

Gallery Example

Wesley Ketchum

What is gallery?

- It's a hook into *art* (*larsoft*) created data products without needing to use all of *art*
 - *art* output files always have been ROOT files, but with difficult structure
 - *gallery* → more natural way to read the ROOT files on their own
 - Allow writing simple ROOT (or PyROOT) macros, and C++ compiled code to view/analyze data “outside” the *art* framework
- Better to just show via tutorial, so we'll get to that soon
- When should you use it?
 - Simple analysis tasks to take a quick look at output data
 - Early development of reconstruction and analysis tools
 - Easy to copy code into an art module
 - For now, no availability of *services* provided in *larsoft*, so some limited use

Let's dive right in

- Clone repository from github:
 - `git clone`
<https://github.com/wesketchum/gallery> `example.git`
 - `cd gallery_example`
 - `git checkout protodune`
- You need to "setup" two UPS products: larsoftobj and gallery
 - `source setup_gallery`
 - Shows available versions, and gives hints on the setup commands
 - `setup larsoftobj v1_08_01 -qe10:prof`
 - `setup gallery v1_03_08 -qe10:nu:prof`
- Have it?

Two options shown here

- Macros
 - Simple ROOT macros
 - To run a macro, you should open ROOT and do:
 - `root [0] .L demo_ReadEvent.C++`
 - `root [1] demo_ReadEvent()`
 - Notes
 - Generally, you need to exit ROOT in order to recompile and run again without errors
- C++ compiled code
 - Simple C++ main programs
 - Makefile exists (you can use it has an example for any new program)
 - Example to compile and run:
 - `make demo_ReadEvent`
 - `./demo_ReadEvent`

OK, so ... let's take a look!

- Comments in the code should be robust/useful
 - Email wketchum@fnal.gov with any questions
- What they are (some of them):
 - macros/demo_ReadEvent.C
 - Simply open a file and see event/run numbers
 - macros/demo_ReadHits.C
 - Open a file and read a collection of hits
 - macros/demo_ReadClusters.C
 - Open a file and read a collection of clusters, and read associated hits
 - cpp/demo_ReadClusters_MakeTree
 - Open a file, read cluster and hit information, and create a simple output tree