K1.8 analyzer README

Rev. 2024.10.23
2020.09.06

It is assumed to work on KEKCC, Red Hat Enterprise Linux release 9.3 (Plow). Use gcc 11.4.1 and root 6.32.04.

Environment setting

See KEKCC.pdf and set environment variables. Anaconda/Python environment is needed to use runmanager.

Unpacker

The unpacker compiled with gcc 11.4.1 is placed in the group directory, /group/had/sks/software/unpacker/e70. This is updated constantly.

If you want to install in local, install as follows.

```
$ git clone ssh://sks@www-online.kek.jp:8022/~/public_html/git/unpacker.git
$ cd unpacker/src
$ cp Makefile.org Makefile
$ make
```

Check if the "unpacker-config" command is available.

```
$ unpacker-config --version
2024-10-21
```

K1.8 analyzer

Install the K1.8 analyzer.

```
$ git clone \
ssh://sks@www-online.kek.jp:8022/~/public_html/git/k18-analyzer.git
$ cd k18-analyzer
$ git checkout e70 # choose branch as you like
$ cp Makefile.org Makefile
$ make
```

e.g.) Hodoscope, Usage: Hodoscope [analyzer config file] [data input stream] [output root file]

```
$ ./bin/Hodoscope param/conf/analyzer_e70_2024_0429.conf \
/group/had/sks/E70/JPARC2024May/e70_2024may/run71244.dat hoge.root
```

runmanager

runmanager is a script for managing jobs on KEKCC. Prepare runlist.yml by referring to the example.yml. Note that the indentation determines the nest depth in yaml.

```
#
# RUN LIST (YAML format)
#
# The allowed keys are
#   queue  <- bsub queue (eg. s, l, etc...)
#   qmerge <- bsub queue for merging job to command "hadd"
#   unit   <- dividing event unit
#   nporc  <- number of process for merging (up to 18)
#   buff   <- intermediate root files will be placed here if nproc is more
#             if not set, use the same "root" directory
#   bin    <- path to executable binary
#   conf   <- path to conf directory/file from the work directory
#   data   <- path to data directory/file
#   root   <- path to the output ROOT output directory/file
#
# Be careful of the indent rule because YAML format is sensitive to it.
# Some problems will happen if there is no default setting declaration.
#

#_____________________________________________________
# work directory path
```

```yaml
# The following paths must be relative to this path

WORKDIR: ~/k18analyzer/pro

#_____________________________________________________
# default setting
# default setting MUST have all items.
# This setting will be inherited unless you explicitly set values
# for the individual cases.

DEFAULT:
  queue: s
  qmerge: s
  unit:  100000
  nproc: 1
  buff:  /group/had/sks/Users/user/buffer
  bin:   ./bin/Hodoscope
  conf:  ./param/conf/analyzer_default.conf
  data:  ../data
  root:  ../rootfile

#_____________________________________________________
# Individual settings
# If you want to adapt the default setting to some runs,
# you only have to list keys.

RUN:

  test: # Any keys are OK unless it overlaps.
    bin:  ./bin/Hodoscope
    conf: ./param/conf/analyzer_default.conf
    data: ./test.dat.gz
    root: ./test.root

  3838:
  # The default setting will be adapted.
  # In this case run03838.dat(.gz) will be loaded.

  3800: # ./param/conf/analyzer_03800.conf will be loaded.
    bin:  ./bin/KuramaTracking
    conf: ./param/conf
    data: ./tmp_data

  hodo:
    data: ../data/run03800.dat.gz
```

Run run.py. Note that the process runs until all jobs have finished. Ctrl-C kills all jobs and terminates the process.

```
$ ./runmanager/run.py runmanager/runlist/foo.yml
```

Run monitor.py on another tty to see the progress of the jobs. The job status is updated in the "stat" directory, using the same name as the runlist in json format.

```
$ ./runmanager/monitor.py runmanager/stat/foo.json
```