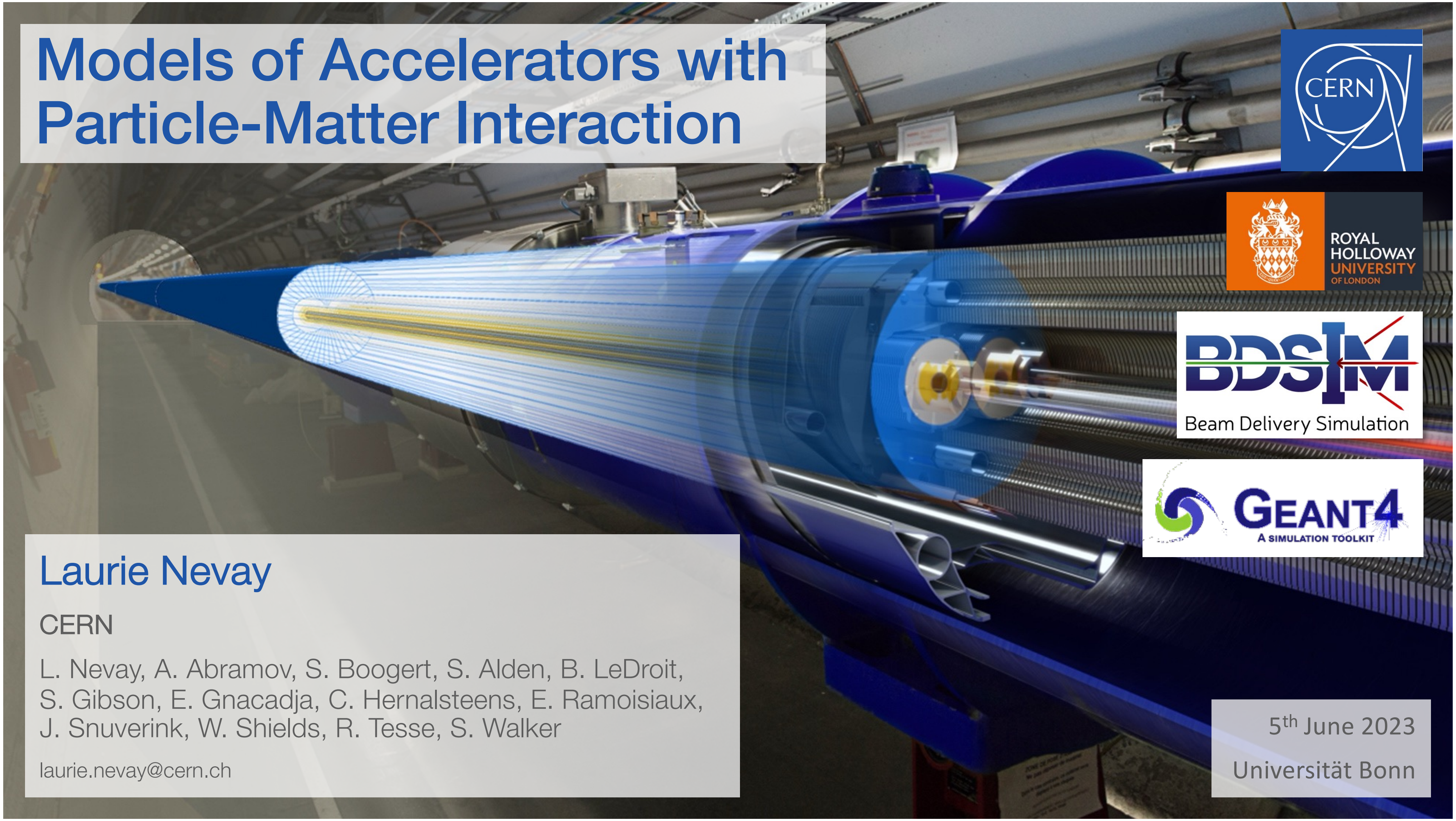
Models of Accelerators with Particle-Matter Interaction



Laurie Nevay

CERN

L. Nevay, A. Abramov, S. Boogert, S. Alden, B. LeDroit, S. Gibson, E. Gnacadja, C. Hernalsteens, E. Ramoisiaux, J. Snuverink, W. Shields, R. Tesse, S. Walker

laurie.nevay@cern.ch

5th June 2023

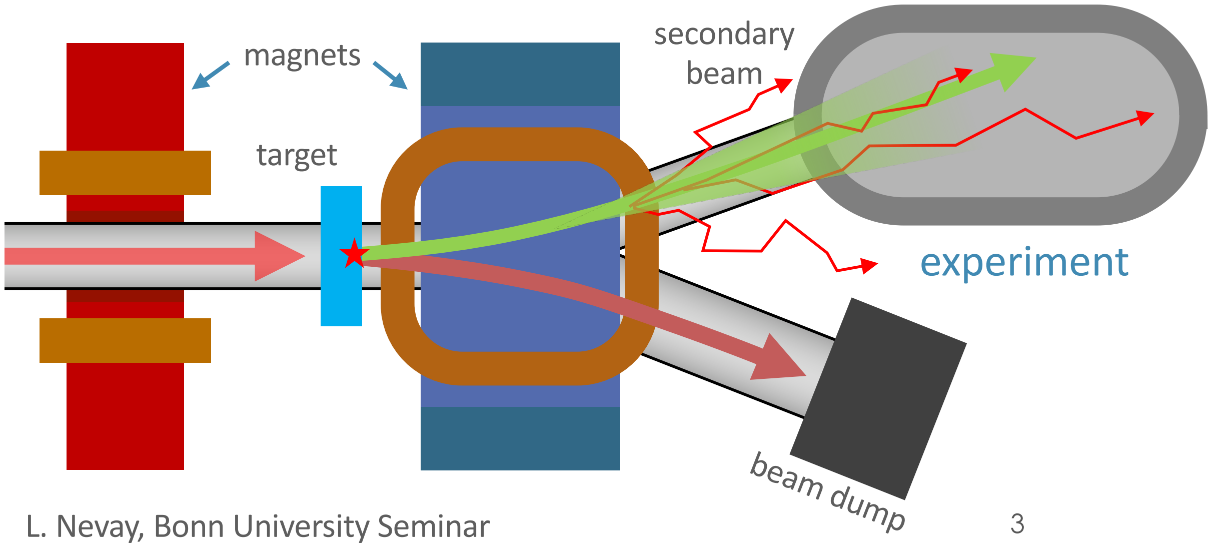
Universität Bonn

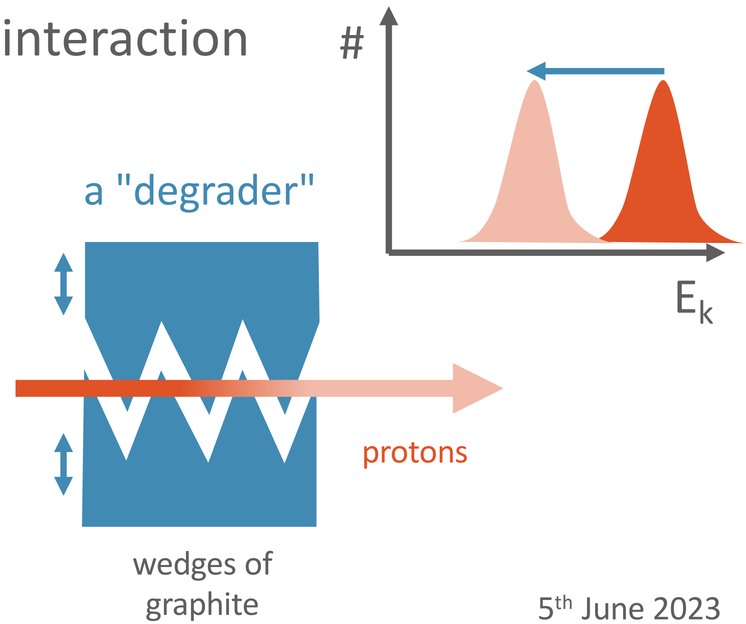
Introduction

1. Conceptual problem of simulating all particles in an accelerator
2. A solution! … and practicalities
3. Geometry handling, conversion and challenges
4. Programmatic model building
5. Examples of applications
   - FASER
   - LHC physics debris and collimation
   - medical hadron therapy
6. Outlook & Conclusions

Beam Losses & Particle-Matter Interaction

- No accelerator contains *all* particles
- Consequences: *none* / *intended* / *catastrophic*
- Many physics experiments in the vicinity of accelerators experience charged particles from the accelerator as unwanted 'background'
- Many (medical) beam lines exploit particle-matter interaction



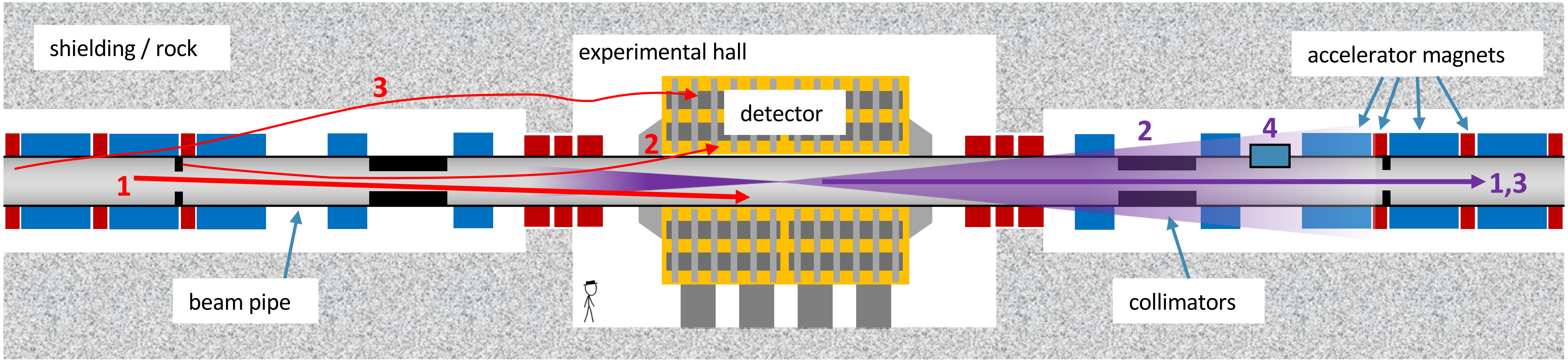


The Machine Detector Interface

- Radiation / particles in both directions - *both* are interesting

Incoming:
1. products from residual gas interaction
2. leakage from the collimation system
3. general secondaries from beam loss

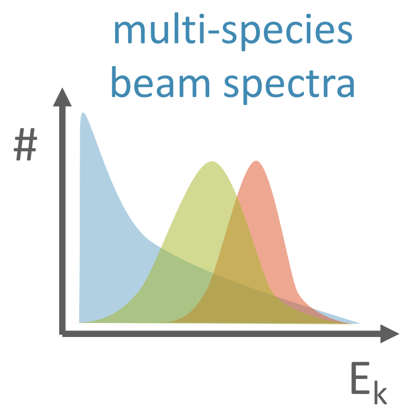
Outgoing:
1. lightly scattered primaries
2. physics debris
3. forward physics
4. forward experiments

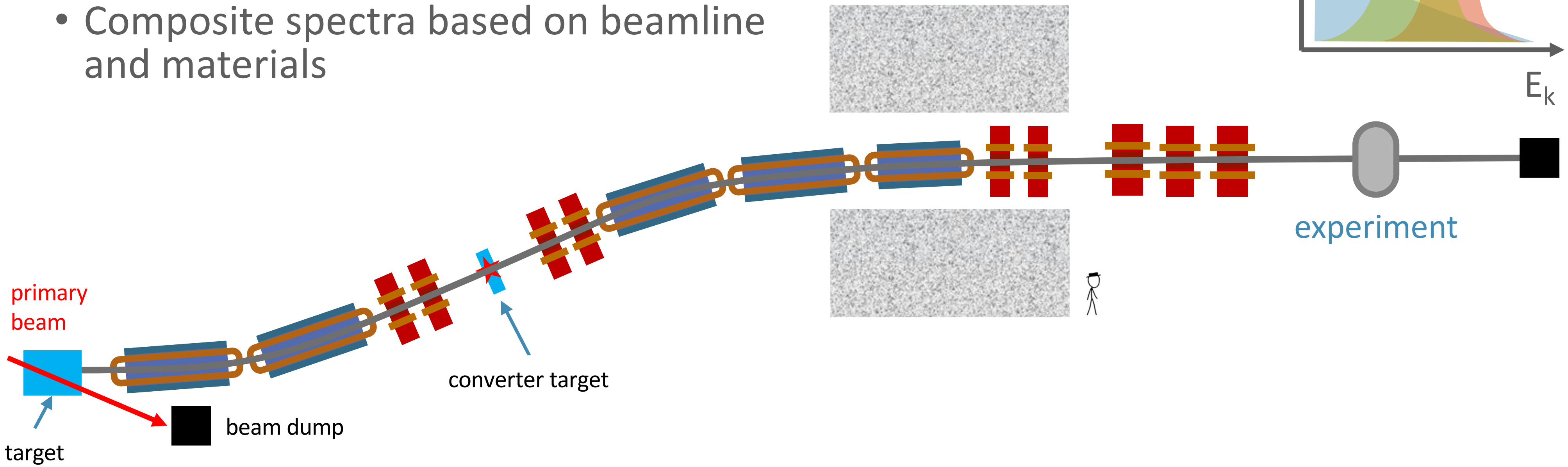


Goal: simulate far reaching particles *in* and *out* of experiment and understand them
Need: accurate magnetic particle tracking *and* interaction with matter

Secondary / Tertiary Beamline

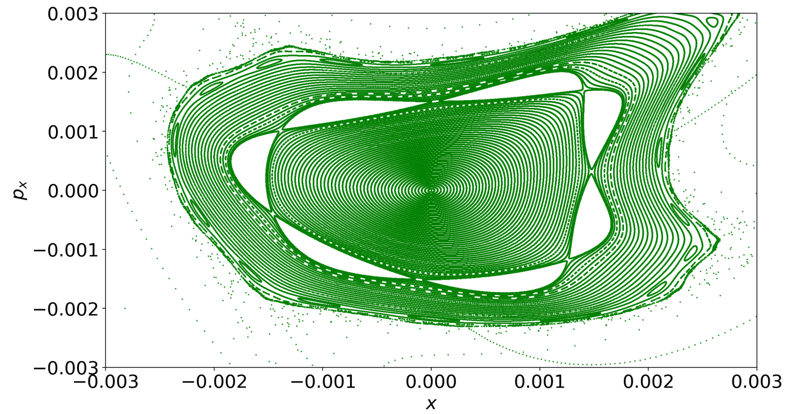
- Secondary or tertiary beams can provide a wide range of particles over a wide range of momenta
  - useful for detector calibration and development
- Exploit particle-matter interaction
- Composite spectra based on beamline and materials



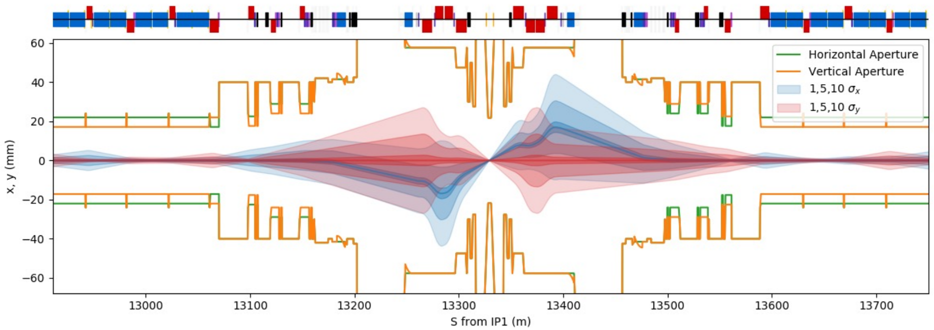


Accelerator Tracking

- Accelerator tracking models motion in many magnetic fields in an accelerator
- Typically *only 1* type of particle
  - as usually only 1 type in the main beam
- 'Losses' are when particle exceeds an amplitude or an aperture radius
- Uses *curvilinear* coordinates
  - small relative difference to orbit for better numerical precision in a computer with fixed precision
- Must be accurate for up to millions of fields or operations in a row for many revolutions
  - for linear machines this isn't too important
- Focus on computational speed vs accuracy
- Compounding transfer maps into 1 map for a ring is not sufficiently accurate for nonlinear fields and doesn't tell you where a particle is lost



example Poincaré map through nonlinear fields



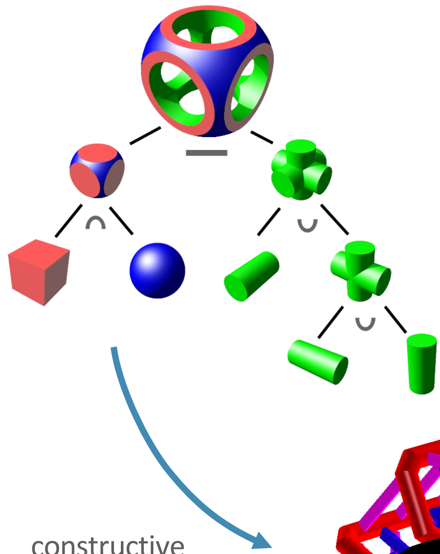
LHC IR5 beam envelope and aperture

3D Detector Models in Particle Physics

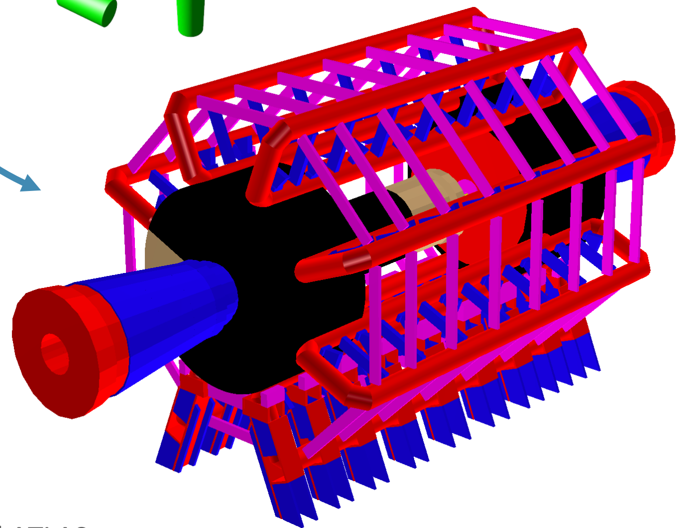
- Use a 3D radiation transport model to predict detector response and compare to data using Monte Carlo
- Detailed 3D geometry - *complex* and *specialised* to an experiment
  - takes many person-years to make and validate
- Often includes magnetic field for particle identification
  - not a uniform or perfect magnetic field
  - use numerical integration for tracking all particles in Cartesian coordinates e.g. 4th order Runge Kutta
- Geant4 is one of the most common software libraries for creating these models
  - open source C++, regular development by community
  - all subatomic particles, huge array of physics models
  - 24 years of work by ~100 people, > 50,000 users,

GEANT4
A SIMULATION TOOLKIT

http://geant4.web.cern.ch



constructive solid geometry



simplified ATLAS geometry

Mixed Tracking Premise

- Need:
  - accurate tracking of all particles in an accelerator up to the detector
  - particle-matter interaction for scattering and secondary production
- Solution:
  - build radiation transport model like a detector
  - provide coordinate transforms between curvilinear and Cartesian
  - provide accelerator-style integrators for particle motion
  - fall back to numerical integration where needed (e.g. non-paraxial particles)
- Tricky bits:
  - making the 3D model takes forever and even then it's hard coded
  - 'thin' things - accelerator tracking uses thin kicks to represent imperfections
  - dipoles - where the curvilinear frame bends - surprisingly more involved
  - angled pole faces (input / output face) on dipoles

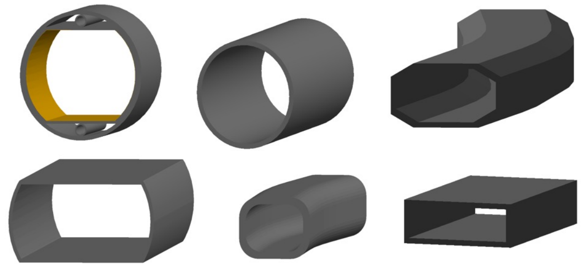
A Solution

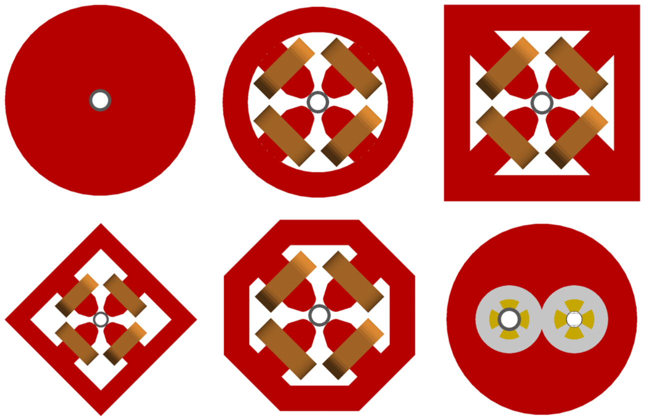
The 3D Model - Complexity & Time

- Accelerators are typically repetitive and similar in design
- Described by list of elements in order:

  drift, dipole, drift, quadrupole, drift, quadruple, drift

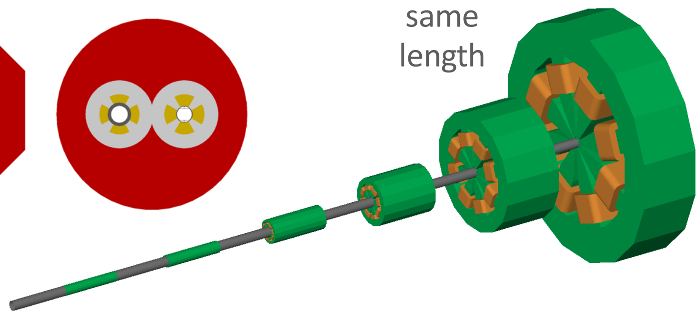
- Solution: provide library of typical accelerator components with adjustable proportions

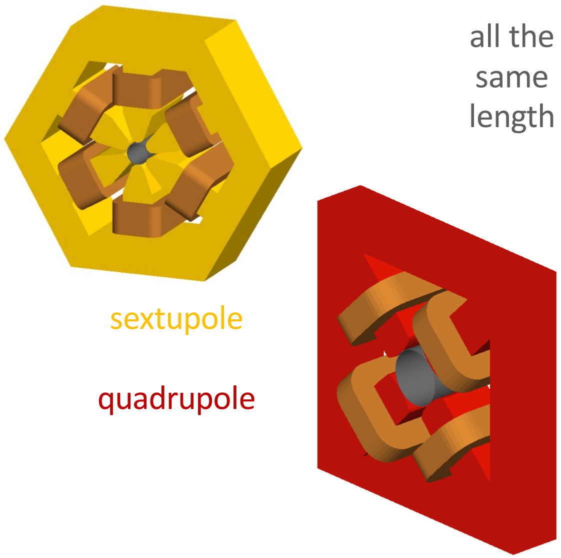




different yoke styles

all the same length

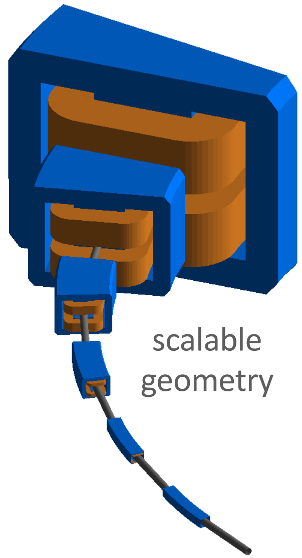




sextupole

quadrupole

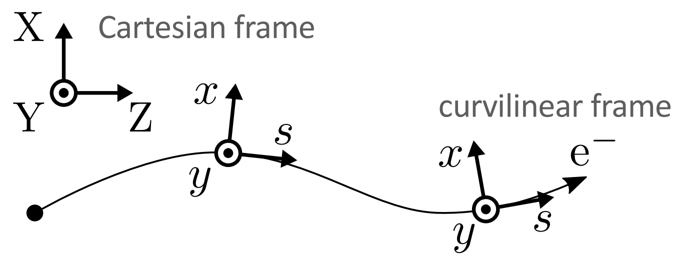
all the same length

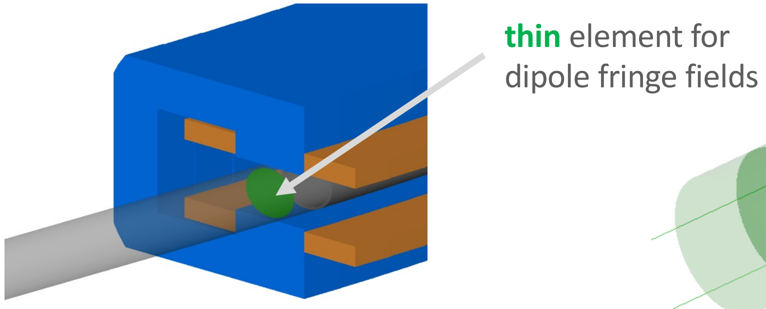


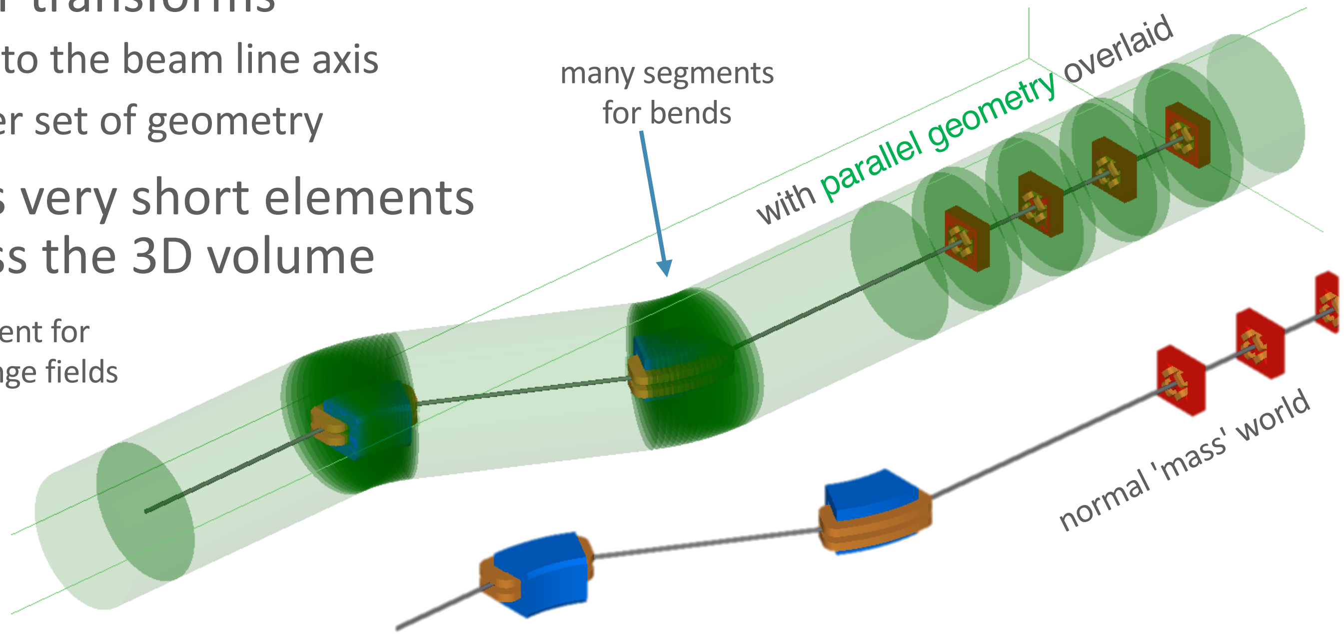
scalable geometry

Coordinate Systems & "Thin" Elements

- Need to convert between coordinate systems
- Geant4 provides a transform from one shape to a global frame, but relies on known geometry
- Use ***parallel geometry*** for transforms
  - cylinders that correspond to the beam line axis
  - parallel means just another set of geometry
- Include 'thin' elements as very short elements with 1 tracking step across the 3D volume

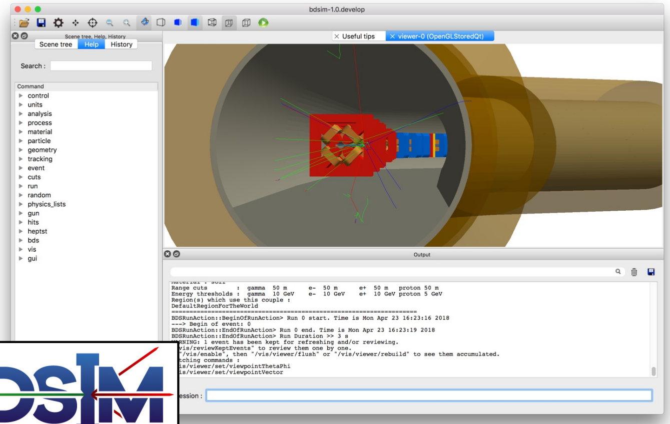




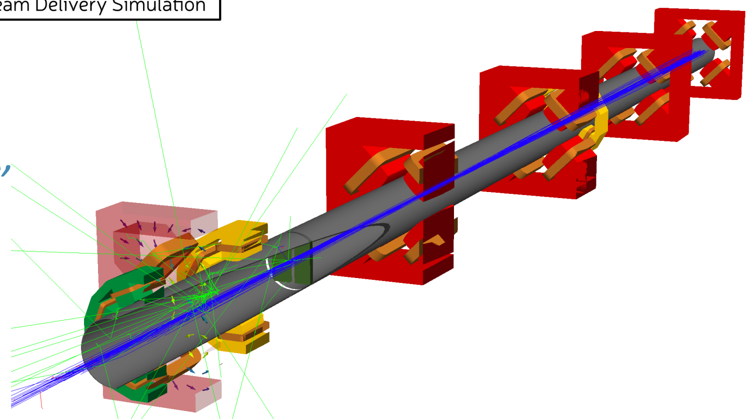


BDSIM - Beam Delivery Simulation

- BDSIM code joins the two seamlessly
- Automatic Geant4 models of accelerators
  - rewritten and modernised since 2013
- Library of scalable accelerator components
  - make a model in minutes to weeks, not years
  - convert from other accelerator codes easily
- Applied to many experiments and machines
  - *CERN North & East Areas, ILC / CLIC, AWAKE, XFEL undulators, LHC collimation, Laserwires, FASER, KLEVER, NA62, ATLAS non-collision backgrounds, MAGIX at MESA, FCC MDI, LhARA*
- Benefit from community-driven Geant4
  - physics models constantly updated and improved



Beam Delivery Simulation



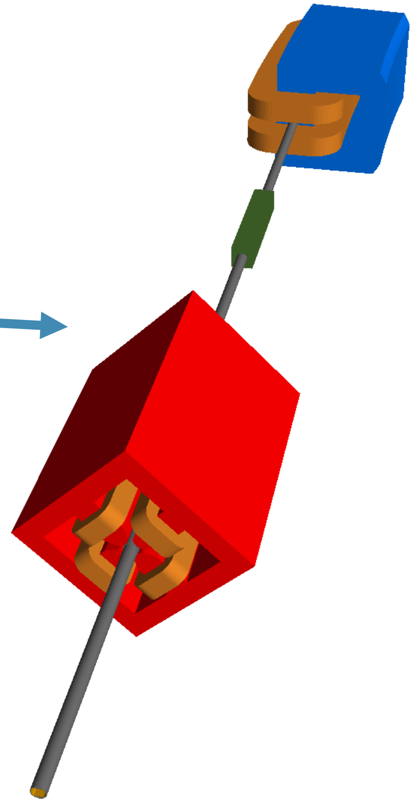
Comp. Phys. Comm. (252), July 2020, 107200

Example BDSIM Syntax

- "GMAD" - Geant4 + MAD (MAD is an accelerator optics code)

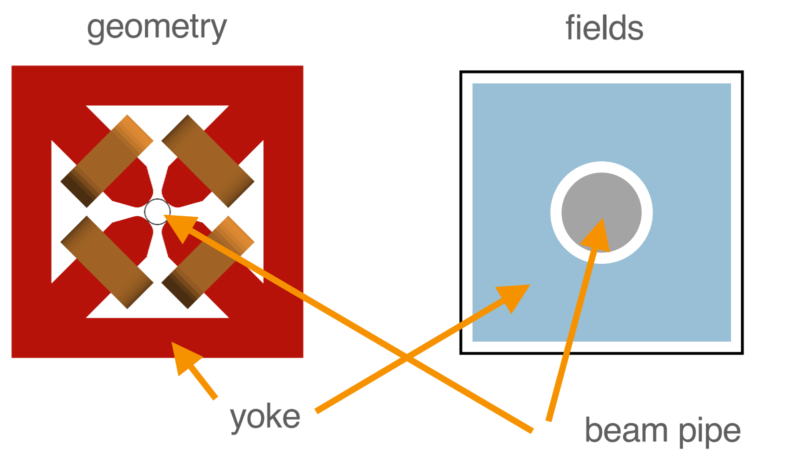
sm.gmad

```
d1: drift, l=1*m;
q1: quadrupole, l=1*m, k1=1.3, magnetGeometryType="polesfacet";
c1: rcol, l=0.6*m, ysize=5*mm, xsize=5*mm, material="Copper", outerDiameter=10*cm;
s1: sbend, l=1*m, angle=0.10;

l1: line = (d1, q1, d1, c1, d1, s1);
use,period=l1;

sample, all;

option, ngenerate=1,
        physicsList="em";

beam, particle="proton",
      energy=10.0*GeV,
      X0=0.001,
      Y0=0.001;

-:---  sm.gmad        All L3    Git:develop  (Fundamental)
Wrote /Users/nevay/physics/reps/bdsim/examples/simpleMachine/sm.gmad
```

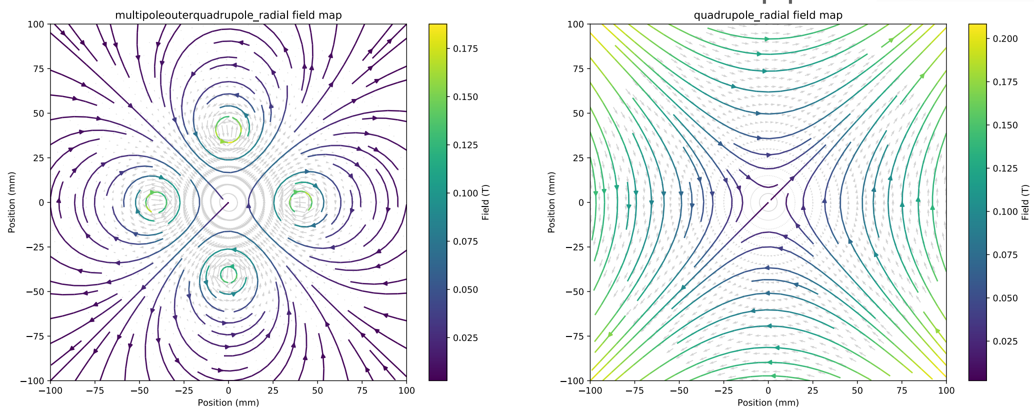


Tracking Implementation

- Custom numerical integrators
  - for 1st order matrix transport maps
- These ignore the field and are constructed with a strength
  - like "k1" for quadrupole
  - scaled with rigidity of each particle
- Fall back to RK4 if…
  - non-paraxial (e.g. sideways)
  - low radius of curvature (spiralling)
- Provide suitable fields
  - pure field in vacuum
  - parameterised yoke field
  - normalise at pole tip
- Requires curvilinear coordinate system



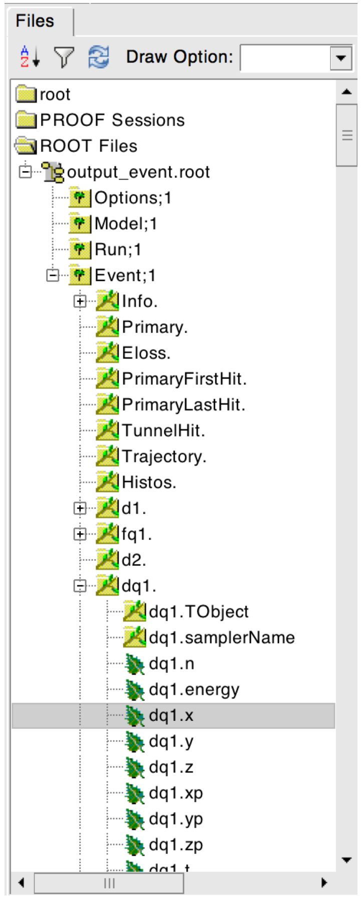
$$B_x = \frac{\partial B_y}{\partial x} y$$
$$B_y = \frac{\partial B_y}{\partial x} x$$
$$B_z = 0$$



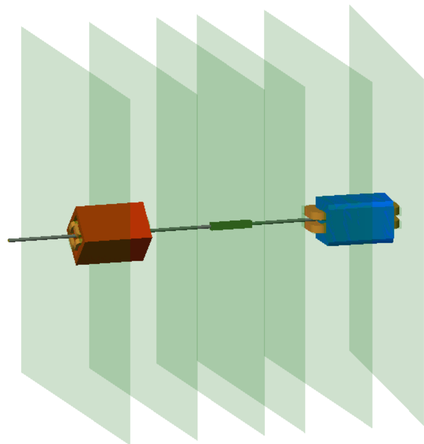
Data & Analysis

- Accelerator simulations: 1 particle in, 0 or 1 out
- Detector simulations: 1 event in, 0 - 10^7 particles out
- Data is stored in ROOT format with per-event structure
  - structure allows filtering of histories
  - understand beam losses, origins and processes involved
- Data format and included analysis tools key to understanding the origin of energy deposition
  - easy filtering / selection in analysis and skimming
- Invisible "sampler" planes to record distributions after an object
  - in another parallel world
- Scoring meshes and energy deposition too

example data tree structure

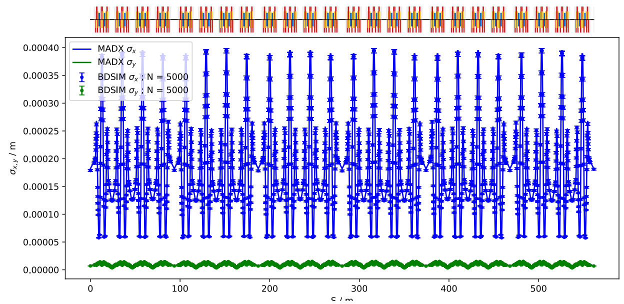


sampling planes after each element (normally invisible)

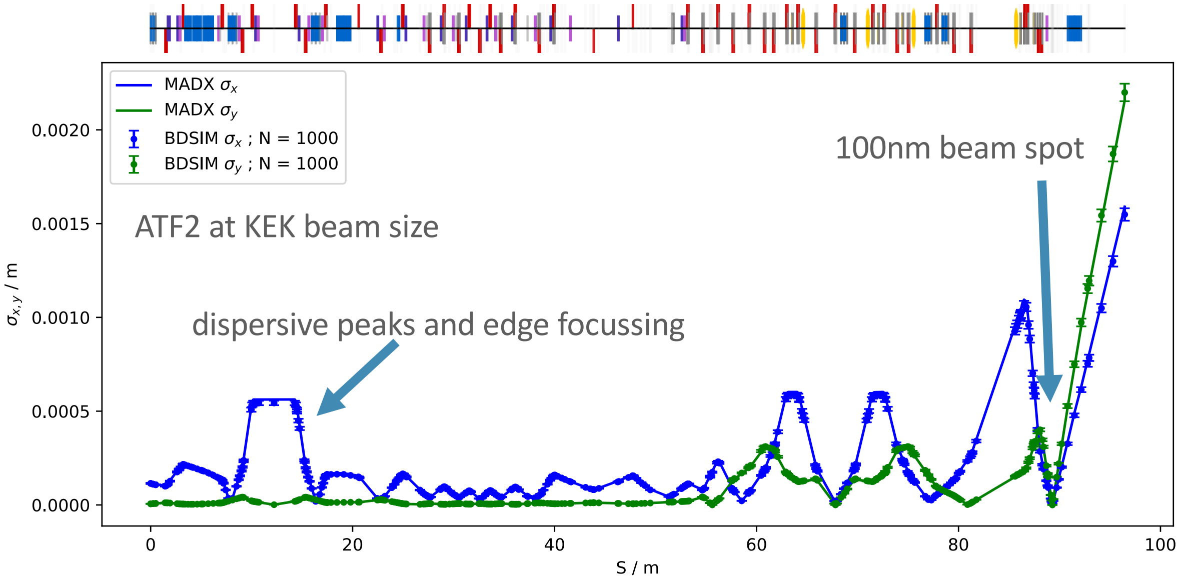


Tracking Validation

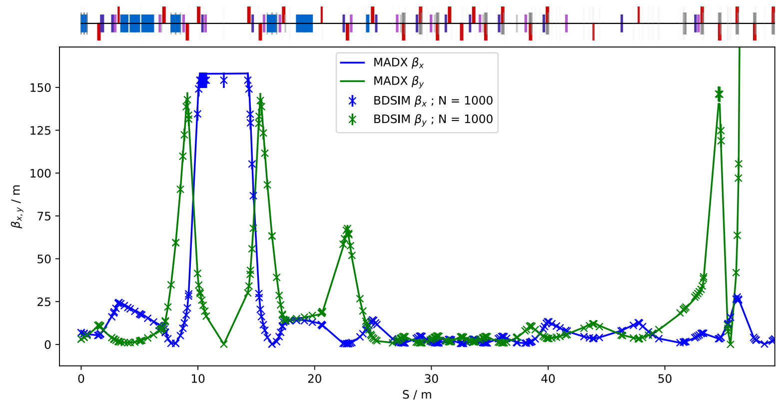
- Crucial to compare 'optics' of machine models
- Compare particle distribution after each element
- Calculate optical functions from particle distribution
  - using (up to) 4th order moments including statistical uncertainty



Diamond Light Source

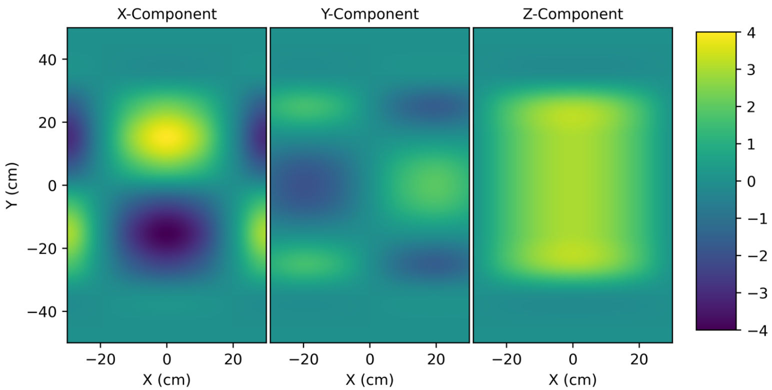


ATF2 Twiss Beta functions

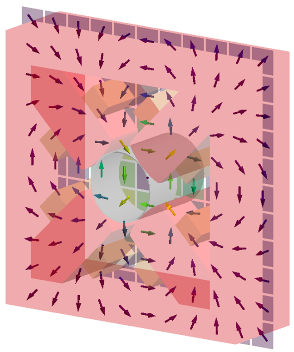


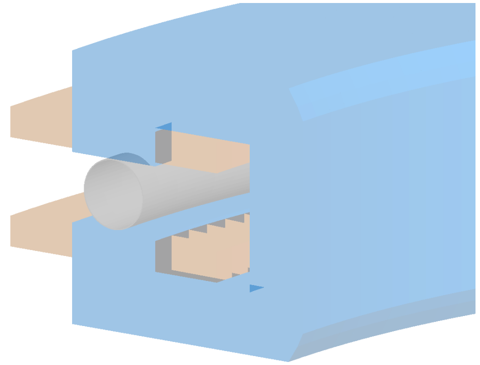
Customisation

- Generic geometry is not always suitable
  - include ability to include custom geometry
- Can overlay E or B or combined EM field maps
  - 1D to 4D field maps supported
  - numerical interpolation 1D to 4D included
- Supports all Geant4 physics lists
  - modular and reference
- Cross-section biasing per volume
  - combine multiple biases (particle:processes)
  - overlay only on vacuum or yokes or world
- Beam distribution generator for accelerators
- 3D scoring meshes and beam loss monitor scoring
- Automatic tunnel building following beam line
- Circular tracking control!
  - stop particles after N turns of circular accelerator



field map interpolation




custom field map attached to geometry

Geometry Handling & Conversion

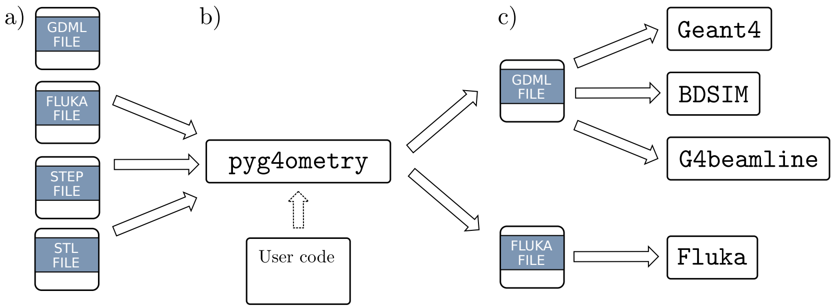
Custom Geometry

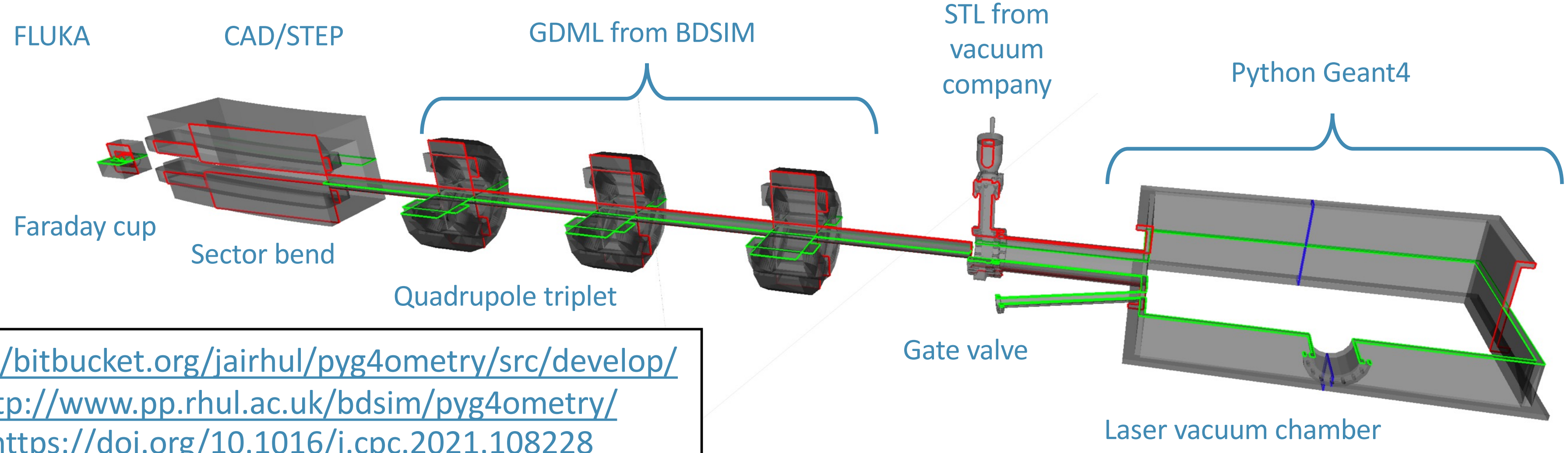
- So far, we've talked about BDSIM's library of scalable geometry
  - predefined components for a common purpose, e.g. quadrupole
  - built into BDSIM and guaranteed to be good to use
- What about other geometry specific to an experiment?
  - not looking to make the detector model - assume this is specialised
  - but some more detailed geometry relevant to get the right outcome
- 3 hurdles:
  - ability to import / place it in a Geant4 model
  - creation of geometry from scratch
  - conversion of detailed geometry from another format (different mathematical form)
- Goal: a PhD student can create reasonable geometry in reasonable time
  - hours to days to weeks, not months to years → what can 1 person do and learn?

pyg4ometry

Stewart Boogert

- Python geometry package:
  - create, convert, composite, compare, validate
- Place custom components in Geant4 / BDSIM
- Have parity with models in Geant4 & FLUKA



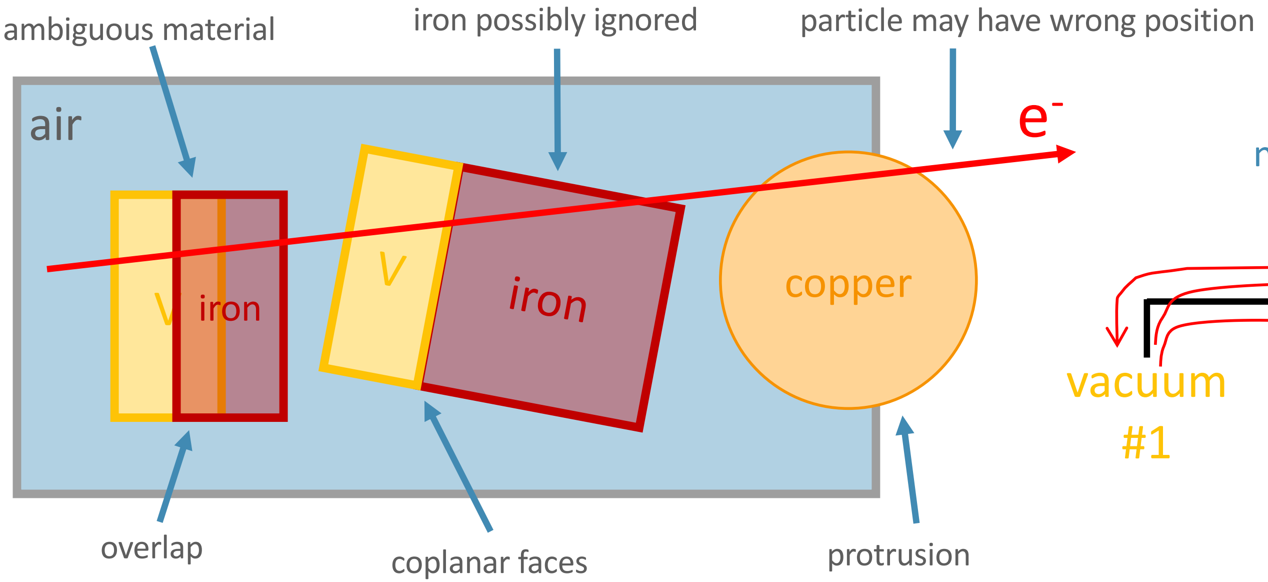


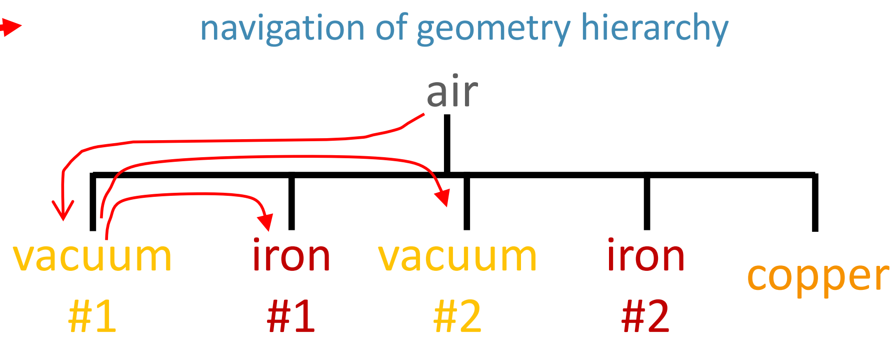
https://bitbucket.org/jairhul/pyg4ometry/src/develop/
http://www.pp.rhul.ac.uk/bdsim/pyg4ometry/
https://doi.org/10.1016/j.cpc.2021.108228

email stewart.boogert@cockcroft.ac.uk

Overlaps & Tracking

- When defining custom geometry, it's entirely possible to define something non-sensical
- Shapes are just instances of classes with translations and rotations
  - shapes are defined by an equation or algorithm (i.e. are you inside or outside it?)
- Unlike the real world, we must not place things 'against' others in constructive solid geometry (CSG)

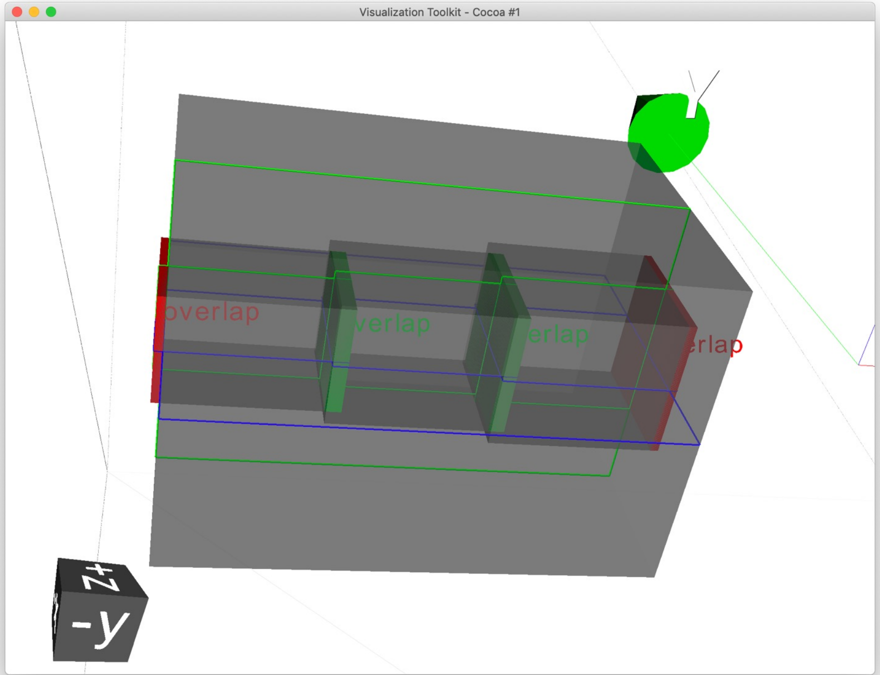




Overlap Identification

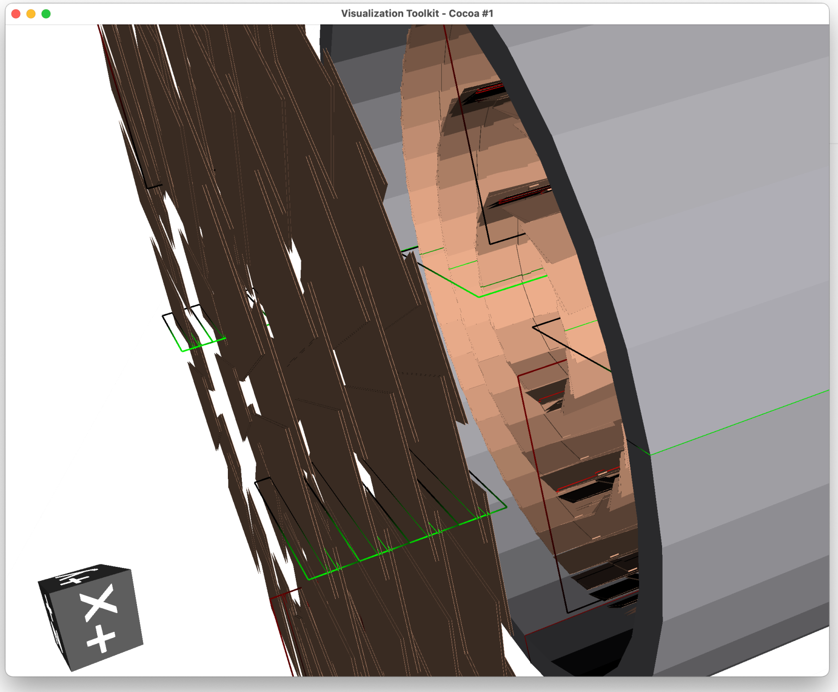
- Use visualisation meshes to identify geometry problems
- e.g. an overlap will form a valid intersection of two meshes
- Use this overlap mesh also for visualisation
  - if the user can see where the problem is it's often easy to fix
- In Geant4 overlap reports can be hard
  - local coordinates etc, → better in 10.7!
  - not always identified

```
Checking overlaps for volume bbrr_pv (G4Orb) ...
-------- WWWW ------- G4Exception-START -------- WWWW -------
*** G4Exception : GeomVol1002
      issued by : G4PVPlacement::CheckOverlaps()
Overlap with volume already placed !
          Overlap is detected for volume bbrr_pv:0 (G4Orb)
          with bbr2_pv:0 (G4Orb) volume's
          local point (562.925,-491.466,564.793), overlapping by at least: 6.32957 cm
NOTE: Reached maximum fixed number -1- of overlaps reports for this volume !
*** This is just a warning message. ***
-------- WWWW -------- G4Exception-END --------- WWWW -------
```

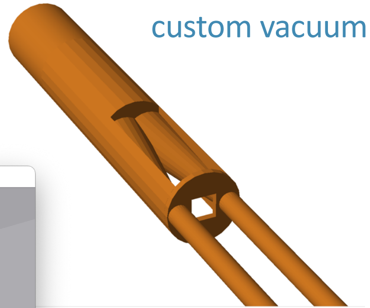


Examples

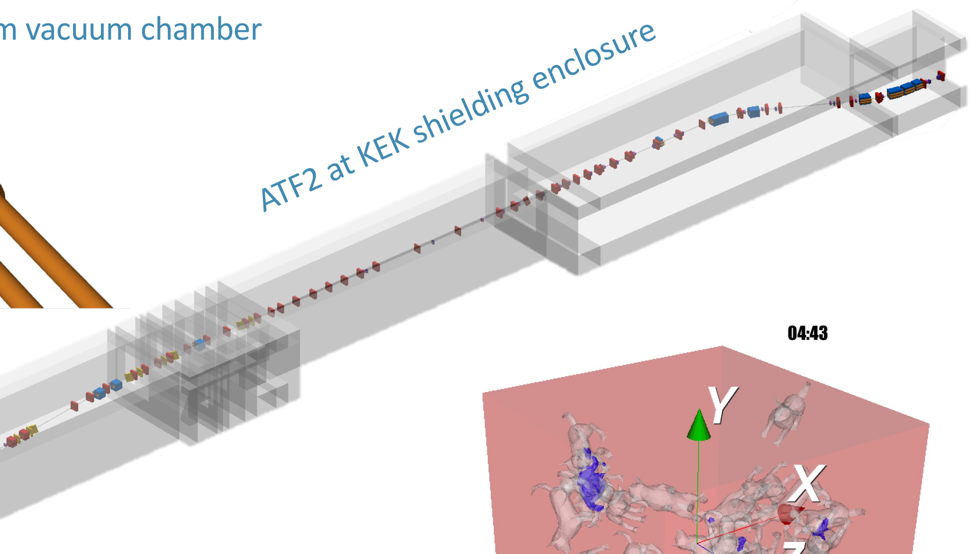
silicon tracker detector

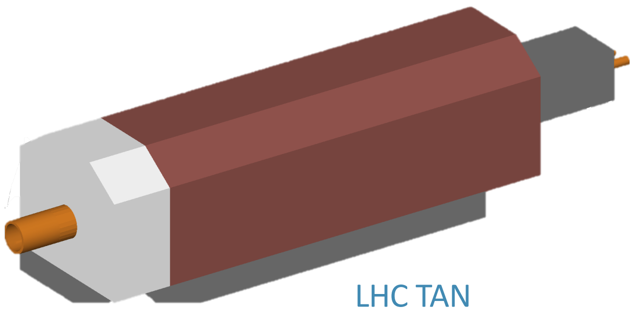


custom vacuum chamber

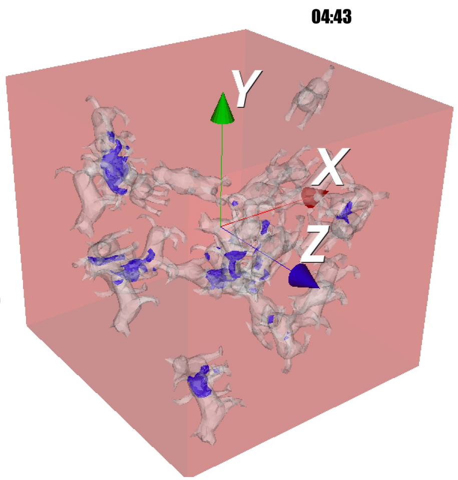


ATF2 at KEK shielding enclosure





LHC TAN

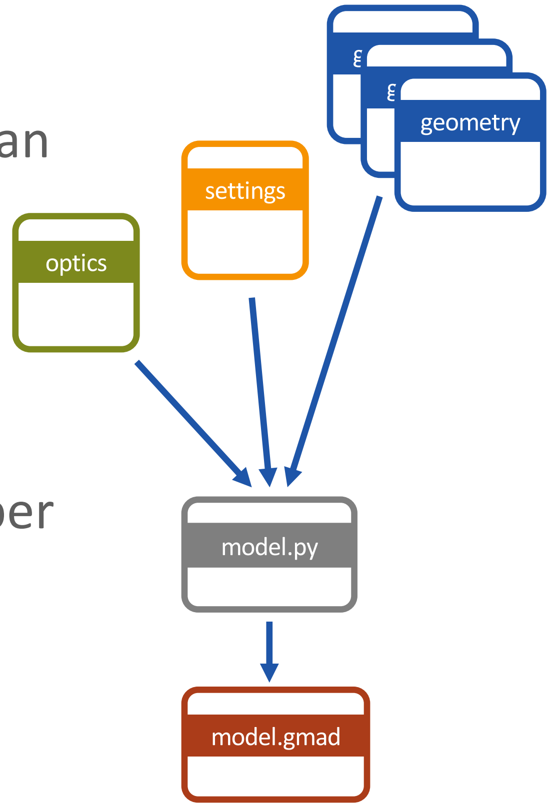


complex overlap identification with STL meshes - a box of dogs

Programmatic Model Building

Sustainability of Creating Models

- A model is rarely used with only 1 configuration
  - magnets, collimators, beam parameters can change
- Even though BDSIM's syntax saves a lot of effort, a model can remain a lot of work
- Premise: models created without any hand edits
- Use Python library / script to create models
  - including stitching or editing of basic conversion
- Re-run construction at any time with even 1 updated number
  - exploit git to ensure only what we want has changed
- Allow others to use and build on your model



Information Flow in Model Building

- Included pybdsim tool gives MADX → BDSIM conversion
- Supply extra information by name of item in optics

parameter = value, parameter = value…

- Build up python dictionaries of information
  - only we can know our own specific information
  - e.g. collimator settings, field map names etc.
- Can write our own python to load our settings

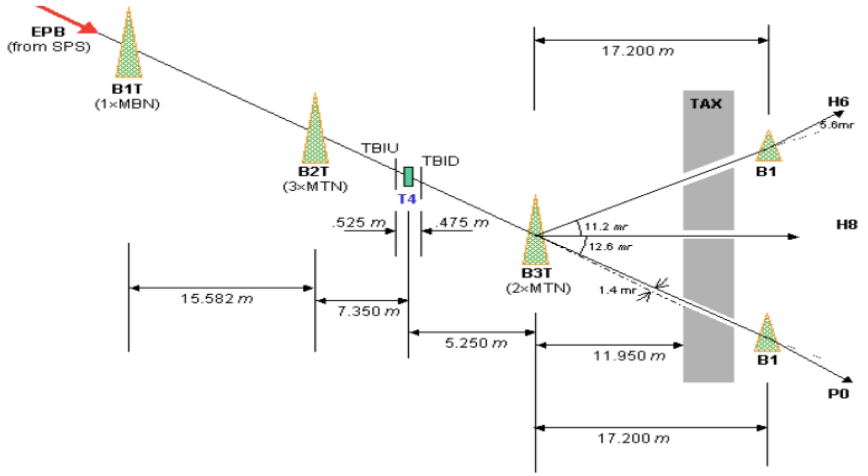
```
d = {'TAX.B1' : {'aper1' : (20,'mm'),
                 'apertureType' : 'circular' },
     'COL1.B1' : {'xsize' : (5,'mm'),
                  'ysize' : (7,'mm')}
     }
```

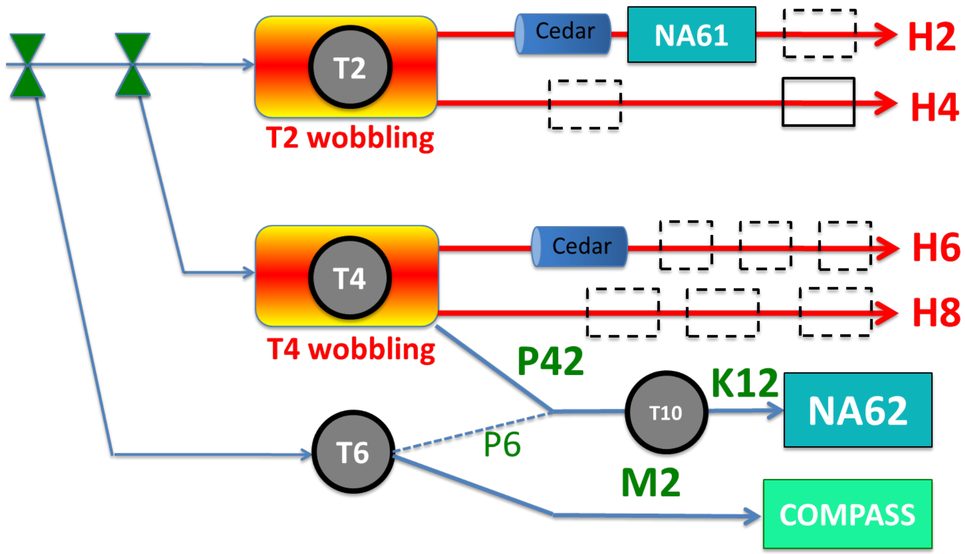
- e.g. for my group we have a python library for field maps and GDML files

Simulation Examples

CERN’s North Area

- Primary 400 GeV/c proton beam slowly extracted from SPS
  - ~4s spill duration
- ‘Wobbling’ is a static bend in the orbit through the target
- Produce 2-3 beams from 1
  - different central momenta for each

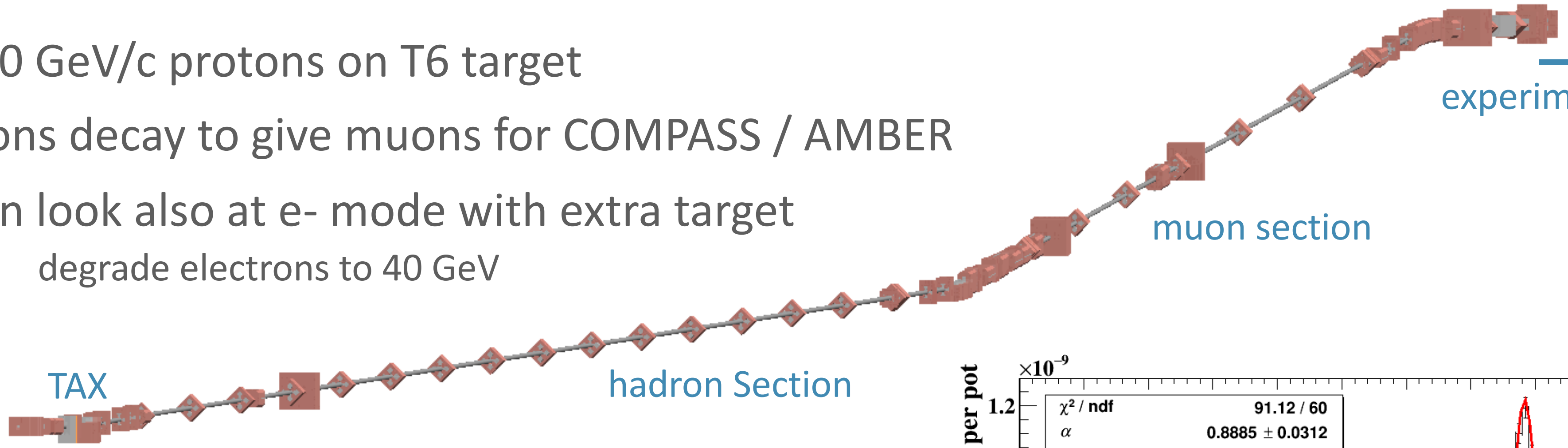




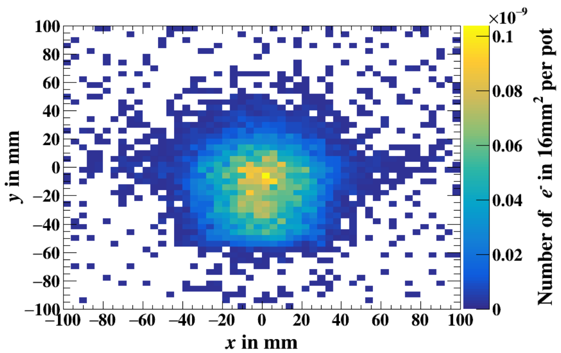
M2 Beamline

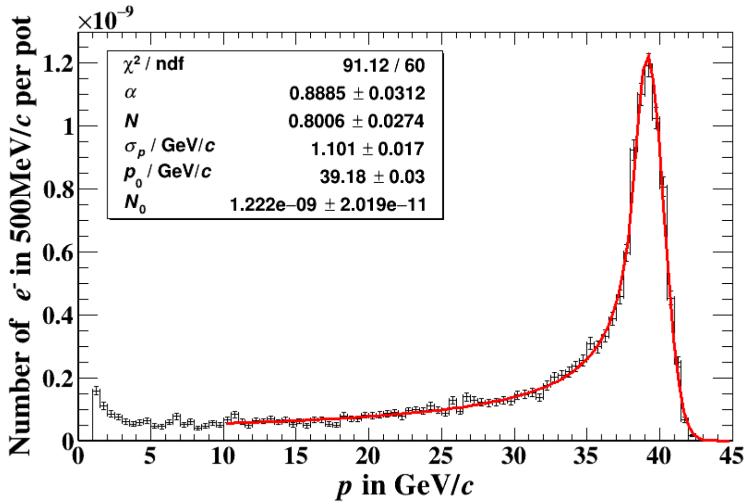
F. Metzger

- 400 GeV/c protons on T6 target
- Pions decay to give muons for COMPASS / AMBER
- Can look also at e- mode with extra target
  - degrade electrons to 40 GeV



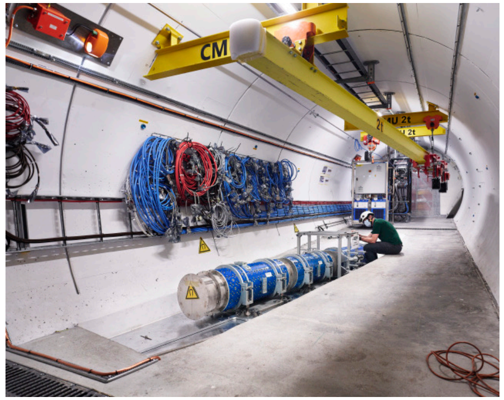
- Multi-stage simulation
- Custom geometry
- Field maps throughout

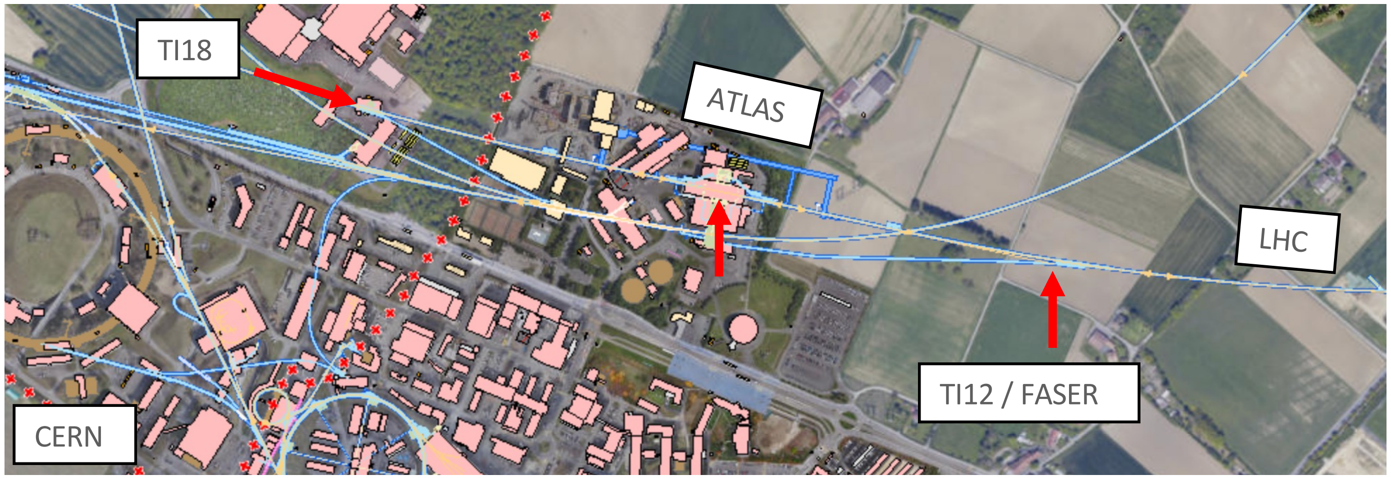




FASER - ForwArd Search ExpeRiment at the LHC

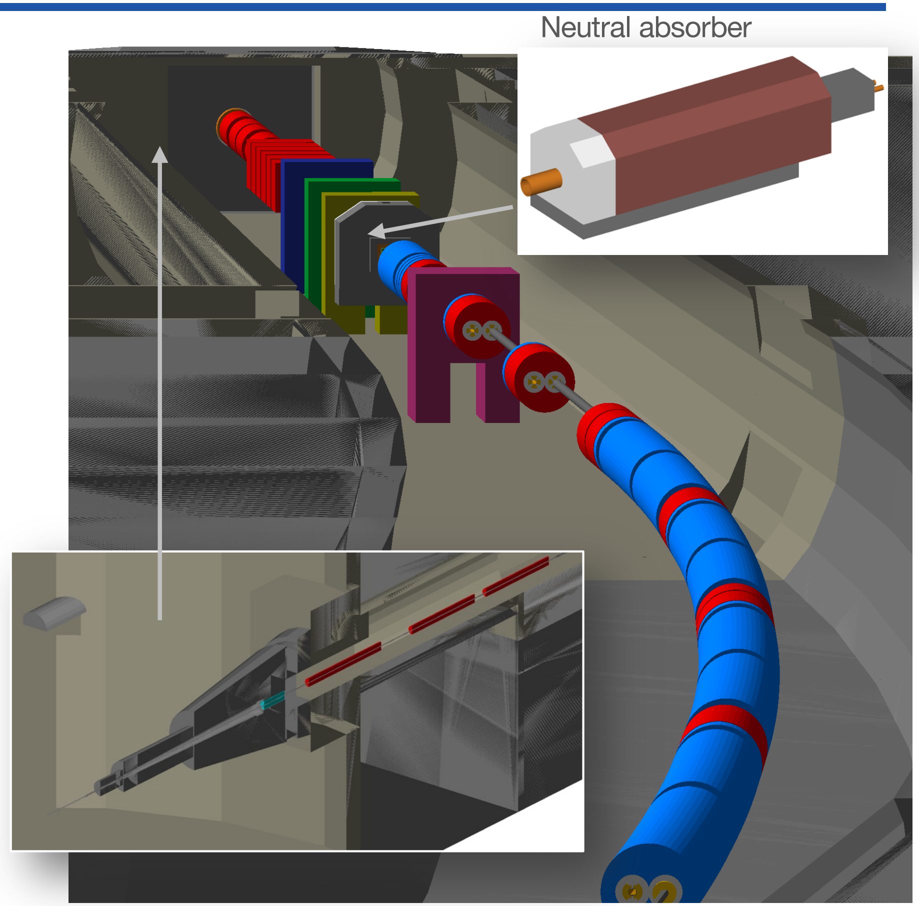
- New search for axion-like dark matter at the LHC
- Look in the very far forward direction along line of sight
  - Z = 475 m from IP1 (ATLAS) collision point
- Conception in 2017, designed and built -> just starting!
- Also study neutrinos in unique energy range
  - higher E_k than reactor sources and lower than astrophysical sources





FASER - A Simulation Challenge

- Want to understand backgrounds (i.e. muons)
  - model propagation and decays for systematics studies
- Need precision of 1 in 10^{12} per p-p collision
- Full "detector-like" model made of the accelerator in Geant4 using BDSIM from IP1
  - can simulate the various magnet settings during the run such as beam crossing-angle and divergence (from collision β*) → luminosity levelling
- Study significant differences seen in event generators - especially for neutrinos
- Understand contribution from secondary showers
  - this grows off-axis from 0 angle
- Study possibility of permanent-magnet sweeper upstream to reduce muon background further over a larger area

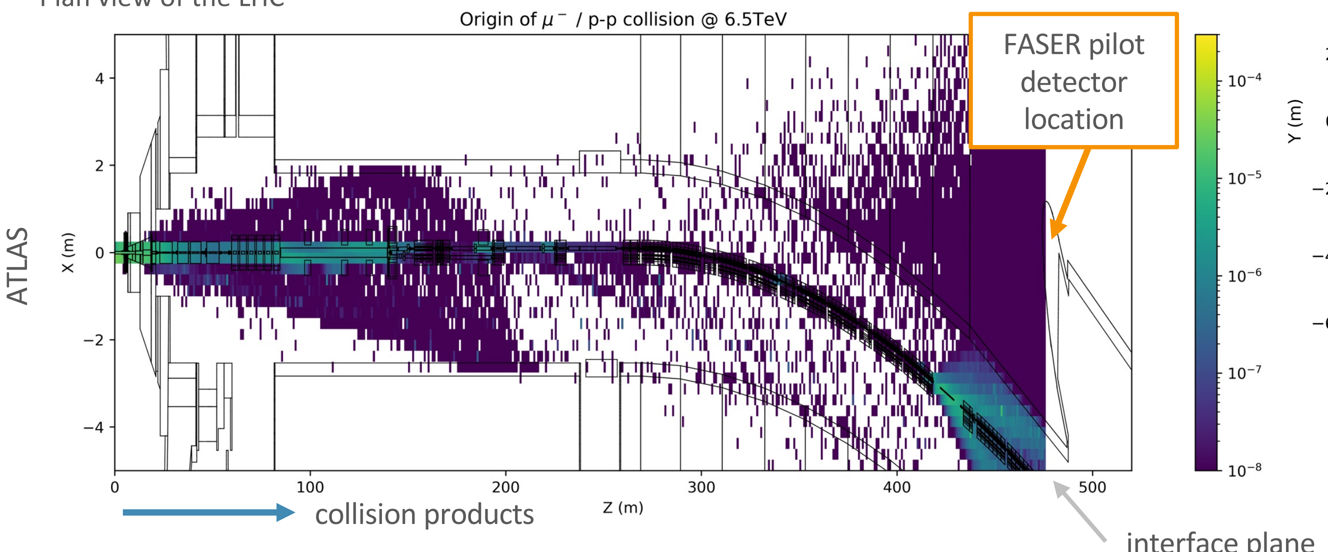


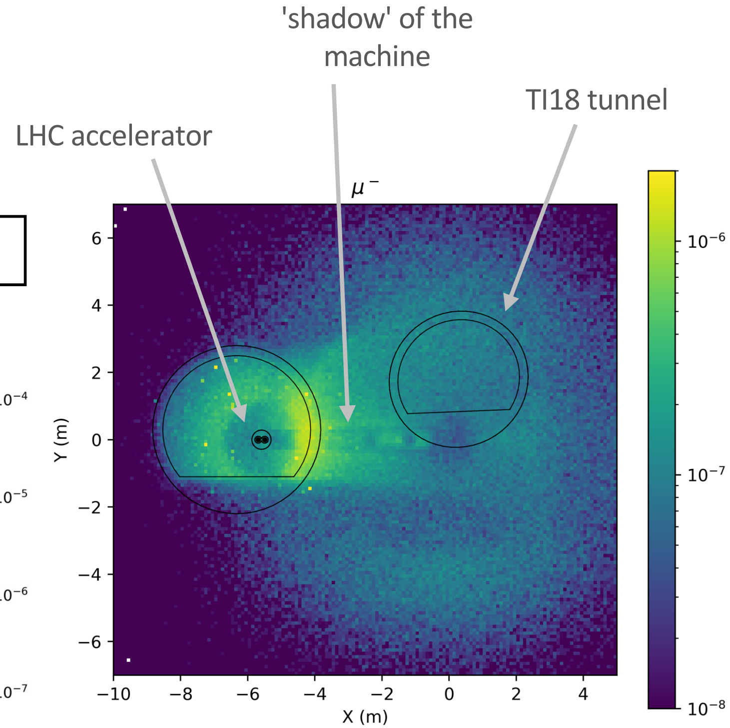
FASER Background & ν Modelling

- Understand the origin of muons reaching FASER
  - and TI18 pilot detector location from LHC Run II
- FASERν observed first neutrinos from the LHC
  - from small pilot detector

[Phys. Rev. D 104, L091101]

Plan view of the LHC

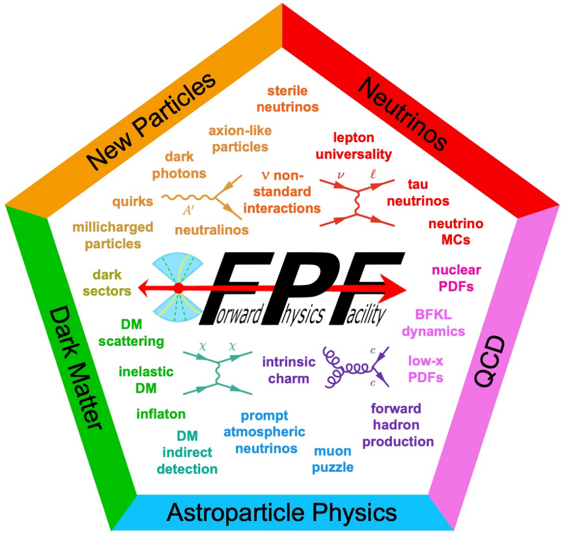


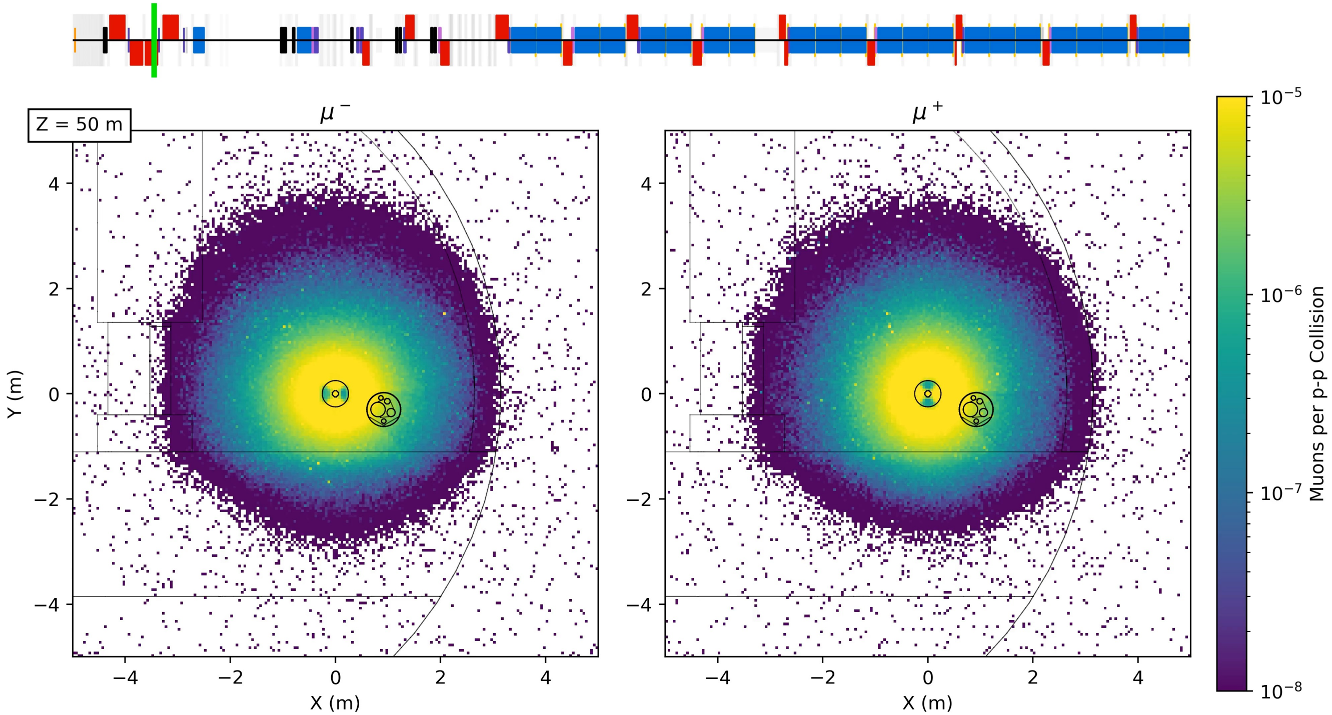


TI18 pilot detector plane simulated μ^- distribution per p-p collision at 6.5 TeV

Forward Physics Facility

- Proposed dedicated facility in new tunnel ~600 m from IP1
  - for HL-LHC era
- Studying sweeper magnet design and location

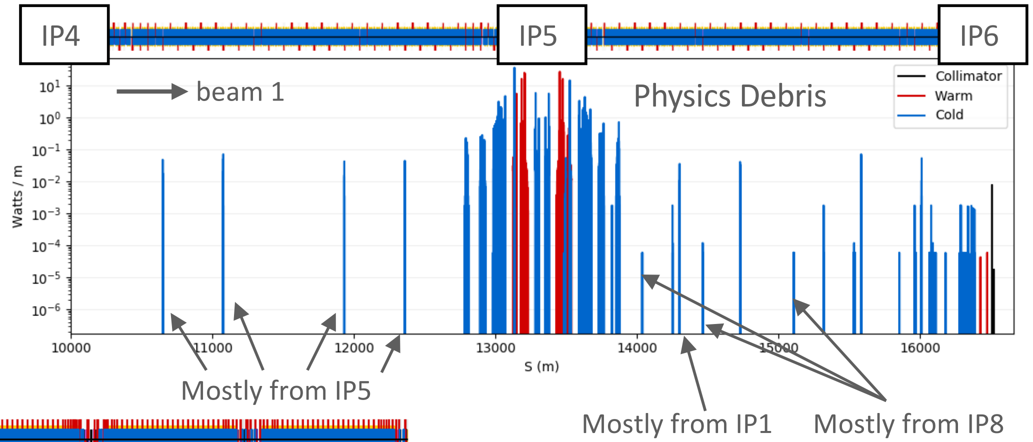


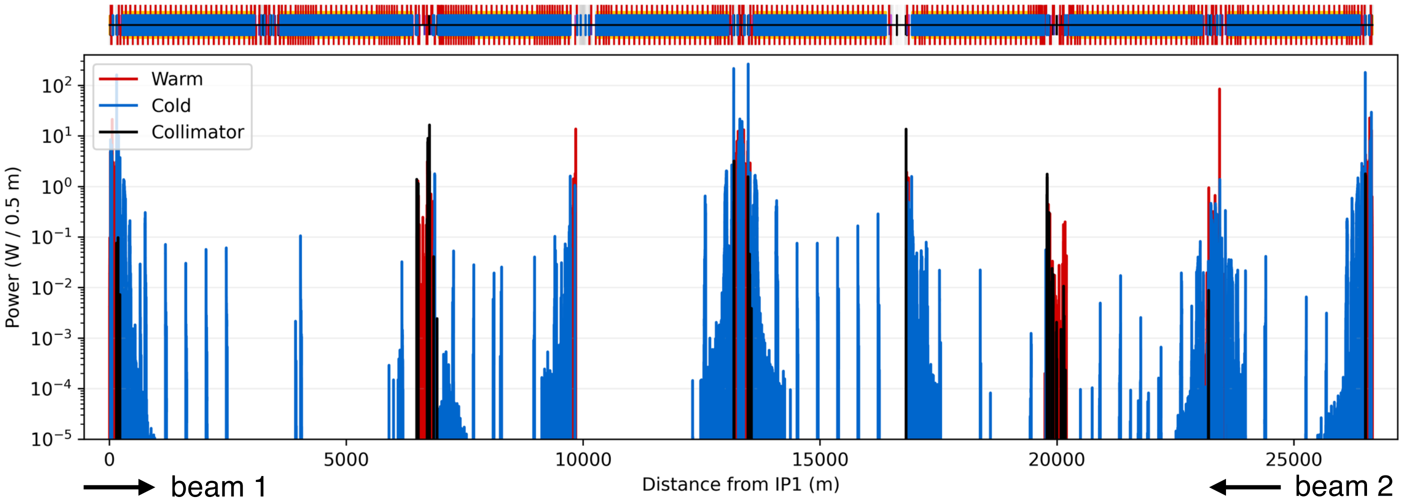


[Snowmass 2022 Paper https://arxiv.org/abs/2203.05090](https://arxiv.org/abs/2203.05090)

LHC Physics Debris Simulations

- A very interesting application is physics debris
  - scattered protons and secondaries can reach far from the experiments into the accelerator
  - certain beam loss monitors are highly correlated with luminosity and not with the stored beam intensity
- Simulate head-on p-p collision with event generator at IPs 1,5 and 8
- Simulation agrees well with BLMs





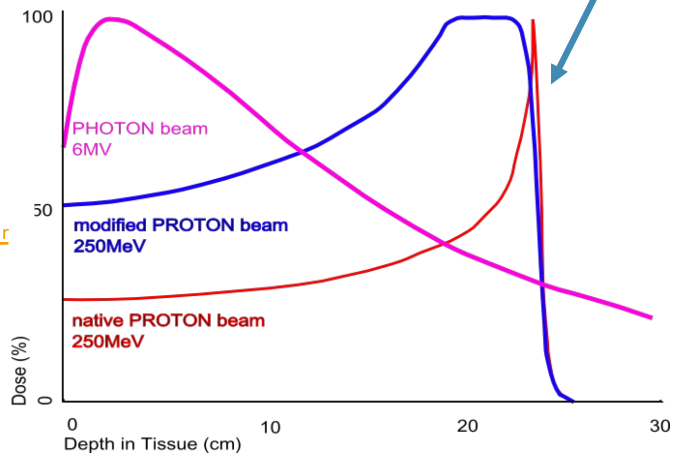
Weighted combination of each study according to luminosity

IP	Luminosity
1	1.5×10^{34}
5	1.5×10^{34}
8	0.05×10^{34}

Medical Hadron Therapy

- Protons and ions are used to treat cancers
  - greater relative biological efficiency (RBE) compared to X-rays
- Accelerator must move around patient
- Low energy + nozzle leads to large beam pipe and magnets
- Large national-level therapy centres
- Societal need greater than availability

Bragg curve



D. A. Miller, https://commons.wikimedia.org/wiki/File:BraggPeak.png

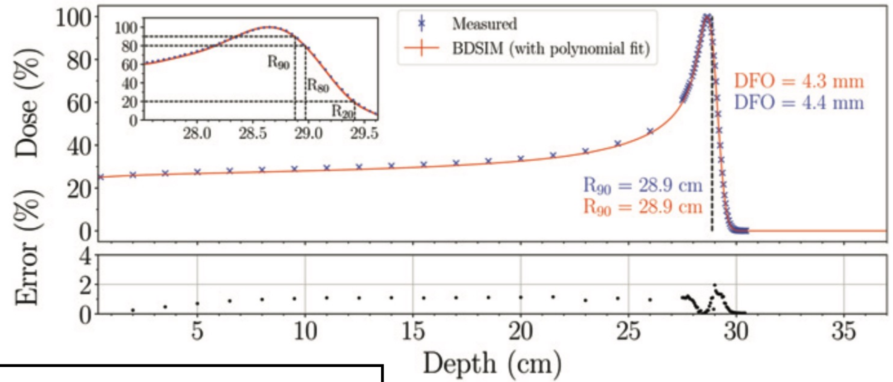
H. Owen et al., International Journal of Modern Physics A (29), 14, 1441002 (2014)

BDSIM Medical Simulations

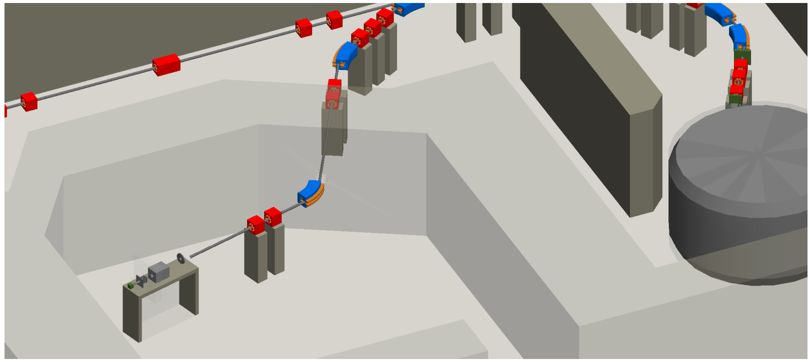
- First start-to-end gantry simulation with particle-matter interaction - from cyclotron to Bragg Peak
  - excellent experimental agreement
- Recent study on increasing dose rate in eyeline

  [Phys. Rev. Research 4, 013114 (2022)]

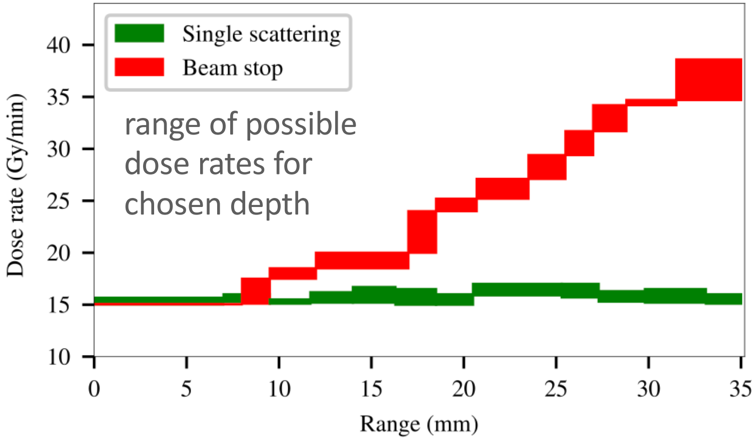
- Results: reduce distal fall-off and reduce treatment time by factor of 3: treatment in < 30s
  - huge improvement for patient holding still
- ULB contributing to new BDSIM features regularly



[EPL, 132 (2020) 50004]

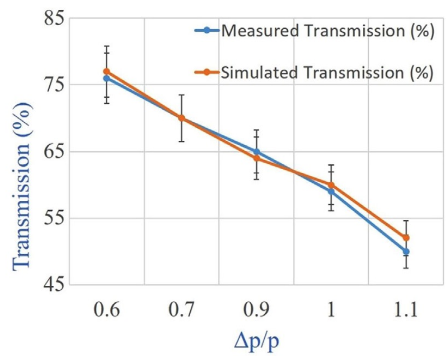


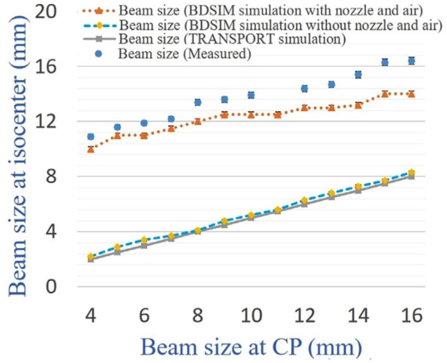
IBA Proteus©Plus Model with eyeline

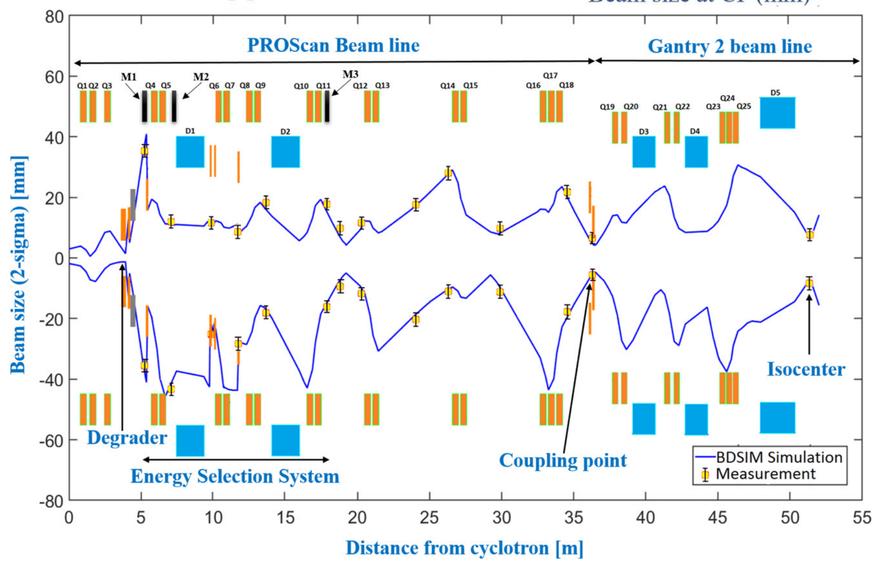


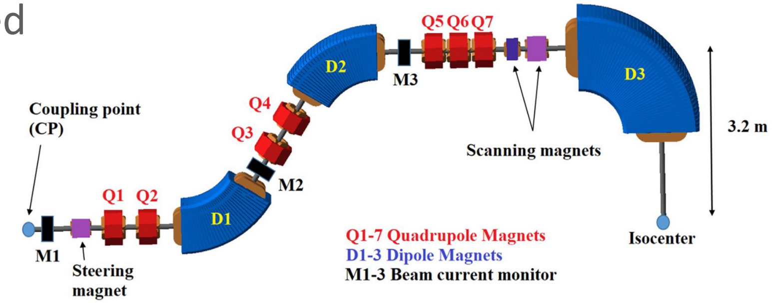
Recent Results by Others with BDSIM

- PSI designed a new emittance selection system to increase transmission by a factor of 6 for goal of FLASH therapy
  - https://doi.org/10.1002/mp.15278
- Also, new optics for PSI Gantry 2 for greater acceptance
  - https://doi.org/10.1002/mp.15505
- Goal of higher transmission
  - new optics and coupling point (CP)
- Excellent agreement with experiments achieved









PSI Gantry2 Model

Conclusions & Outlook

- Combined particle physics and accelerator tracking simulations
- Many current and future experiments are not in an isolated environment and link with the accelerator
- The coupling between the two cannot always be ignored
- Tools presented for geometry preparation and conversion
  - a single person can achieve and learn something quickly!
  - it is possible to have parity between FLUKA & Geant4 models

Thank you & Questions?

LAURIE NEVAY

laurie.nevay@cern.ch

GEANT4
A SIMULATION TOOLKIT

BDSIM
Beam Delivery Simulation

BDSIM - website - manual - paper

[1] Title slide image credit; CERN + BDSIM model

Backup

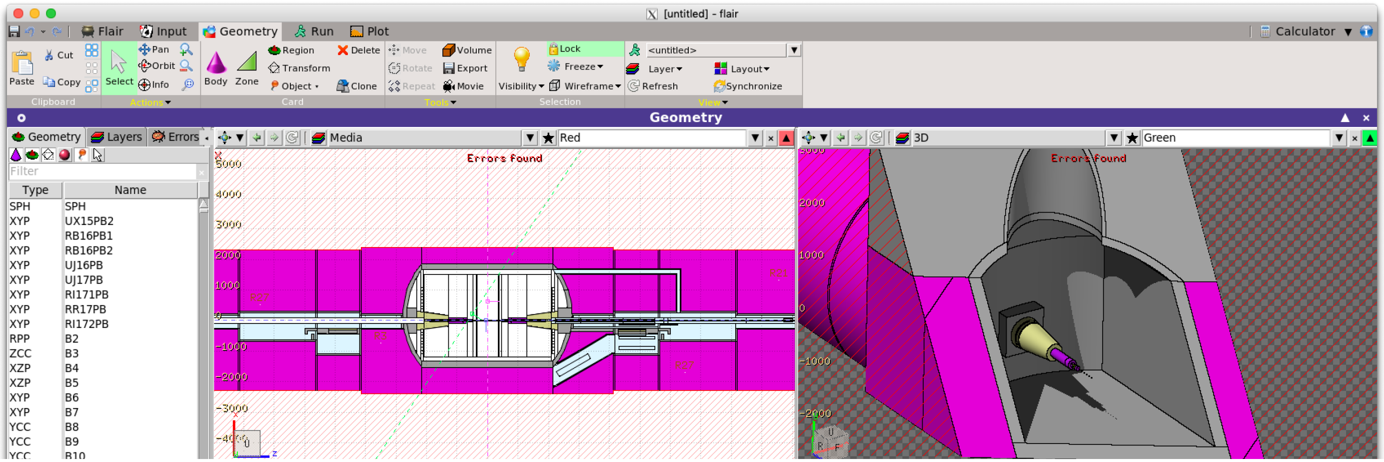
Geometry Formats - Possible Sources

- GDML
  - persistency format of Geant4 - made for single file dump of geometry
  - XML
  - NB solids parameterised different to Geant4 C++
- Geant4 C++
  - developer writes C++ in their application using Geant4
  - requires compilation to change (typically)
  - Geant4 constructors of solids register themselves in static registries (1 complete geometry)
- FLUKA
  - text input - a bit hard to read, but comes with FLAIR GUI to create, inspect and debug
  - can have coplanar faces as faces can be used by both 'sides' of a shape
  - need to 'cut' out air as flat hierarchy
- CAD
  - many formats - common features not often present in radiation transport models, e.g. chamfering
- Tessellated Solids / Meshes
  - surface represented by series of polygons - commonly used for finite element simulations
  - potentially huge mesh and slow tracking (lots of research on this!)
  - problems if mesh has holes (waterbag)

FLUKA Geometry

- FLUKA is an alternative to Geant4 and typically used for radiation shielding applications and machine protection
  - also used for variety of nuclear applications
- Certified for radiological protection by many organisations
  - therefore they often require a FLUKA model at some point
- A lot of geometry in this format at CERN for example
- Uses combinatorial geometry with *infinite-half* solids and *no hierarchy*

https://fluka.cern



Multi-Stage Simulations

- Several tools included for skimming, combining, analysis etc.
- Complex workflows easily possible