LArSoft - Necessary Maintenance #18136

Fix design flaw in geo::ChannelMapAlg interface

11/07/2017 12:32 PM - Gianluca Petrillo

Status:	Closed		**Start date:**	11/07/2017
Priority:	Low		**Due date:**	
Assignee:	Paul Russo		**% Done:**	0%
Category:	Geometry		**Estimated time:**	4.00 hours
Target version:			**Spent time:**	0.00 hour
Experiment:	-			

Description

geo::ChannelMapAlg::SignalType has two signatures, both virtual and subject to redefinition from derived classes.
When only one of them is overridden in a derived class, the other one is hidden.
This is in general not a concern since the derived classes are expected to be accessed only via the abstract interface, but it still is a design flaw.
A standard solution is to have the methods exposed as interface in geo::ChannelMapAlg non-virtual, and have them call protected virtual implementation methods. These methods might have also different names to avoid the hiding problem. This solution is a breaking change to the implementation of the interface: all implementations which currently override any of the two methods will need to be fixed (the fix is trivial: move to protected and rename to the new name the overriding methods).

History

#1 - 11/07/2017 01:03 PM - Gianluca Petrillo

- Subject changed from Fix design flaw in @geo::ChannelMapAlg@ interface to Fix design flaw in geo::ChannelMapAlg interface

#2 - 11/13/2017 10:38 AM - Lynn Garren

- Status changed from New to Assigned

- Assignee set to Gianluca Petrillo

#3 - 04/13/2018 03:14 PM - Katherine Lato

- Assignee changed from Gianluca Petrillo to Paul Russo

#4 - 05/30/2018 04:28 PM - Paul Russo

The SignalType() virtual functions have been replaced with two non-virtual member functions:

```
geo::SigType_t SignalTypeForChannel(raw::ChannelID_t const channel) const;
geo::SigType_t SignalTypeForROPID(readout::ROPID const& ropid) const;
```

one for each of the original two overloads. Use these two functions in place of calling SignalType(). Which one to use depends on the argument passed (in the obvious way).

A derived class implementer may customize the behavior of these functions by providing overrides for:

```
virtual geo::SigType_t SignalTypeForChannelImpl(raw::ChannelID_t const channel) const = 0;
virtual geo::SigType_t SignalTypeForROPIDImpl(readout::ROPID const& ropid) const;
```

which are called by the base class implementations of:

```
geo::SigType_t SignalTypeForChannel(raw::ChannelID_t const channel) const;
geo::SigType_t SignalTypeForROPID(readout::ROPID const& ropid) const;
```

Note that the implementer is **required** to provide an override for:

```
virtual geo::SigType_t SignalTypeForChannelImpl(raw::ChannelID_t const channel) const = 0;
```

as it is pure virtual.

#5 - 05/30/2018 04:35 PM - Paul Russo

dunetpc, icaruscode, and lariatsoft were updated to adapt to this breaking change.

Released in v06_78_00.

#6 - 05/30/2018 04:35 PM - Paul Russo

- Status changed from Assigned to Resolved

#7 - 09/20/2018 03:11 PM - Katherine Lato

- Status changed from Resolved to Closed