XBOA – offline analysis tool

Chris Rogers,
ASTeC,
Rutherford Appleton Laboratory

Overview

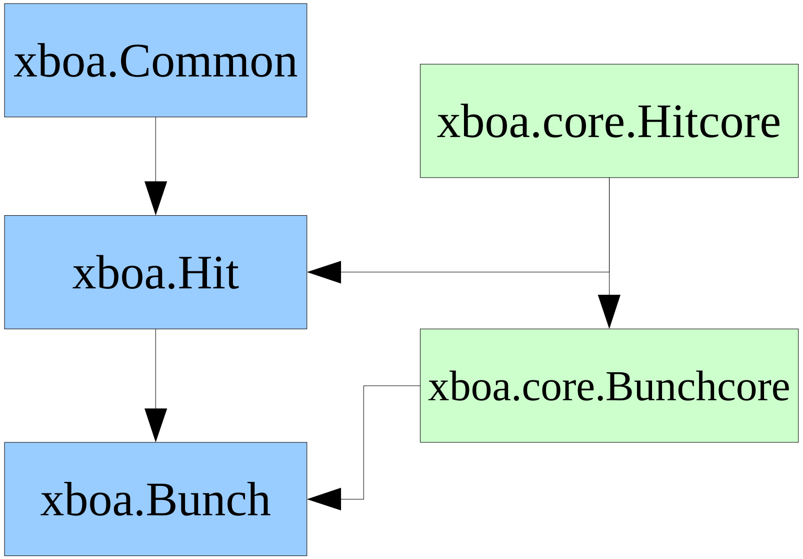
- Analysis computational needs
  - Phase space analyses
    - Emittance, beta functions, etc
  - Beam selection
    - Statistical reweighting, etc
  - General analysis stuff
    - Wrappers and add-ons to ROOT
  - Track matching handled by “Global reconstruction” MAUS package
  - Reconstruction of detector handled by MAUS
- Plan is to use tool xboa driven by Rogers
  - Targeted at python
  - With C++ backend for speed
  - Not as fast as ecalc9f
  - But more versatile

XBOA

- G4MICE Analysis package
  - Developed by Rogers as part of graduate studies
  - Needed a refactor
- XBOA
  - Developed to support Neutrino Factory design study
  - Developed outside of G4MICE framework
    - G4MICE was dying
  - Aim to make “plot emittance vs z” type needs easy:
  - Three lines of code:
    - Import library
    - Load file
    - Make the plot
  - Aim to make more complicated things easier
    - Cuts/statistical weighting
    - Amplitude calculations and plots
    - 2D/4D/6D
- Available for people to use now
  - Comes packaged with MAUS

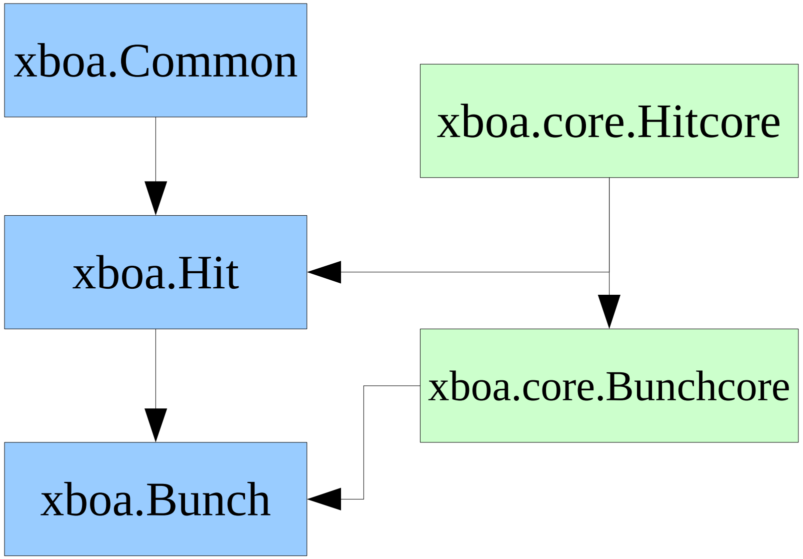
XBOA - Common

- Common
  - ROOT wrappers
    - Automatic histogram ranges
    - Convenience interface to graphs
    - ...
  - “Useful stuff”
    - Particle masses
    - Few maths routines



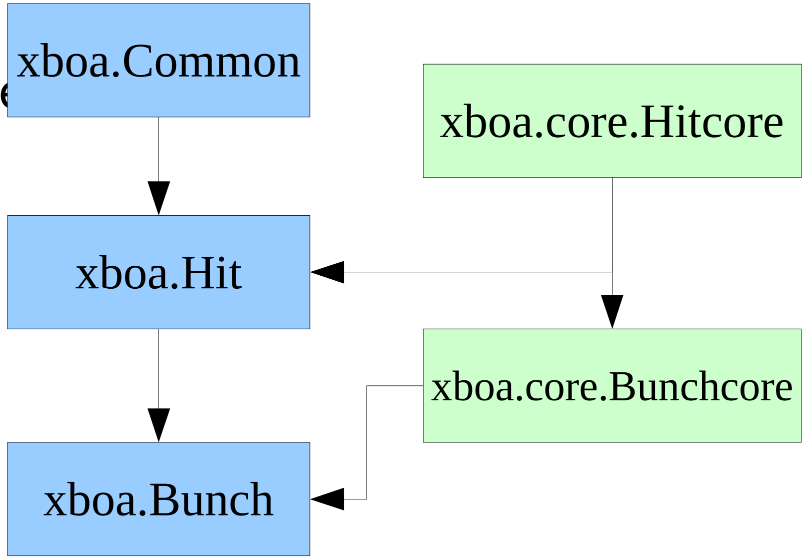
XBOA - Hit

- Hitcore
  - C backend to Hit
  - Represents a detector/MC output plane “hit”
  - Container for data
  - position, momentum, etc
  - “Global (statistical) weight”
  - Local (statistical) weight
- Hit
  - I/O routines per output “hit”
  - Utilities
    - Transformations
    - Mass shell condition
    - Equality operator

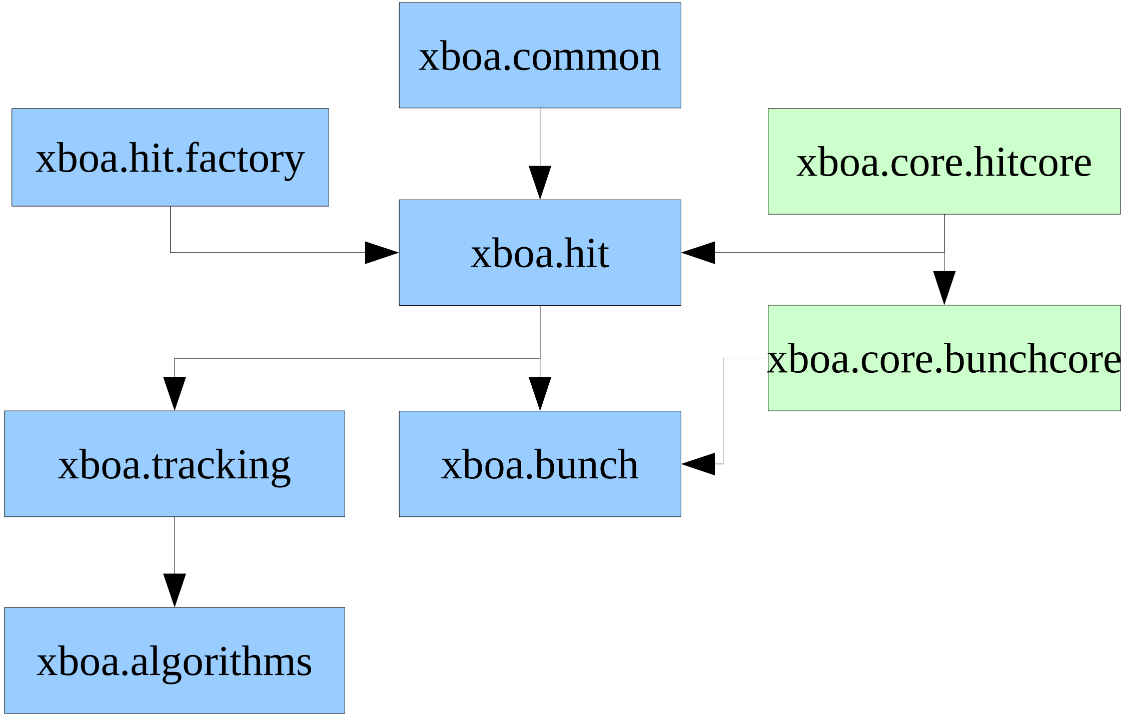


XBOA - Bunch

- Bunchcore
  - C backend to Bunch
  - Collection of Hitcore objects to make up a “Bunch”
  - Container for data
  - Calculation of moments
- Bunch
  - I/O routines at “bunch” level
  - Transformations
  - Moment calculations
  - Optical function calculations
  - Amplitude calculations
  - Plotting
    - 1D histograms, scatter plots, graphs
  - Cuts



Recent additions



- Core libraries migrated from C to C++
  - C++ code separated from Python interface code
- Read routines moved to “factory” classes
  - Easier to extend/add new read routines
- Added “algorithms” library
  - Closed orbit finder, tune finder, general data analysis tools
- Added “tracking” interface library
  - Wraps external tracking libraries e.g. MAUS G4 routines

Problems/Feature Requests

- Targeted at python; this doesn't suit everyone
  - Probably too big a job to refactor into C++
- Model is “load data into memory; then do stuff with it”
  - Memory usage is too high for larger data sets
  - Rogers job to design a more MAUS-like model
    - Build up e.g. covariance matrix on a spill-by-spill basis
    - Cuts on e.g. chi2 or amplitude require multiple read operations
- Only has I/O routines for MC data at the moment
  - Easy to add interface to recon data
- Code is available to use but not mandatory
  - **Folks should publish analysis scripts with their papers for Step IV papers**
    - Step I papers?