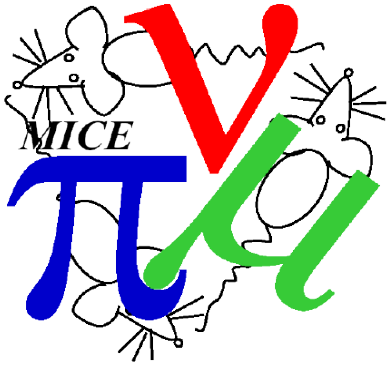
MICE Software and Analysis

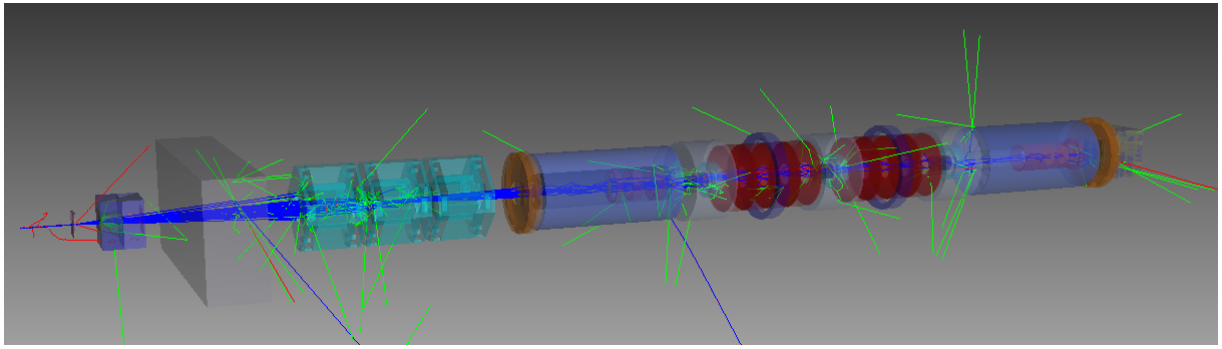


Chris Rogers,
ASTeC,
Rutherford Appleton Laboratory

Overview

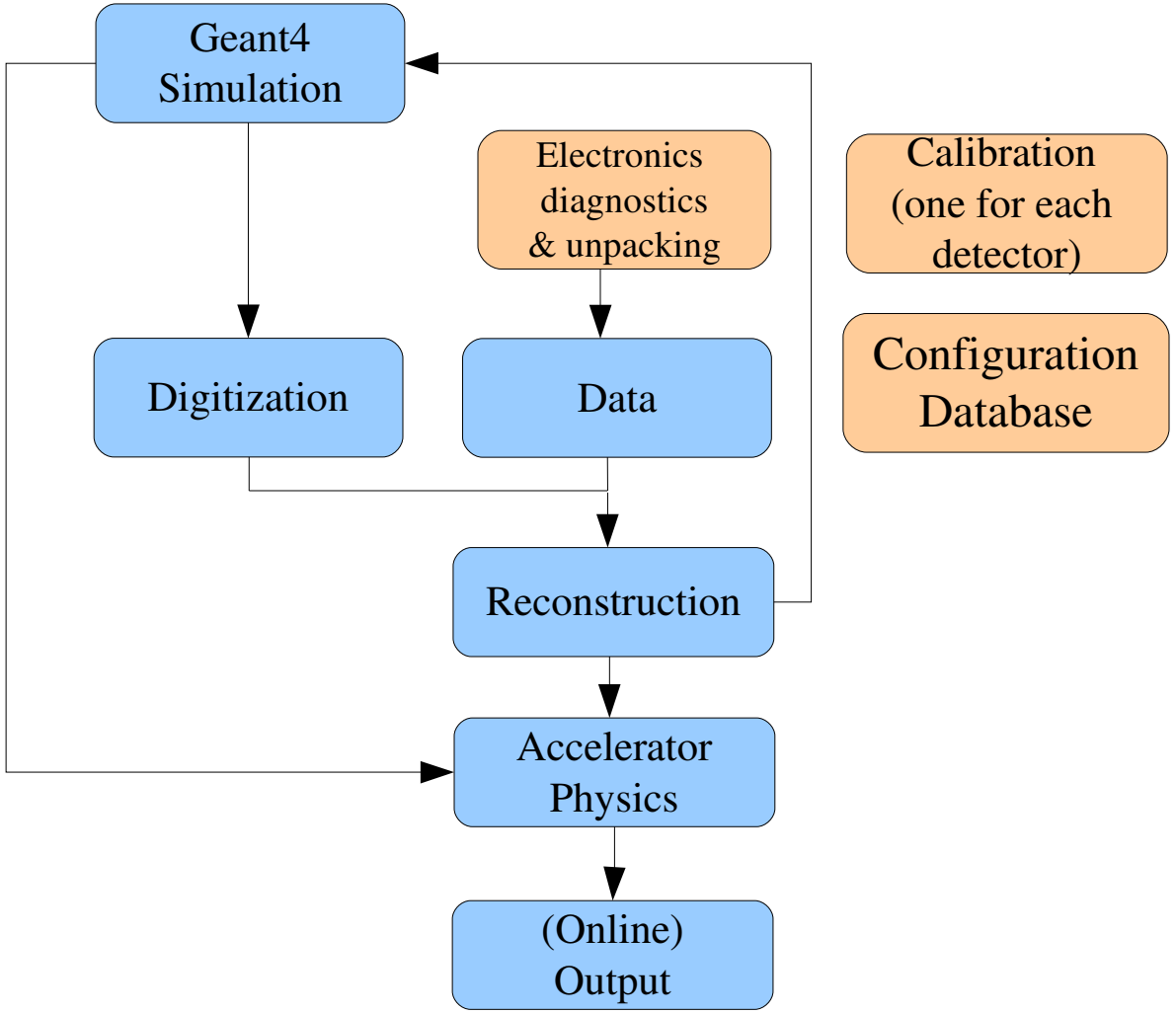
- MICE physics software refurbishment
  - Modular system to plugin different detector reconstructions
  - Distributed processing for online operation
  - Accelerator physics routines
  - Continuous Integration
- Data analysis results
  - Reconstructed phase space plots
  - Quality of match into the cooling channel

MICE Software aims

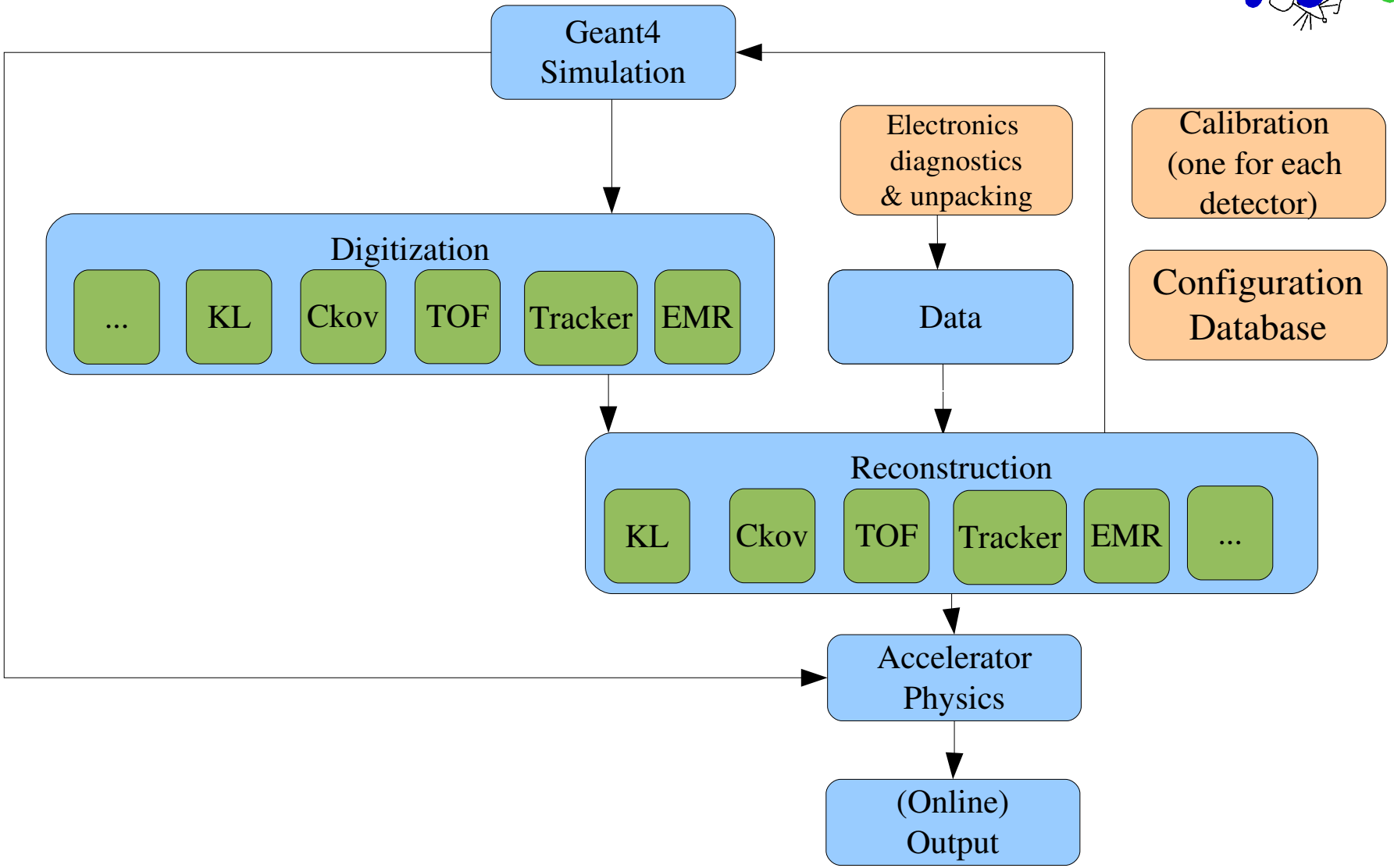


- Detector reconstruction
- Global reconstruction
- Tracking simulation of MICE
- Physics analysis tools
- Data quality check (online)
- Detector response
- Fast simulation of MICE (transport matrices)
- QA process
- Documentation
- Geometry model

Data Flow


Geant4
Simulation
Electronics
diagnostics
& unpacking
Calibration
(one for each
detector)
Digitization
Data
Configuration
Database
Reconstruction
Accelerator
Physics
(Online)
Output

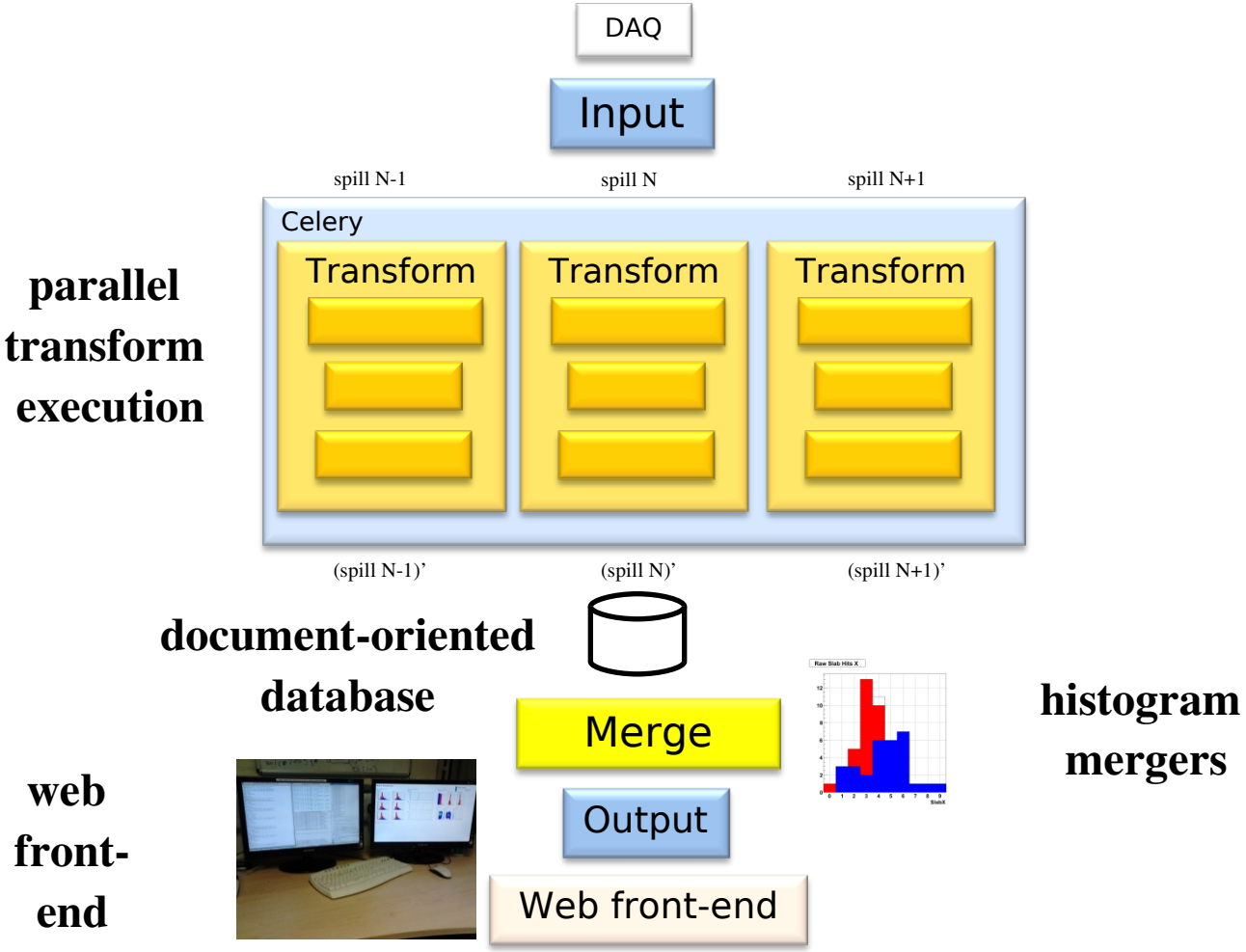
Data Flow


Geant4
Simulation
Electronics
diagnostics
& unpacking
Calibration
(one for each
detector)
Digitization
...
KL
Ckov
TOF
Tracker
EMR
Data
Configuration
Database
Reconstruction
KL
Ckov
TOF
Tracker
EMR
...
Accelerator
Physics
(Online)
Output

API

- Plugin design
  - Inspired by map-reduce framework (Hadoop/google, etc)
  - input-transform-merge-output
- **Input**: Read in data
  - Access the socket
  - Read in a binary DAQ data file
  - Read in a beam file for Monte Carlo
- **Transform**: Process a spill; return modified spill data
  - Monte Carlo simulation
  - Reconstruction for each detector
- **Merge**: Summarise data from many spills
  - Histograms for monitoring each detector performance
  - Some accelerator analysis functions?
- **Output**: Write out data
  - Write in ROOT format
  - Write in Json format

Distributed Processing (Jackson)



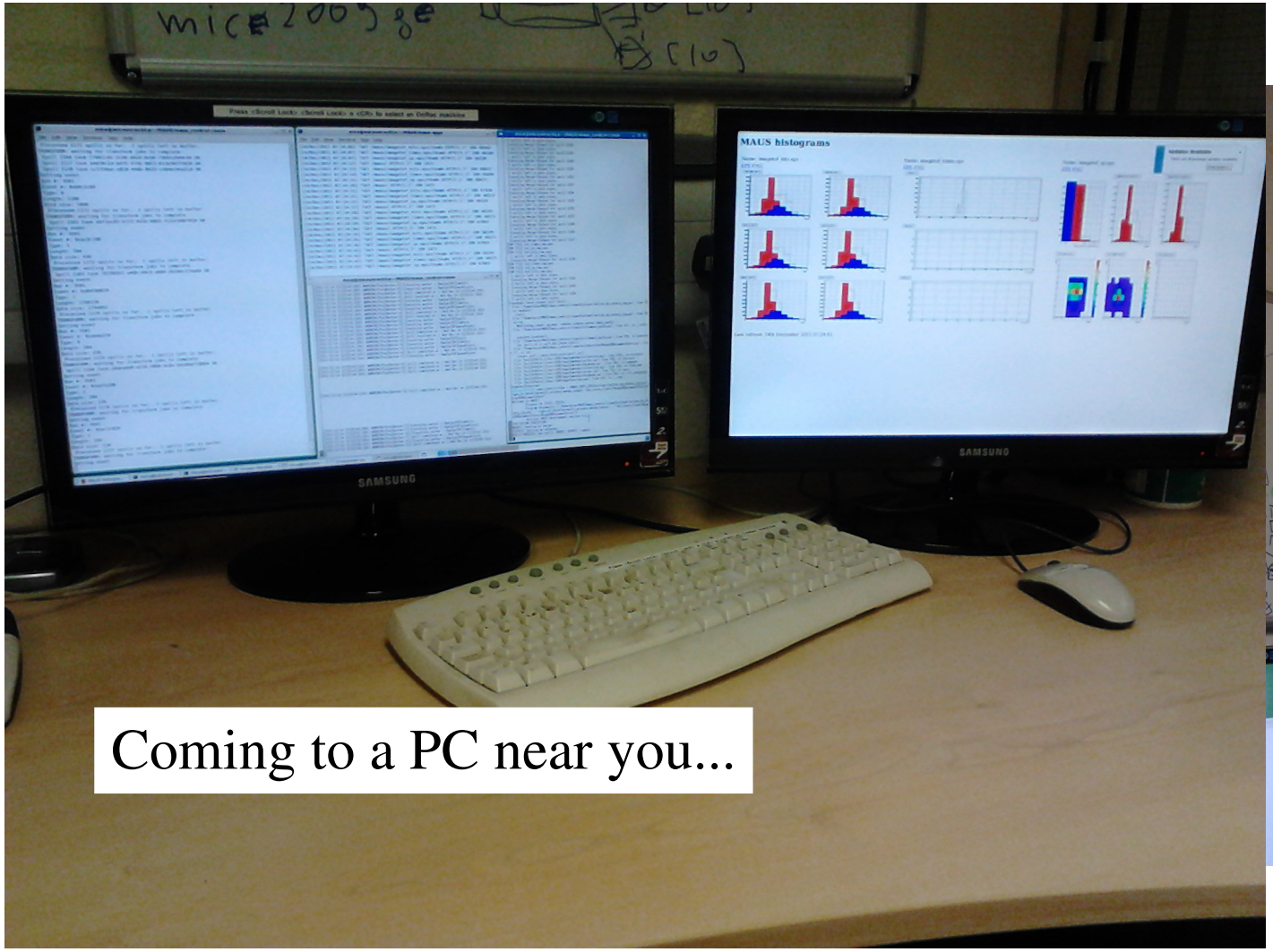
parallel transform execution

document-oriented database

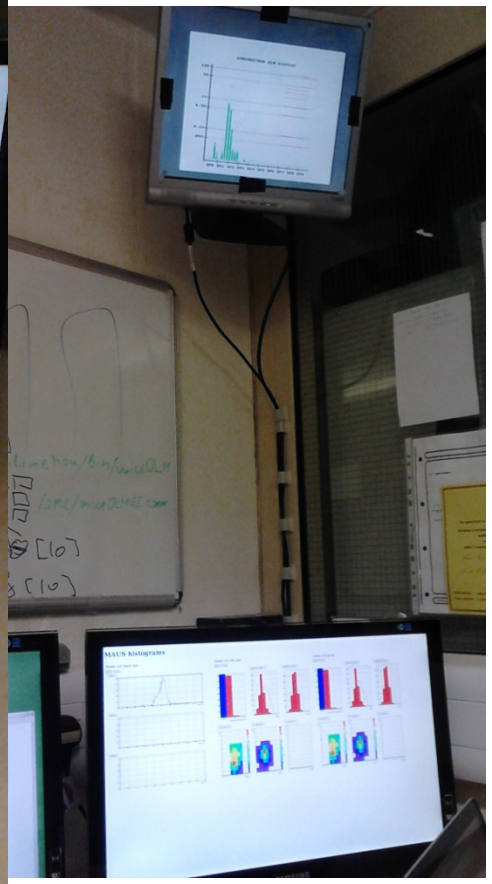
web front-end

histogram mergers

Current state (Jackson)

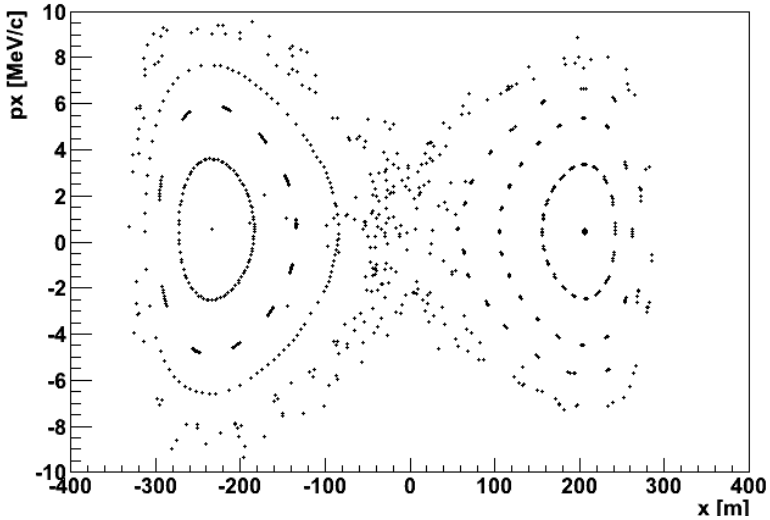


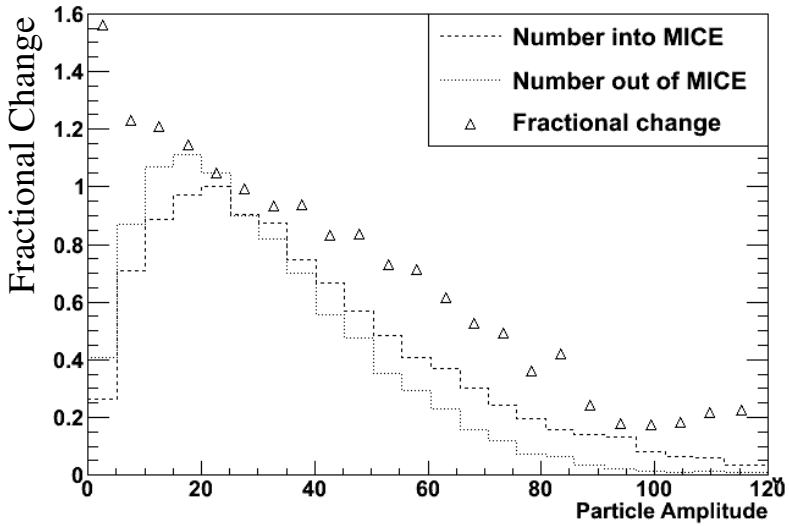
Coming to a PC near you...



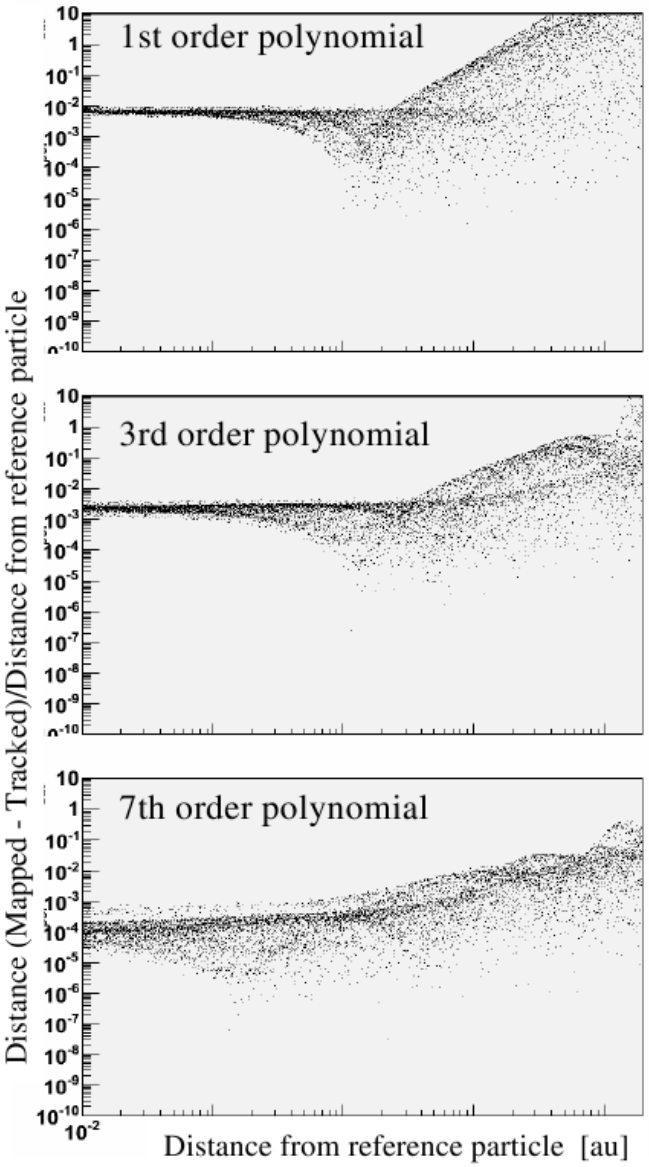
Beam Physics Routines

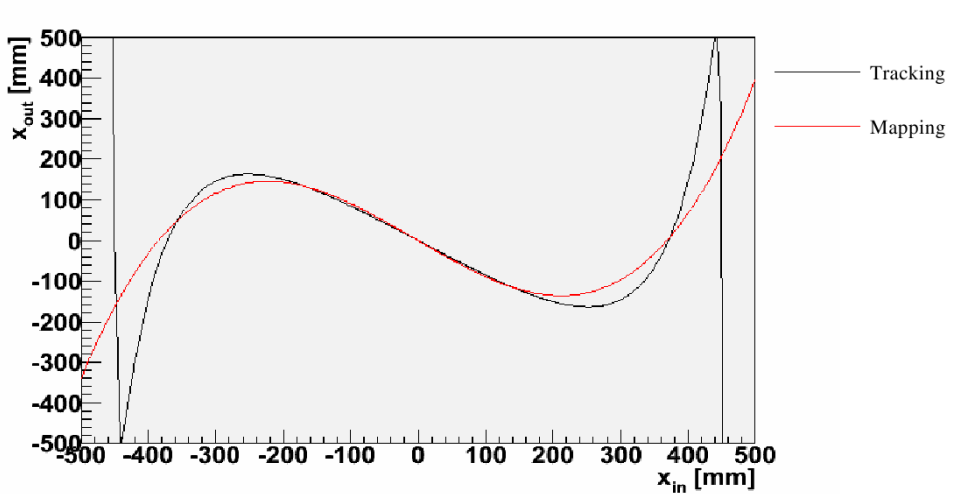
- Beam analysis and plotting routines
  - Scriptable interface (python)
  - Plotting bindings to ROOT and matplotlib
  - Calculate emittances, twiss functions, etc
    - Fully 6D
  - Cuts
  - File IO in MAUS, G4BL, ICOOL, MARS, etc





Accelerator Optics Routines

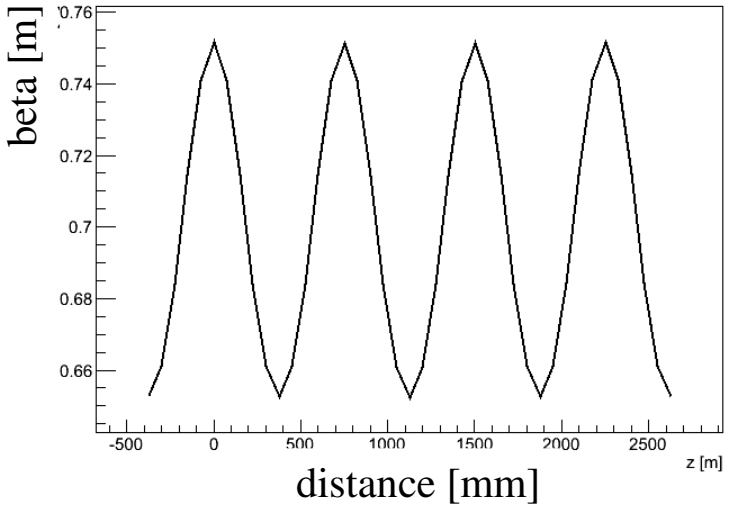


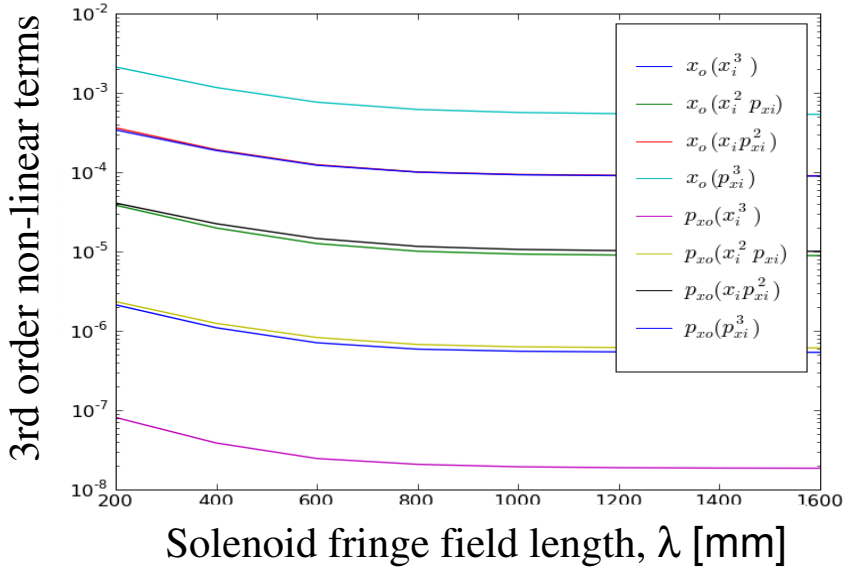


- Extract transfer maps from tracking
  - Use numerical differentiation of tracking
  - Use least squares fit to tracking
  - Fit to arbitrary order
  - Work in arbitrary dimensional phase space
  - Limit is numerical precision of tracking

Matching Routines

- Sort of plots we can make
  - β function down the beamline
  - Response to solenoid fringe field
- Set of scripts including optimiser
  - (Powered by minuit)
  - Optimise for current, twiss parameters, penn parameters, ...
- But needs some cleanup
  - Would be nice to plugin to G4BL, OPAL, GPT
  - In the tar pit



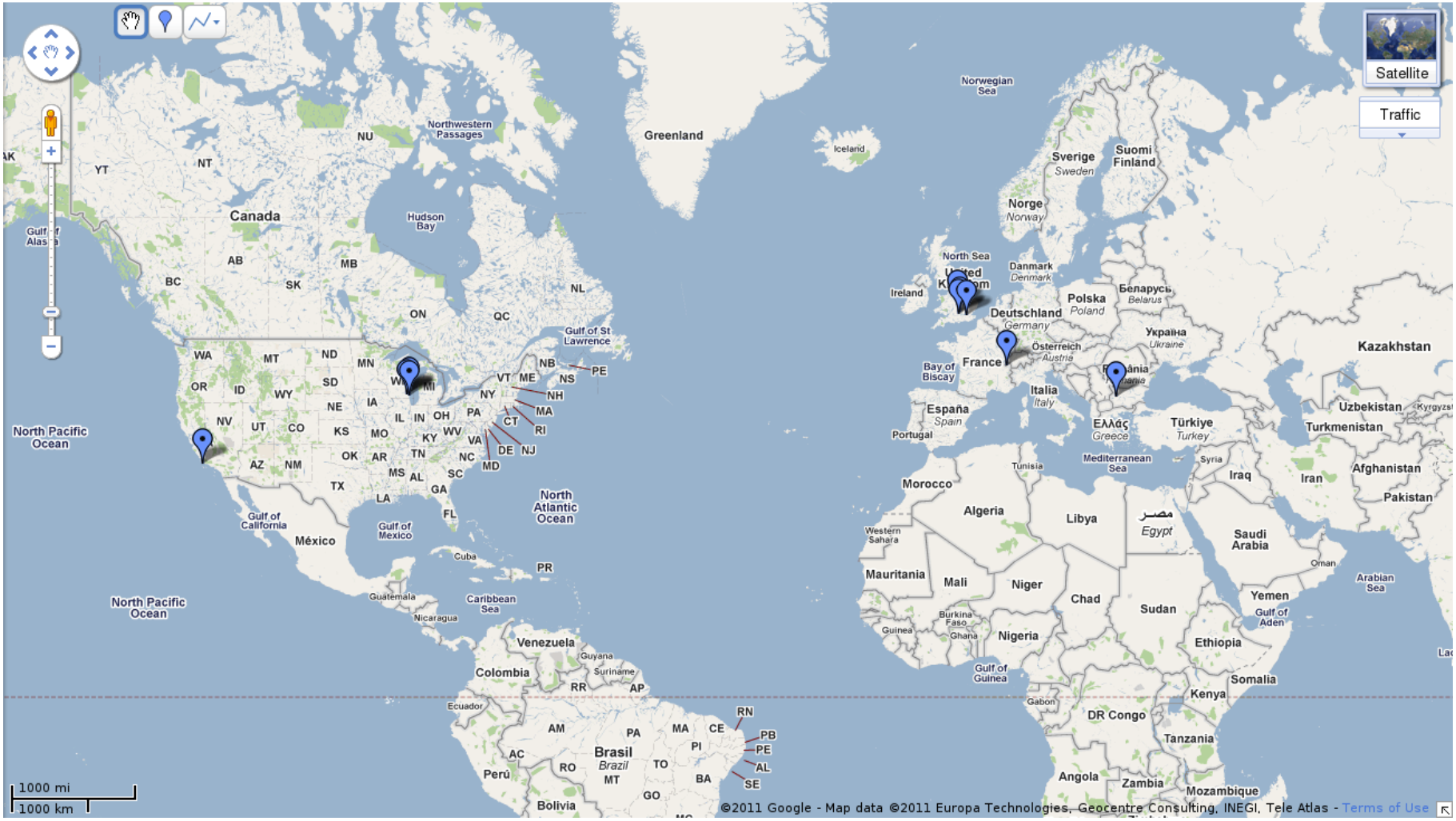


The Tar Pit

No scene from prehistory is quite so vivid as that of the mortal struggles of great beasts in the tar pits. In the mind's eye one sees dinosaurs, mammoths, and sabertoothed tigers struggling against the grip of the tar. The fiercer the struggle, the more entangling the tar, and no beast is so strong or so skillful but that he ultimately sinks.

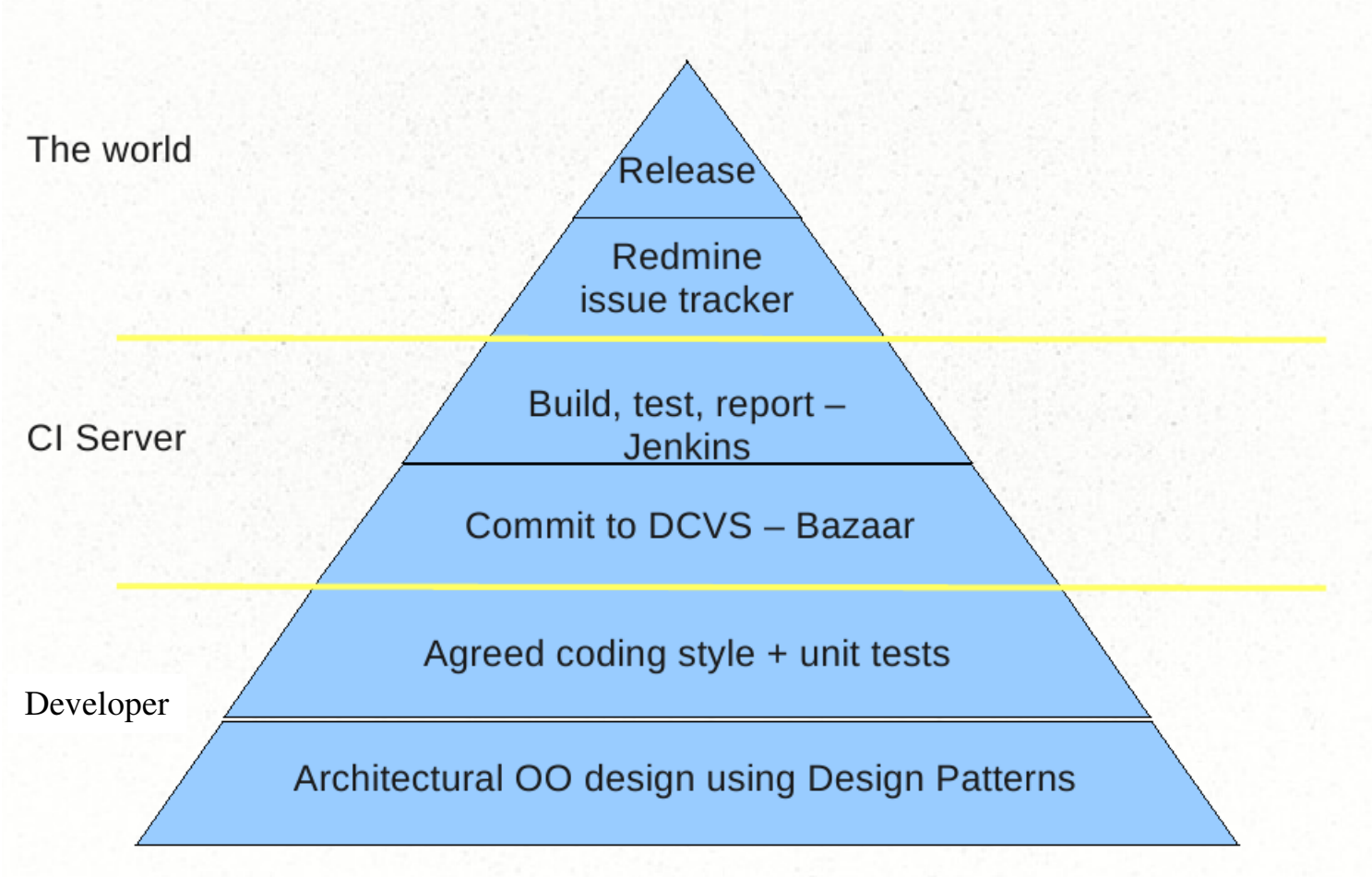
Large-system programming has over the past decade been such a tar pit, and many great and powerful beasts have thrashed violently in it. Most have emerged with running systems—few have met goals, schedules, and budgets. Large and small, massive or wiry, team after team has become entangled in the tar. No one thing seems to cause the difficulty—any particular paw can be pulled away. But the accumulation of simultaneous and interacting factors brings slower and slower motion. Everyone seems to have been surprised by the stickiness of the problem, and it is hard to discern the nature of it. But we must try to understand it if we are to solve it.

MAUS developers



Managing Complexity

- 10-20 developers, most with no experience working part time on the wrong side of the world
- Few hundred components, each with several interfaces



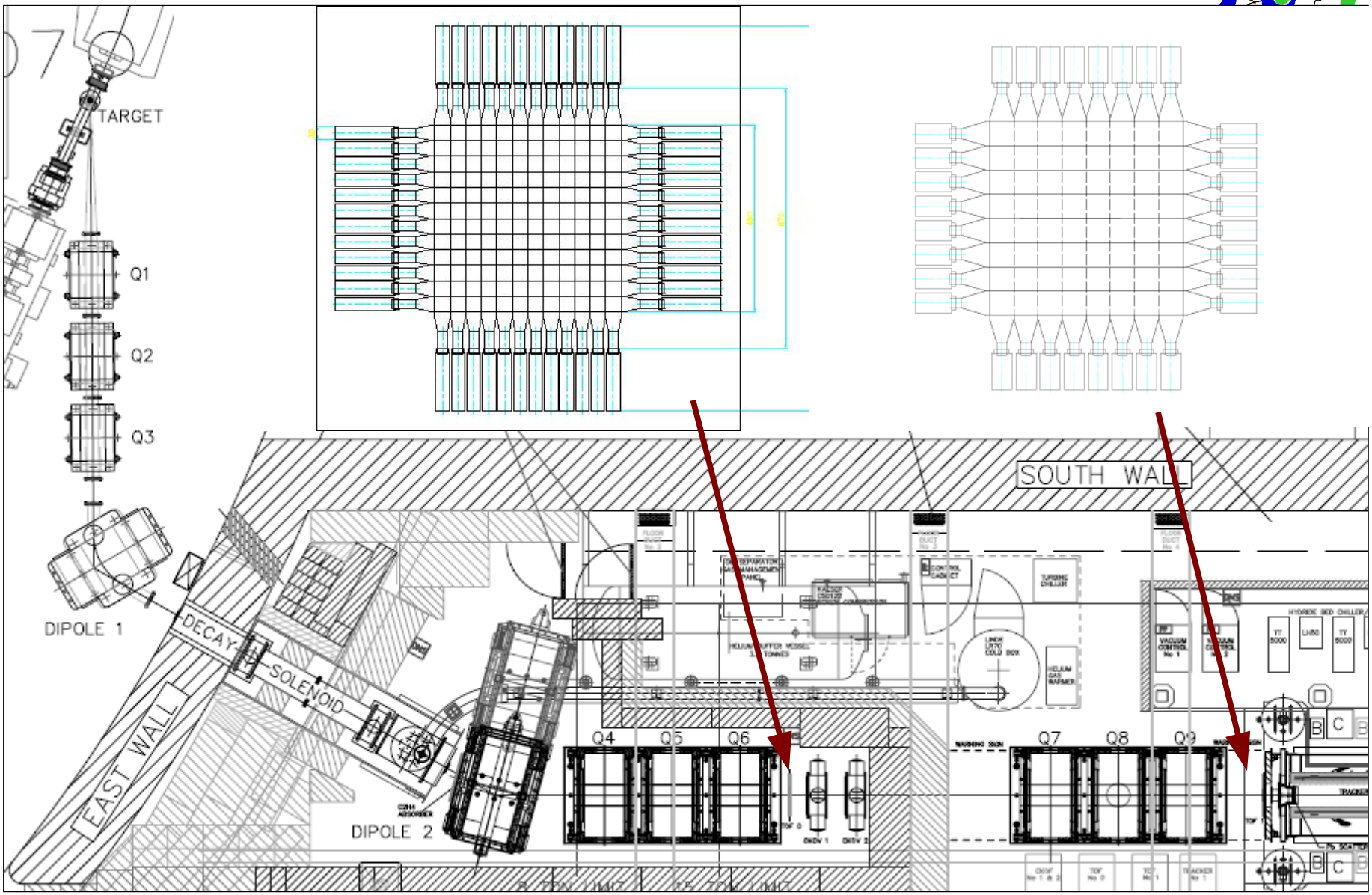
Software Status

- Software is migrating to a more manageable framework
- API (Application Programmer Interface)
  - Distributed processing
- “Continuous Integration” stack
  - Unit Testing
  - Documentation
  - Integration Testing
  - Release management
- Online reconstruction is working well
- Developing offline “turn key” reconstruction
- Ready by Q4 2012?

Data Analysis

- Data taking continues
  - Debugging hardware
  - Monitoring installed equipment
  - Eager to get on with Step IV
- Analysis of data taken in summer 2010 continues
  - Preparation of beamline hardware paper (Bonesini, Soler)
  - **Preparation of beamline physics paper (Rayner)**
    - Using old code (not the new framework)

Beamline Geometry

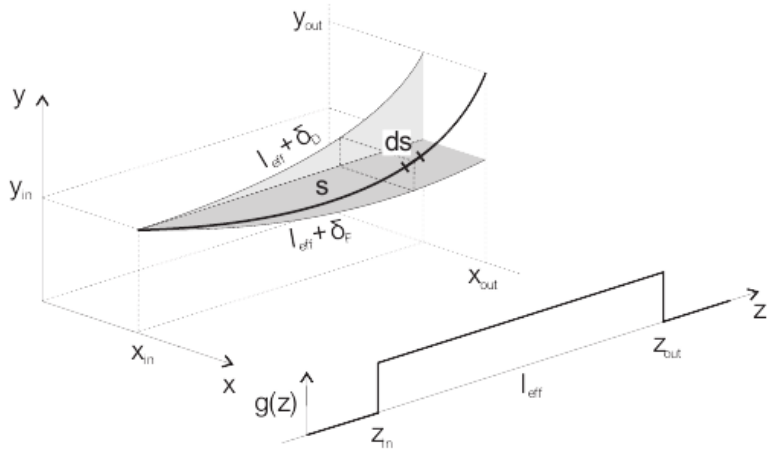


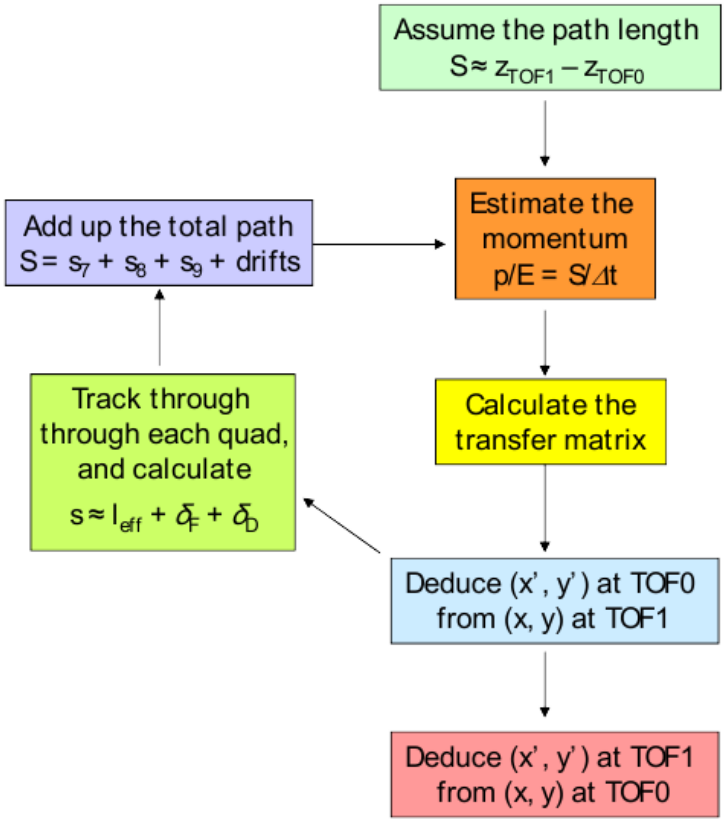
Reconstruction Routine (Rayner)

- Use transfer matrix inversion to unfold x', y' despite presence of quads
- Iteration to deal with big momentum spread (and chromatic aberration)

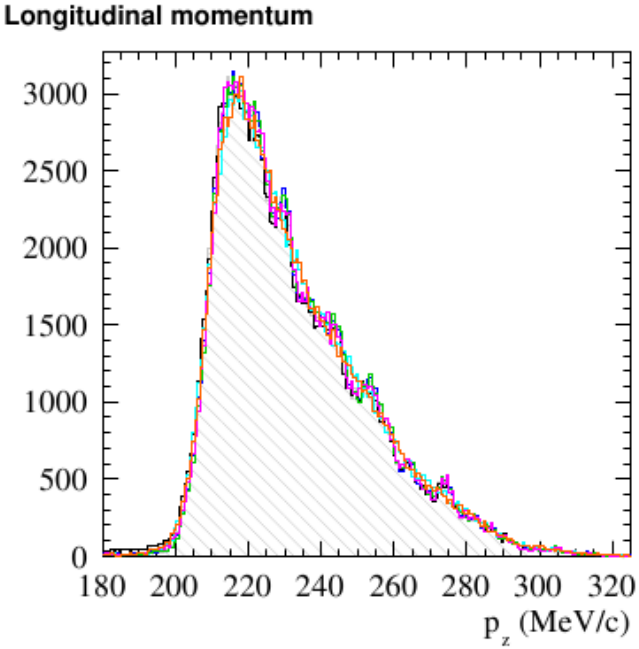
An iterative method is used to remove path length bias

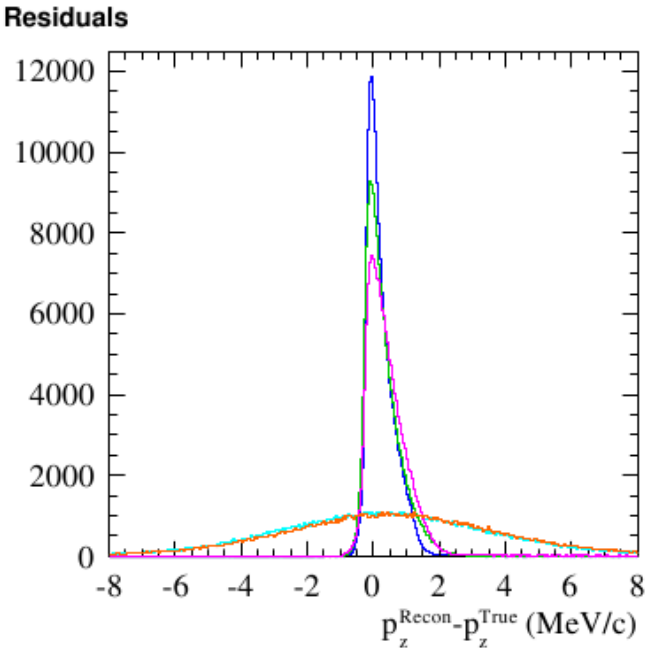
Particles are tracked using a thick edge quadrupole model





Momentum resolution



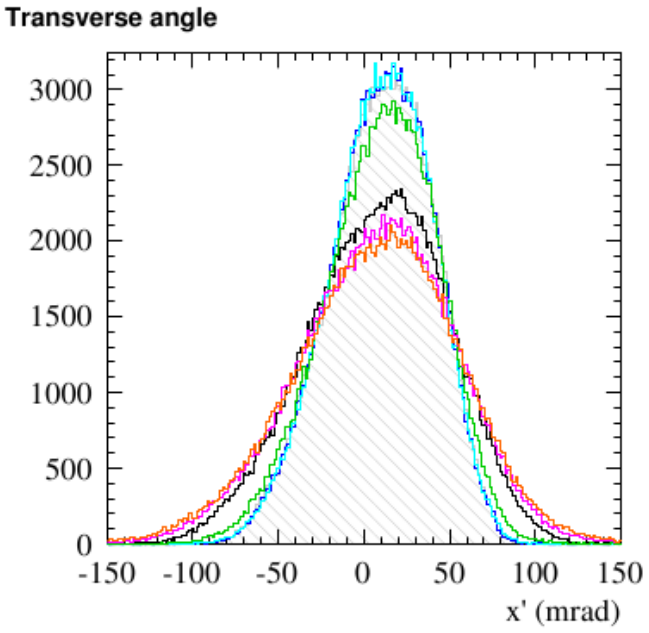


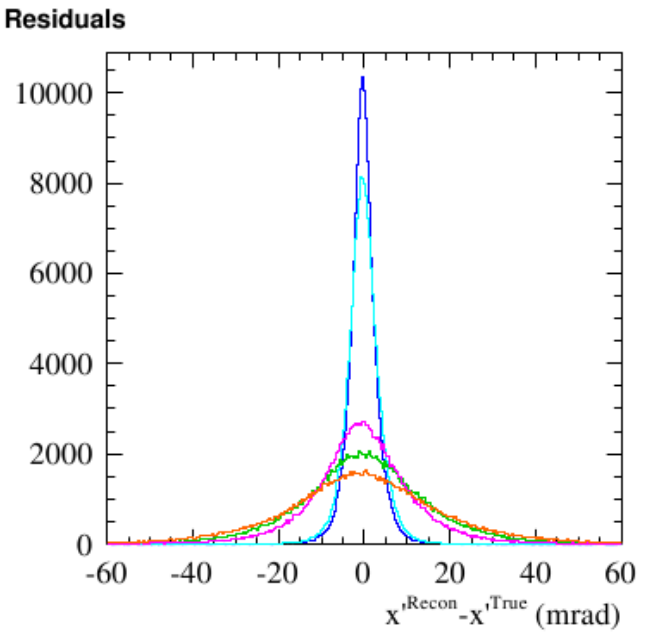
- Truth
- Truth (no scattering)
- No effects
- Time resolution
- Space resolution
- Scattering
- All effects

The final measurement

- Bias = 0.49 MeV/c
- Error = 3.12 MeV/c

x' resolution



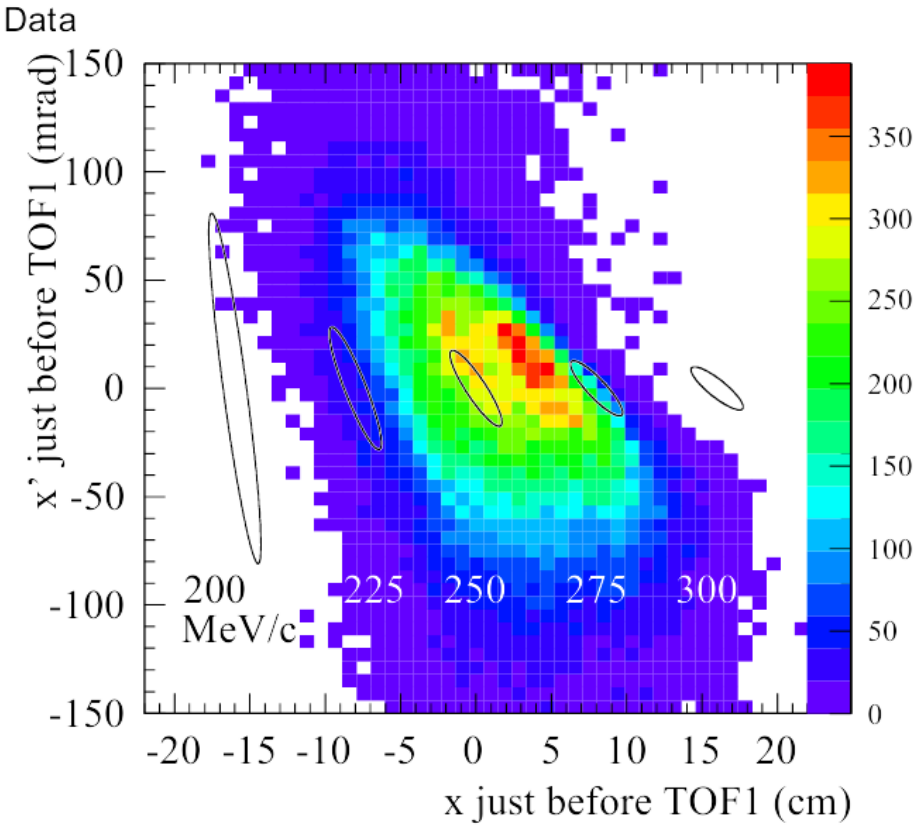


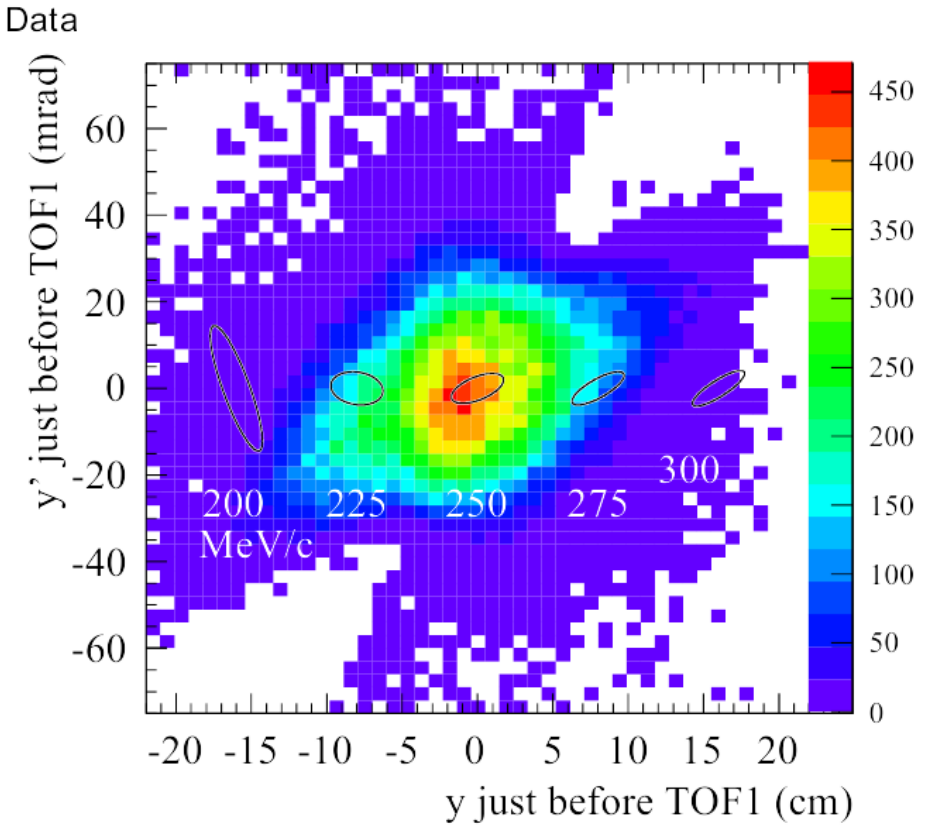
- Truth
- Truth (no scattering)
- No effects
- Time resolution
- Space resolution
- Scattering
- All effects

The final measurement

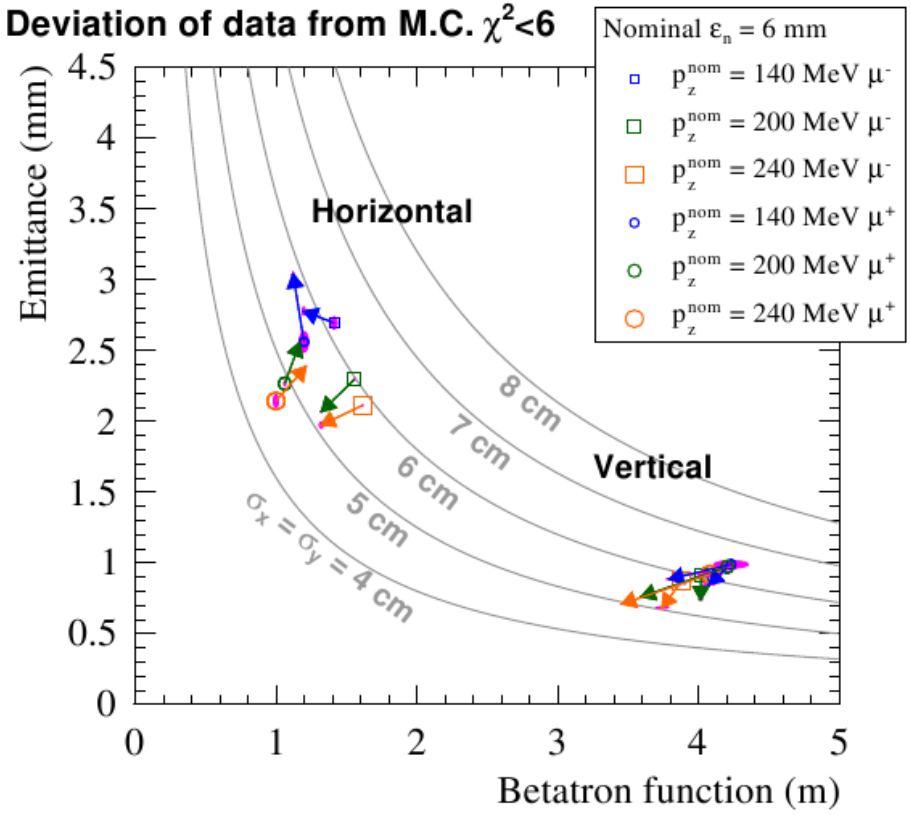
- Bias = −0.40 mrad
- Error = 18.4 mrad

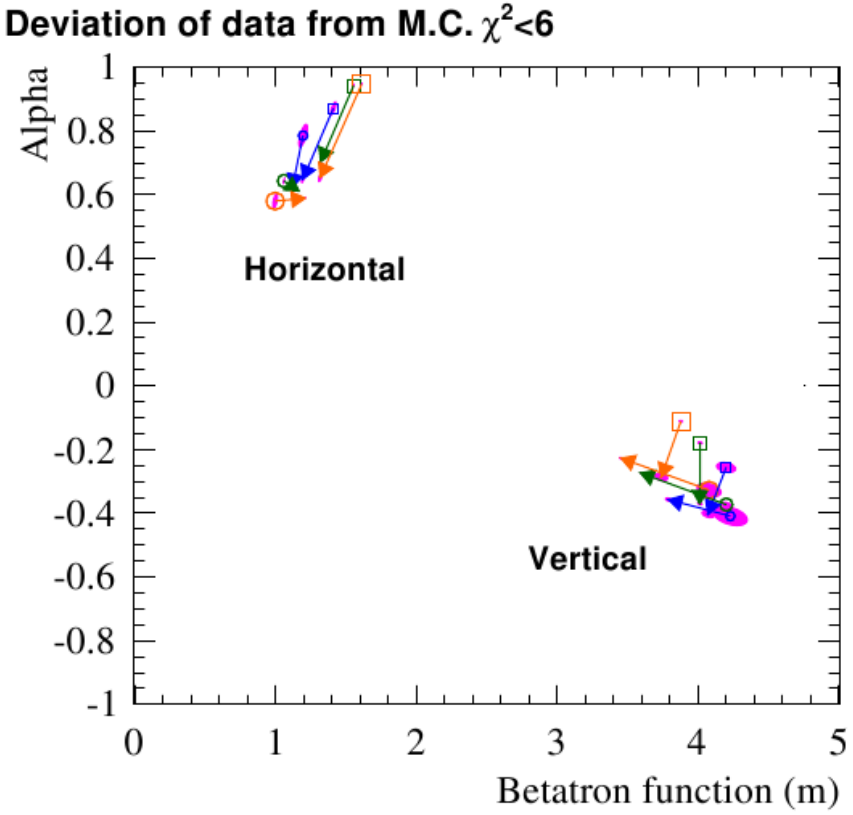
Transverse Phase Space





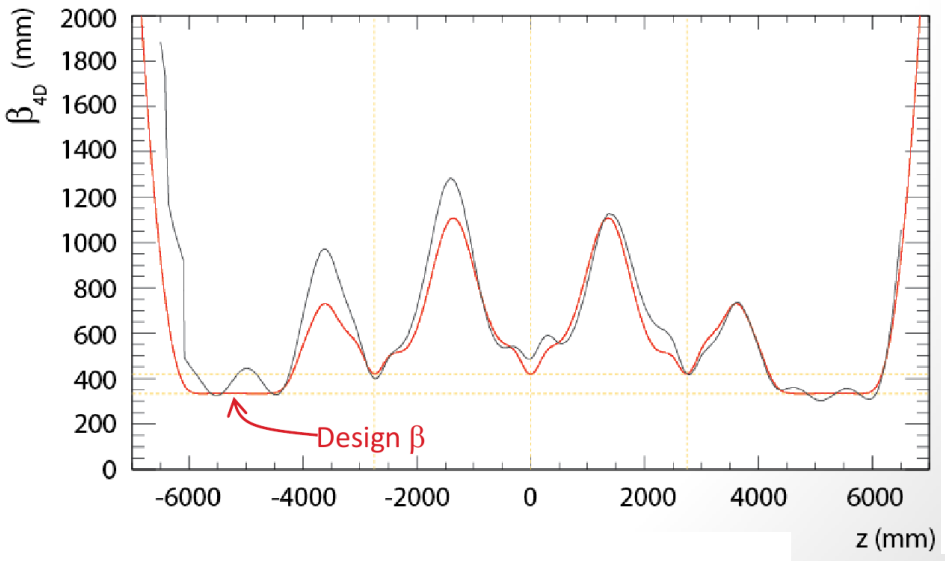
Validation of G4Beamline MC

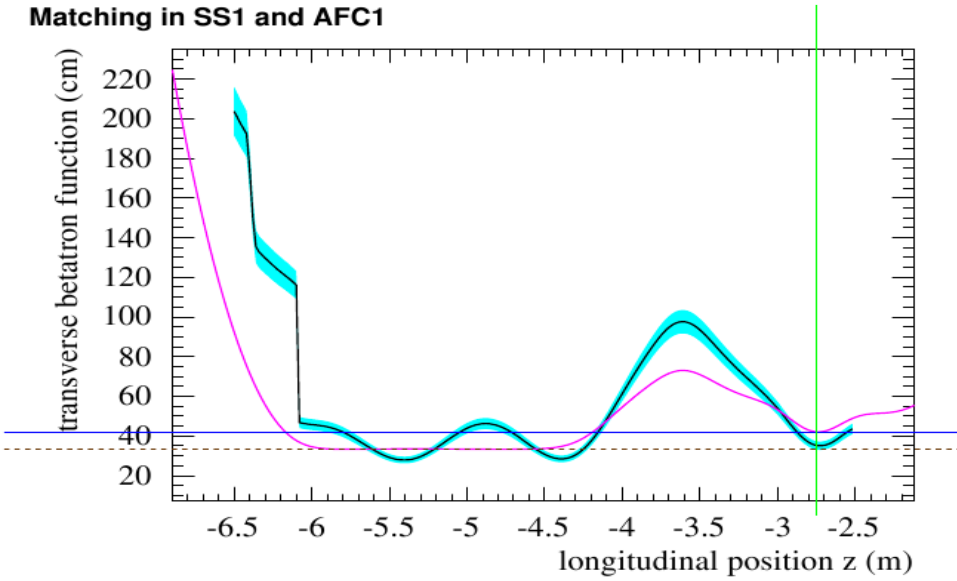




- Arrows point from reconstructed data to monte carlo truth
  - ~ 10 % - 20 % discrepancy in beta and emittance

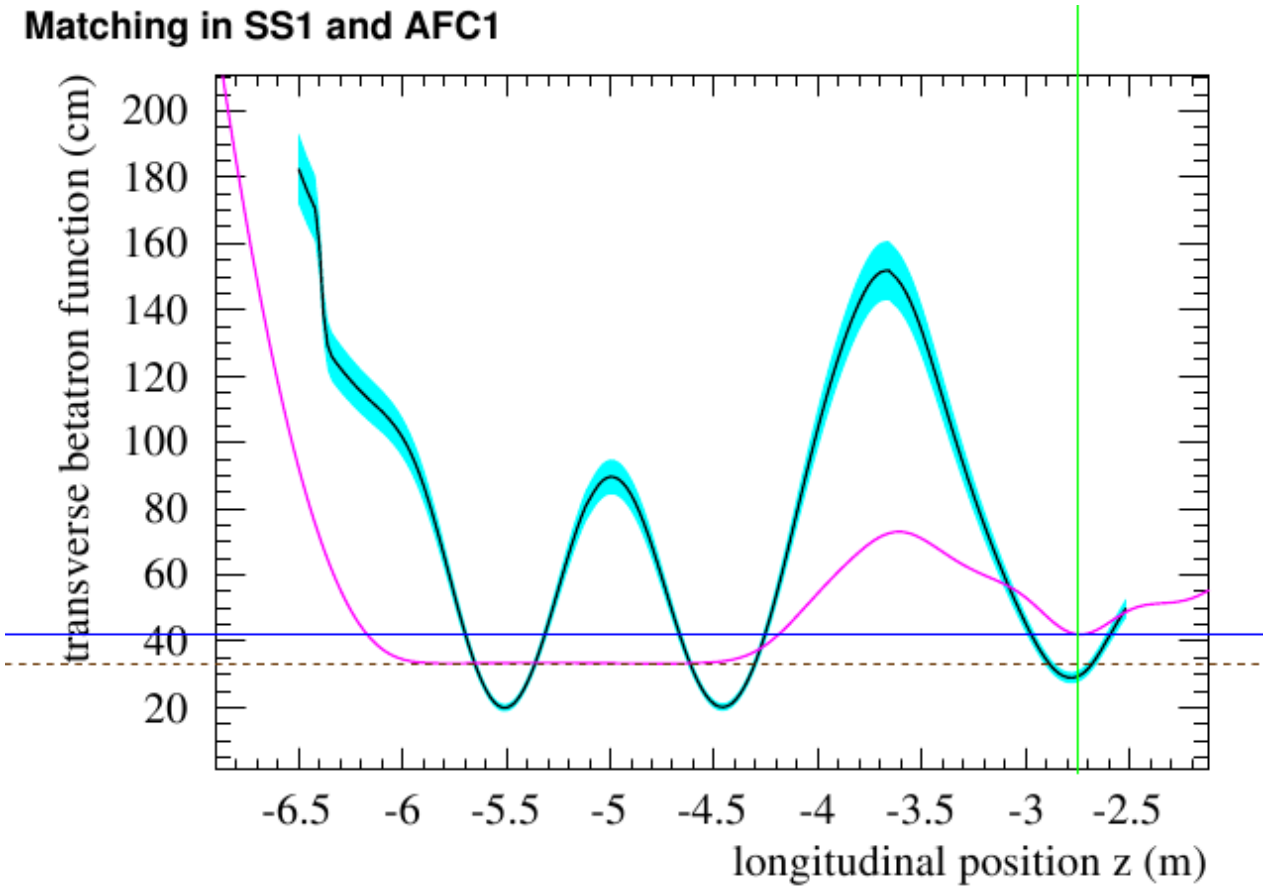
Quality of match - 6 mm





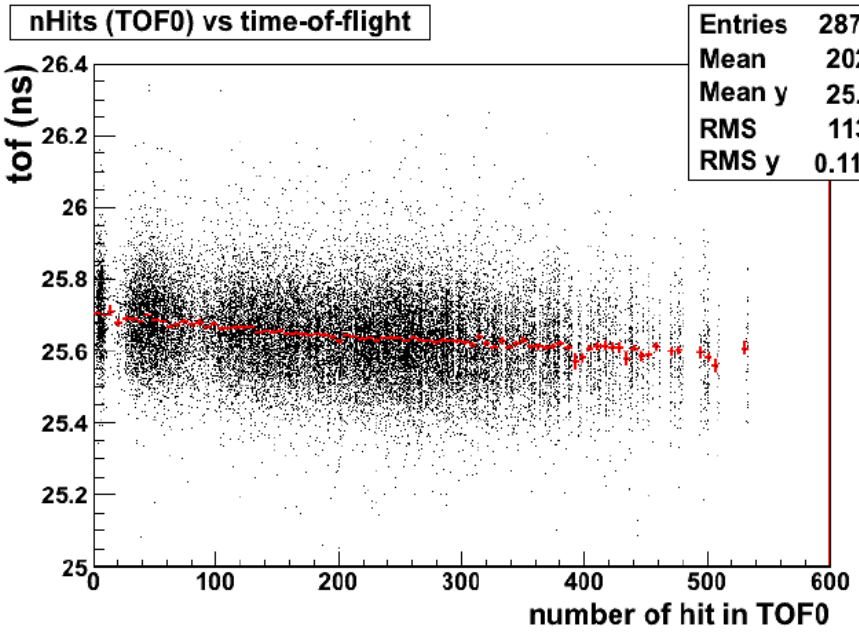
- Take beam from data
- Simulate into the MICE cooling channel
- Match looks pretty good for “standard” 6 mm, 200 MeV/c beam
  - True of all the beams emittance >~ 6 mm

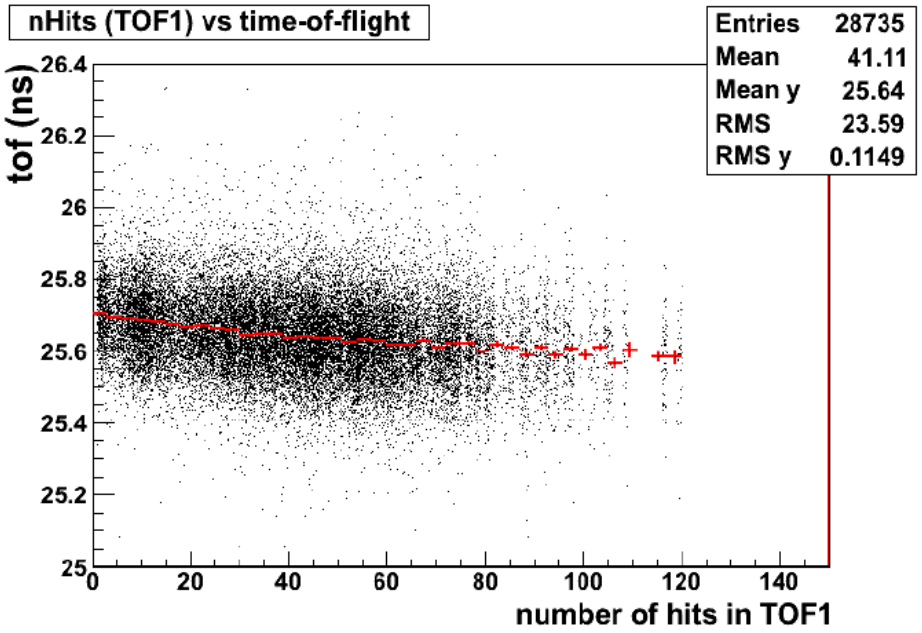
Quality of match - 3mm



- Match looks pretty bad for nominal 3 mm case
- Beamline cannot go below 3 mm
  - Probably need some sampling routines below 3 mm
  - Equilibrium emittance for “standard” configuration is ~ 2.5 mm

Caveat: Rate issue in TOF (Karadzhov)





- Look at electron time of flight between TOF0 and TOF1
  - As rate increases, time of flight gets shorter
  - Probably bug in TOF
  - Effect ~ magnitude of TOF resolution
  - Indicates efficiency issue also? (which will be position dependent)
    - So central TOF panels miss some particles

Data analysis summary

- Analysis of data taken in summer 2010 continues
  - Some nice reconstruction routines
    - (But only work for this particular geometry, detector set up)
  - Looks like the beam is matched for the higher emittance cases
  - Looks like we need to do some clean up or sampling for the lower emittance cases
    - Sampling routines are work in progress
    - But low priority