

NanoRC status

Pierre Lasorak

- NP04 specific RC
- Credentials management
- Run number database
- Run registry database
- Parallel commands
- Subsystem tree

- 2 executables:
 - nanorc → standalone, with no relation with databases
 - nano04rc → full data-taking ready run control

- Status: ready, on develop

- To accessing databases, one requires socket / username / password
- Stored in a config file called `.nanorc.json`
- Default location in the home directory
 - Configurable:
`nano04rc --dotnanorc /where/is/.nanorc.json config_dir`
- Status: ready, on develop

- Thanks to Enrico, we have a DB that gives us run number
- Used in `nano04rc` only
- No need to tell `start` which run we are taking, we use the DB for that

- Status: ready, on develop
 - However we will probably need to spin another run number DB, right now we use a test DB

- Thanks to Roland, we have a run registry DB
- Entries in the DB:
 - Run number ← filled on start
 - Run start timestamp ← filled on start
 - Run end timestamp ← filled on stop
 - Run type ← filled on start
 - Detector id ← filled on start
 - Configuration blob ← filled on start
- On `nano04rc`, the `start` command takes the run-type
 - That can only be “**TEST**” or “**PROD**” (all caps)
- Status: in PR31: <https://github.com/DUNE-DAQ/nanorc/pull/31>
 - Instructions to test it: <https://github.com/DUNE-DAQ/nanorc/wiki#how-to-runtest-the-run-registry-implementation>
 - Also will need to drop the test DB and use the real one

- Alessandro changed the way the commands are dispatched to the applications
- If order is specified, sequentially send the command to the apps, otherwise send them all in one go, and wait for answers
- Status: ready, on develop

- Now possible to use a `top_level.json` instead of the configurations
 - This file contains all the configuration folders that we want to use, and we can create a hierarchy of them or group them
- Translated into a tree structure within `nanorc`
- User can then send commands
 - `init = init /root = init /`
 - `conf /root/lrevs` ← find the correct lrevs node and send conf to whatever is underneath
 - `start` and `stop` ← always uses `/root` (i.e. sends command to everything)
- Status: in PR30
 - <https://github.com/DUNE-DAQ/nanorc/pull/30>
 - Testing is encouraged! Instructions: <https://github.com/DUNE-DAQ/nanorc/wiki/Testing-subsystem-trees>

top_level.json

```
{
  "lrevs": {
    "lr1s": {
      "conf-dir": "listrev_2_conf"
    }
  },
  "daq": {
    "conf-dir": "mdapp_fake"
  }
}
```

Apps					
name	host	alive	pings	last cmd	last succ. cmd
root					
├─ lrevs					
│ └─ lr1s					
│ └─ lr0	np04-srv-010	True	True	init	init
│ └─ lr1	np04-srv-010	True	True	init	init
└─ daq					
│ └─ dataflow	np04-srv-010	True	True	init	init
│ └─ hsi	np04-srv-010	True	True	init	init
│ └─ ruemu0	np04-srv-010	True	True	init	init
│ └─ trigger	np04-srv-010	True	True	init	init

- Evolving documentation
- <https://twiki.cern.ch/twiki/bin/view/CENF/NanorcRunControl>