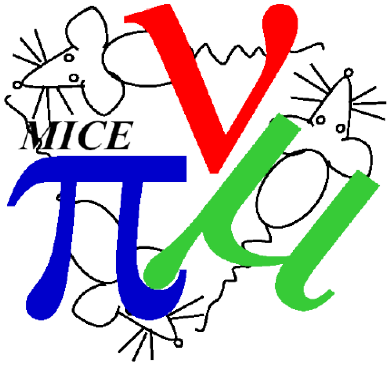
MAUS Overview



Chris Rogers,
ASTeC,
Rutherford Appleton Laboratory

Responsibility/Block Diagram

Project management
Rogers

Build system
~~Tunnell~~ => Rogers? (Urk)

QA
Rogers

Documentation
Rogers

Geometry + fields
Rogers/Littlefield

Geant4 Simulation
Rogers

Data flow/API
Rogers/Richards

TOF
~~Karadzhov~~ Rajaram

Tracker
Dobbs/Santos et al

Ckov
Cremaldi/Kafka

KL
Bogomilov?

Data Unpacking
Karadzhov

EMR
Karadzhov/Ruslan

Detector Integration
Rogers/Lane

Accelerator physics analysis
Rogers/Lane

Aims for CM32

Unpacking tests	Karadzhov	01/10/11	**01/01/12***
Simulation refactor	Rogers	01/10/11	**01/01/12**
TOF tracks	Karadzhov	01/01/12	01/01/12
Prototype Ckov recon	Cremaldi	01/01/12	01/01/12
Online histos finished	Tunnell	01/01/12	01/01/12
Visualisation UI finalised	Robinson	01/01/12	01/01/12

- Simulation refactor
  - Progressing slowly
  - Low priority
- Visualisation UI
  - Now to be an external application
  - Online group responsibility
- TOF tracks
  - Probably this becomes a global reconstruction task
  - Potentially implement after implementing a Kalman fitter
- Others ... we hear about later

Python Test Coverage

Name	Cover	Responsible
ReducePyTOFPlot	4	Rajaram
MapPyTOFPlot	12	Rajaram
ReducePyROOTHistogram	18	Jackson
CouchDBDocumentStore	28	Jackson
mausloader	29	Jackson
MongoDBDocumentStore	42	Jackson
MapPyScalersDump	45	Karadzhov
Go	69	Tunnell -> Rogers
InMemoryDocumentStore	100	
InputPyEmptyDocument	100	
InputPyJSON	100	
InputPySpillGenerator	100	
MapPyDoNothing	100	
MapPyFakeTestSimulation	100	
MapPyGroup	100	
MapPyPrint	100	
MapPyRemoveTracks	100	
OutputPyDoNothing	100	
OutputPyImage	100	
ReducePyDoNothing	100	
ReducePyHistogramTDCAD	100	
ReducePyMatplotlibHistogra	100	

Needs distributed computing libraries on test server

Cpp Test Coverage

Directory	Line Coverage		
src/common_cpp/DetModel		91.7 %	11 / 12
src/common_cpp/DetModel/SciFi		34.4 %	45 / 131
src/common_cpp/Simulation		90.8 %	824 / 907
src/common_cpp/Utils		65.7 %	463 / 705
src/input/InputCppDAQData		60.9 %	337 / 553
src/map/MapCppPrint		90.5 %	19 / 21
src/map/MapCppSimulation		96.1 %	49 / 51
src/map/MapCppTOFDigits		79.7 %	145 / 182
src/map/MapCppTOFSlabHits		93.3 %	111 / 119
src/map/MapCppTOFSpacePoints		90.1 %	146 / 162
src/map/MapCppTrackerDigitization		88.0 %	169 / 192

Dobbs

See below

Karadzhov

Filename	Line Coverage		
CppErrorHandler.cc		78.6 %	33 / 42
CppErrorHandler.hh		100.0 %	8 / 8
DAQChannelMap.cc		50.5 %	46 / 91
DAQChannelMap.hh		83.3 %	10 / 12
JsonWrapper.cc		89.6 %	60 / 67
MAUSEvaluator.cc		77.1 %	64 / 83
PyMausCpp.cc		93.8 %	15 / 16
TOFCalibrationMap.cc		59.4 %	129 / 217
TOFCalibrationMap.hh		100.0 %	8 / 8
TOFChannelMap.cc		50.7 %	73 / 144
TOFChannelMap.hh		100.0 %	17 / 17

Karadzhov

Rajaram

Rajaram

Documentation

- Tracker documentation in progress
- Some prototype simulation documentation
  - Aim to excercise the workflow
  - Understand how we layout the documentation

Build System

- Scons files clean up
  - Was ~ 300 line main function + class + few random functions
  - Now 4 modules
    - environment_tools handles environment set up
      - Check/add 3rd party packages
      - Any other environment variables etc
    - core_builder builds libMausCpp
    - module_builder builds library for each of Input/Map/Reduce/Output
    - SConstruct file drives the whole thing
  - All commented up properly
  - Added dependencies between modules
  - No tests yet
  - Future functionality (low priority)
    - Specify external (3rd party) dependencies of modules
    - C++ tests for modules (rather than SWIG/python)?
- Third party libraries
  - Support for shared third_party library (Dobbs)
  - Explicitly we only support SL/CENTOS
  - Other distros are supported on a “best effort” basis

Geometry + Fields

- Report from Littlefield on CAD import
- Discussion with Jason Tarrant on geometry implementation
  - South Mezzanine comes first
  - Get a “best effort” by March running
  - Rolling issue – it will be a lot of effort to get a decent geometry model for every iteration/survey of MICE
    - An issue I have been struggling with for 2 years!
- Verification of geometry implementation
  - (see Analysis talk)
  - Easy in CAD to accidentally get material wrong/whatever
  - “Survey is accurate to microns, errors O(cm)”
  - Detector experts are responsible for verifying their detector
  - Beamline expert is responsible for verifying beamline geometry
    - Supported by MAUS group
    - Interface with beamline is upstream of D2
  - Analysis group is responsible for verifying cooling channel geometry
- First attempt at a full iteration in March...

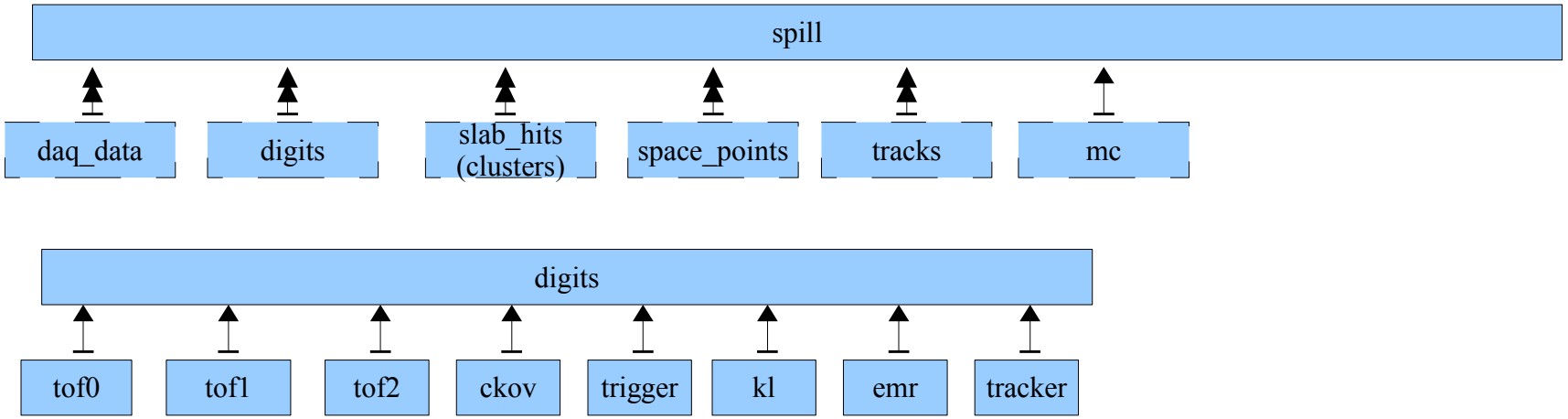
GEANT4 Simulation

- Added ability to read file as beam input
  - Takes G4Beamline, ICOOL, MAUS/json as input
- PhysicsList controls available in py datacards
- Prototype documentation added
- Plans:
  - Clean up of MICEDetectorConstruction/FillMaterials
  - Additional integration tests (e.g. G4MICE physics list tests)
  - Low priority

Data flow / API

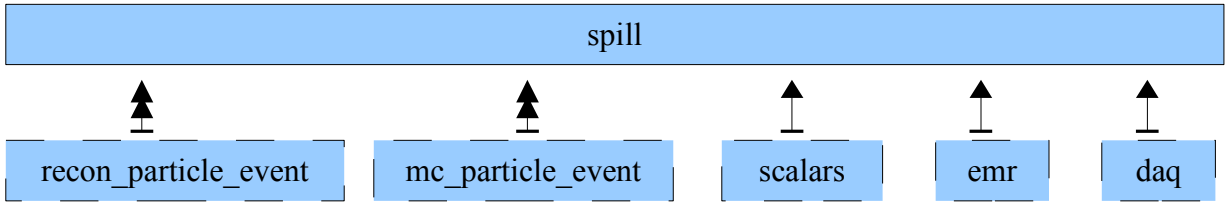
- Implementation of ROOT IO + cpp class structure
  - Framework is in place
  - In testing phase
  - Need to implement the actual data structure
  - Blocked – struggling to reach agreement on data structure
  - Analysis group request explicitly documentation on datastructure

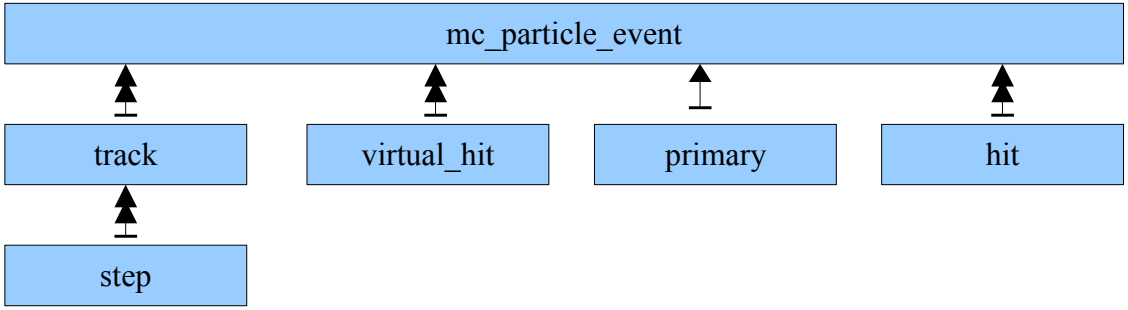
Current data structure

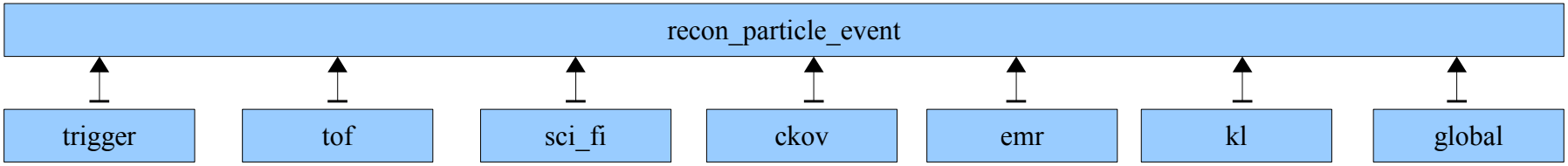


Etc...

Potential data structure







Data flow / API Plans

- Future plans
  - Abstraction layer for the module structure
  - Generic converter between data representations
  - Python logging + interface to Cpp logging (Squeak)
  - Module control at runtime
    - We include this recon or that recon at runtime

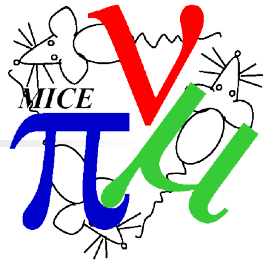
Detector Integration

- Plans
  - Constants and configuration handling
    - geometry, fields, calibrations, cabling, ...
    - Replaces legacy “MICERun” object
    - Full spec in issue 839
  - Trigger model
  - Global reconstruction (see analysis talk)

Accelerator Physics analysis

- Optics tool
  - See analysis talk
- Need to understand what else Analysis group wants online – and who will do it
  - Possibly work as a postprocessor (i.e. not MAUS responsibility)
  - Possibly work as a hybrid
    - MAUS puts e.g. beta functions etc in output data stream and some other tool plots them as a function of e.g. magnet current
  - But maybe Rogers does it...

Project Management



- For the items I have responsibility
  - I am starting to work up task specification for major development
  - Specifying task before implementation... mostly...
  - Would like to have decent specification for the whole project before going any further
  - Would like to have proper detector integration in place before detectors reconstruct any more
    - Project momentum means I struggle to keep up
    - Not enough manpower to turn it around
    - Now hard to bring in more manpower
- For other items would like specification and design before implementation
  - You need to make sure that your proposed implementation fits with the rest of the project
  - This is not happening
- Results in **RISK**
- Results in **REFACTORING**