

fhicl-cpp - Feature #10828

Python fhicl interface

11/05/2015 05:07 PM - Herbert Greenlee

Status:	Closed	Start date:	11/05/2015
Priority:	Normal	Due date:	
Assignee:	Kyle Knoepfel	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	2.07.01	Spent time:	5.00 hours
Description			
Python interface should be able to:			
<ol style="list-style-type: none">1. Parse a fcl file or text string representing a fhicl parameter set into something like a python dictionary.2. Read fhicl parameter sets from an artroot file sqlite database and produce a python dictionary.3. Encode a python dictionary representing a fhicl parameter set into parseable fcl text.			
Related issues:			
Related to art - Feature #16342: Facility to convert fhicl::ParameterSets fro...		Accepted	04/26/2017

History

#1 - 11/23/2015 11:31 AM - Marc Paterno

- Status changed from New to Feedback

We would like to meet to discuss more details about the requirements, such as what dependencies are acceptable for this new facility. We will follow up through email.

#2 - 04/17/2017 09:08 AM - Kyle Knoepfel

- File *fclmodule.cxx* added

- File *CMakeLists.txt* added

The attached files have been provided by Herb for creating a Python dictionary from a C++ fhicl::ParameterSet object.

Herb, the code contains what we would hope to see--a final dictionary that is independent of the fhiclcpp details. We would be in a better position to take on this facility if you could convert the code to using the fhicl::detail::ParameterSetWalker interface, which admittedly is not well-known. You would write a derived class (e.g. PSetToPythonDict), which overrides functions for processing FHiCL atoms, sequences, and tables. After the derived class is defined, you could do (e.g.):

```
PSetToPythonDict converter;
pset.walk(converter);
PyObject* pset_dict = converter.result();
```

without having to do any explicit tree-walking yourself.

Unfortunately, we do not have the effort available to make this conversion ourselves. However, by making this conversion, we can then adopt the code and guarantee that any future changes made in the fhicl::ParameterSet implementation will be appropriately reflected in the Python dictionary.

Please look at the documentation in fhiclcpp/detail/ParameterSetWalker.h and/or contact us with questions. Note that the fhicl::ParameterSet::walk method is currently private (and spelled with a trailing underscore). However, there is no reason it cannot be made public.

Your code is a big step forward.

#3 - 04/19/2017 09:23 PM - Herbert Greenlee

As requested, I made another version of fclmodule.cxx which is implemented using ParameterSetWalker. This implementation has the problem that ParameterSet method "walk_" is private (had to use "#define private public").

The updated version is saved in redmine git repository ubutil on branch feature/greenlee_pset_walk.

https://cdcvns.fnal.gov/redmine/projects/ubutil/repository/raw/fclmodule/fclmodule.cxx?rev=feature%2Fgreenlee_pset_walk

#4 - 04/20/2017 08:09 AM - Kyle Knoepfel

Excellent, Herb. Thank you. Yes, we will make the ParameterSet::walk method public.

#5 - 04/25/2017 02:38 PM - Kyle Knoepfel

- Status changed from Feedback to Accepted

We intend to move item 2 to its own issue since it involves moving some art infrastructure to canvas, as well as new development. Items 1 and 3 are provided by the code that Herb provided. We will integrate Herb's Python extension module to fhicl-cpp, which also involves moving the Python dependency from canvas down to fhicl-cpp.

#6 - 04/26/2017 11:32 AM - Kyle Knoepfel

- Related to Feature #16342: Facility to convert fhicl::ParameterSets from art/ROOT file to Python dictionary added

#7 - 04/26/2017 11:34 AM - Kyle Knoepfel

We have added an additional issue that refers to part 2 of this request. After we have finished migrating Herb's module to fhicl-cpp, which addresses parts 1 and 3, we will mark this issue as resolved.

#8 - 04/27/2017 01:39 PM - Kyle Knoepfel

- Target version set to 1209

#9 - 05/03/2017 11:04 AM - Kyle Knoepfel

- Status changed from Accepted to Assigned

- Assignee set to Kyle Knoepfel

- % Done changed from 0 to 80

- SSI Package - added

Largely implemented with commit [fhicl-cpp:2ce61ba](#).

Some discussion is required regarding how the module library should be packaged/"set up" for use.

#10 - 05/08/2017 01:08 PM - Kyle Knoepfel

- Status changed from Assigned to Resolved

- % Done changed from 80 to 100

Implemented with commit [fhicl-cpp:b4cc5ad](#). The extension module is called fhicl, and PYTHONPATH is appropriately set whenever the fhiclcpp UPS product is set up.

#11 - 05/25/2017 09:10 AM - Kyle Knoepfel

- Target version changed from 1209 to 2.07.01

#12 - 05/25/2017 09:17 AM - Kyle Knoepfel

- Status changed from Resolved to Closed

#13 - 05/31/2017 12:53 PM - Kyle Knoepfel

- Project changed from art to fhicl-cpp

Files

CMakeLists.txt	283 Bytes	04/17/2017	Kyle Knoepfel
fclmodule.cxx	12.5 KB	04/17/2017	Kyle Knoepfel