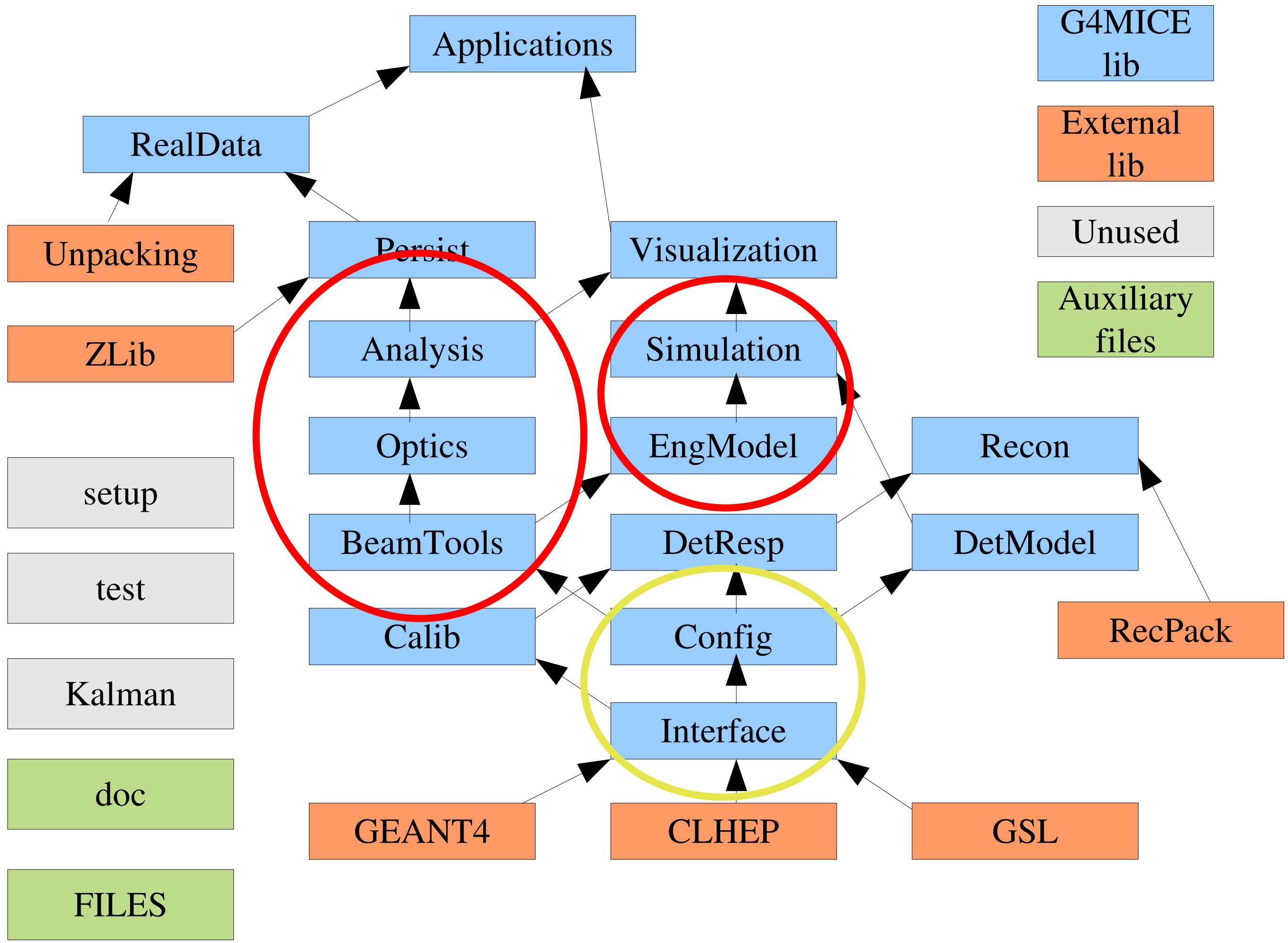
Accelerator Tools - Status

Chris Rogers,
ASTeC,
Rutherford Appleton Laboratory

G4MICE Accelerator Physics

- Simulation
  - Renovation of **multipole** field map
  - Implementation of **physics list tests**
- Optics
  - New **PolyFit** algorithm
- Analysis
  - Python tools **XBOA**
- Each topic is probably ~ 1 hour talk


Applications
RealData
Unpacking
Persist
Visualization
ZLib
Analysis
Simulation
Optics
EngModel
Recon
BeamTools
DetResp
DetModel
Calib
Config
RecPack
Interface
GEANT4
CLHEP
GSL
setup
test
Kalman
doc
FILES
G4MICE lib
External lib
Unused
Auxiliary files

G4MICE

- Simulation
  - Renovation of multipole field map
  - Implementation of physics list tests
- Optics
  - New PolyFit algorithm
- Analysis
  - Python tools

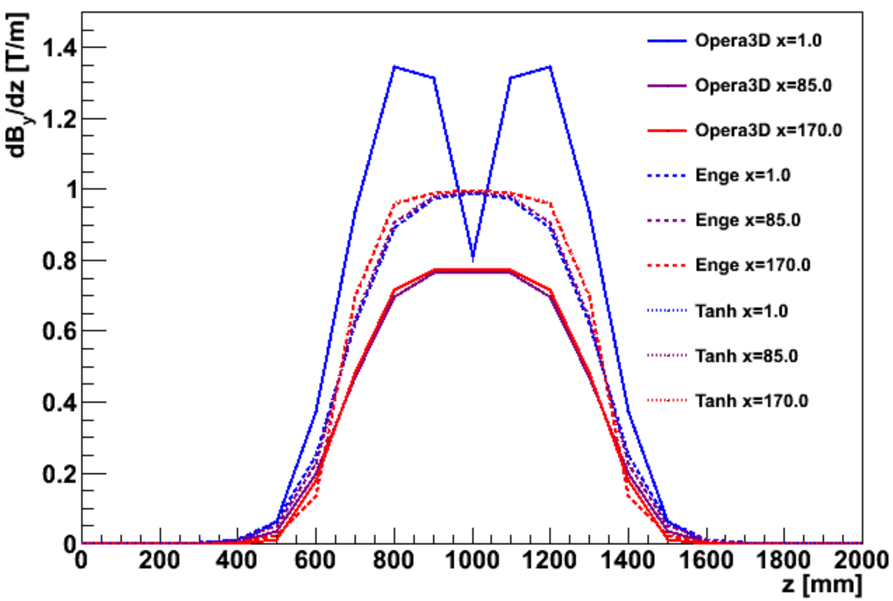
Multipole field map

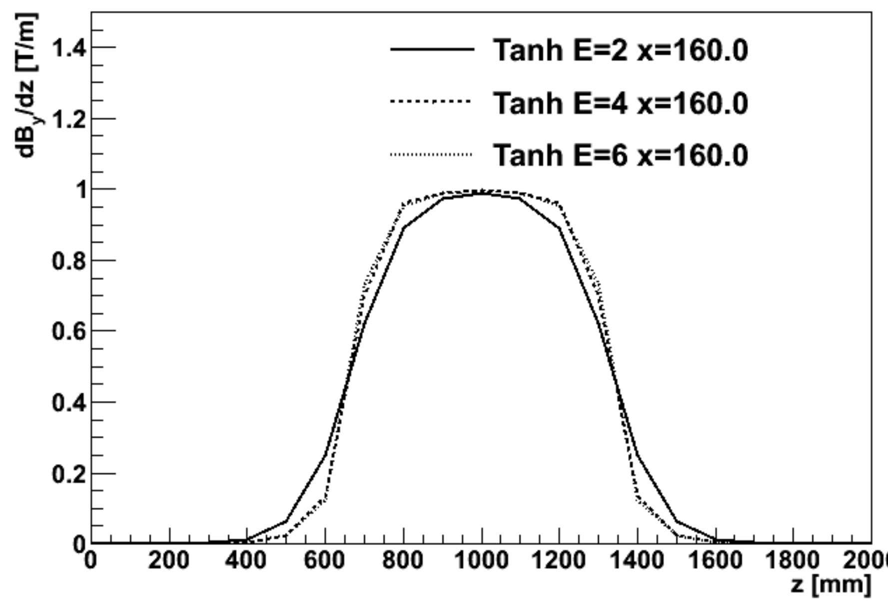
- Issue with QC field map
  - Generated in ancient history by Opera3d
  - Non-physical dipole field was found on axis
- Fall back to analytical model for multipole field
  - Scalar magnetic potential like:

$$V_n = \sum_{q=0}^{q_w} \sum_{m=0}^{n} n!^2 \frac{G^{(2q)}(s)(r^2+y^2)^q \sin\left(\frac{m\pi}{2}\right) r^{n-m} y^m}{4^q q!(n+q)!m!(n-m)!}$$

  - *G(s)* is some function that tells how the field gradient drops along the axis
  - r is horizontal displacement, s is longitudinal displacement, y is vertical displacement, n indexes pole type (dipole, quad, sextupole, ...)
  - Recursion relation from Div **B** = Div (Grad(V))=0 to get magnetic field off axis
    - i.e. apply Maxwell's laws
- Algorithm was unfortunately broken and as I dug, it became clear major refurbishment was necessary

Multipole field map



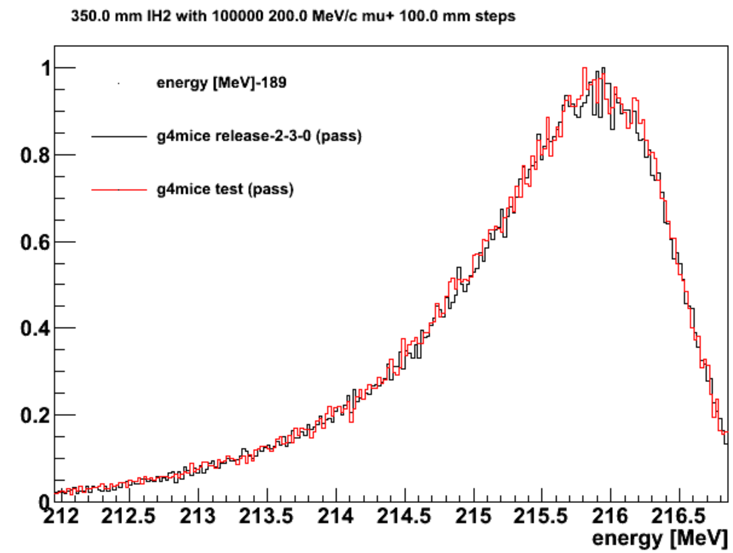


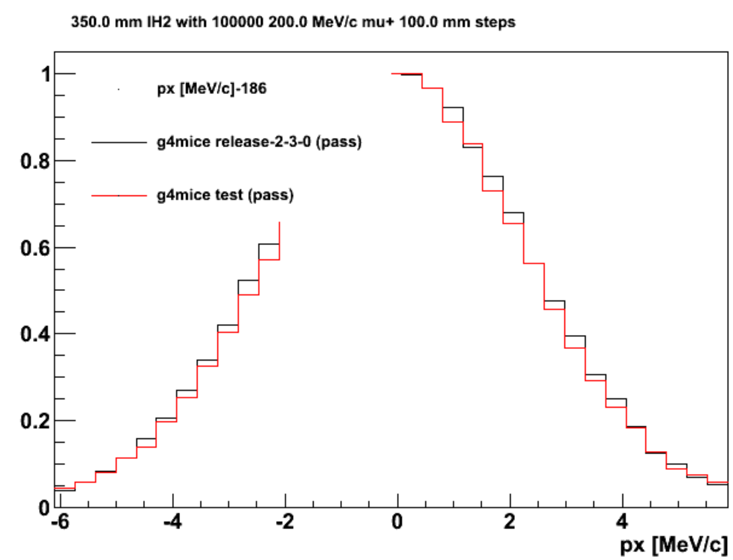
Multipole Model Status

- Implemented
  - Enge model
    - I regard this as too slow - needs some optimisation work
  - Tanh model
    - Acceptable speed
      - 1e3 μ
      - 6 quads
      - 10 m channel
      - 10 mm step size
      - Takes ~30 s on my laptop
    - Further optimisation is possible if it's required
- New code is reasonably well commented
- ~30 Unit tests implemented in gunit test framework
- Application tests in Integration/Simulation/Multipoles
  - Fire neutrons through field map and look at field reported in VirtualPlane output
  - Compare reference output with new output with tolerance 1 mT

Physics List tests

- G4MICE was previously not able to simulate pions
  - Physics processes loaded had some problem with hadronic interactions
- Implement G4 physics package
  - Default is QGSP_BIC
- Make physics list regression tests
  - Compare versions against some reference version
  - Apply Kolmogorov-Smirnov test to determine if distributions are the same
  - Produce reference plots posted to web
- Code can be run against G4MICE, ICOOL, G4Beamline, etc





Optics::PolyFit algorithm

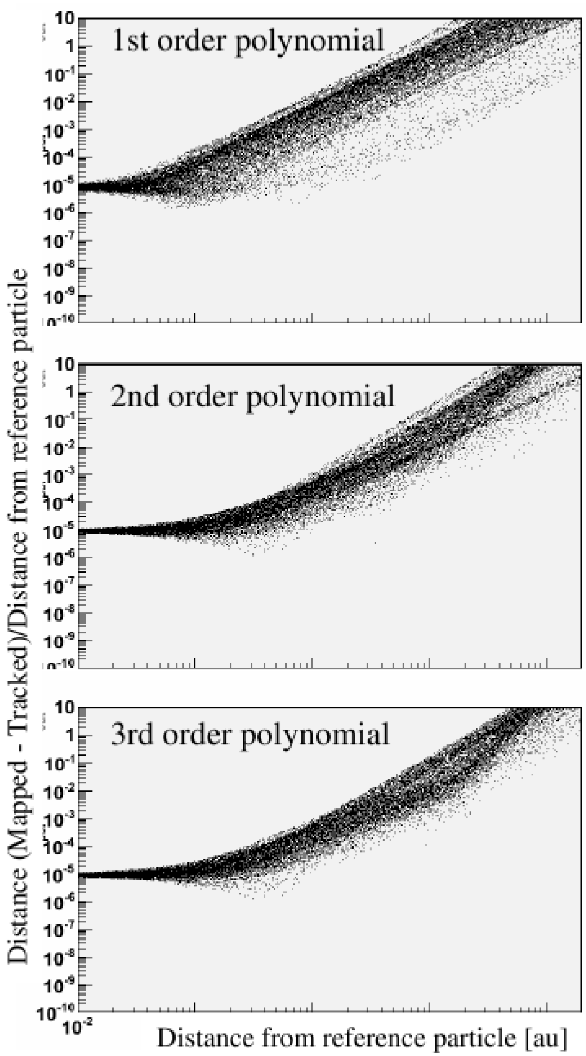
- Fundamental problem in (low intensity) accelerator physics is to find mapping that transports particles from some point at z_1 to some other point at z_2
  - From this mapping, we can transport beam ellipses etc
- Optics *TrackingDerivative* algorithms calculate Jacobian numerically, dx_{out}/dx_{in}, dp_{out}/dx_{in}, and so forth
- Allows us to calculate transfer matrix directly from Geant4 tracking
  - Nice test -> Are (complex) eigenvalues on unit circle
  - Allows to transport beam envelopes
  - Allows to transport single particles
- New algorithm *PolyFit* has now been implemented

Optics::PolyFit algorithm (2)

- New algorithm *PolyFit* has now been implemented
  - Here we fit a polynomial to tracking output
  - So we generate a polynomial $(time,x,y,px,py,energy)_{out}$ as a function of $(time,x,y,px,py,energy)_{in}$
  - Use PolynomialVector - arbitrary order, arbitrary dimension polynomial routine
    - With arbitrary order LLS fitting
- Idea is to find higher order terms in the transfer map
  - *TrackingDerivative* is not really stable beyond the linear mapping
  - Inputs:
    - step size
    - tolerance in residuals between tracked data and mapped data
    - polynomial order
  - Dynamically seeks a transfer map with appropriate properties

Optics::PolyFit algorithm (3)

- E.g. for 6D (right)
  - *Amplitude of the residual* vs amplitude
  - Amplitude of the residual is amplitude of distance between mapped particles and actual particles (in 6d)
  - For some RF/magnet lattice
- Get some improvement from 2nd order
  - No improvement really at 3rd order
- Nice tool for e.g. comparing measured data transfer matrix with calculated transfer matrix

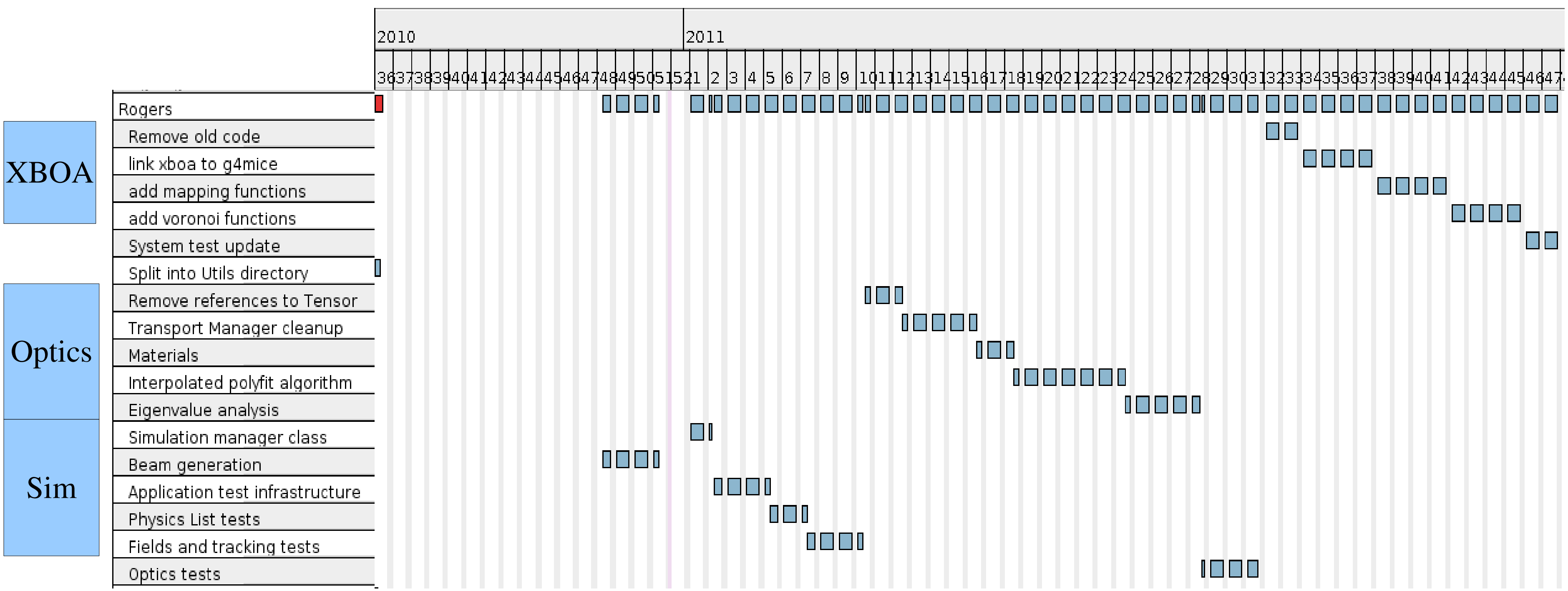


Introducing XBOA

- MICE analysis software requirements
  - Physics analysis routines for MICE control room
    - Beta, alpha, emittance, plotting
    - Need fast turnaround of plots
- Current physics analysis routines are not fit for purpose
  - Written by some rubbish graduate student as a first coding project
- Instead use XBOA physics analysis library
  - ~ few thousand lines of code
  - Good test coverage
  - Well documented (function-by-function + worked examples)
  - Local expertise
  - Written by some rubbish RAL staff guy
- Allows for quick and dirty physics analysis online

Accelerator Side workplan to 2012

- Most functionality is already in place
  - So structure looks rather flat (dependencies are already “done”)
  - Still a lot to do
  - Additional tasks:
    - MAUS integration not listed but is expected to be ~ 2 months work and is not listed
    - Arbitrary 3D meshing routines for measured field maps need to go in



Final comment

- Significant amount of development work in last 4 months has been bug fixing
  - Either we have most of our functionality in place
  - Or most of our code is broken
- Focus on improving test coverage and analysis routines