EventLoop Introduction

Nils Krumnack (Iowa State University)

EventLoop Origins

- first version of EventLoop was developed to create groups of stack plots quickly/easily
  - together with SampleHandler and MultiDraw packages

- should be efficient for very light weight jobs:
  - fast as TTree::Draw() to read a single variable, fill one histogram
  - can have a separate algorithm for each histogram filled
- integrate well with plotting macros:
  - can run on many datasets in one job
  - easy to access histograms when job ends

- later merged with project for better support for grid running
  - added support for batch systems as well
- usually focus more on running "heavy" jobs these days
- not doing any plot-making exercises in the tutorial anymore
  - functionality is still there though…

EventLoop Driver

- a typical analysis often uses a large quantity of data
  - looping over events often slow → parallel processing advised
- EventLoop implements an event loop for you
  - works for local running, batch systems and the grid
  - handles a lot of submission and merging details for you

- have so called Driver object to describe where you are running
  - separate class for each available backend
  - can switch target location by replacing Driver object
- fairly straightforward to add support for new batch systems
  - typically a couple of hours to implement a new Driver

- entire job gets described by Job object:
  - contains datasets, algorithms, options, etc.

- https://twiki.cern.ch/twiki/bin/viewauth/AtlasProtected/EventLoop

SampleHandler

- it is not uncommon to need ~100 datasets for a single analysis
  - can be difficult to track them all
  - also need to track associated metadata, e.g. luminosity, k-factor…
- SampleHandler can do all that bookkeeping for you
  - can store the location and metadata for each dataset
  - can discover available datasets for you
  - can handle different storage systems (including the grid)
  - can group and select data samples in various ways
- some notes:
  - if SH doesn't match your workflow or setup, you can/should ask for an adapter
  - SH doesn't do magic, you still need to define your own metadata
  - most people only use SH to pass file lists to EventLoop

- https://twiki.cern.ch/twiki/bin/view/AtlasProtected/SampleHandler

Tool-Kit Philosophy

- aim for set of independent tools/packages instead of framework
  - i.e. fairly easy to use one software package without the others
  - lesson from history: using Athena was/is mostly all-or-nothing

- however: please try to use the common tools where possible
  - a lot of experience went into them
  - most are well-tested with large user base
    - fewer bugs, easier to find help
  - often have features that are useful later on
  - easier to exchange code with others

- still, feel free to replace common tools with your own, but:
  - please try the common tools first
  - ask an expert whether the feature you want is already there
    - or whether we can implement it as a new feature
  - consider contributing to the common tools

Dual-Use Philosophy

- put strong emphasis on dual-use code:
  - allow running analysis code inside Athena unchanged
  - only a question of recompiling in the new environment

- original motivation: allow running CP tools in Athena
  - many CP groups only released code for AnalysisBase in run 1
  - some tools are also used/needed for offline/online
  - also: some people huge fans of doing analysis in Athena

- side benefit: learn Athena basics/concepts without using Athena
  - main difference: job configuration/management
  - some things not (yet) dual-use: services, algorithm sequences…

- side benefit: can develop code in preferred environment
  - e.g. even people who know Athena often prefer AnalysisBase

Differences to Athena

- for a lot of tasks both environments are about equally well suited
  - not trying to push you one way or another
  - mostly pointing out some special edge cases here

- EventLoop is (somewhat) faster
  - mostly relevant for shorter/lighter jobs
  - for grid jobs you probably won't notice a difference

- have some advanced/niche features in AnalysisBase, e.g.
  - running on MacOS without VM/docker
  - use SampleHandler+PlotMaker for quickly making stack plots
  - dataset discovery with SampleHandler

- harder to get feature/merge requests into athena/gaudi
  - analysis not their main/only use case
  - not a problem if you do things the "Athena way"

Some Caveats

- in AnalysisBase you need to match releases to file versions:
  - e.g. release 21.2.* reads offline release 21.2 files
- AnalysisBase doesn't apply AODFix (i.e. some basic corrections):
  - to get this, you need to run on derivations (or in Athena)
- files written in AnalysisBase can't be read in Athena

- not all EDM objects are accessible in AnalysisBase
- Athena has extra services, e.g. for conditions data access

- athena-only features generally not needed for "normal" analysis