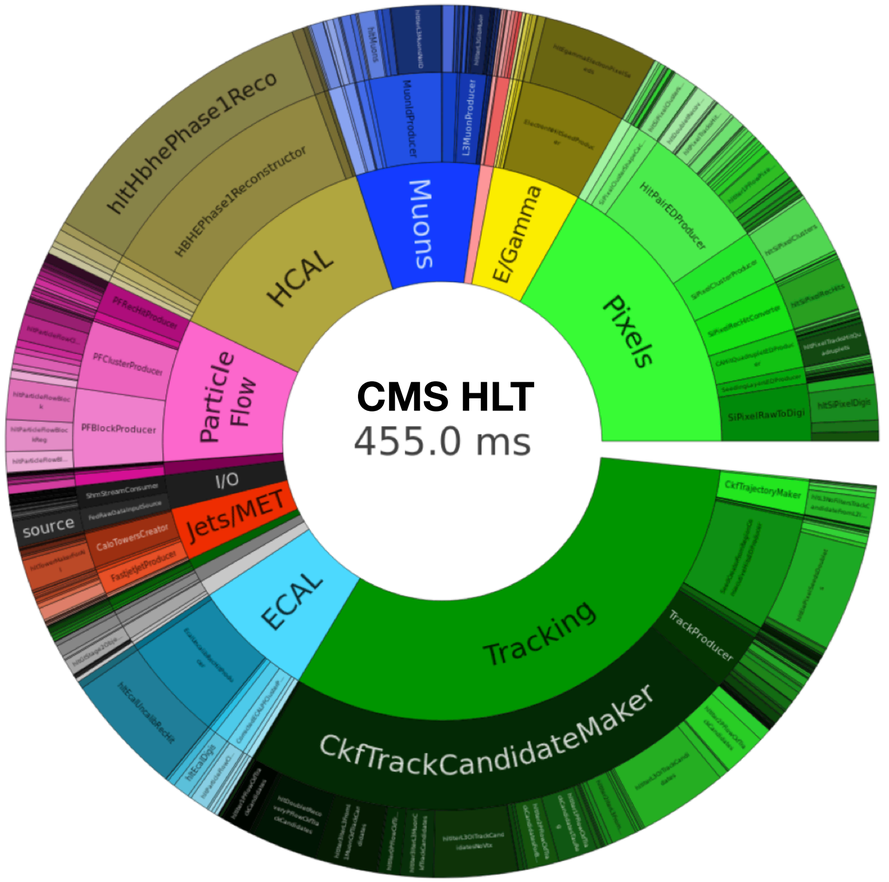
GPU



11

Or do we just need to accelerate?

- PataTrack as a full replacement of CMS HLT Pixel tracking
  - Written entirely in CUDA
  - technology bound at this stage, I CUDA to any other GPU-aware language will be way more easy



One mayor job done:

- adapt your Event Data Model

(usually speeds of CPU code)

Circles

- Eigen turned out to be a great choice as it runs natively CPU/GPU

Or do we just need to accelerate?

Vladimir Vava Gligorov

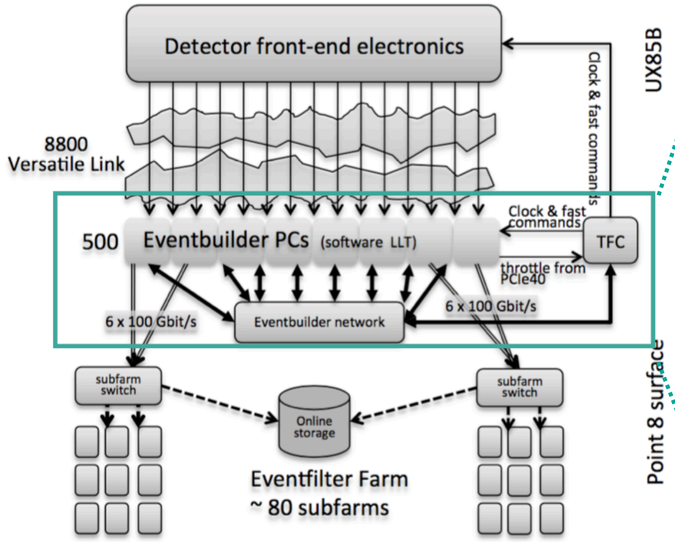
- Proposal to replace HLT1 with GPUs
  - practically cost-neutral to HW solution

And we also developed a GPU HLT1!

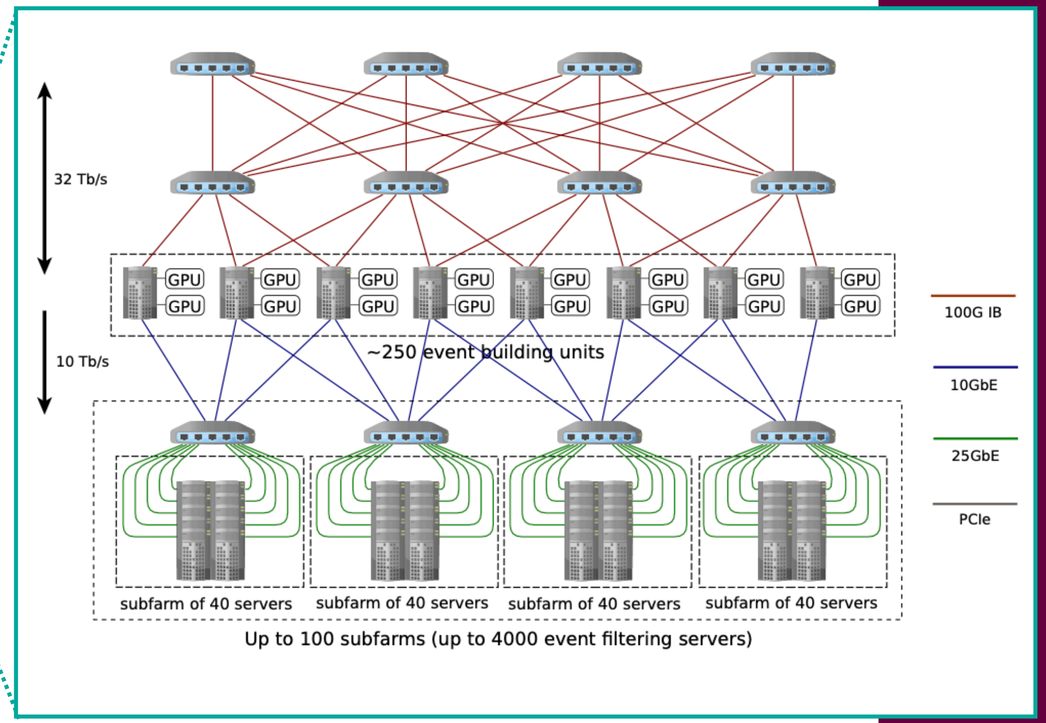
LHCb-ANA-20XX-YYY
May 31, 2019

Proposal for an HLT1 implementation on GPUs for the LHCb experiment

R. Aaij[1], J. Albrecht[2], M. Belous[a,3], T. Boettcher[4], A. Brea Rodríguez[5], D. vom Bruch[6], D. H. Cámpora Pérez[b,7], A. Casais Vidal[5], P. Fernandez Declara[c,7], L. Funke[2], V. V. Gligorov[6], B. Jashal[9], N. Kazeev[a,3], D. Martínez Santos[5], F. Pisani[d,e,7], D. Pliushchenko[f,3], S. Popov[a,3], M. Rangel[10], F. Reiss[6], C. Sánchez Mayordomo[9], R. Schwemmer[7], M. Sokoloff[11], A. Ustyuzhanin[a,3], X. Vilasís-Cardona[8], M. Williams[4]



Exploits flexibility of our Run 3 DAQ by implementing HLT1 directly in the servers receiving the data from the detector. Judged viable by external review, full cost-benefit analysis ongoing to decide if we will use this in Run 3. 38



Or just need to write better code?

David Chamont

A picture of me

A picture relevant to my work

- HEP code is often > 20 years old
  - AND it shows
- HPCs are becoming a player
  - Portability of code is an issue
  - There are new products (at least promised)
- Functional programming !!!

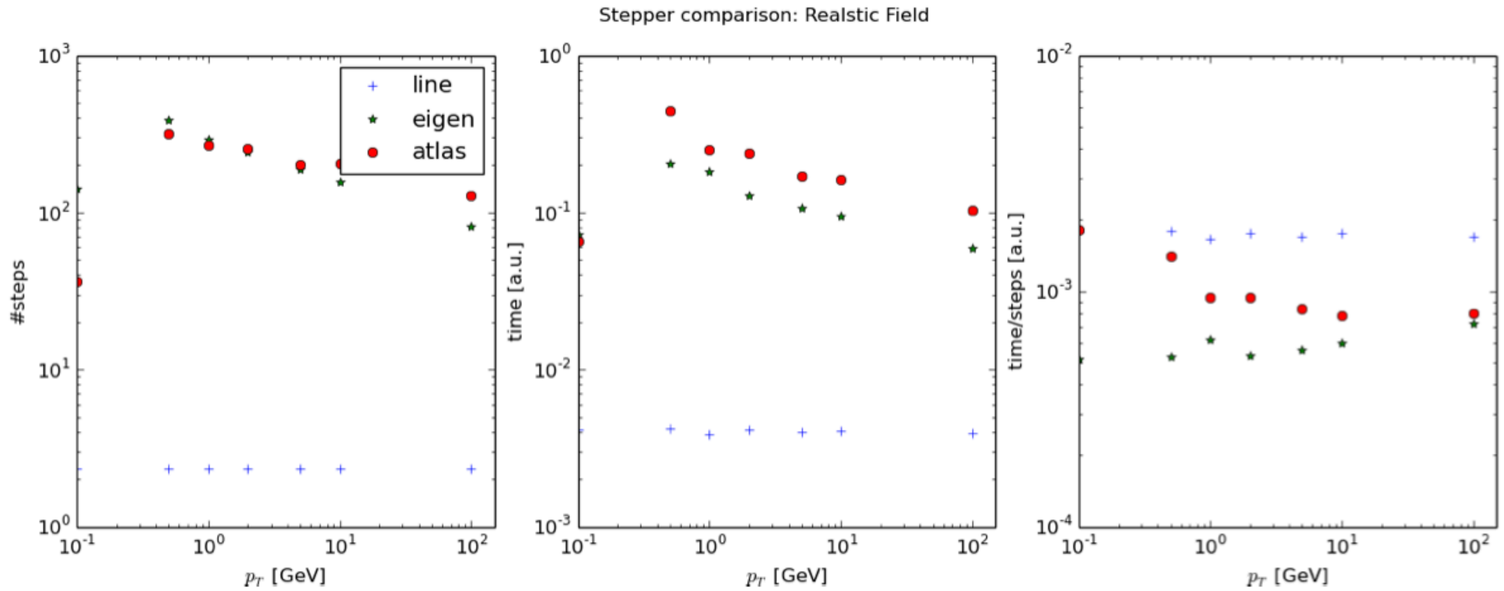
Future C++

```
auto open_file( string const & filename )  -> ifstream {
    return ifstream(filename) ;
 }

auto count_lines( ifstream file )  -> int {
    using ifiterator = istreambuf_iterator<char> ;
    return count( ifiterator(in), ifiterator(), '\n') ;
}

auto count_lines_in_files( vector<string> const & files ) -> vector<int> {
    return files | transform(open_file) | transform(execution::par,count_lines) ;
}
```

48 / 41

Or just need to write *better* code?

- Status of the Acts software package
  - C++/CPU targeted
  - EDM restructuring should help to move (later) to accelerator



36

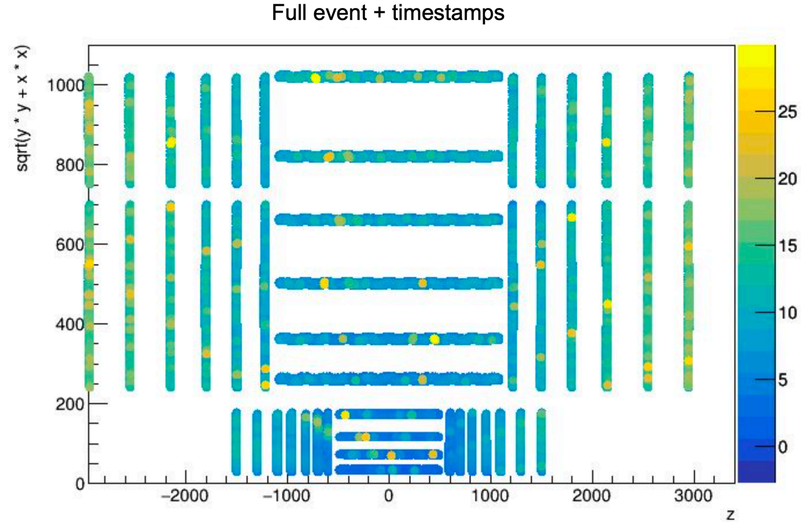
Or just get better detectors?

- Timing resolution has become good enough for track timing tagger
  - For Phase-2 ATLAS/CMS will install timing detectors for vertex identification
- Timing in reconstruction?

Corentin Allaire

Fabian Klimpel

Event view @<μ>=200



Follow-up:
- OpenDataDetector dataset with truth time (for smearing)

TrackML overview

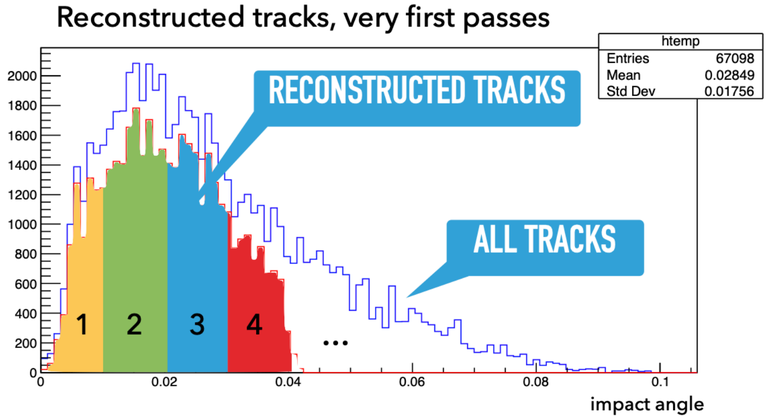
Moritz Kiehn

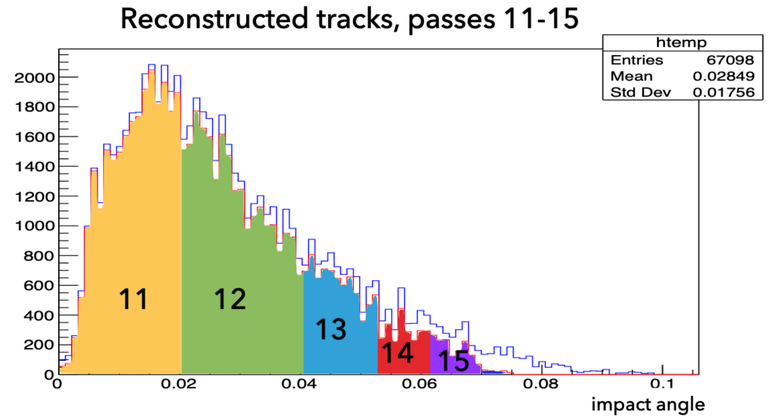
A picture of me

- TrackML aftermath
  - main solution have been pretty much digested
  - still many submissions not even looked at
    - did we miss something that has been submitted
  - Trying to convert this into a components that play together
    - see later
  - Dataset format established

Solutions - Mikado

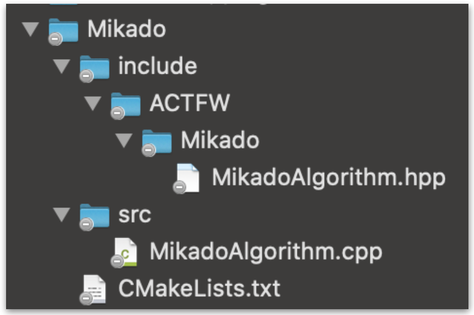
- Phase-2 winner
  - sub Hz solution with very high efficiency





To investigate & ideas:
- Reinforcement learning for parameter optimisation on Mikado
- Layer link-list auto-generation in acts ?

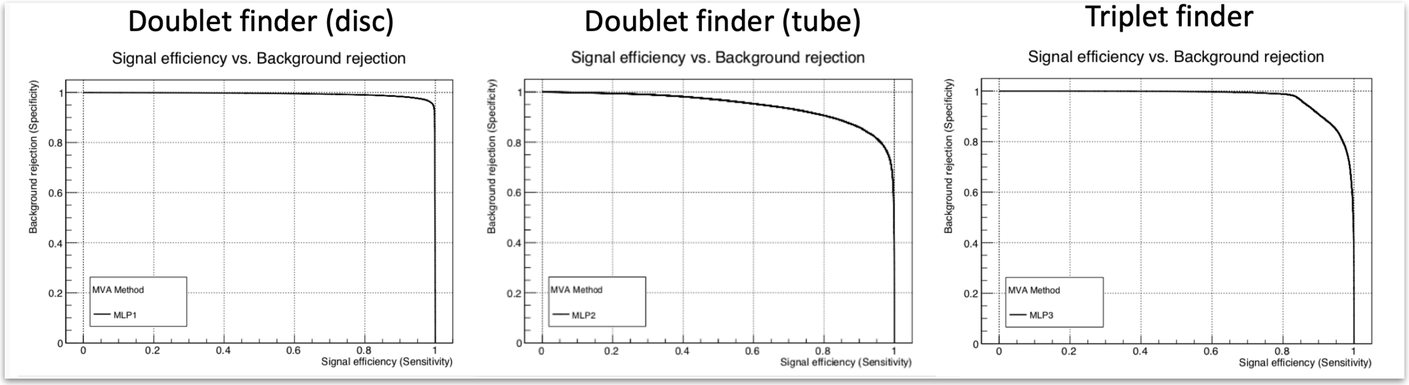
Started Mikado integration in acts



Solutions - *cloudkitchen*

- Based on winning solution of Phase 1
  - Which would have never scored high in Phase 2 (too slow)
  - Beauty of a challenge: *“and make it better”*

- Voxelisation & DAGs followed by NNs to find doublets/triplets



Marcel Kunze

A picture of me

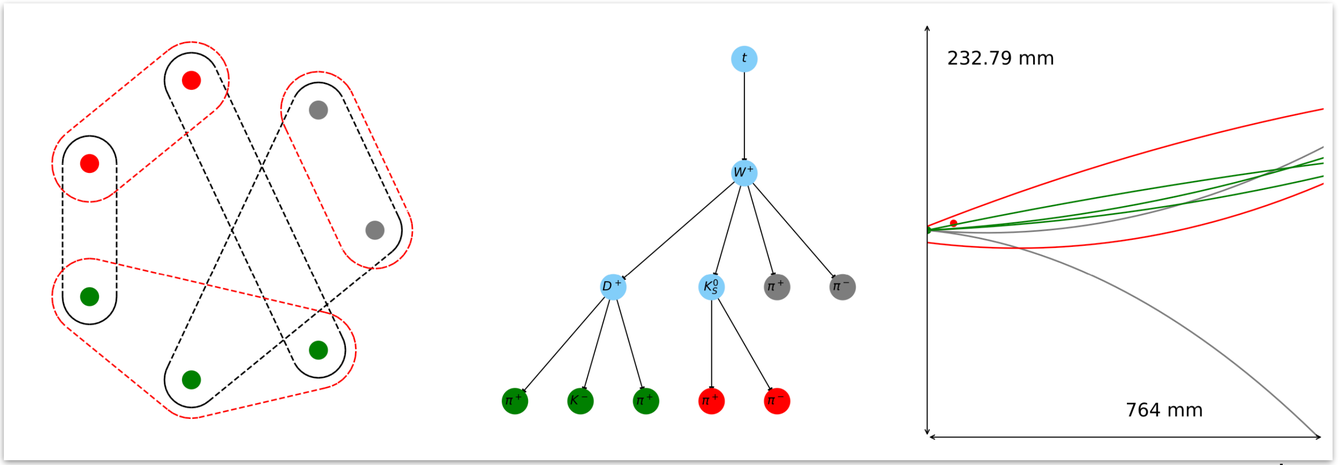
Graph network survey paper:
https://arxiv.org/pdf/1901.00596.pdf

Can we use Marcel’s DAGs as alternative buckets ?

Graphs, graphs, graphs

- Hep.TrkX and Exa.TrkX projects
  - Graph network approach for end-to-end tracking

- Graph network for vertex reconstruction



Jean-Roch Vlimant

Jonathan Shlomi

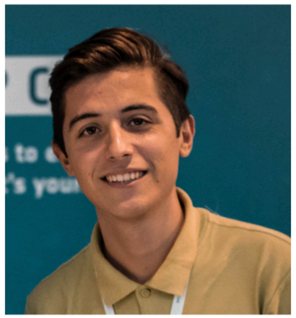
And how we look at things

- Event displays
  - From geometry and algorithm developing to physics object visualisation
  - The eye is impressive when it comes to pattern recognition

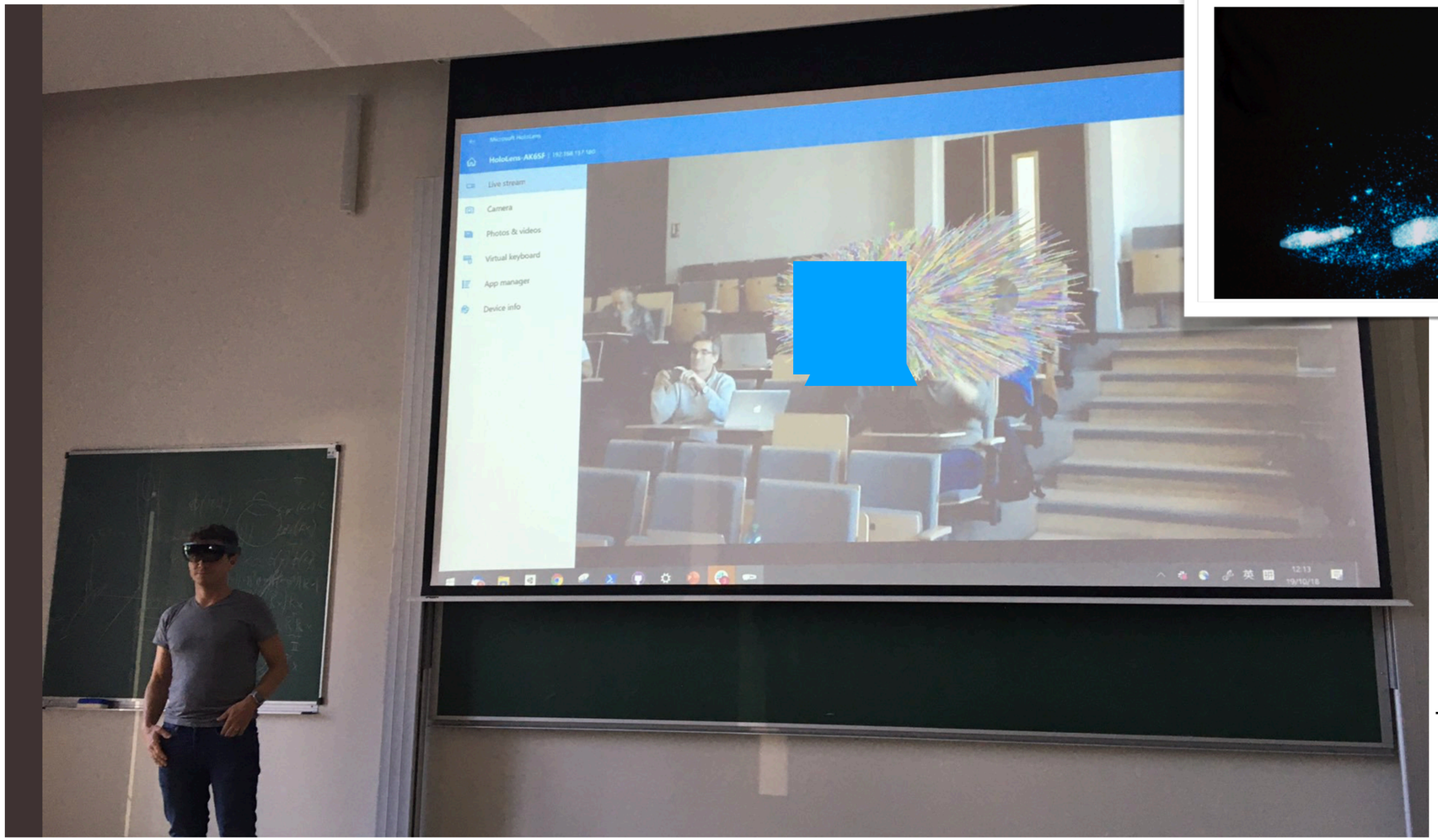
- A browser based Event display project supported by HSF
  - Common languages between experiments would be the key

Edward Moyse

Emilio Cortina



And how we look at things



Tobias Isenberg

sharing the know-how

David Rousseau

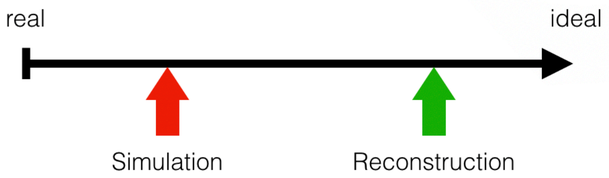
A picture of me (OK a few wrinkles missing)

A picture relevant to my work:

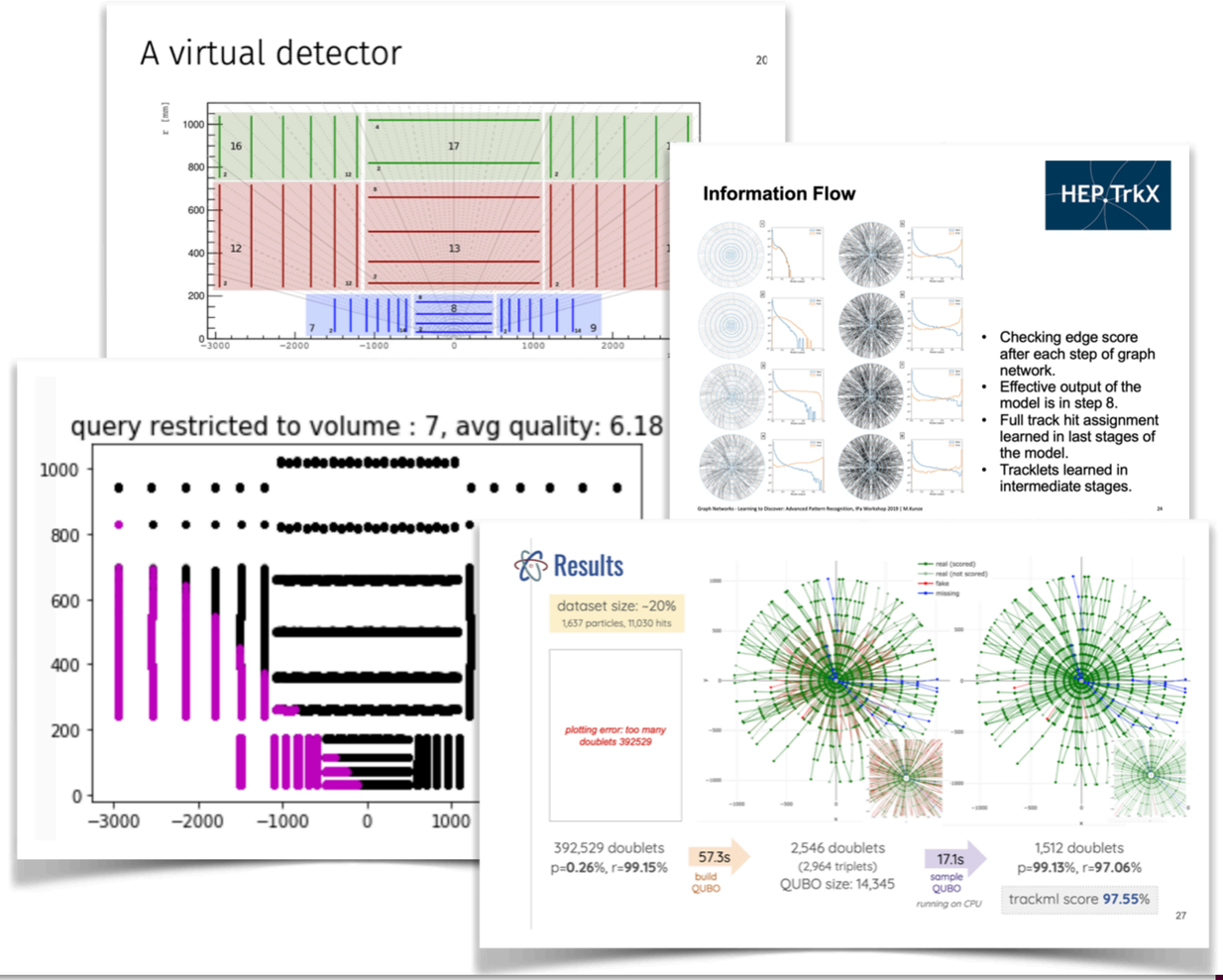
- Running data challenges is scientifically powerful
  - We have gained some experience now with HiggsML & TrackML
  - It takes significant person power to set this up
- A few musts
  - A good score that can not be tricked
  - Enough statistics to train
  - Be active and responsive before/while/after the challenge

TrackML Dataset

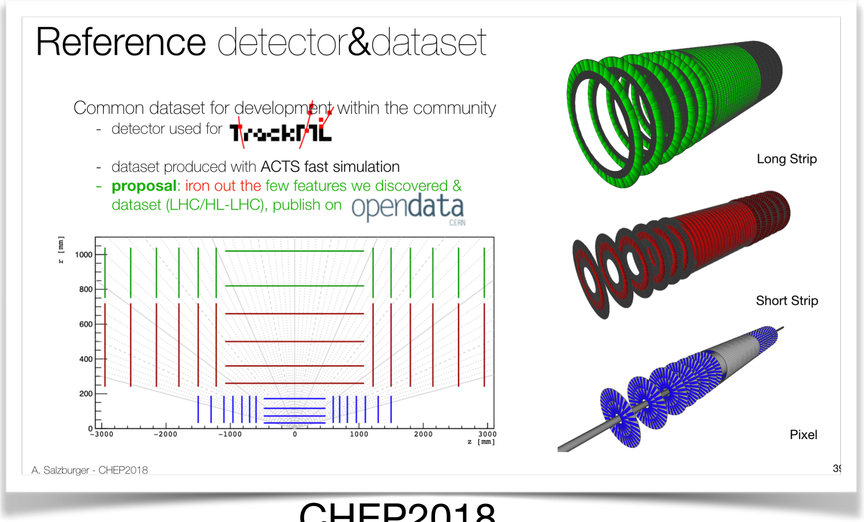
- The TrackML dataset proved to be super useful
  - now used in several areas, particularly in ML R&D in the field
  - Output format is (albeit a bit awkward) pretty established
  - however, it has it flaws



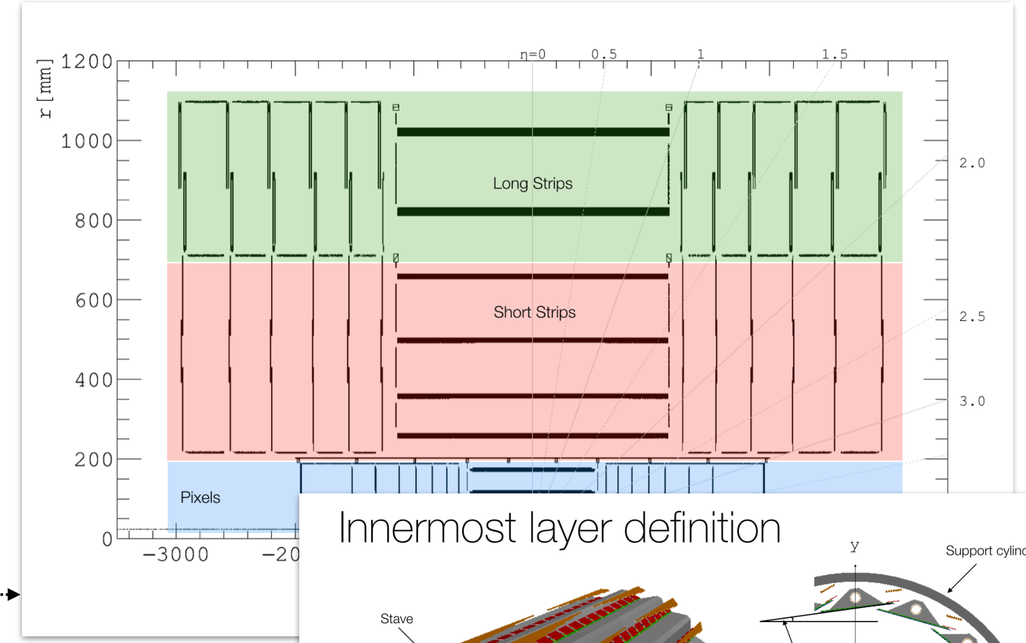
From this workshop only

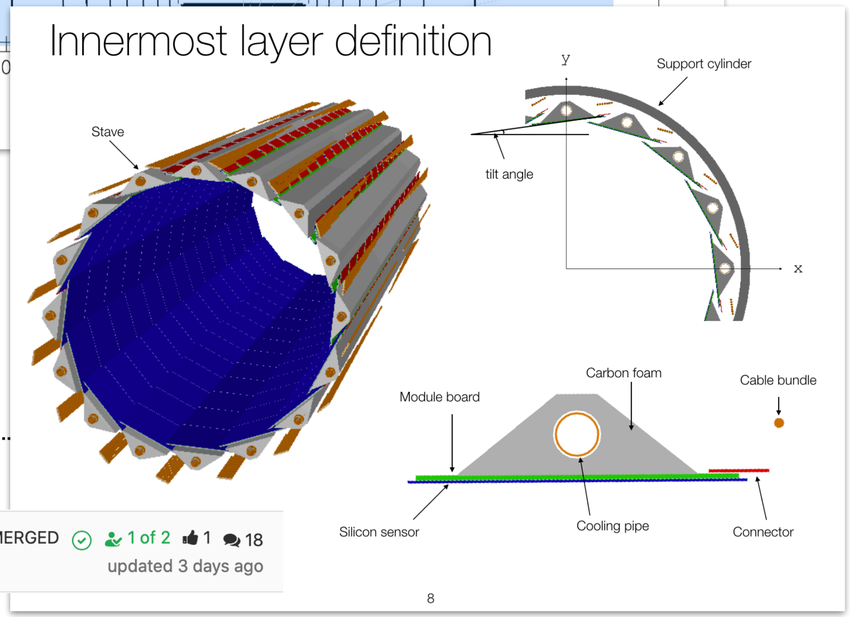


Dataset to Dataset



CHEP2018





- Merged into `acts-framework` during this workshop

Resynchronising master, add open data detector
!164 · opened 2 weeks ago by Andreas Salzburger Improvement New Feature
MERGED 1 of 2 1 18
updated 3 days ago

follow-ups: hashing

Sabrina Amrouche

- Complexity downscaling by hashing
  - Use (approximate) nearest neighbourhood to find buckets and run reconstruction in those
  - Use metric learning to find better representations

Fig 1 Learned (u/v) space

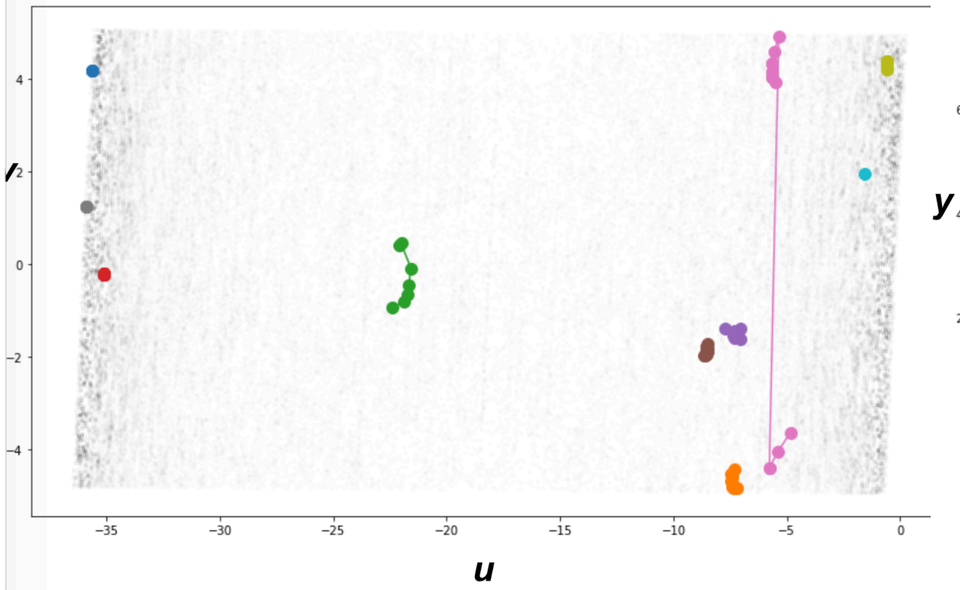
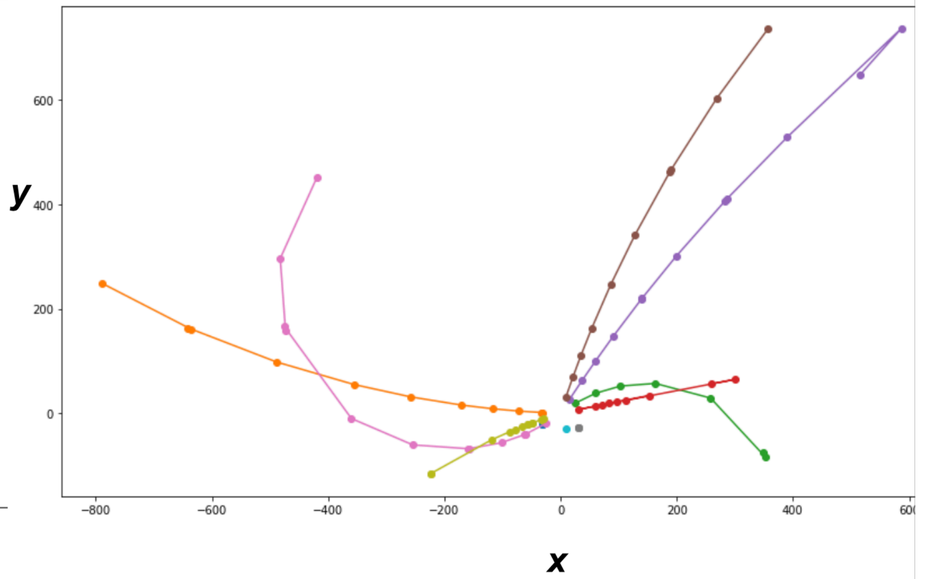


Fig 2 Transverse view (x/y)



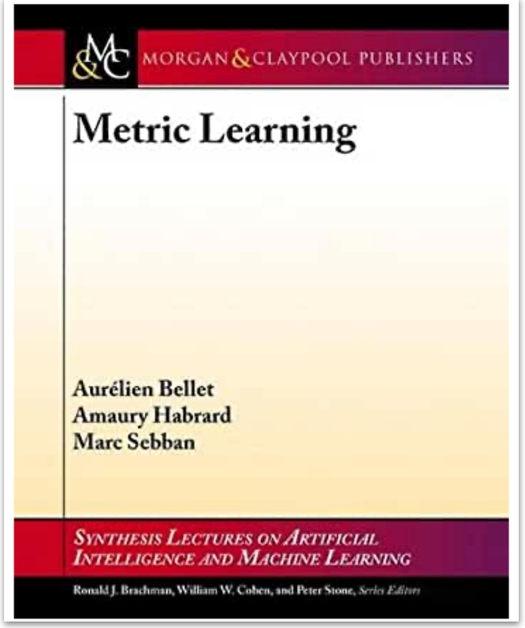
Metric learning

- Basics and details of metric learning
  - Obvious:
    in the right metric all tracks cluster

  - Some great dissuasions afterwards
    on the corridor

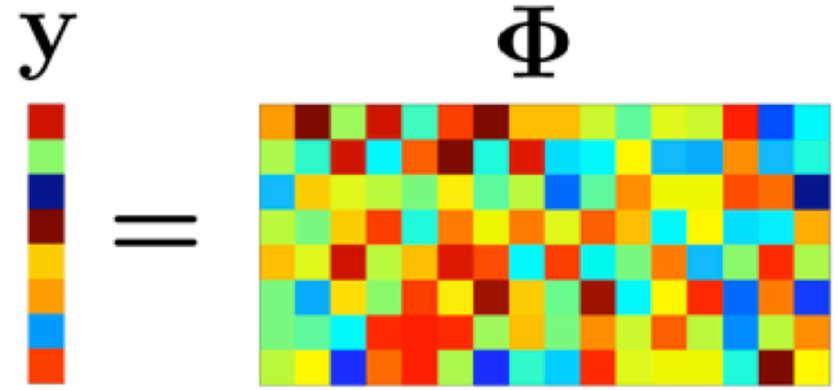
Amaury Habrard

A picture of me



Local metric learning
- different metrics in differents part/sectors of the data

TrackML/we meet random/compression matrices

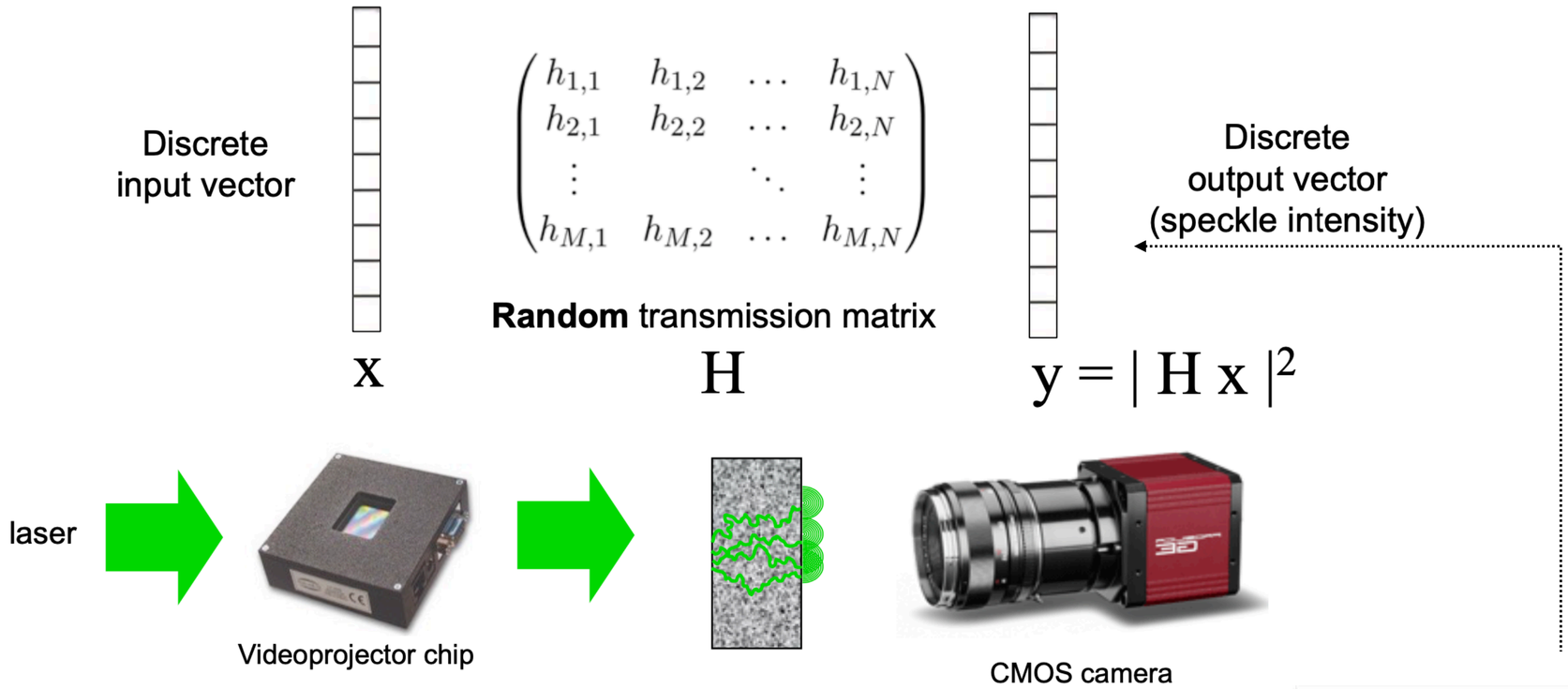


Vasily Sazonov

Slava Voloshynovskiy

A picture of me

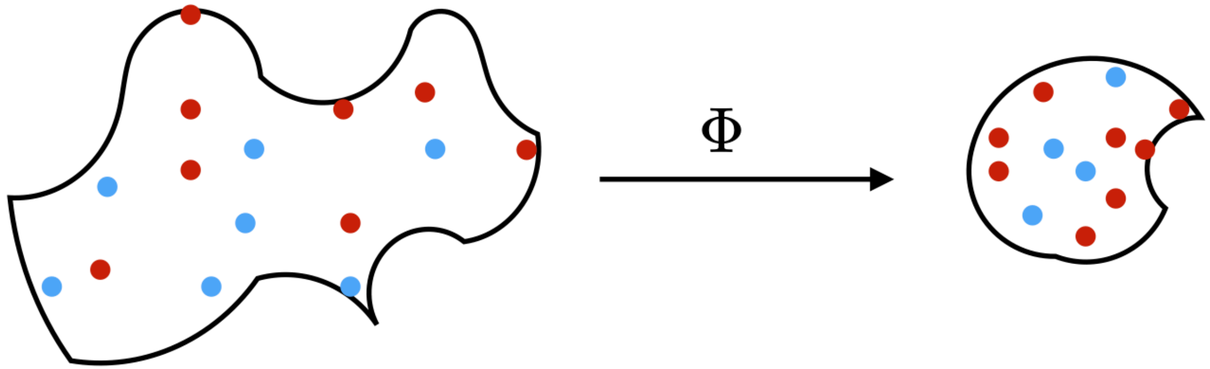
Lighton
We bring Light to AI

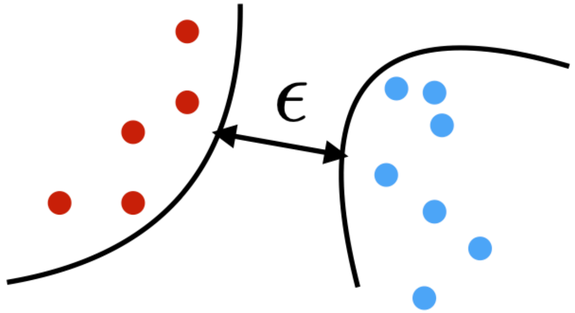

Discrete input vector
$\begin{pmatrix} h_{1,1} & h_{1,2} & \dots & h_{1,N} \\ h_{2,1} & h_{2,2} & \dots & h_{2,N} \\ \vdots & & \ddots & \vdots \\ h_{M,1} & h_{M,2} & \dots & h_{M,N} \end{pmatrix}$
Random transmission matrix
Discrete output vector (speckle intensity)
x
H
$y = | H x |^2$
laser
Videoprojector chip
CMOS camera

TrackML/we meet scattering matrices

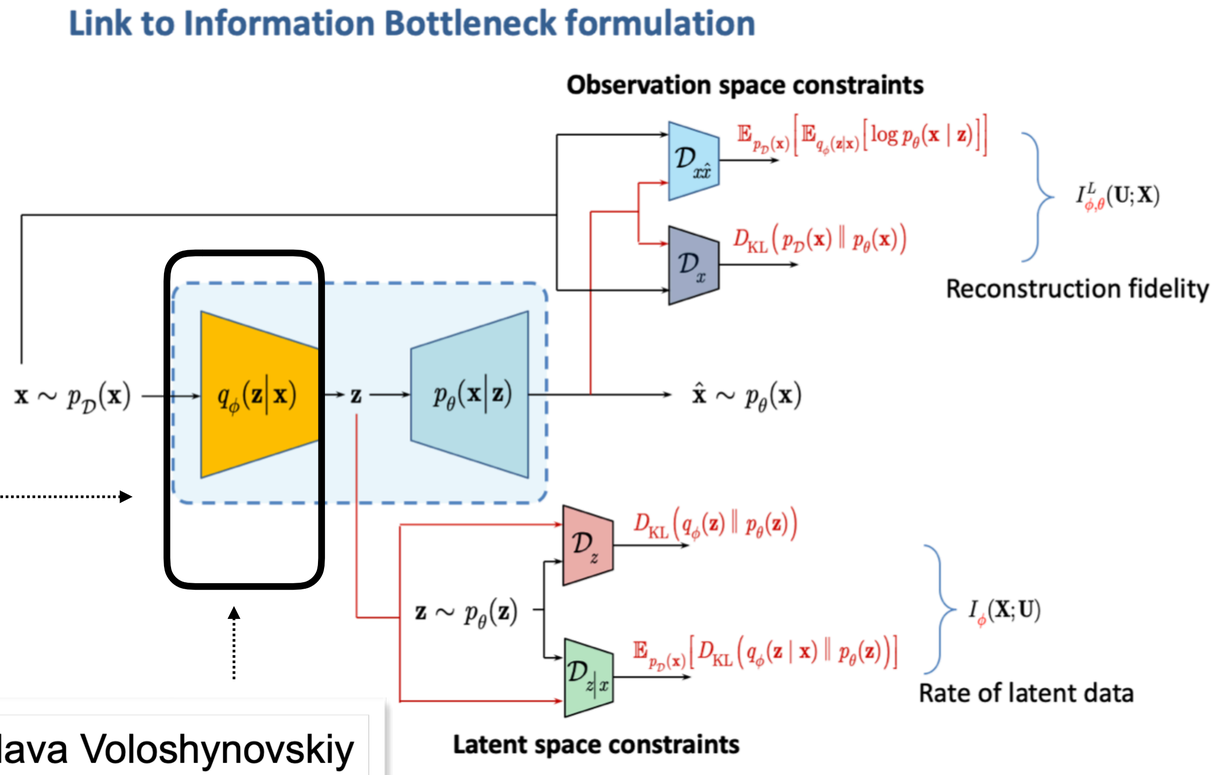
Edouard Oyallon

- Random (small) scattering matrix preserves principal distances/clustering properties
  - Central limit theorem works in our favour (and that's why tracking works)





- Doesn't have to be random matrix
  - e.g. PCA



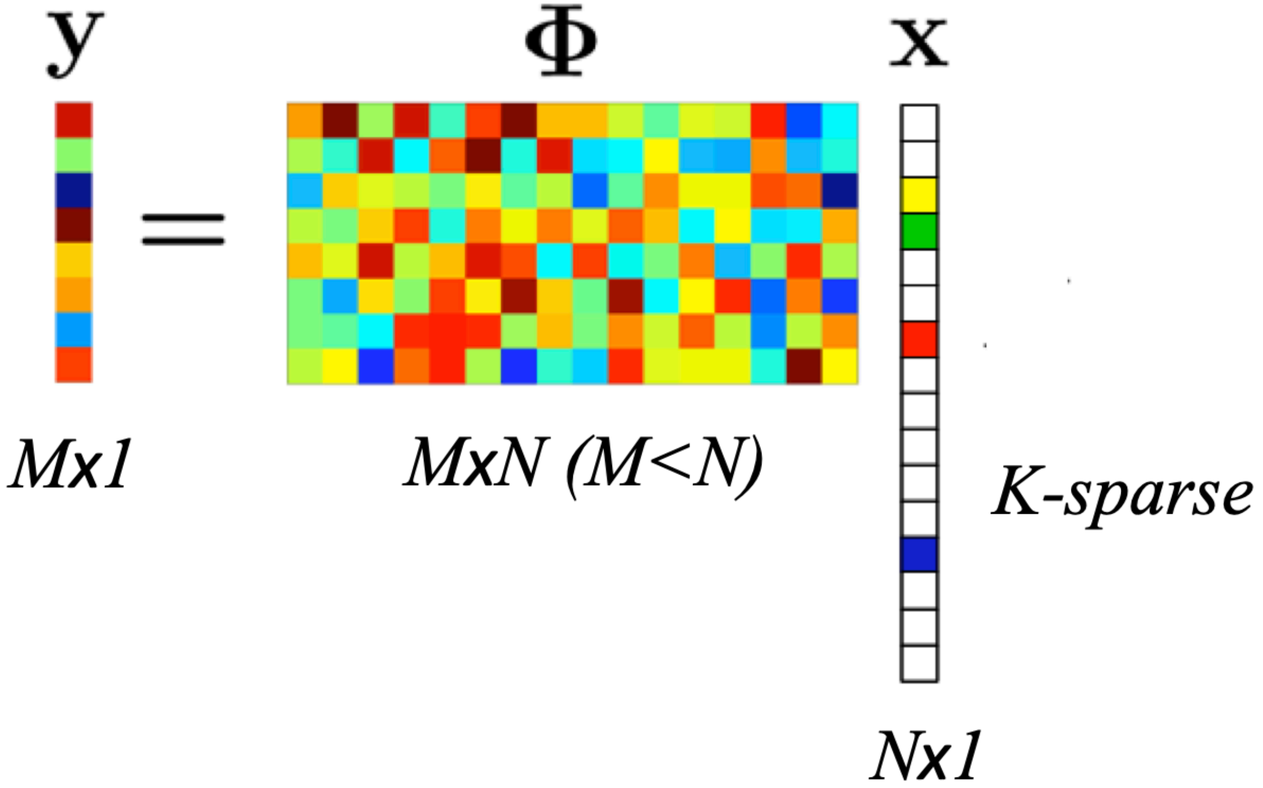
Vasily Sazonov

Slava Voloshynovskiy

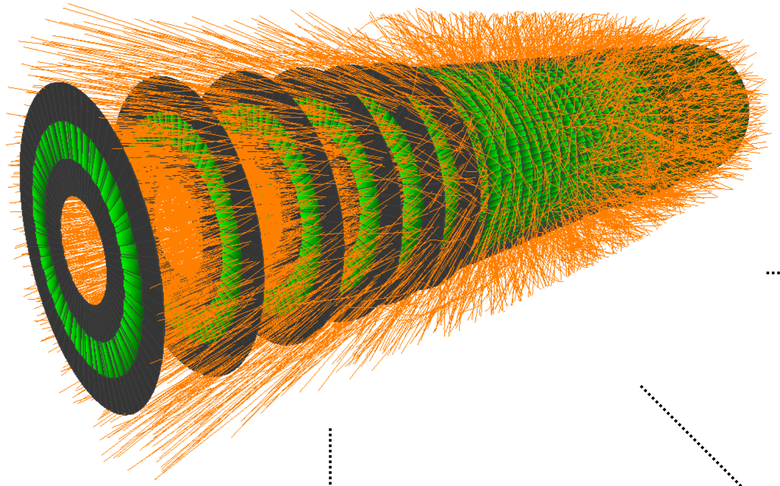
A picture of me

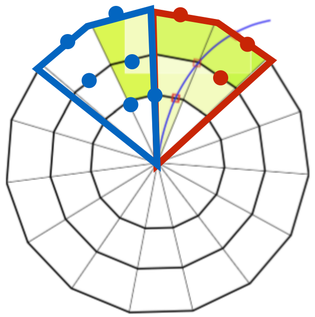
TrackML/we meet random/compression matrices

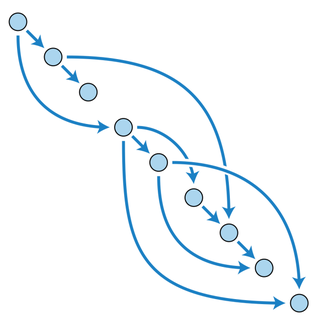
- How does this relate to HEP/track reconstruction ?
  - Aren't we already in the sparse space?

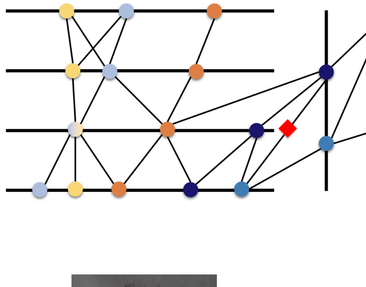


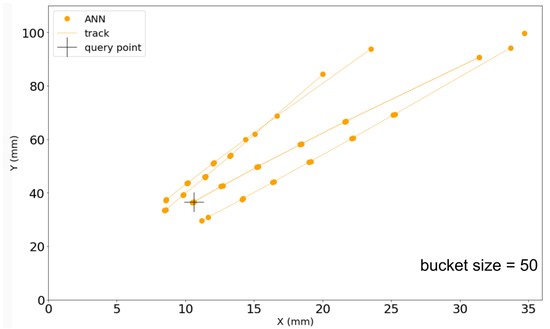
Hashing / Indexing / Graphs / Trees

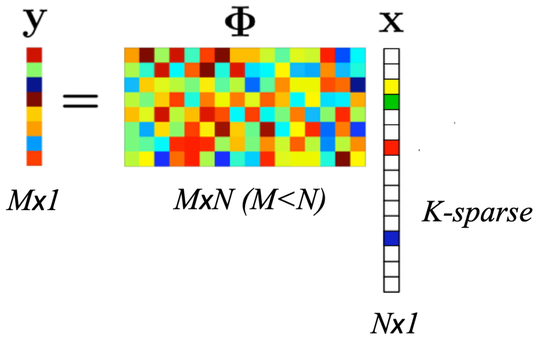




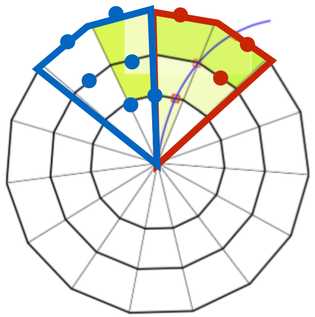


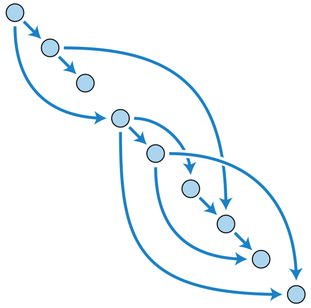


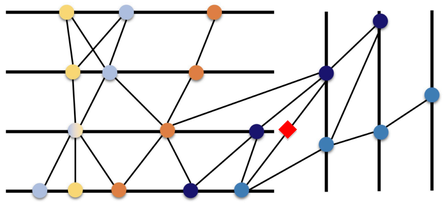


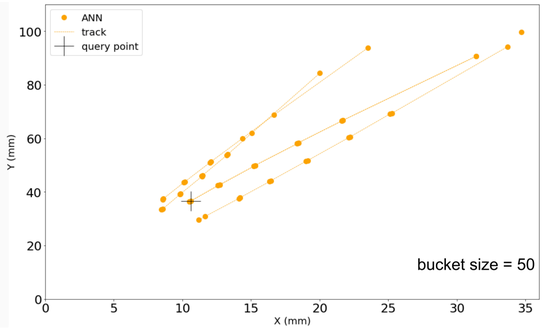


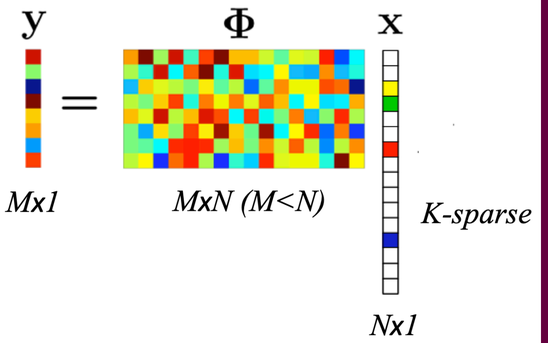
Filter / fitting / inference

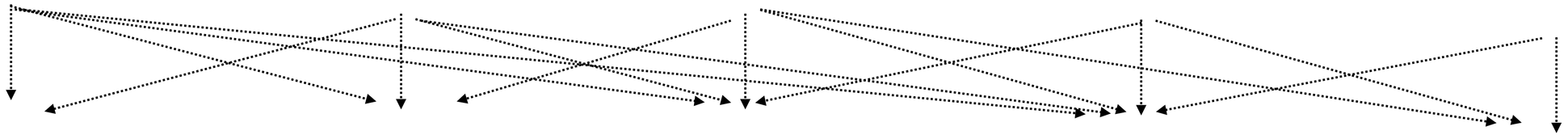


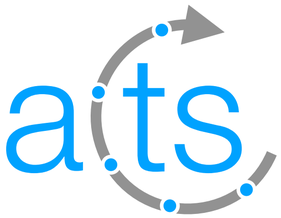


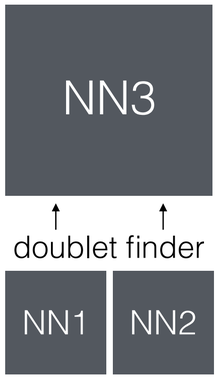


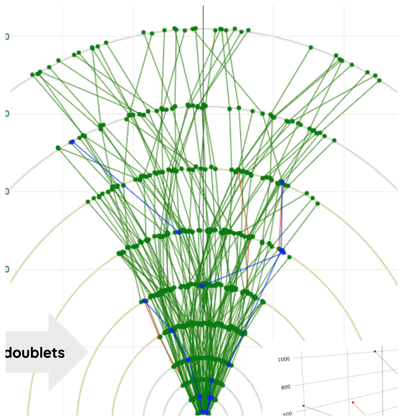


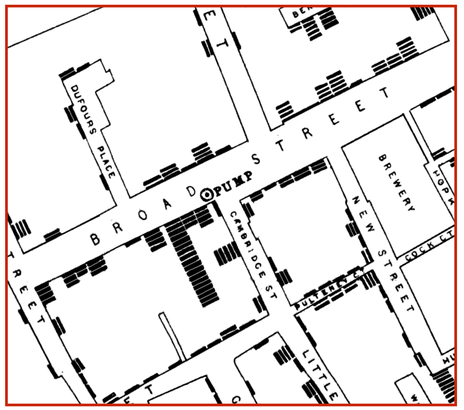












?

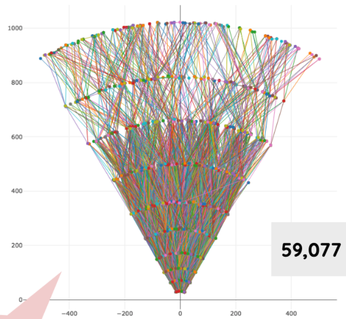
TrackML meets Quantum Computing

- Quantum computing has found its entry into HEP
  - Quantum annealing on DWave

Performances at low Pt

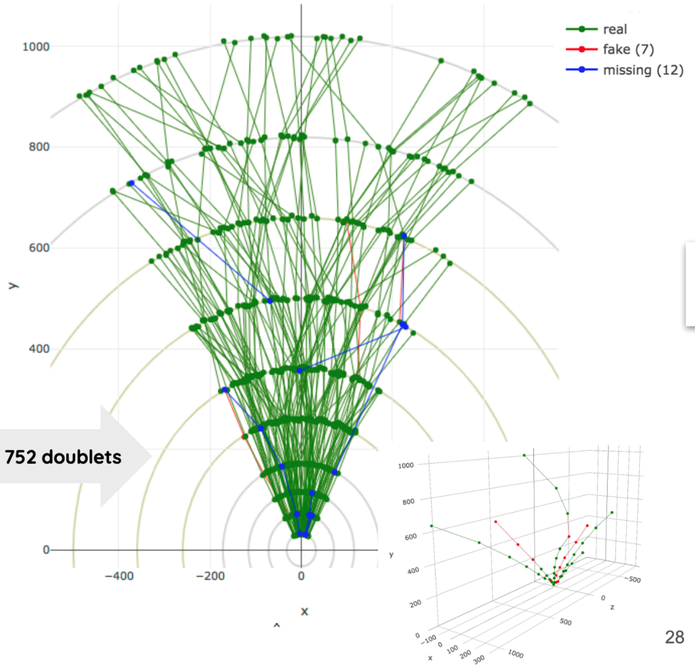
186 particles in a phi slice of π/3

precision (%): 98.5, recall (%): 98.4, trackml score (%): **98.35**



QUBO size
68,043

59,077 ⇒ 752 doublets



28

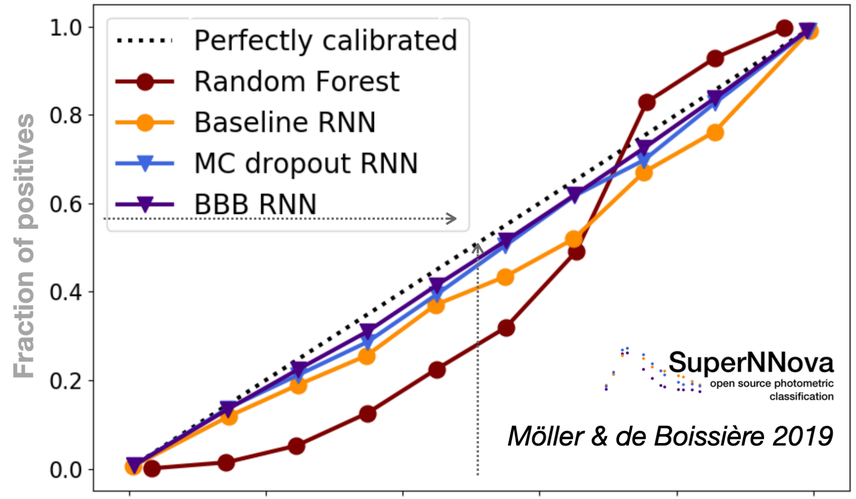
Lucy Linder

- Can we run an alignment algorithm on D-Wave ?
- Leap - if you wanna try out 1 minute of D-Wave time :-)

Finally, we reached for the stars

- We are not alone (in HEP)
  - Real time data streaming
  - Classification, augmenting, Baysian networks
  - Kaggle challenges

ML probabilities for statistical analyses?



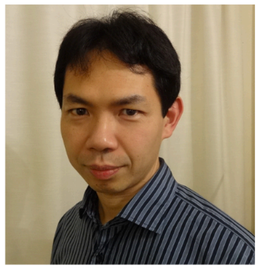
Anais Möller

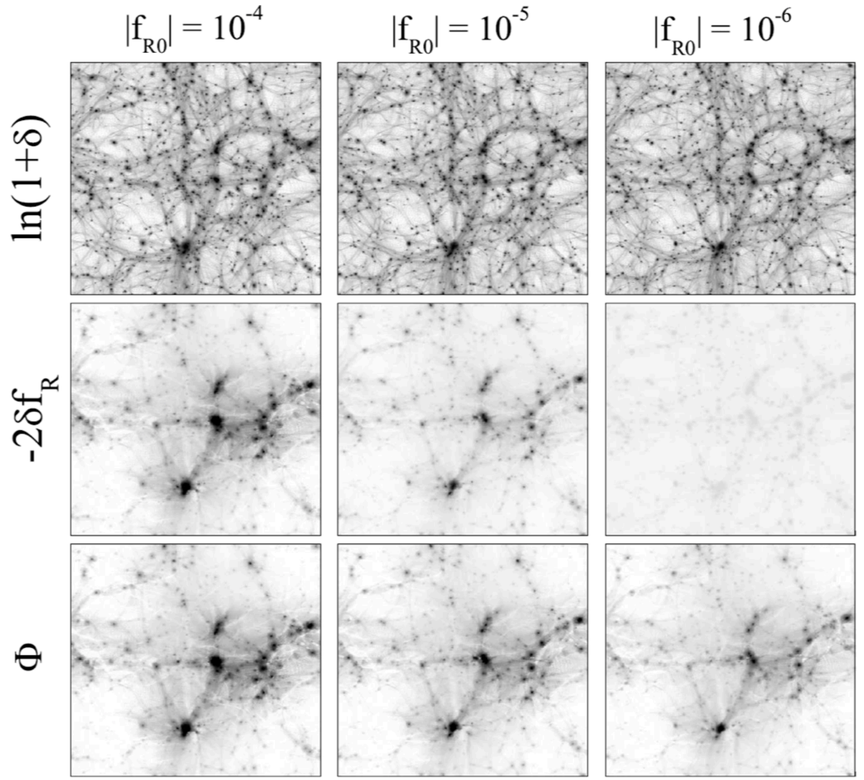
Traceability of the trained network with changing datasets
Sphynx - code commenting

Finally, we reached for the stars

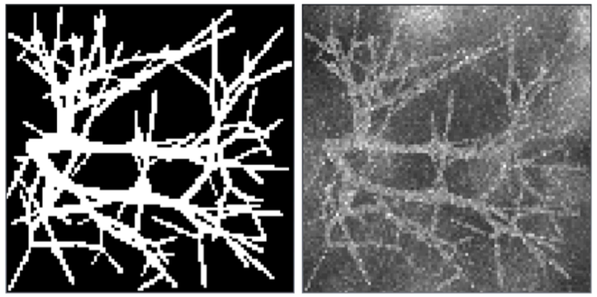
- N-body simulation for DM and galaxy building
  - Small and large scale simulation with different correlations
- GANs (as in HEP) becoming more popular
  - Application of GAN created simulations discussed

Kazuya Koyama



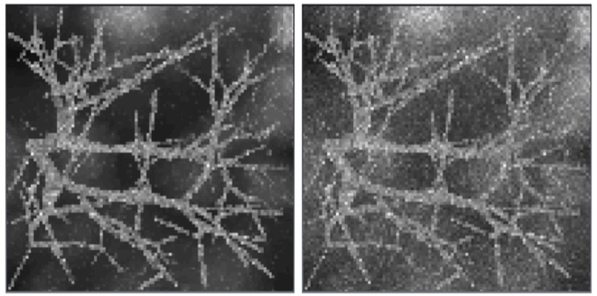


And from the stars to the brain



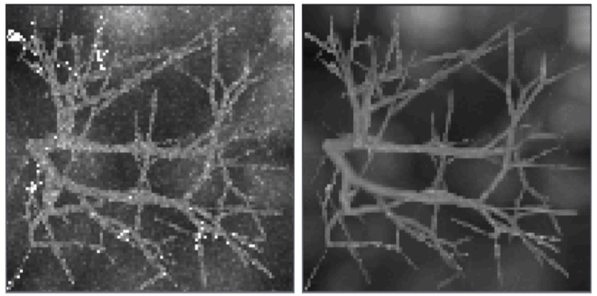
(a) Ground-truth

(b) Initial image ($g = 15$)



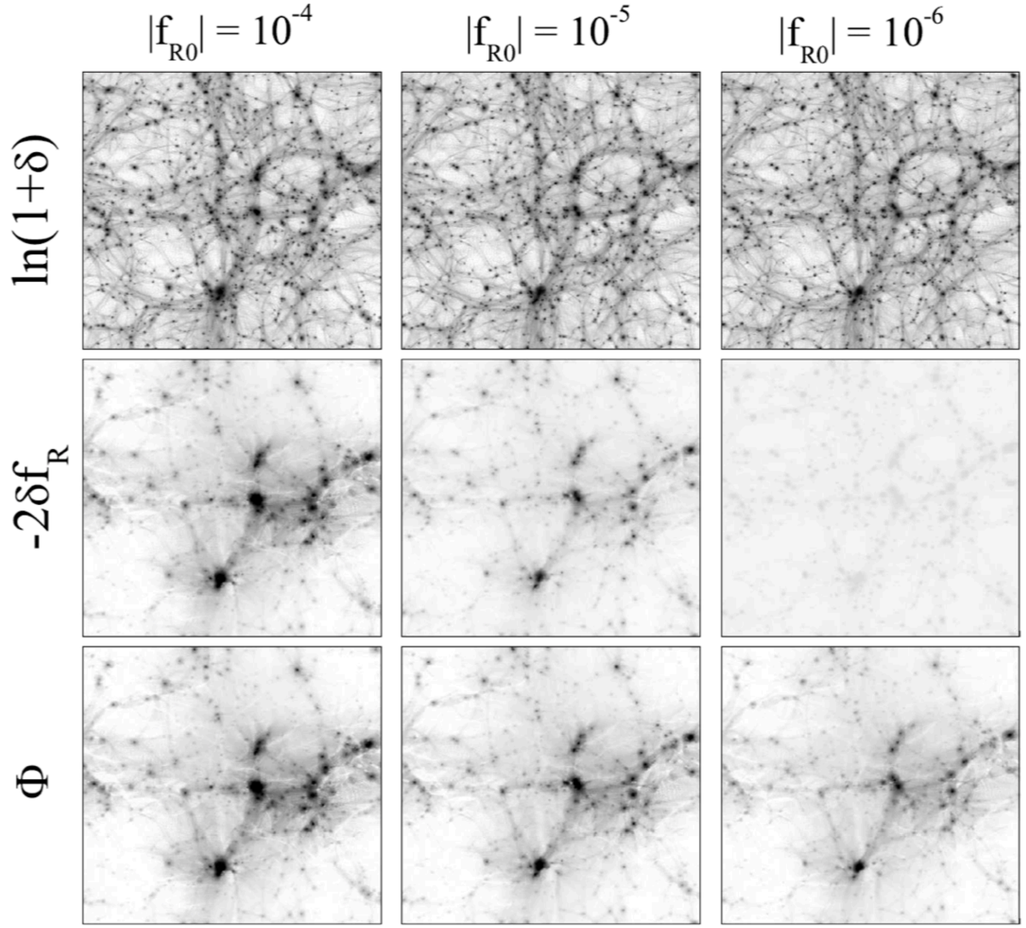
(c) Directional regularization

(d) ROF [22]



(e) HDCS [46]

(f) BM4D [40]

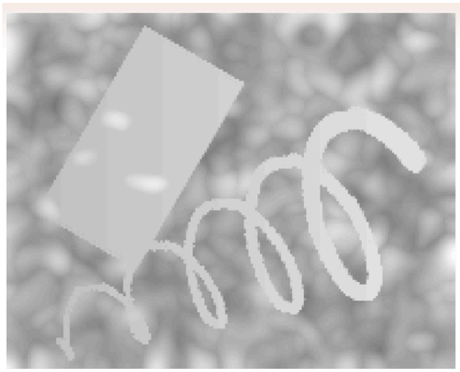


Blood and neutrinos

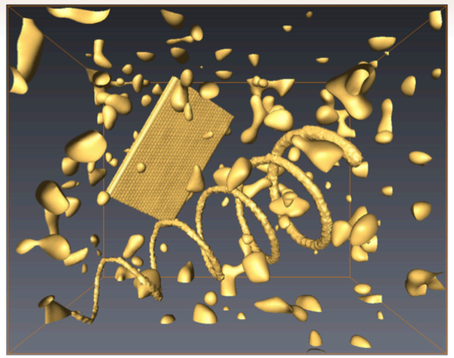
Hugues Talbot

université PARIS-SACLAY
INSTITUT PASCAL

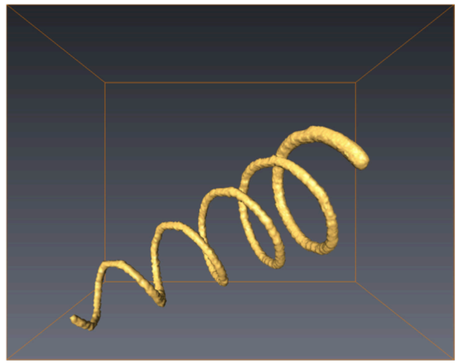
- Blood vessel finding in MRIs
  - 3D feature finding problem



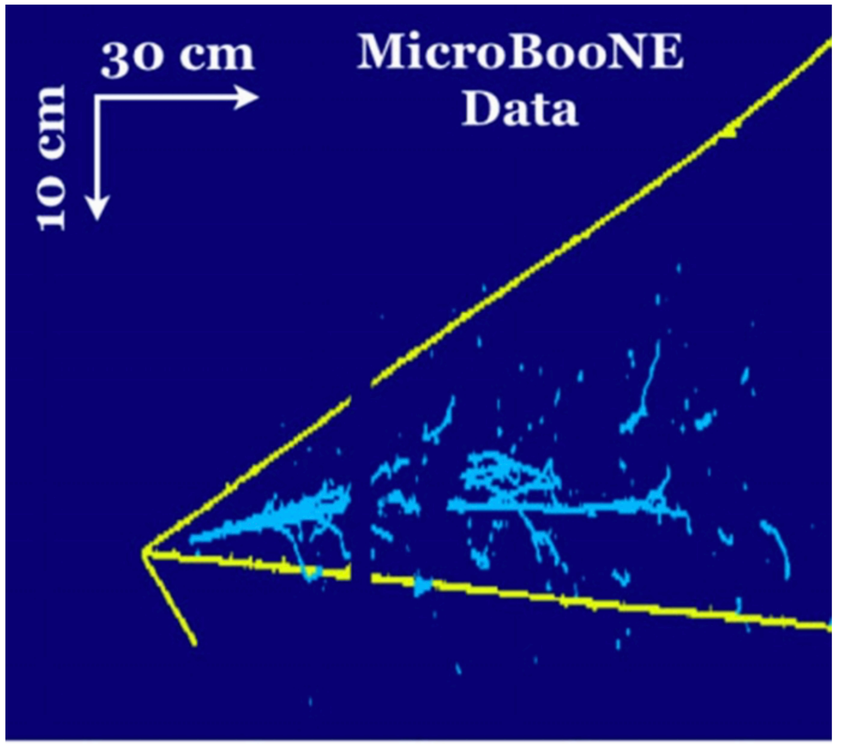
(a) CCM=0.605, Dice=0.634



(b)



(c)



And a big THANK YOU

- To the submitters of the *Learning to Discover* proposal
  - The initiators can be found [here]
- To the INSTITUT PASCAL for accepting this posposal
  - And of course for the stunning location & excellent local organisation
- To you for participating
  - And it was indeed participating not just **attending**

A warm welcome

- Learning to Discover - a series of three workshops:

1. 15-26 Jul 2019: Real time analysis workshop
2. 14-25 Oct 2019: Advanced Pattern recognition

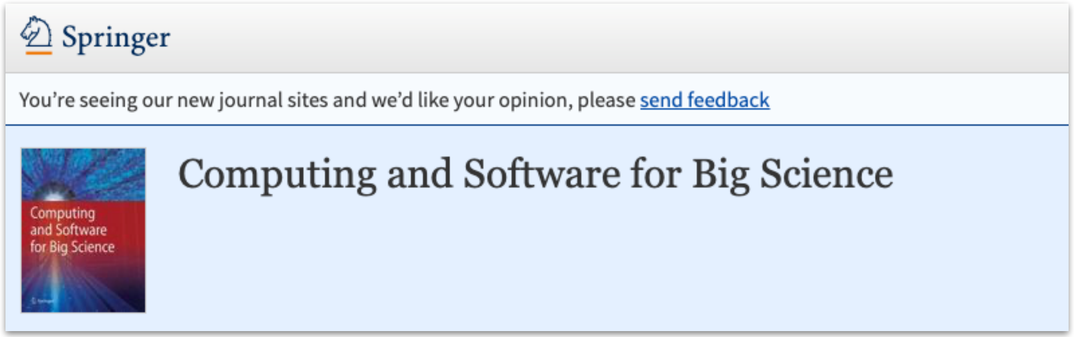
Organisation committee: Andreas Salzburger (CERN), David Rousseau (LAL Orsay), Jean-Roch Vlimant (Caltech), Cherifa Sabrina Amrouche (University of Geneva), Cecile Germain (Université Paris-Sud), Slava Voloshynovskiy (University of Geneva), Marco Rovere (CERN), Marc Schoenauer (INRIA Saclay), Paolo Calafiura (LBNL) & Sabrina Soccard (Institut Pascal)

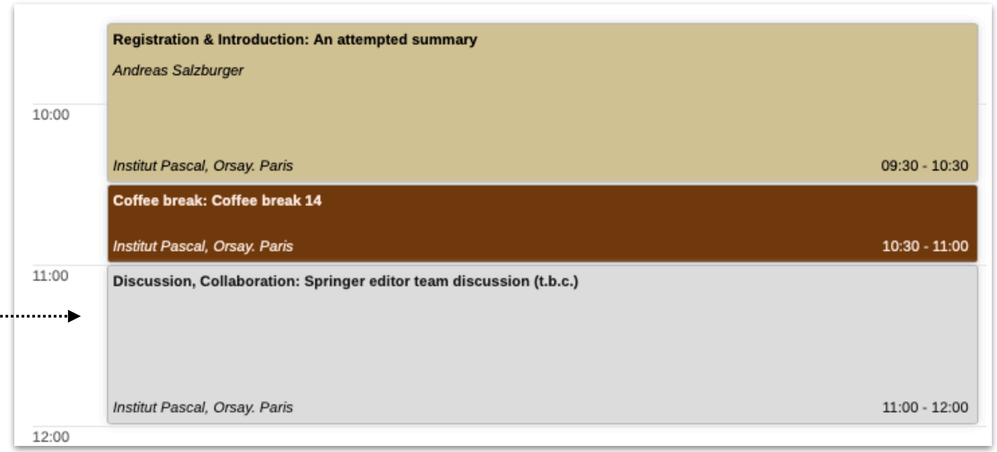
Stay tuned

3. 20-31 Jul 2020: Learning to Discover

Springer & us

- Learning, Discover & Publish
  - **Springer** offered to cover this workshop in a special issue
    - Terms are relatively free
    - We'd need to form an editorial team
    - Publishing should happen roughly 6 months after the workshop
    - **I will contact all the contributors individually**





Eventual discussion on Friday 25/10

Last words for today

- I hope you enjoyed *Learning to Discover*
  - I personally learned a lot
  - And discovered some interesting paths up to the Saclay plateau

Save travels!

After Slava's talk (and what he showed about superresoluation) I will never laugh about these TV scenes anymore

https://www.youtube.com/watch?v=Vxq9yj2pVWk