



art news

Kyle J. Knoepfel
art stakeholders meeting
13 July 2017



Today's meeting

- *art* exec linking against ROOT libraries
- Multi-threading diagnostic information
- Proposed interface changes for *art* 2.08
- GDB 8

art executable linking against ROOT libraries

- All *art* execs (**art**, **gm2**, **lar**, **mu2e**, **nova**) link against 13 ROOT libraries for convenience to users.
 - For experiments that do not use closed linking of their libraries, the unconditional loading of these ROOT libraries can provide the necessary symbols for users if the appropriate ROOT library dependencies have not been specified.
 - For experiments that rely on closed linking, providing these extra libraries is unnecessary.
- Are there any experiments that do not use closed linking that wish to retain unconditional loading of these libraries at runtime?

Multi-threading diagnostic information

- To alert users of changes to *art* required by multi-threading, we would intend to provide diagnostic messages (e.g.):
 - At some point, it will be mandatory to specify the products that are consumed by a module. We will be providing a utility that tells the user (to a reasonable approximation) which statements are necessary to include in the source code:

```
=====  
The following consumes statements are missing from  
module label: 'a1' of class type 'CheckProducts'  
-----  
consumes<arttest::CalibConstants, art::InSubRun>("reco:CalibConstants:reco");  
consumes<arttest::TrackEfficiency, art::InSubRun>("reco:TrackEfficiency:reco");  
consumes<unsigned int, art::InSubRun>("prod:nParticles:eventGen");  
consumes<arttest::Geometry, art::InRun>("sim:Geometry:particleSim");  
consumes<unsigned int, art::InRun>("prod:nPOTs:eventGen");  
=====
```

- Our intention is to provide a messagefacility category (e.g. `MTinfo`) so that all `MTinfo` messages (like the one above) are logged to a given file.

Multi-threading diagnostic information

- We can:
 - Provide a `--mt-diagnostics` program option that enables the system to log such messages
 - Unconditionally make an `mt_info.log` file for every *art* process.
 - Any other options you have....

Proposed interface changes for *art* 2.08

- **fhicl::parse_value** will be removed.
 - It is deprecated and not used in source code; not used in experiment code.
- **art::ProcessConfiguration** schema has changed:
 - **art::PassID** data member (and class!) will be removed.
- **canvas/Utilities/TestHelper** facility will be removed.
 - It does not appear to be used anywhere.
- **RootInput fileMatchMode** configuration parameter will likely be removed.
 - It is not conducive to allowing processing of inconsistently-produced *art*/ROOT files.
- **art::JobMode** enumeration will be removed.

GDB 8.0

- We currently distribute GDB 7.12
- GDB 8.0 was released last month.
 - The GDB versions we distribute are built with the system compiler
 - *This is the first GDB release to require a C++11-compliant compiler* (GCC 4.8 or newer)
 - This is not possible for SLF6 (sys. compiler GCC 4.4)
- Our current approach is to *not* distribution GDB 8.0.
 - This cannot be the long-term solution, however, as the compiler constraint will not go away.
- We could build GDB 8.0 with the GCC we distribute, but it introduces extra complications for users (e.g GCC would need to be e-qualified).