



# Status of code migration

Saba Sehrish

LArSoft Coordination Meeting

10/09/2018

# Overview

- What are we doing?
  - The plan is to identify all the relevant pieces of code that corresponds to algorithms that can be moved to the LArObj suite (LAr\*alg repositories).
- Why are we doing this?
  - To be able to use the algorithms that do not depend upon the core framework features outside of the framework context (e.g. with gallery).
- How will this affect the user code?
  - Breaking change: only need to update the headers corresponding to the moved files, scripts are provided
  - Some new dependencies may be introduced e.g. LAr\*obj

# What has been done so far

- The following repositories were looked at to identify what code needs to be migrated so that it can be used in gallery:
  - `lardata`, `larana`, `larcore`, `larreco`, and `larsim`
- Our criteria is to move the files that
  - are not *art* modules, services, *etc.*
  - do not use any of the framework functionality directly or indirectly
- We currently have no deadline for this migration.
- The next step is to create redmine issues corresponding to the proposed plan.
  - If you do not see a ticket for some algorithm code that you would like to use inside gallery environment, please submit a ticket.

# A few example candidates from each of the repositories

- `lardata`
  - Most of the utility code in `lardata/Utilities` and `ArtDataHelper/Dumpers`
  - #17179: Move `DetectorInfo` to `lardataalg` [completed]
- `larana`
  - `TruncatedMean/Algorithm`, a couple of algorithms from `CosmicRemoval`, `OpticalDetector` and `ParticleIdentification`
- `larcore`
  - nothing needs to be moved
- `larreco`
  - Algorithms in `MCComp`, `Calorimetry`
  - `RecoAlg`, `SpacePointSolver` needs a second pass
- `larsim`
  - Algorithm in `IonizationScintillation`
  - #18281: Move `MCDumpers` from `larsim` to `lardataalg` [completed]