NanoRC developments

Pierre Lasorak

- At least 4 improvements that are likely to make it in v2.8.1

  - General division between *np04-real* nanorc and *standalone* nanorc

  - Run database interface

  - Config saving/snapshots

  - Parallelisation of commands

- Another one, that I haven't started yet:

  - Subsystem, multiple configuration directory

- 2 executables for running:

  - nanorc: similar to what was used before

  - nano04rc: same as before but with extra production environment configuration

- Both can be fed a `.nanorc.json`, which contains the addresses of things and authentification

  - `--dotnanorc ~/.nanorc.json`

- Differences explained further on

```
(dbt-pyvenv) np04-srv-010:~/test_dunedaq > nano04rc listrev_2_conf
./
                        Shonky NanoRC
    This is an admittedly shonky nanp RC to control DUNE-DAQ applications.
        Give it a command and it will do your biddings,
        but trust it and it will betray you!
    Use it with care!

Loading /nfs/home/plasorak/.nanorc.json
shonky np04rc>
boot        exit        init        quit        scrap       status      terminate
conf        help        pause       resume      start       stop        wait
shonky np04rc> quit
[09:55:30] WARNING   NanoRC context cleanup: Terminating RC before exiting
           WARNING   Terminating
(dbt-pyvenv) np04-srv-010:~/test_dunedaq > nanorc listrev_2_conf
                        Shonky NanoRC
    This is an admittedly shonky nanp RC to control DUNE-DAQ applications.
        Give it a command and it will do your biddings,
        but trust it and it will betray you!
    Use it with care!

shonky rc>
boot        exit        init        quit        scrap       status      terminate
conf        help        pause       resume      start       stop        wait
shonky rc>
```

- Service that accesses a DB to get the run number

  - Supervised and in ansible (thanks Bonnie!)

  - Runs behind gunicorn/gevent as of 1.5 hour ago.

  - Dummy authentication credentials → maybe need something a bit better when we switch to production DB.

  - np04rest has 3 commands, and that's ok, right?

    - Get run number

    - Increment run number

    - Stop run

- Interface in nanorc:

  - Configurable credentials in `.nanorc.json`

    - Maybe good time to think about how we are going to store password…?

  - Start command doesn't need the run number anymore for nano04rc

    - Still needed for standalone nanorc

  - Uses a run number manager in the code

- Config saved in the `${PWD}/RunConf_${RunNumber}`

  - Path configurable with `--cfg-outdir /whatever/you/want`

  - Saves the resume and start commands when they are issued

- Done by Alessandro

- Separated the send_command now only sends command, and implemented a check_response and send_command_and_wait in AppSupervisor

  - send_command sends all the commands without waiting for anything

  - check_response queries if the app has executed the command

  - send_command_and_wait sends command and wait for the response.

- Desired behaviour depends whether an command order is specified in the json.

- 2 ways:

  - Nicely with "subsystems"

  - Hacky by merging json directories