Auto-processing of data

Content

There are few tasks which need in processing of all runs, e.g. check data quality, check detector pixel status etc. This tool is intended to automatically process each data run with some pre-defined command line.

This note has a short description of entire system and references to code.

Control scripts

cron_auto_proc

Job control is performed using cron-job, currently resides in ~/bin/

```
~/bin/cron_script_1h.sh # hourly kicking ass of the next script, output goes to ~/test_crontab_1hour
~/bin/cron_auto_proc.sh # sets environment and runs python script proc_new_datasets (PSCalib/app)
```

can be moved to `/etc/cron.hourly/`

proc_new_datasets

`PSCalib/app/proc_new_datasets`

- defines a list of datasets (experiment and run pairs) which need in processing
- loop over list and if number of allowed jobs does not exceed a number of running jobs in batch starts in subprocess `proc_control`

`proc_control`

PSCalib/app/`proc_control`

- submits job (currently for procname='pixel_status') for a single dataset processing in batch and checks its status with some period untill it is not completed in any ways.
- Check-loop is running for status in (None, 'RUN', 'PEND') and terminated in status in ('EXIT', 'DONE')
- if status != 'DONE' - log file name is appended by the status.

Process name

procname='pixel_status' stands for butch execution of the command line `det_ndarr_data_status`

det_ndarr_data_status

`Detector/app/det_ndarr_data_status`

- dataset processing for specified detectors with aim to find bad pixels and save a calib status file of type pixel_datast (similar to pixel_status for dark runs).

Subprocess utilities

- PSCalib/src/SubprocUtils.py

Datasets processing DB

File system based DB is used to keep info about processed runs. Everything lives under master directory, for now `/reg/g/psdm/logs/run_proc`

/reg/g/psdm/logs/run_proc/<process-name>/

/reg/g/psdm/logs/run_proc/pixel_status/experiments.txt - list of experiments which need to be processed

/reg/g/psdm/logs/run_proc/pixel_status/<instrument>/ - subdirectory for each instrument

/reg/g/psdm/logs/run_proc/pixel_status/<instrument>/<experiment>/ - subdirectory for batch submission and processing log-files

/reg/g/psdm/logs/run_proc/pixel_status/<instrument>/<experiment>-proc-runs.txt - file with list of processed runs, e.g.

/reg/g/psdm/logs/run_proc/pixel_status/CXI/cxil2316-proc-runs.txt

Run for processing finding utilities

- PSCalib/src/RunProcUtils.py

Commands

`Manual launch of the scripts in stead of cron-job:`

- ~/bin/cron_script_1h.sh
- ~/bin/cron_auto_proc.sh

See current processing status

- proc_new_datasets
- proc_new_datasets -s # submit jobs

Print / control DB status

- `python PSCalib/src/RunProcUtils.py 1` # list of non-processed runs for experiments in the list `.../pixel_status/experiments.txt`
- `python PSCalib/src/RunProcUtils.py 2` # list of non-processed runs for all experiments
- `python PSCalib/src/RunProcUtils.py 4` # list of runs listed in logs but removed from xtc directory
- `python PSCalib/src/RunProcUtils.py 5` # print statistics about all available instruments, experiments, runs.
- `python PSCalib/src/RunProcUtils.py 40` # move processed runs from `<experiment>-proc-runs.txt` to `<experiment>-arch-runs.txt for removed xtc files`
- `python PSCalib/src/RunProcUtils.py 10` # the same as 1, but runs are marked as processed in `.../pixel_status/<instrument>/<experiment>-proc-runs.txt`
- `python PSCalib/src/RunProcUtils.py 20` # the same as 2, but runs are marked as processed in `.../pixel_status/<instrument>/<experiment>-proc-runs.txt`

- `python PSCalib/src/SubprocUtils.py 1,2,4 #` the same as above

References

- Detector
- PSCalib