

# Initial Experience with Pocket

☰ README.md

## Pocket

Setup a demo installation of DUNE DAQ in a variety of setups.

Pocket is compatible with Linux, MacOS (intel based), and Windows (through WSL2).

### Quick-start

For a cluster with all built-in services enabled:

```
make setup.local
# equivalent to
# SERVICES=ECK,opmon make setup.local
```

This will setup your local (one-node) cluster, and print out available default services and their access credentials.

#### Languages

- Makefile 49.3%
- Shell 33.9%
- HCL 13.3%
- Dockerfile 3.5%

Alan Watson, Ioannis Katsioulas,  
Juraj Bracinik

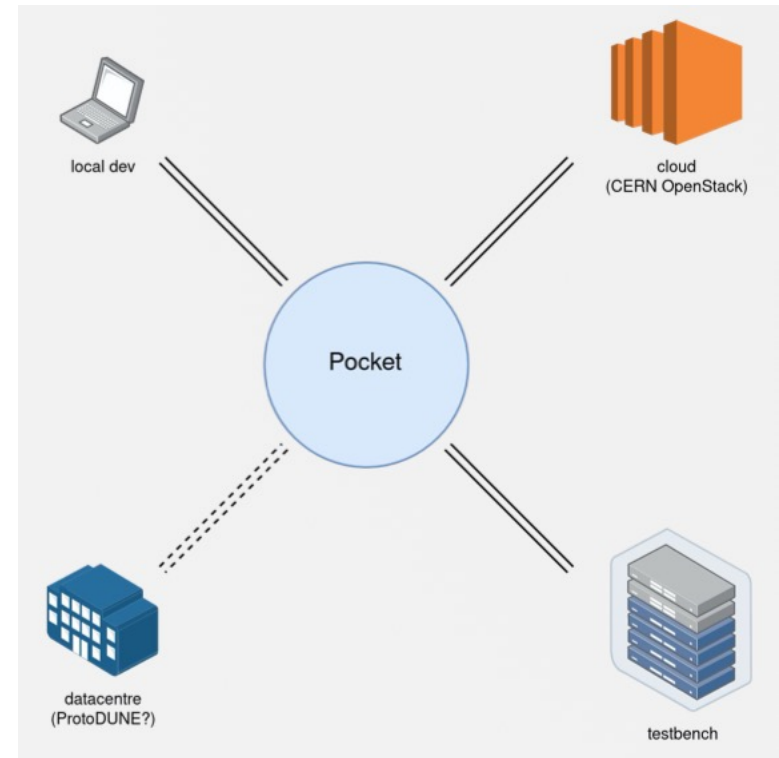


UNIVERSITY OF  
BIRMINGHAM



# Pocket

- A portable DUNE DAQ environment for development and testing
- Motivations
  - Faster development, feedback
  - Closter to real world
  - Easy for different teams to sync with each other
  - Give the feel of a 'real' environment
  - End-to-end testing
- Requires:
  - Locally: docker, make
  - CERN OpenStack
- Described by Glenn in the last meeting, <https://indico.fnal.gov/event/49261/>
- Birmingham group, who are new to DUNE DAQ software, have been trying pocket as “test subjects” using the provided opmon example



# Environments

- Platforms tested
  - MacOS 10.15, Fedora 32, Ubuntu 20.04 (VM)
- Docker availability
  - Sysadmin wary of docker installation on main cluster due to security management
  - Have been working using laptops, though could get a dedicated development machine set up
    - Considered whether could make it run using singularity, but leave that for now unless needed
- Kubernetes
  - If you are not familiar with it, getting kube installed and configured can be a bit of a fiddle
    - Probably took more time than running pocket itself



# A Few Platform-Specific Notes

- MacOS:
  - The optional eval command to use the included binaries rarely works, but you can run without it
  - MacOS doesn't support the "ip" command, but "brew install iproute2mac" solves that
  - The README.md notes that pocket is compatible with "MacOS (intel based)" – is there a known issue with M1 Macs?

# Running daq\_application\_opmon Example

- Once you have docker and (mini)kubernetes running (ideally without needing sudo), running the opmon example was rather painless
  - Kudos to Glenn!

```
daq_application_opmonlib --- ~/DUNE/FD_DAQ/CCM/pocket/pocket/examples/daq_application_opmonlib --- -bash --- 125x38

(dbt-pyenv) [root@example-daq-app-manually exampleapp]#
(dbt-pyenv) [root@example-daq-app-manually exampleapp]# curl -LO https://raw.githubusercontent.com/DUNE-DAQ/listrev/develop/test/list-reversal-app.json
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left   Speed
100 3448  100 3448    0     0    7309    0 --:--:-- --:--:-- --:--:-- 7305
(dbt-pyenv) [root@example-daq-app-manually exampleapp]#
(dbt-pyenv) [root@example-daq-app-manually exampleapp]# daq_application -n batman -c stdin://list-reversal-app.json -i 'influx://opmonlib-proxy.monitoring:80/write?db=influxdb'
2021-Jun-29 22:59:47,902 LOG [dunedaq.opmonlib::makeOpmonService(...) at /scratch/dev/sourcecode/opmonlib/include/opmonlib/OpmonService.hpp:66] SERVICE = influx://opmonlib-proxy.monitoring:80/write?db=influxdb
2021-Jun-29 22:59:49,134 LOG [stdinCommandFacility::stdinCommandFacility(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:55] Loading commands from file: list-reversal-app.json
2021-Jun-29 22:59:49,138 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:83] Available commands: | init | conf | start | stop
init
2021-Jun-29 23:00:04,655 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:94] Executing init command...
2021-Jun-29 23:00:04,658 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:83] Available commands: | init | conf | start | stop
2021-Jun-29 23:00:06,569 LOG [stdinCommandFacility::completion_callback(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:111] Application batman executed command init with result: 1 OK
f
2021-Jun-29 23:00:07,801 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:94] Executing conf command...
2021-Jun-29 23:00:07,803 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:83] Available commands: | init | conf | start | stop
2021-Jun-29 23:00:07,813 LOG [stdinCommandFacility::completion_callback(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:111] Application batman executed command conf with result: 1 OK
start
2021-Jun-29 23:03:41,952 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:94] Executing start command...
2021-Jun-29 23:03:41,953 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugins/stdinCommandFacility.cpp:83] Available commands: | init | conf | start | stop
2021-Jun-29 23:03:41,955 LOG [dunedaq.listrev::ListReverser::do_start(...) at /scratch/patch-dev/sourcecode/listrev/plugins/ListReverser.cpp:72] lr successfully started
```

# A Very Simple Monitoring Demo

- Starting and stopping DAQ application in one terminal
- Viewing status via influx in another

```
daq_application_opmonlib — @example-daq-app-manually/workdir/exampleapp — kubectl exec -it exa...
/sourcecode/listrev/plugins/ListReverser.cpp:134] Reversed list #98, new contents {796, 293, 989, 368
, 290} and size 5. DAQModule: lr
2021-Jun-29 23:05:18,994 DEBUG_63 [dunedaq::listrev::ReversedListValidator::do_work(...) at /scratch/
patch-dev/sourcecode/listrev/plugins/ReversedListValidator.cpp:159] Validating list #98, original con
tents {290, 368, 989, 293, 796} and reversed contents {796, 293, 989, 368, 290}. DAQModule: rlv
op2021-Jun-29 23:05:19,989 DEBUG_63 [dunedaq::listrev::RandomDataListGenerator::do_work(...) at /sca
tch/patch-dev/sourcecode/listrev/plugins/RandomDataListGenerator.cpp:163] Generated list #99 with con
tents {744, 145, 830, 391, 683} and size 5. DAQModule: rd1g
2021-Jun-29 23:05:19,989 DEBUG_63 [dunedaq::listrev::ListReverser::do_work(...) at /scratch/patch-dev
/sourcecode/listrev/plugins/ListReverser.cpp:134] Reversed list #99, new contents {683, 391, 830, 145
, 744} and size 5. DAQModule: lr
2021-Jun-29 23:05:19,989 DEBUG_63 [dunedaq::listrev::ReversedListValidator::do_work(...) at /scratch/
patch-dev/sourcecode/listrev/plugins/ReversedListValidator.cpp:159] Validating list #99, original con
tents {744, 145, 830, 391, 683} and reversed contents {683, 391, 830, 145, 744}. DAQModule: rlv
2021-Jun-29 23:05:20,483 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugin
s/stdinCommandFacility.cpp:94] Executing stop command...
2021-Jun-29 23:05:20,483 LOG [stdinCommandFacility::run(...) at /scratch/dev/sourcecode/cmdlib/plugin
s/stdinCommandFacility.cpp:83] Available commands: | init | conf | start | stop
2021-Jun-29 23:05:20,492 INFO [dunedaq::listrev::ListReverser::do_work(...) at /scratch/patch-dev/sou
rcecode/listrev/plugins/ListReverser.cpp:160] : Exiting do_work() method, received 99 lists and succe
ssfully sent 99. DAQModule: lr
2021-Jun-29 23:05:20,493 LOG [dunedaq::listrev::ListReverser::do_stop(...) at /scratch/patch-dev/sour
cecode/listrev/plugins/ListReverser.cpp:81] lr successfully stopped
2021-Jun-29 23:05:20,990 INFO [dunedaq::listrev::RandomDataListGenerator::do_work(...) at /scratch/pa
tch-dev/sourcecode/listrev/plugins/RandomDataListGenerator.cpp:201] : Exiting the do_work() method, g
enerated 99 lists and successfully sent 198 copies. DAQModule: rd1g
2021-Jun-29 23:05:20,991 LOG [dunedaq::listrev::RandomDataListGenerator::do_stop(...) at /scratch/pat
ch-dev/sourcecode/listrev/plugins/RandomDataListGenerator.cpp:100] rd1g successfully stopped
2021-Jun-29 23:05:20,994 INFO [dunedaq::listrev::ReversedListValidator::do_work(...) at /scratch/patc
h-dev/sourcecode/listrev/plugins/ReversedListValidator.cpp:182] : Exiting do_work() method, received
99 reversed lists, compared 99 of them to their original data, and found 0 mismatches. DAQModule: rlv
2021-Jun-29 23:05:20,994 LOG [dunedaq::listrev::ReversedListValidator::do_stop(...) at /scratch/patch
-dev/sourcecode/listrev/plugins/ReversedListValidator.cpp:84] rlv successfully stopped
2021-Jun-29 23:05:20,994 LOG [stdinCommandFacility::completion_callback(...) at /scratch/dev/sourceco
de/cmdlib/plugins/stdinCommandFacility.cpp:111] Application batman executed command stop with result:
1 OK
```

```
daq_application_opmonlib — ~/DUNE/FD_DAQ/CCM/pocket/pocket/examples/daq_application_opmonlib...
162500790000000000 true 42 0 global_batman
> select * from "dunedaq.rcif.runinfo.Info"
name: dunedaq.rcif.runinfo.Info
time running runno runtime source_id
-----
162500759900000000 false 0 0 global_batman
162500760900000000 false 0 0 global_batman
162500761900000000 false 0 0 global_batman
162500762900000000 false 0 0 global_batman
162500763900000000 false 0 0 global_batman
162500764900000000 false 0 0 global_batman
162500765900000000 false 0 0 global_batman
162500766900000000 false 0 0 global_batman
162500767900000000 false 0 0 global_batman
162500768900000000 false 0 0 global_batman
162500769900000000 false 0 0 global_batman
162500770900000000 false 0 0 global_batman
162500771900000000 false 0 0 global_batman
162500772900000000 false 0 0 global_batman
162500773900000000 false 0 0 global_batman
162500774900000000 false 0 0 global_batman
162500775900000000 false 0 0 global_batman
162500776900000000 false 0 0 global_batman
162500777900000000 false 0 0 global_batman
162500778900000000 false 0 0 global_batman
162500779900000000 false 0 0 global_batman
162500780900000000 false 0 0 global_batman
162500781900000000 false 0 0 global_batman
162500782900000000 false 0 0 global_batman
162500783900000000 true 42 0 global_batman
162500784900000000 true 42 0 global_batman
162500785900000000 true 42 0 global_batman
162500786900000000 true 42 0 global_batman
162500787900000000 true 42 0 global_batman
162500788900000000 true 42 0 global_batman
162500789900000000 true 42 0 global_batman
162500790900000000 true 42 0 global_batman
162500791900000000 true 42 0 global_batman
162500792900000000 false 0 0 global_batman
```

(Is there a clean way of exiting this?)

# Where Next?

- Have shown that someone other than the authors (indeed quite inexperienced users) can set up and run daq examples in pocket
- So, what would be an interesting/useful thing to apply this to next?