

LBRUlibs updates.

Krzysztof Furman – 07.09.2021

Unit Tests

- First unit test for ZMQ Pub-Sub with a Poller is now operational
- To create and run a unit test:
 - Create a .cxx unit test file in the unittest folder of your plugin
 - Add it to CMakeLists.txt as a unit test
 - When building the program add “--unittest” to have it run unit tests for locally downloaded plugins in sourcecode folder
 - It is not required to perform a “--clean” build to run unit tests (saves a lot of time when debugging them)

```
=====  
RUNNING UNIT TESTS IN /users/furman/DAQ/2.8/build/./lbrulibs/unittest  
=====  
Start of unit test suite "/users/furman/DAQ/2.8/build/./lbrulibs/unittest/ZMQPubSub_test"  
Running 1 test case...  
Entering test module "ZMQPubSub_test"  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(20): Entering test suite "ZMQPubSub_test"  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(22): Entering test case "SendReceiveTest"  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(39): info: check m_publisher_connected has passed  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(40): info: check m_subscriber_connected has passed  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(55): info: check rcvcd != 0 has passed  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(56): info: check msg.size() == 4 has passed  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(58): info: check received == "TEST" has passed  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(22): Leaving test case "SendReceiveTest"; testing time: 20004648us  
/users/furman/DAQ/2.8/sourcecode/lbrulibs/unittest/ZMQPubSub_test.cxx(20): Leaving test suite "ZMQPubSub_test"; testing time: 20004687us  
Leaving test module "ZMQPubSub_test"; testing time: 20004721us  
  
*** No errors detected  
End of unit test suite "/users/furman/DAQ/2.8/build/./lbrulibs/unittest/ZMQPubSub_test"  
  
Testing complete for package "./lbrulibs". Ran 1 unit test suites.  
  
Test results are saved in /users/furman/DAQ/2.8/log/unit_tests_Wed_1_Sep_12_59_27_BST_2021.log  
(dbt-pyenv) furman@heppc500:~/DAQ/2.8/sourcecode/lbrulibs/unittest$ █
```

Basic ZMQ Pub-Sub unit test passed.

Unit Tests – outstanding issues

- Attempting to unit test the Pacman Frame and its get() functions showcases a design flaw:
 - Our get() functions take a Pacman message (data) as an argument
 - Not an issue in regular operation, but means trying to make a dummy instance of the frame for testing requires creating a fake message
 - We can create a “test” message, but it would seem more appropriate to make the get function entirely generic, just retrieving data members of the Frame
- Need to investigate specifically testing the working thread/data receiver, without loading in all the parameters
- Testing a RAW TCP (STREAM socket based) connection has some syntax challenges
 - At least two possible ways of doing the same thing, it might take some time to find the best solution in C++
 - Despite that a python generator-and-readout is already on a feature branch for when we need this implemented

New Grafana dashboards for readout

- Dashboard for all forms of readout are now available:
 - Using Grafana and requiring some login, which we need to ask about
 - Not sure if login is machine or user specific
 - The core setup is in the pocket repository: <https://github.com/DUNE-DAQ/pocket>
 - This specific 2.8 dashboard is here: <https://github.com/DUNE-DAQ/pocket/blob/develop/images/grafana/dashboards/dunedaq-2.8.0-pocket.json>
 - Not urgent to set up, but useful for presentations/thesis work/publications.



Example readout dashboard for Felix.

Next steps

- Continued development of ZMQ based unit tests
- Developing unit tests specifically calling parts of our plugin
 - Hard coded arguments/parameters
 - Using what was learnt in general ZMQ unit tests to send test messages to them
- Unit tests of the Pacman Frame – with possible design changes to the get() functions
- Taking part in the readout repo reorganization
- Looking into the Grafana dashboards

Thank you



Queen Mary
University of London