

art - Feature #10969

Provide VERSION macro exports

11/20/2015 09:50 AM - Eric Flumerfelt

Status:	Closed	Start date:	11/20/2015
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Scope:	Internal	SSI Package:	
Experiment:	-		

Description

Some *artdaq* code is dependent on specific interfaces in *art*. When these interfaces change (sometimes without notice), we need to be able to conditionally compile code in a way so as to be able to support older versions of *art* as well as the current version. The traditional way of dealing with this problem is to define version variables that can be used in the preprocessor to change our interface usage.

History

#1 - 11/20/2015 10:54 AM - Lynn Garren

This needs discussion. The general view is that we do not support compile time `#if` statements based on versions. Perhaps we can arrange a meeting sometime next week.

#2 - 11/23/2015 11:50 AM - Marc Paterno

- Status changed from *New* to *Feedback*

The UPS package for art already provides an environment variable `ART_VERSION`.

In *artdaq*'s top-level `CMakeLists.txt` file is an example of how to use it to define a feature macro for the C preprocessor:

```
check_ups_version(art "$ENV{ART_VERSION}" v1_14_00 PRODUCT_OLDER_VAR EVENTID_HAS_EXPLICIT_RUNID)
if (EVENTID_HAS_EXPLICIT_RUNID)
  add_definitions(-DARTDAQ_ART_EVENTID_HAS_EXPLICIT_RUNID)
endif()
```

There is a use of this macro in [artdaq:source:artdaq/ArtModules/detail/RawEventQueueReader.cc#L134](https://artdaq.org/source/artdaq/ArtModules/detail/RawEventQueueReader.cc#L134)

Does this facility meet your need?

#3 - 08/29/2016 12:07 PM - Kyle Knoepfel

- Status changed from *Feedback* to *Closed*

Closed due to lack of feedback.