The Common Tracking Software Forum

Frank Gaede, Benedikt Hegner, Markus Elsing

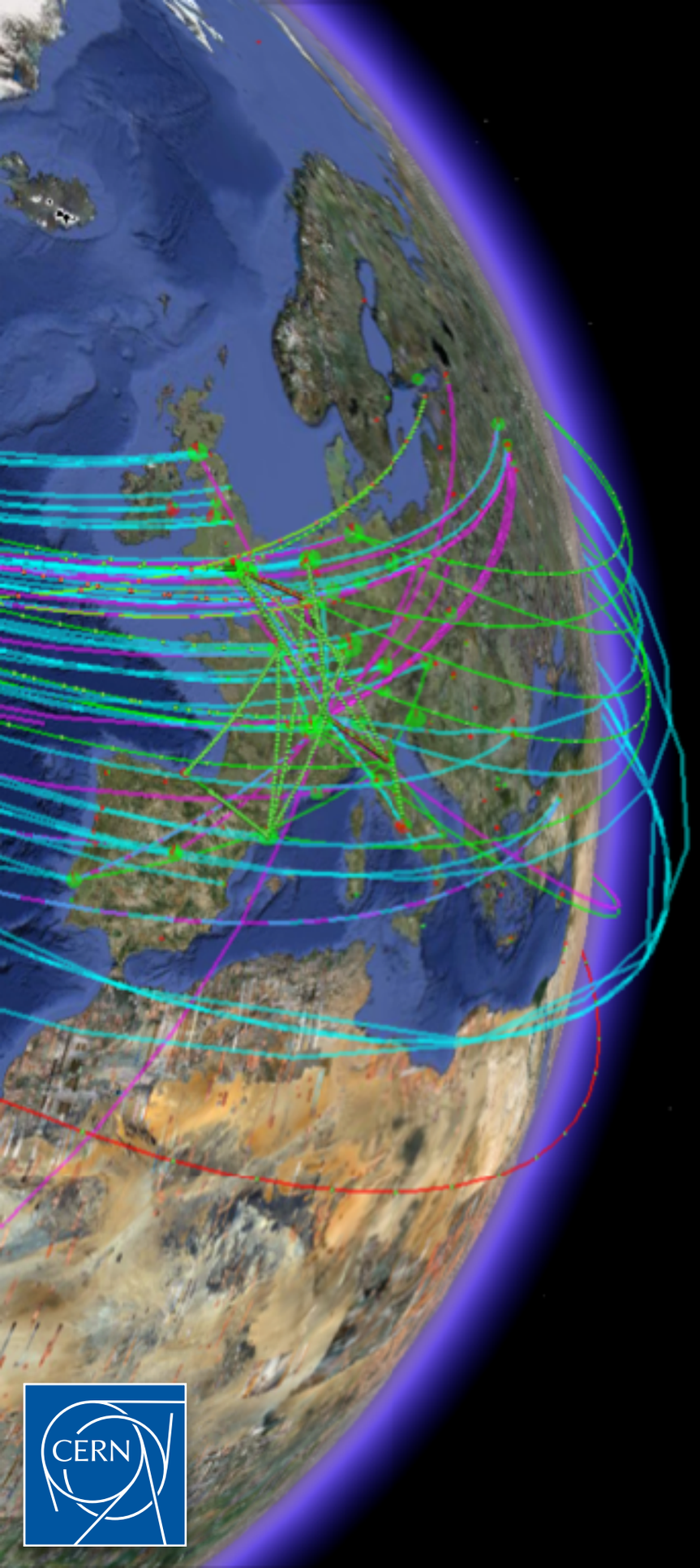
A "forum" across experiments to
discuss and promote the development
of (common) tracking software

Goals of the Common Tracking Forum

- provide a more regular forum to discuss developments in HEP tracking software, to
  - ➡ enable exchange of software ideas and concepts
  - ➡ share best practices
  - ➡ facilitate code re-usage
  - ➡ support possibly emerging common tracking software projects
    - including software for novel pattern recognition techniques

- created formally under the umbrella of HSF
  - ➡ webpage: http://hepsoftwarefoundation.org/
  - ➡ indico: https://indico.cern.ch/category/5816/

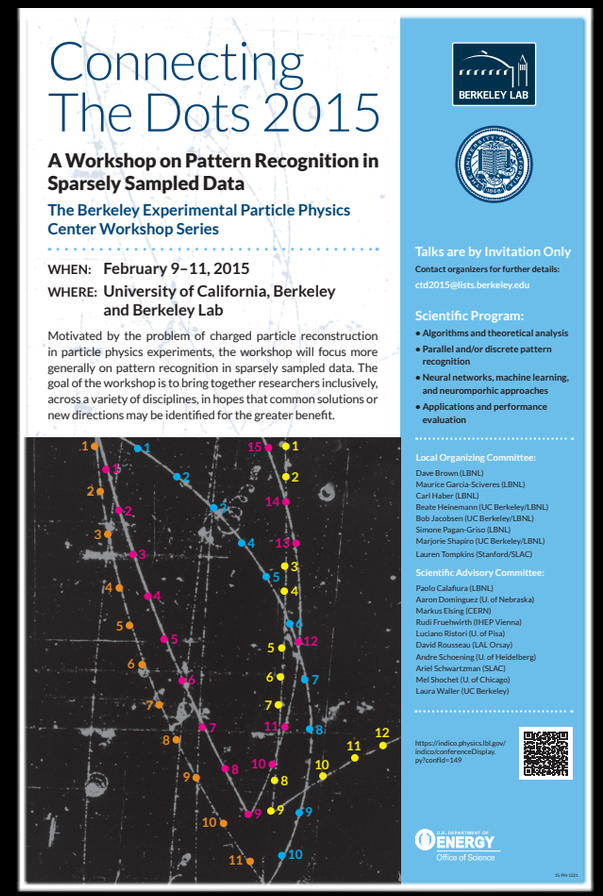
- aiming for monthly meetings
  - ➡ this Wednesday we'll have the 3rd meeting
  - ➡ possibly adjust frequency in future (if low activity)
  - ➡ mailing lists:
    - Detector-Technology-Pattern-Recognition@cern.ch (CTD mailing list)
    - hep-sf-reconstruction@googlegroups.com (HSF mailing list)

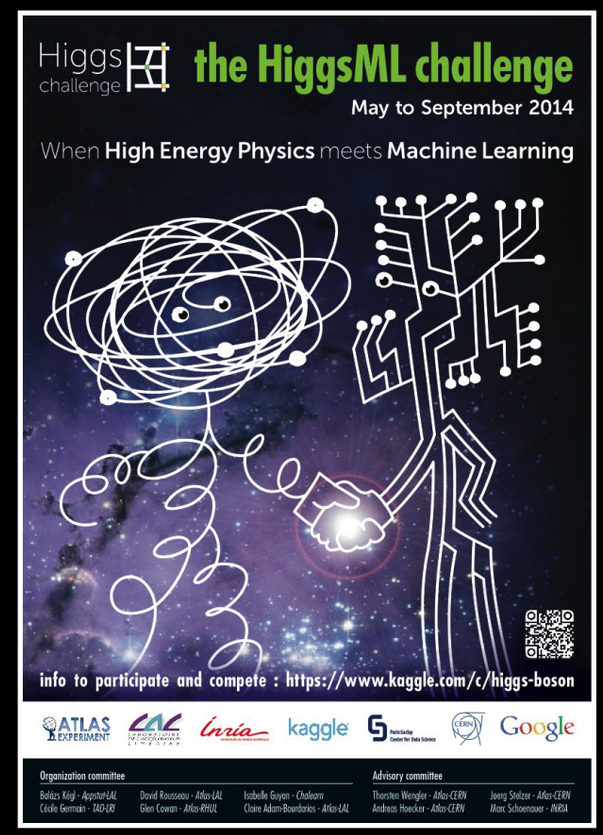


Major Tracking Workshops and Schools ?

Connecting the Dots Workshops

- dedicated to pattern recognition techniques
  - ➡ first workshop in Berkeley, next in Vienna in Feb.2016
- first workshop well received by community
  - ➡ 55 participants across all LHC/BeIle-II/Future Collider experiments, as well contributions from theory and even non-HEP
  - ➡ 2nd workshop this week in here in Vienna
- 4 main workshop subjects
  - ➡ mathematical algorithms and theoretical analysis
  - ➡ parallel and/or discrete pattern recognition techniques
  - ➡ neural networks, machine learning, neuromorphic approach
  - ➡ applications and performance evaluation of existing applications
- follow up initiative(s)
  - ➡ e..g. work towards a machine learning tracking challenge for Kaggle

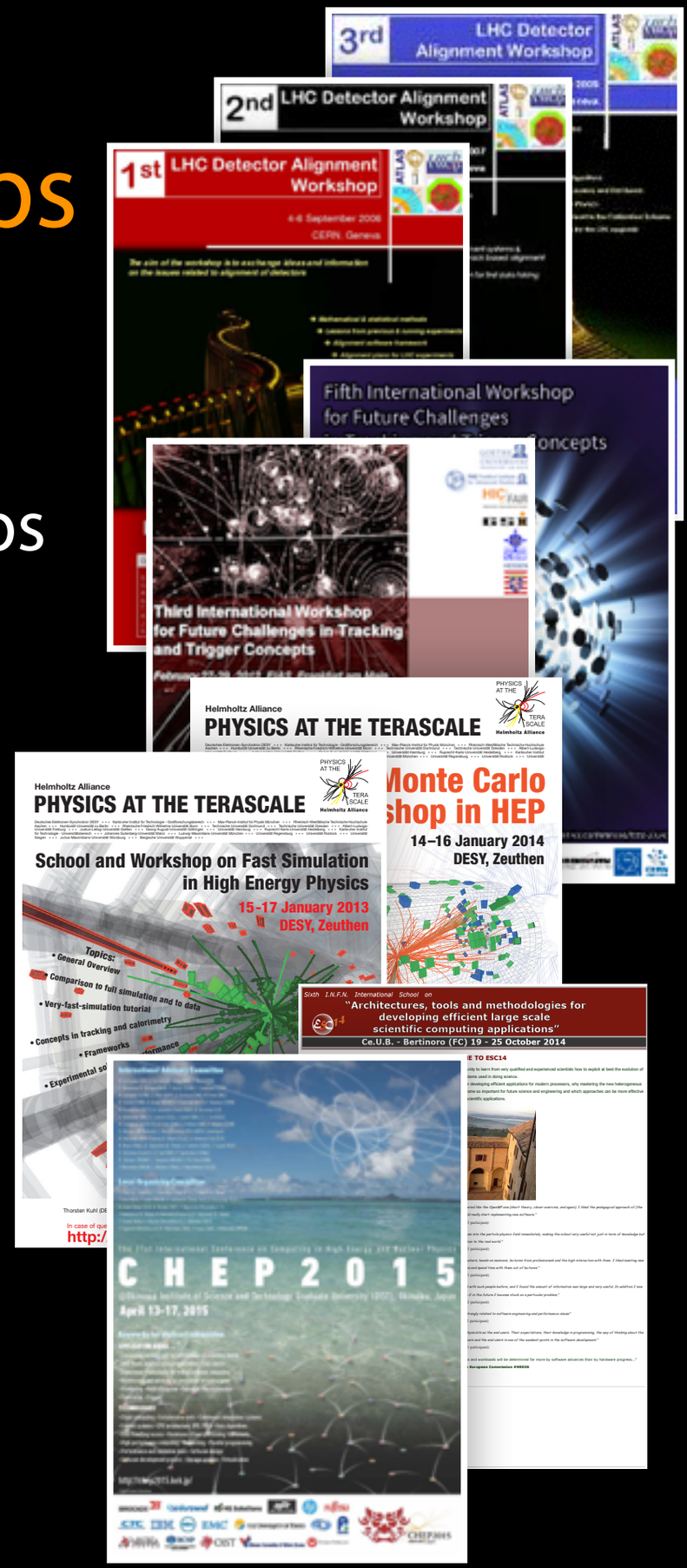




Other Tracking related Workshops

- successful LHC alignment workshops
  - ➡ alignment algorithms and strategies in 2006-2008
- GSI/FAIR future tracking and trigger workshops
  - ➡ served as well as broader forum for algorithm discussion
    - recently focus shifting towards FairRoot
- related fast simulation workshops
  - ➡ fast simulation engines reuse tracking codes...

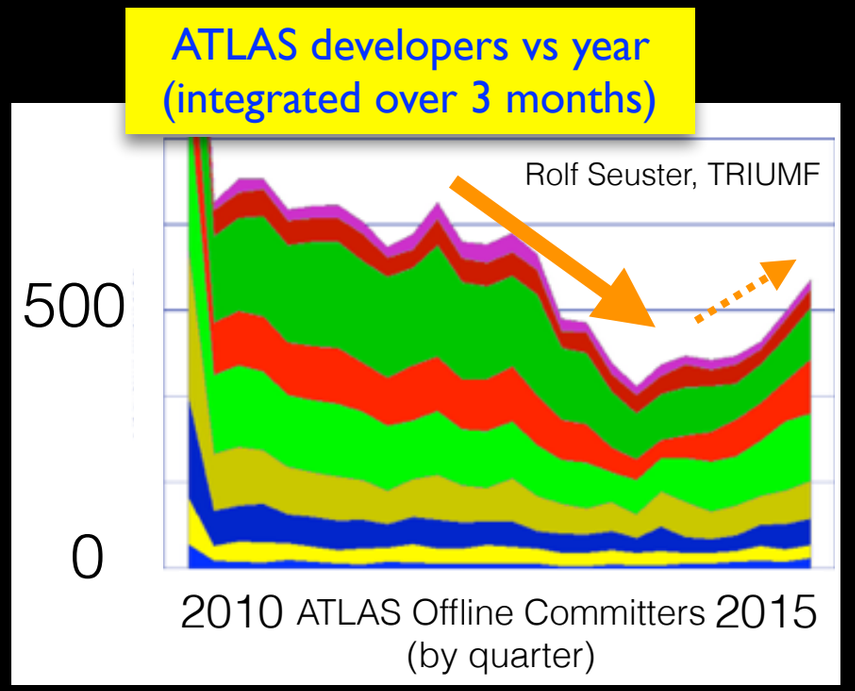
- examples for more general conferences:
  - ➡ ACAT and of course CHEP
  - ➡ Vertex - silicon hardware oriented with some software talks
- examples for more general schools:
  - ➡ CERN schools of computing
  - ➡ others like the ESC INFN computing schools

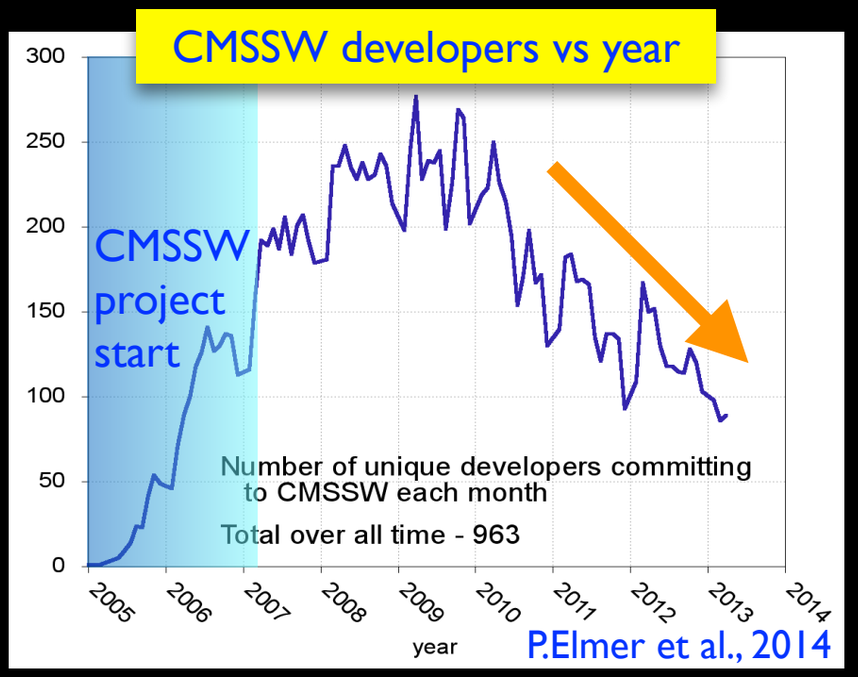


... list is not complete, but short ...

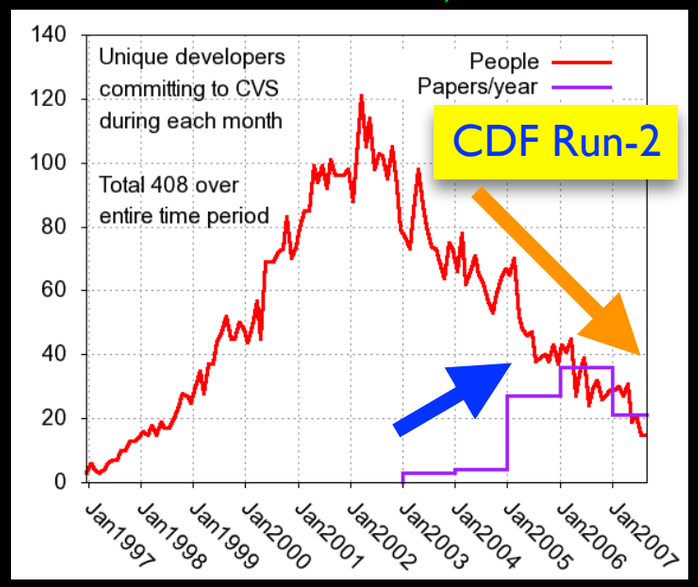
Software and Manpower

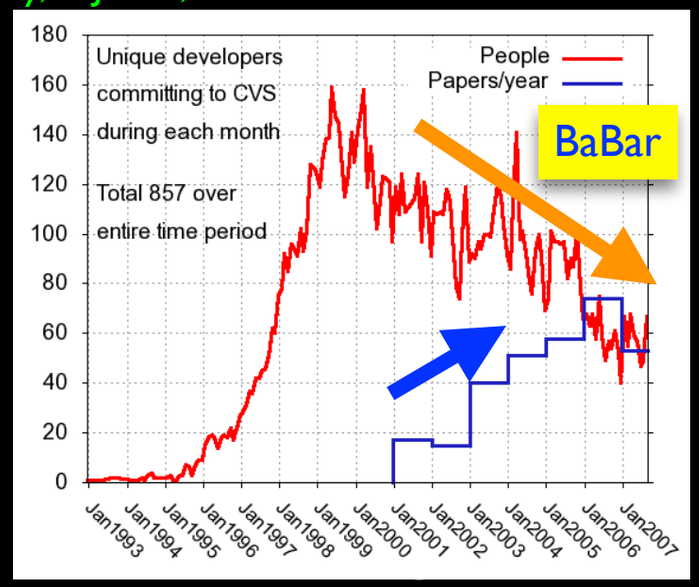
- software follows a natural life cycle
  - ➡ building up the software for an experiment
  - ➡ start of experiment operations and data taking
  - ➡ data analysis and detector upgrades
- loss of software manpower at LHC
  - ➡ (mostly) students and postdocs moved on to do physics
  - ➡ need to attract and train new people to face challenges
- tracking for new/future HEP experiments
  - ➡ similar tracking manpower and training needs
  - ➡ can we re-use existing solutions ?

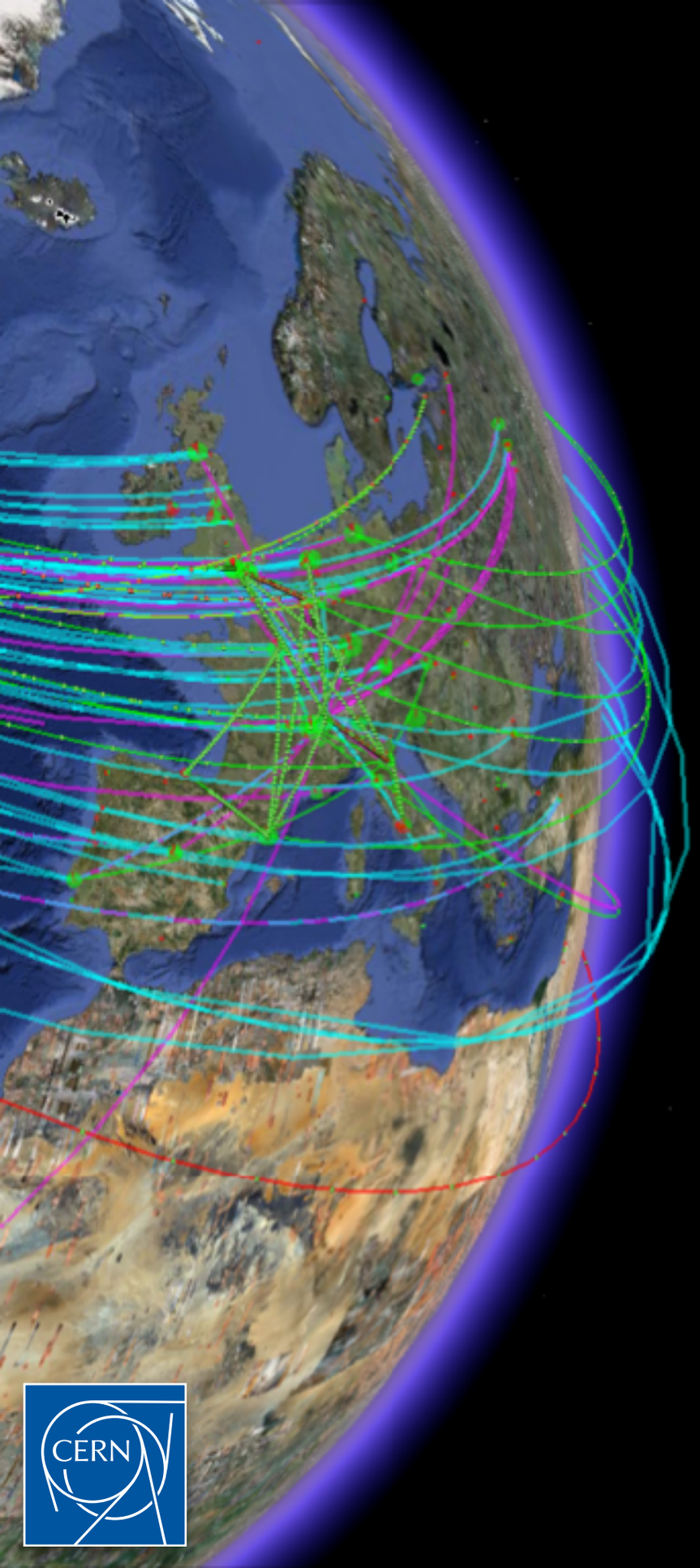




P.Elmer, L.Sexton-Kennedy, C.Jones, ICHEP 2007



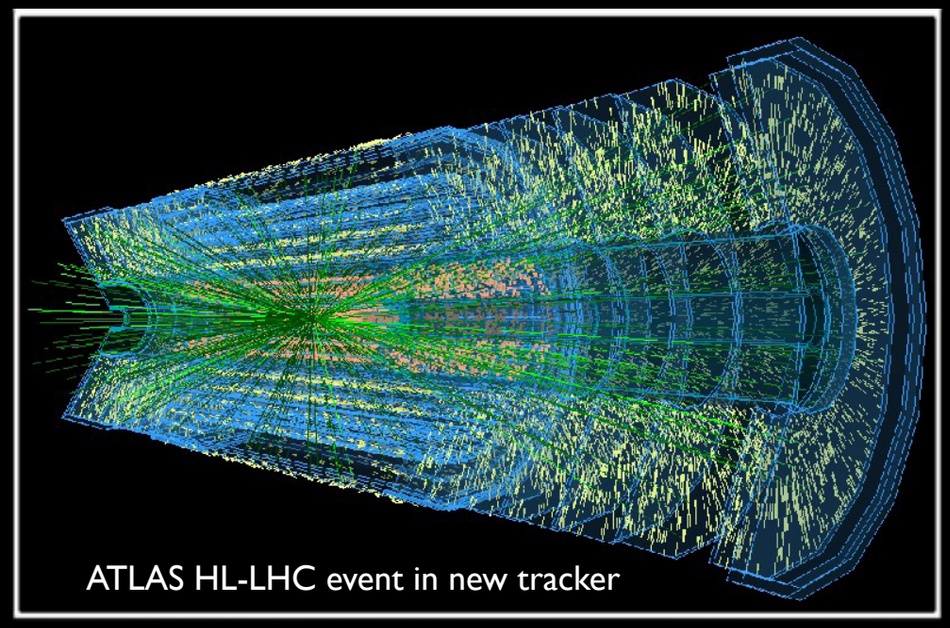




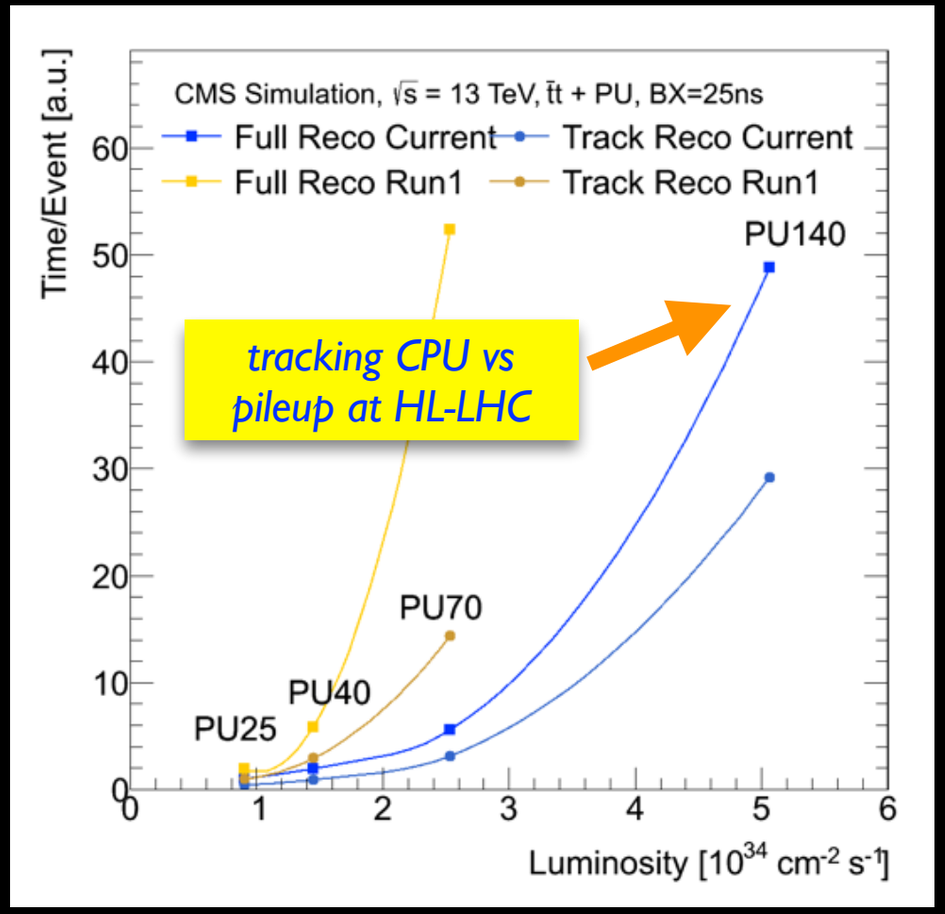
The Challenges we face

The Experiments' Tracking Software Challenges

- ATLAS/CMS - million dollar question:
  - ➡ how to reconstruct HL-LHC events with 200 pileup
  - ➡ how to keep the physics performance up
  - ➡ and do it within the computing resources we'll have...
- tracking is reconstruction CPU driver
  - ➡ not new, we knew this would be the problem
  - ➡ will aim to improve on already highly optimised code

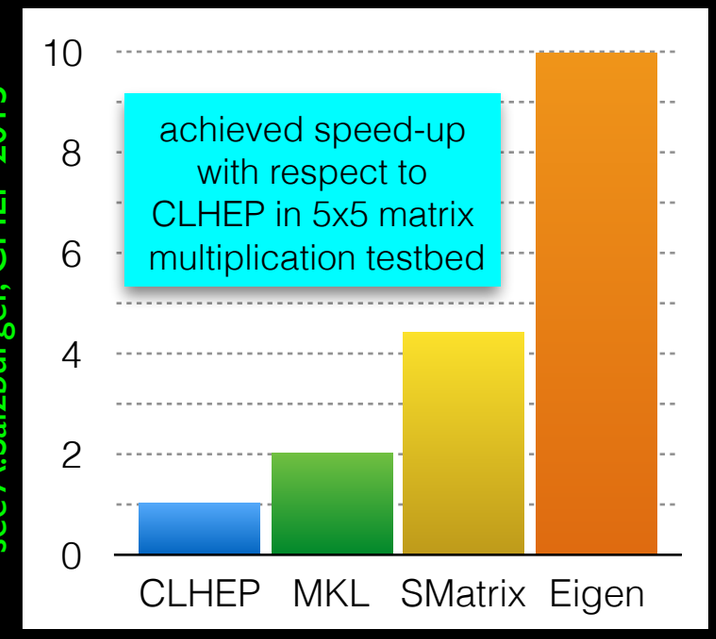

ATLAS HL-LHC event in new tracker

- LHCb and ALICE trigger-less readout
  - ➡ processing/filtering done in online trigger farms
  - ➡ offline quality reconstruction online to achieve needed data reductions
- Belle-II is about to start data taking
  - ➡ raw data volumes comparable to LHC
- Future Collider studies (ILC, CLIC, FCC)



Lessons from Tracking Work for LHC Run-2

- ATLAS and CMS focus on technology and strategy to improve CURRENT algorithms
  - ➡ improve software technology, including:
    - simplify EDM design to be less OO ("hip" 10 years ago)
    - ATLAS migrated to Eigen - faster vector+matrix algebra
      (CMS was already using SMatrix)
    - vectorised trigonometric functions
      (CMS: VDT or ATLAS: intel math lib)
    - work on CPU hot spots
      (e.g. ATLAS replaced F90 by C++ for B-field service)
  - ➡ tune reconstruction strategy (very similar in ATLAS and CMS):
    - optimise iterative track finding strategy for 40 pileup
    - ATLAS modified track seeding to explore 4th Pixel layer
    - CMS added cluster-shape filter against out-of-time pileup

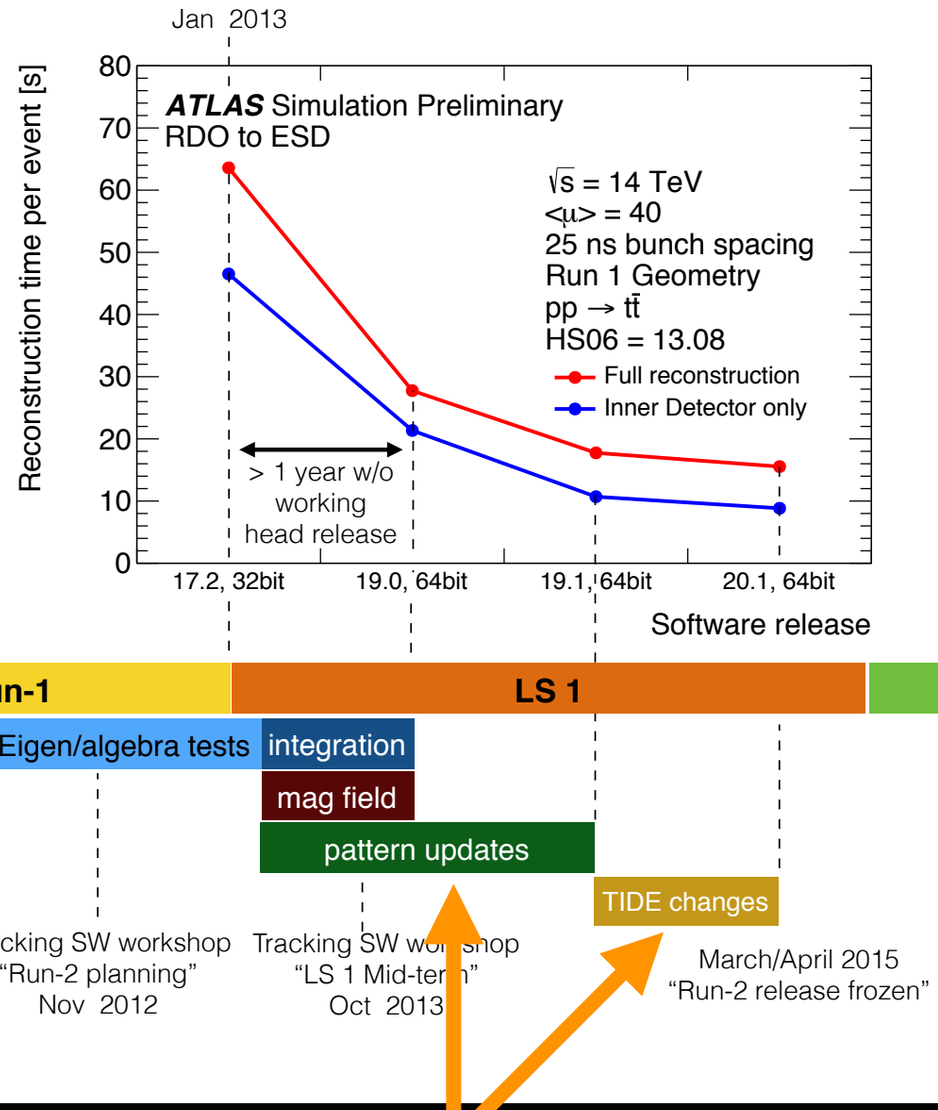


- hence, mix of SIMD and algorithm tuning
  - ➡ huge program in ATLAS, more than 1000 packages fixed
  - ➡ CMS made their tracking as well thread-safe

Huge Improvements e.g. in ATLAS

A.Salzburger, CHEP 2015

The factor 4 - a planning & deployment exercise

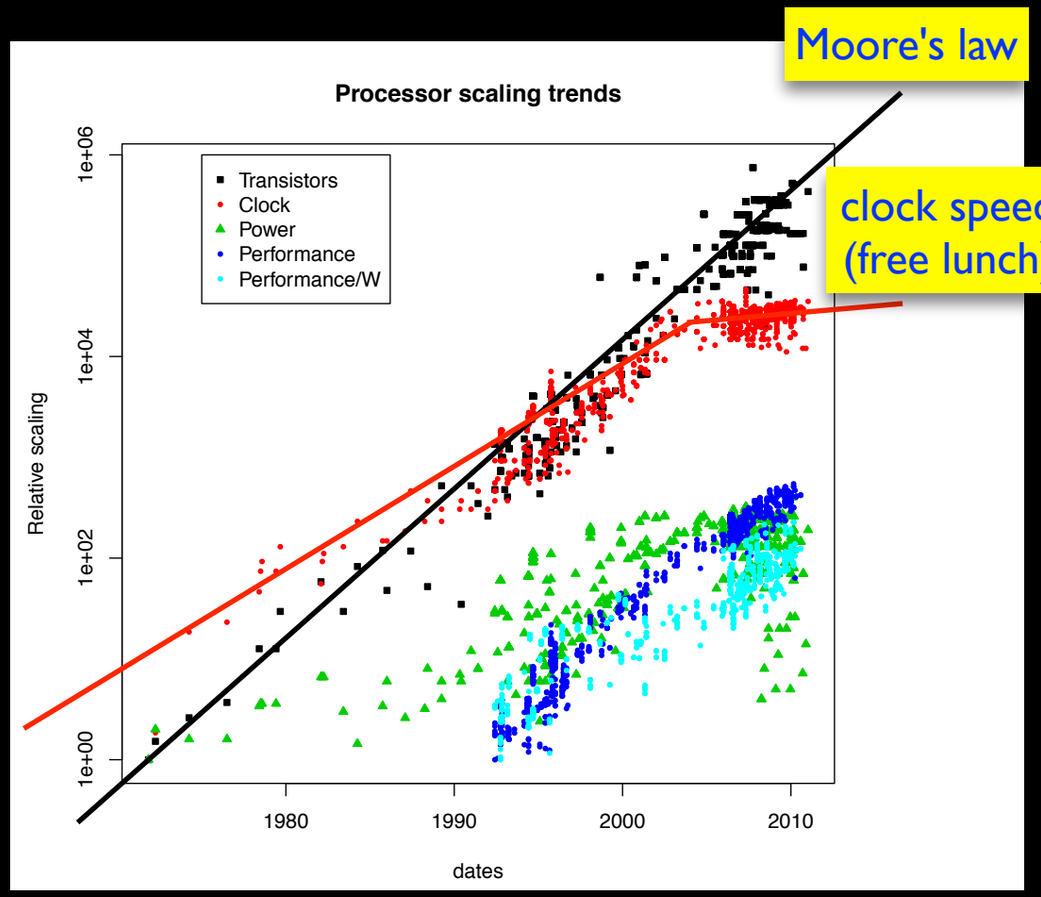


- biggest gain is in algorithms
  - ➡ similar gains achieved by CMS

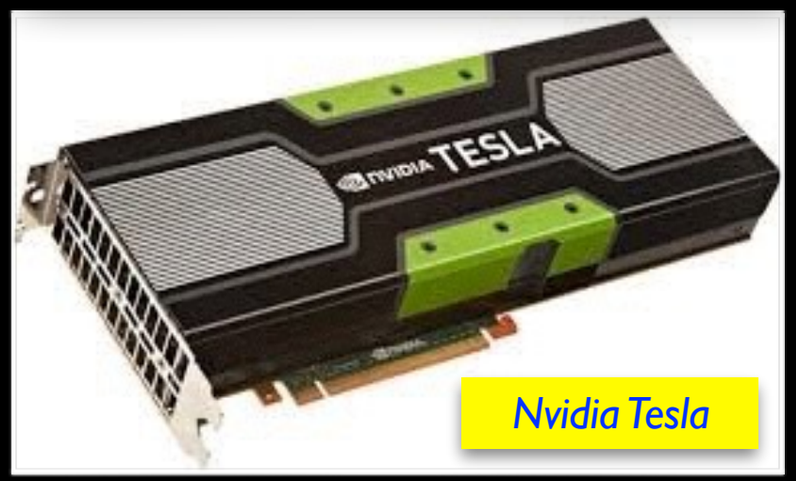
Technology Challenges

- Moore's law is still alive
  - ➡ number of transistors still doubles every 2 years
    - no free lunch, clock speed no longer increasing
  - ➡ lots of transistors looking for something to do:
    - vector registers
    - out of order execution
    - hyper threading
    - multiple cores
  - ➡ many-core processors, including GPGPUs
    - lots of cores with less memory
  - ➡ increase theoretical performance of processors

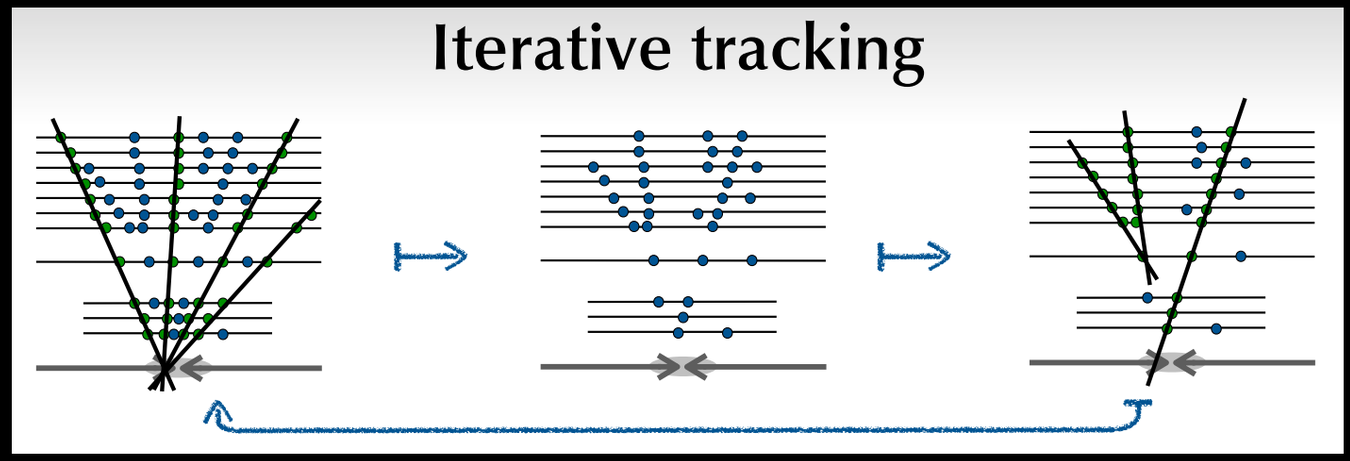
- challenge will be to adapt HEP software
  - ➡ hard to exploit theoretical processor performance
    - many of our algorithm strategies are sequential
  - ➡ need to parallelise applications (multi-threading)
    (GAUDI-HIVE and CMSSW multi-threading a step in this direction)
    - change memory model for objects, more vectorisation, ...
    - link to HSF Concurrency Forum



see G.Stewart, CHEP 2015



Massively parallel Tracking ?



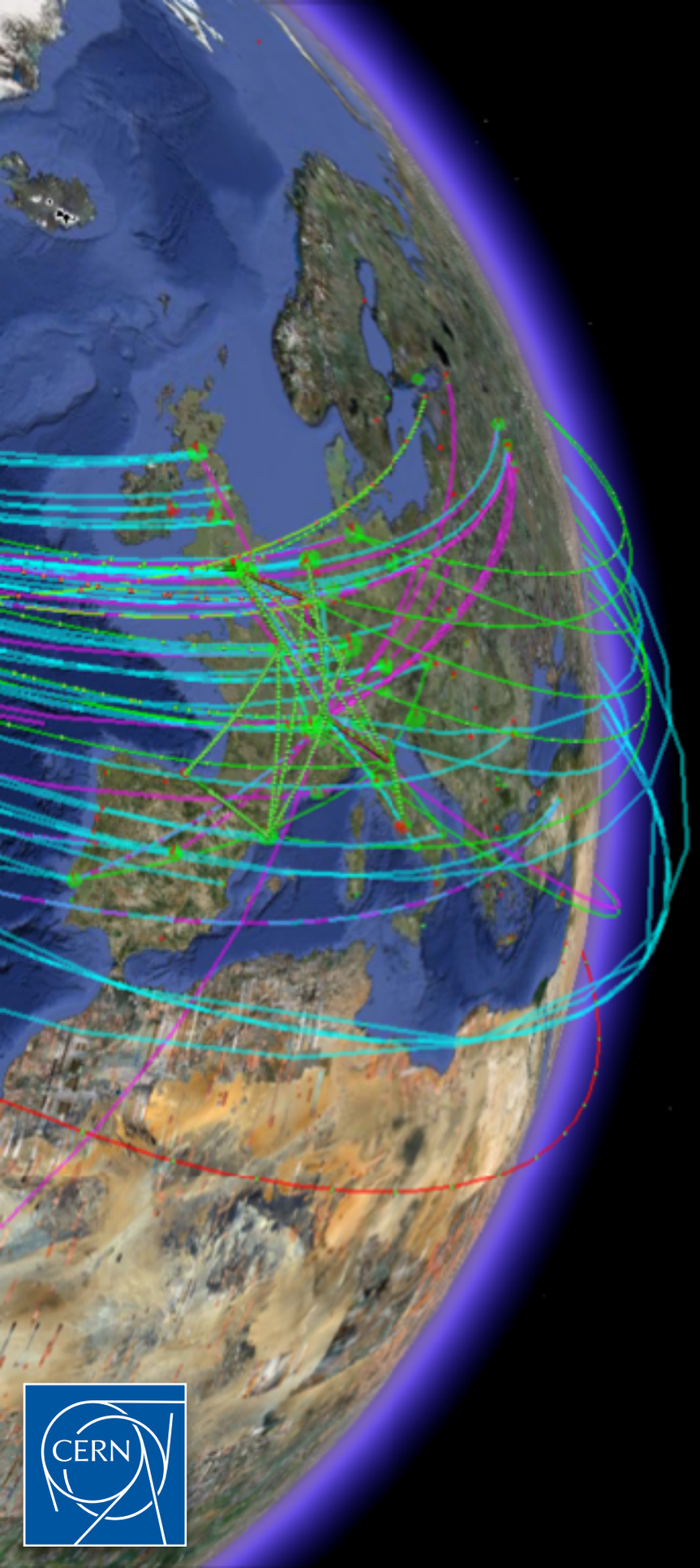
- nearly all tracking strategies today are for early rejection
  - ➡ iterative tracking: avoid combinatorial overhead as much as possible !
    - early rejection requires strategic candidate processing and hit removal
  - ➡ not a heavily parallel approach, it is a SEQUENTIAL approach !

- implications for making it massively parallel ?
  - ➡ Amdahl's law at work:

$$\text{Time}_{||} = \text{Para} / N + \text{Seq}$$

  - ➡ iterative tracking: small parallel part Para, heavy on sequential Seq
    - hence, if we want to gain by a large N threads, we need to reduce Seq

- hence we need to re-think the algorithmic strategy
  - ➡ having concurrency in mind from the very start
    - connection to Concurrency Forum
  - ➡ novel ideas like deep learning part of the solution ?
    - see presentations this week at CTD

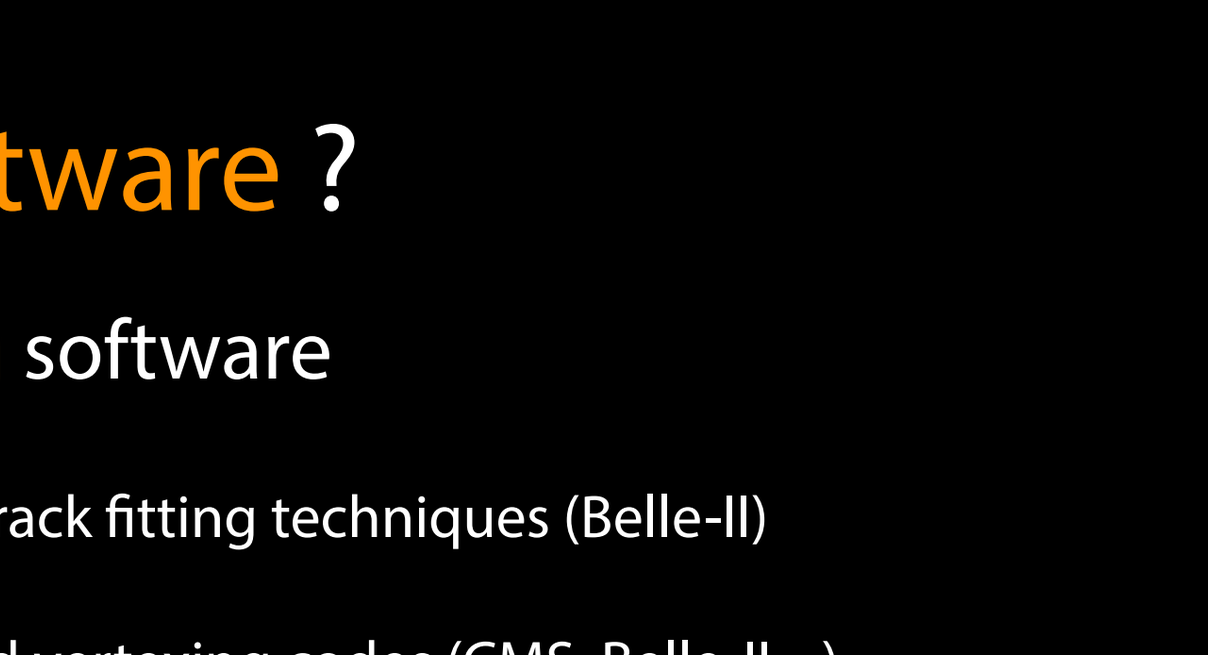


Common Tracking Software

(?)

Common Tracking Software ?

- examples for common tracking software
  - ➡ aidaTT - primarily targeting ILC
  - ➡ GenFit - an implementation of standard track fitting techniques (Belle-II)
  - ➡ Millepede - track alignment page
  - ➡ CMS vertexing suite - package of standard vertexing codes (CMS, Belle-II,...)
  - ➡ VDT, SMatrix, Eigen - vector algebra and math libs
- current attempts for a common tracking implementation
  - ➡ aidaTT is building one common solution
  - ➡ ATS from ATLAS tracking group
    - make tracking/vertexing/fastsim suite public for FCC, build with Gaudi/Athena
  - ➡ GenFit is aiming at a common solution
- are there obstacles ?
  - ➡ experiments already have a solution
  - ➡ integration means picking a data model
    - determines Jacobians in math formulars
  - ➡ integration means framework interfaces
  - ➡ best physics performance ?
    - optimal pattern strategy depends on detector setup and physics requirements



Community driven common Developments ?

- "one-solution-fits-all" seems not an option
  - ➡ question is how to best build upon existing initiatives ?

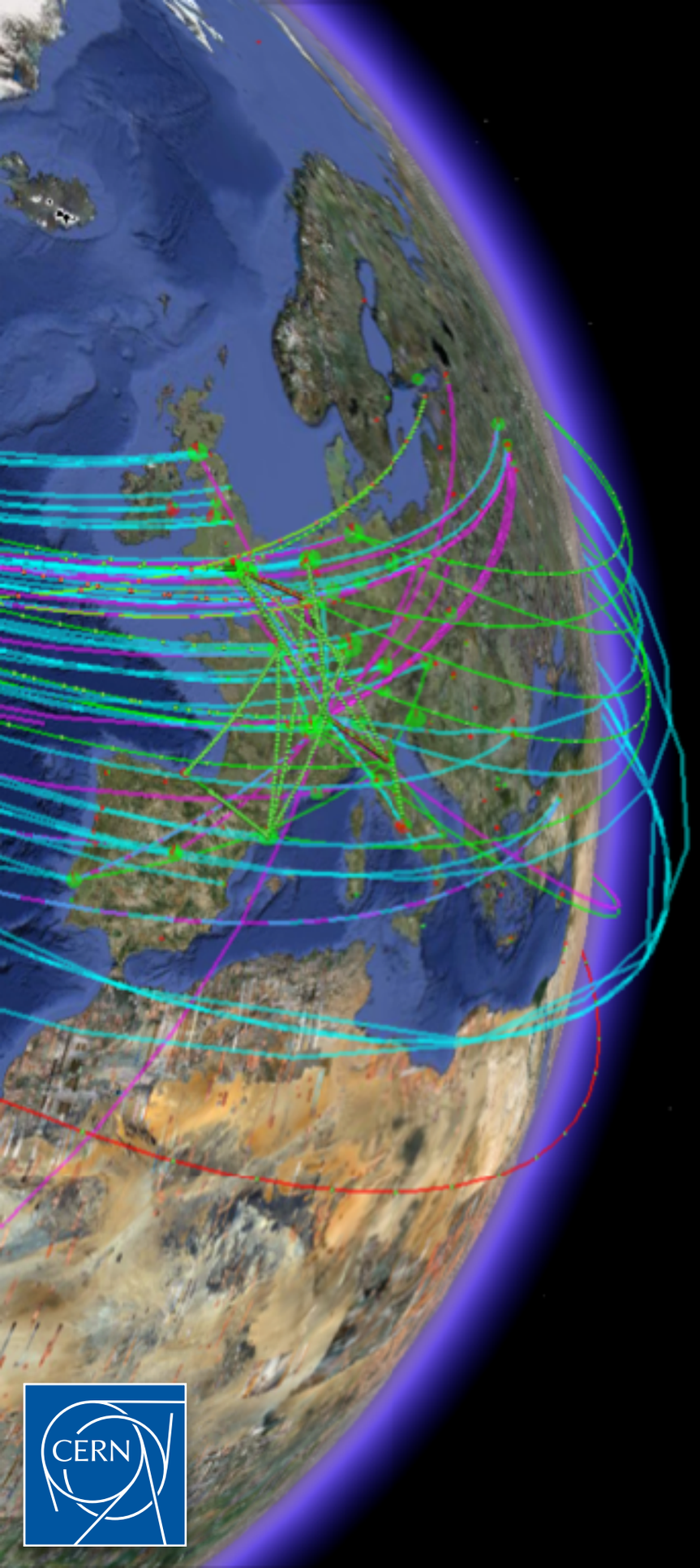
- provide software distributions ?
  - ➡ either centrally provide builds of aidaTT, GenFit, ATS, ...
  - ➡ or refer to already public release sites

from Graeme Stewart

- in a next step, aim at repackaging into a suite of tools
  - ➡ nucleus for something like the old "CERNLIB" idea
  - ➡ Runge-Kutta propagators (ATS), Eigen based data model primitives (ATS), ...
    - we probably will find more such tools that can be isolated

- address more difficult problems
  - ➡ aim at isolating aidaTT or ATS from framework dependencies
    - without creating our own "tracking" framework !!!
  - ➡ or e.g. provide a tracking geometry with internal navigation as a package (a la ATS) ?

- will succeed if community committed because of own need



The Forum is just starting...

Summary of first Meeting

- fruitful meeting, talks from
  - ➡ LC, Belle-2, ATLAS, LHCb and SFT
- triggered discussions
  - ➡ re-use of Runge-Kutta propagator
  - ➡ "cernlib2"
  - ➡ software licenses
  - ➡ (dis)advantage of light-weight approach to tracking
  - ➡ ...
- general agreement to continue the series !
- another follow up
  - ➡ setting up independent builds of tracking software tools

minutes: https://indico.cern.ch/event/459865

Thursday, 3 December 2015

15:00 - 15:10 **Introduction: Goals of the Common Tracking Software Forum** *10'*
Speaker: Markus Elsing (CERN)
TrackingForum.pdf

15:20 - 15:35 **Tracking Tools for the Linear Colliders** *15'*
Speaker: Frank-Dieter Gaede (Deutsches Elektronen-Synchrotron (DE))
gaede_trackreco_l...

15:45 - 16:00 **GenFit and its Application for the Belle-II Experiment** *15'*
Speakers: Johannes Rauch (Technische Universität München), Martin Heck (KIT)
GenfitTrackRecoFo...

16:10 - 16:25 **Future of the ATLAS Common Tracking Project and its Application for FCC** *15'*
Speaker: Andreas Salzburger (CERN)
CDOT-2015-Dec-S...

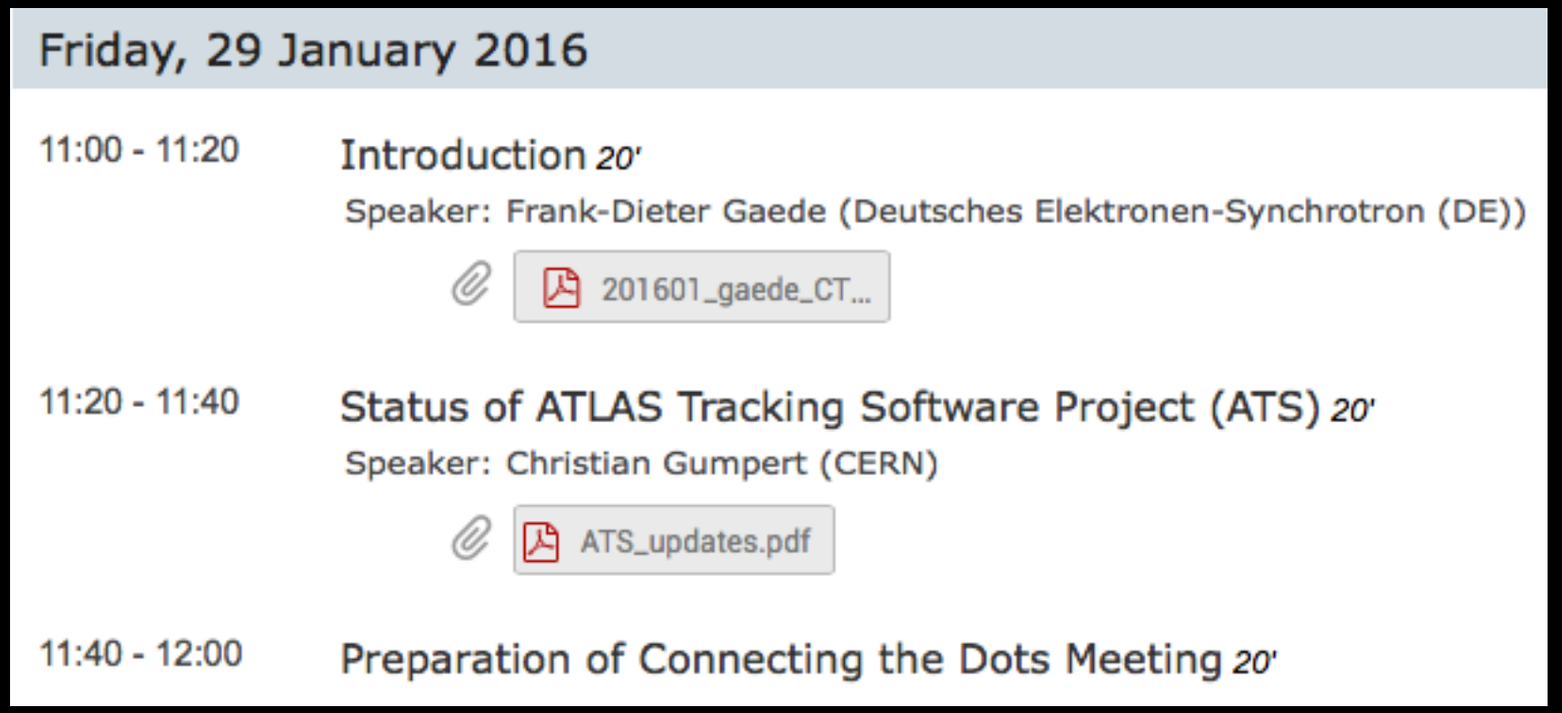
16:35 - 16:50 **Tracking Software in LHCb** *15'*
Speaker: Michel De Cian (Ruprecht-Karls-Universitaet Heidelberg (DE))
20151203.pdf

17:00 - 17:15 **Ideas about Software Repository, Nightly Build System and Software Licences** *15'*
Speaker: Benedikt Hegner (CERN)
HSFTrackingWG.pdf

Summary of second Meeting

- short meeting last months
  - ➡ preparation of this weeks meeting + 1 talk

- status report of ATS
  - ➡ toward a build of a full prototype standalone ATLAS tracking code independently
  - ➡ need to put release infrastructure in place
    - possible first step towards broader common tracking builds
  - ➡ more this week...



Examples for Presentations

Johannes Rauch

Design of GENFIT

Technische Universität München

Modular Design

- Measurements
  - E.g. strip-, pixel-, wire-, spacepoint-measurements.
  - Provide (virtual) detector planes and measurement coordinates and covariance projected into that plane.
- Track representations (“TrackReps”)
  - Track parametrization.
  - Extrapolation through material and magnetic field.
  - Also propagates time of flight.
  - Particle hypothesis.
- Track fitting algorithms
  - Use measurements and TrackReps to calculate fit results.
  - Start value for fit needed, e.g. from pattern recognition.

Track
- Contains measurements (can be from different detectors).
- Can be fitted with several TrackReps (e.g. for fitting different particle hypotheses).

1 — GENFIT and its Application for the Belle II Experiment 5

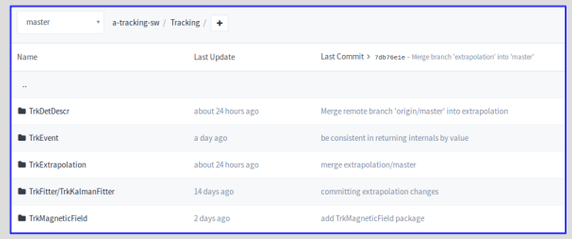
Christian Gumpert

Package structure (ATS)

Tracking					
Geometry (TrkDetDescr)	Event Data Model (TrkEvent)	Extrapolation (TrkExtrapolation) A	Fitting (TrkFitter) A	Calibration, general (TrkTools) A	Alignment (TrkAlignment) A
Done	(Done)	Done	Started	Not yet started	

under discussion

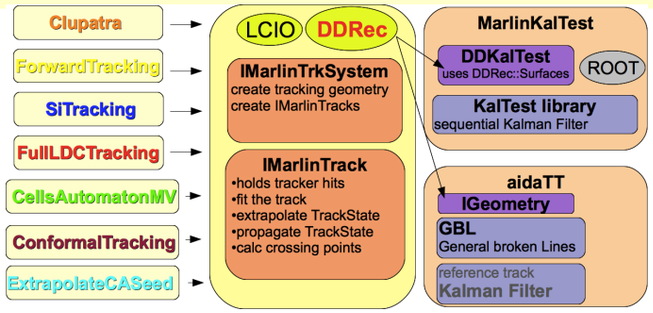
Plugins
- e.g. geometry converters



29.01.16 Christian Gumpert, CERN 5

Frank Gaede

lMarlinTrk - LC Tracking Tools

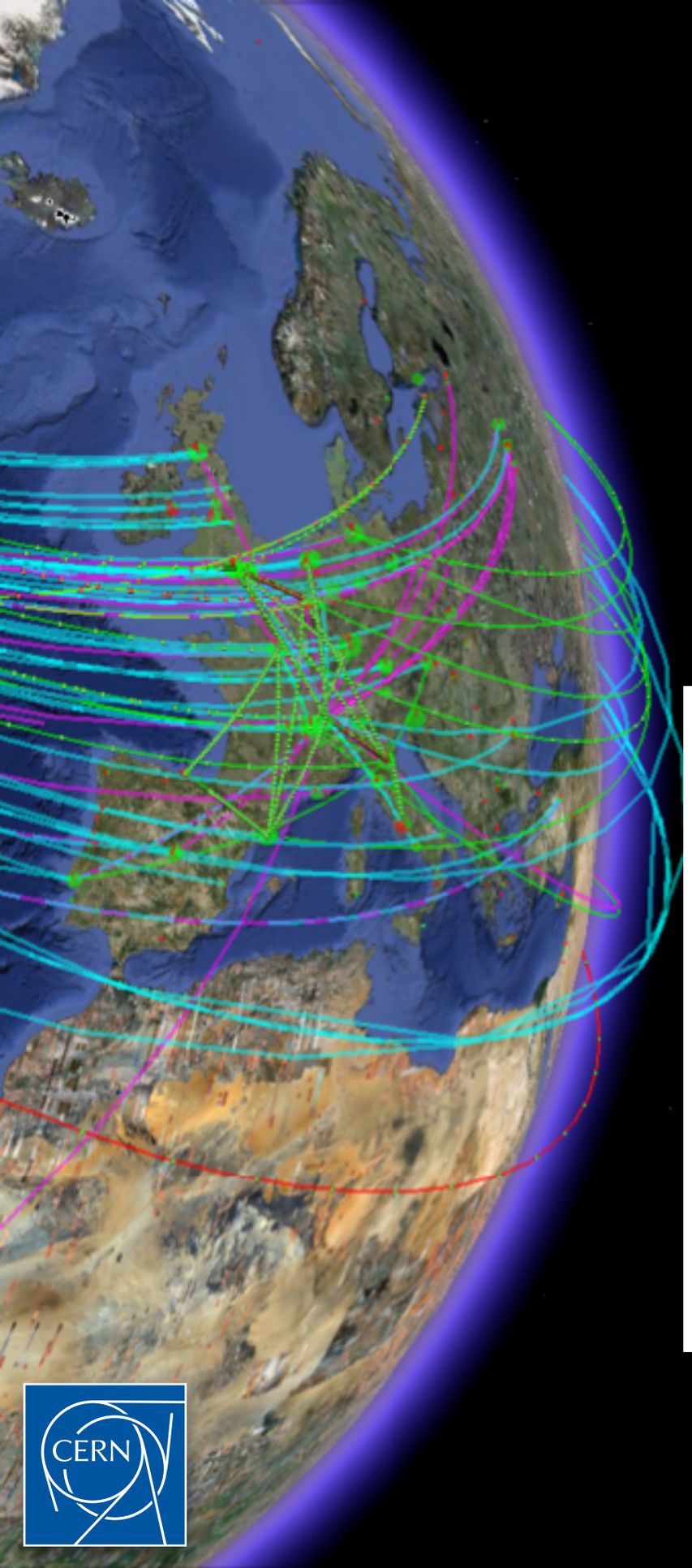


- IMarlinTrk: interface that separates pattern recognition code from actual fitter implementation
  - only dependencies LCIO, DDRec (DD4hep) - NB: no Marlin dependency !
- pattern recognition algorithms have been written to a large extend in plain vanilla C++ (no LCIO, geometry, etc.), e.g.
  - topological clustering, CA libraries, conformal mapping,...
  - currently code lives in iLCSoft libraries - could be extracted to standalone libraries

Benedikt Hegner

Development infrastructure and nightly builds

- To share software with others one has to make sure it compiles, runs and yields proper physics…
  … outside the environment it was originally developed in!
- Multiple free nightly build services for open-source projects available, like Travis CI
  - Nicely integrate with GitHub / GitLab
  - Allow compilation and simple tests
- They do not easily cover
  - (CPU) performance studies or validation do not fit into that model
  - Multiple platform support
  - “Exotic” machines
  - Direct debugging of failures
- Doing it properly involves some boring setup and maintenance work
  - People rarely have time for that!
- Idea by HSF contributors is to set up a basic build and test cluster at CERN the various tracking software projects can take advantage of
  - Do the work only once!
  - Taking advantage of jenkins and docker containers
  - Allowing interactive access for debugging



Join us on Wednesday afternoon for the next Forum meeting

Common Track Reconstruction Software Forum

Convener: Markus Elsing (CERN)

14:45 **Introduction** *10'*
Speakers: Dr. Markus Elsing (CERN), Dr. Benedikt Hegner (CERN)

14:55 **CLIC tracking software and common parts with other ILC developments** *30'*
Speaker: Dr. Rosa Simoniello

15:25 **Software designs of GENFIT3** *30'*
Speaker: Dr. Elisabetta Prencipe

15:55 **CMS Tracking Software Report** *30'*
Speaker: Dr. Vincenzo Innocente

16:25 **ATS walkthrough** *20'*
Speaker: Dr. Andreas Salzburger

16:45 **Overview of ATS project and its use for FCC** *30'*
Speaker: Mrs. Julia Hrdinka