Data analysis using the *TWIST* batch system

Technical note #77

Andrei Gaponenko

February 26, 2003

Abstract

This note suggests a convenient way to analyze sets of raw data files using the facilities of the *TWIST* Run Database and batch job management system.

Introduction

The suggested steps to start an analysis are:

1. Prepare a MOFIA executable and a kcm file.
2. Decide on a list of run numbers you want to analyze. Create a "run request" from the list.
3. Decide where to place the output files and submit the jobs using the `submit_mofia.pl` script with proper arguments.

Note that there is no need to wait till the requested runs are read from tapes. You can submit jobs immediately after creating a run request, and the system will take care of the rest. Moreover, the cases when a data set to be analyzed is too big to fit on disk are also handled. See the `--cancel` option of `submit_mofia.pl`.

The MOFIA and KCM files

Compiling MOFIA and preparing a KCM file is documented elsewhere. You can use a self contained KCM file which includes the "analyze" and "quit" commands. Otherwise, if your KCM file does not terminate MOFIA, the program will get "analyze", "show errors" and "quit" commands on STDIN.

Creating a run request

Go to `https://e614db.triumf.ca/~e614data/` in a Web browser, click on "Create a new request" on the "Run requests" page, and follow the instructions to enter your list of run numbers. After the request is accepted you will see it listed on the "Run Requests" page. Notice the number of your request, you will need it to submit analysis jobs.

Using the `submit_mofia.pl` script.

The script submits jobs to the batch system, one job per run. You can find `submit_mofia.pl` in the `scripts/` directory of the MOFIA CVS tree. It is self-documented: if you run the script without any arguments, or give the `--help` switch among others on the command line, it will print out a usage message and exit.

The required arguments to start an analysis are:

–reqnum=*number* Request number. This is the number shown on the "Run Requests" web page for your request. This argument defines the set of runs to be analyzed.

–mofia=*file* The MOFIA executable to use for the jobs. You can use either a relative (`--mofia=dec08wire`), or an absolute (`--mofia=/some/dir/mofia`) path name.

–kcm=*file* The KCM file. Both relative and absolute path names are accepted.

–outdir=*directory* Where to place the output. Both relative and absolute path names are accepted. For every run, a new subdirectory inside outdir will be created, and MOFIA will run in the subdirectories. The batch script does not overwrite old data: if there is already a directory for a run inside outdir, the corresponding batch job will complain. Check errorlog* files in the directory where you submitted the jobs from.

Important: do not alter the binary or the kcm file till all the jobs are finished. I suggest to rename `mofia` or to move it outside of `MOFIA_USER` and use the copy to run the analysis. So that if you recompile `mofia` your jobs are not affected. The same applies to the KCM file: do not use your working version, it's safer to use a copy.

The optional arguments accepted by `submit_mofia.pl` are:

–local Requires that analysis of run XXX is done on the node which has run XXX on its local disk. By default the job can start anywhere and read the raw data file via NFS. There is an explicit `--nfs` option which is mutually exclusive with `--local`.

–cancel As soon as analysis of a given run is finished, the run is removed from the request. Then the data file can be removed from disk by the Run Database, and another run can be read from tape. This allows to process arbitrarily large data sets with a limited amount of disk space. There is an explicit `--keep` option which is mutually exclusive with `--cancel`.

–jobname=*prefix* Sets a prefix for the names of your jobs. The default is "mofia". Job name is made by appending the request number and the run number to the prefix, e.g. "mofia_32_11380".

–pbsopt=*'opt1 opt2 ...'* This allows you to pass any additional options directly to the PBS `qsub` command. For example, to disallow the use of slow machines for the analysis: `--pbsopt='-l nodes=1:cpu1000'`.

Be careful to not over-restrict your jobs. For example, if you want `--local`, and also requested cluster nodes only, but a data file sits on a desktop machine, a job on that file can never run.

Example

Note: I split a single command into multiple lines with the '\' shell continuation symbol to make it fit on the page. It all can be typed on a single line, of course.

```
rach $ ~/e614soft/triumf/mofia/devel/source/scripts/submit_mofia.pl \
    --reqnum=36  --mofia=./mofia \
    --kcm=/twist/data15/fall2002anal/fall2002prod.kcm \
    --outdir=/twist/data17/andr/pbstest/request36

Processing run 12765: PBS job 2321.e614db.Triumf.CA
Processing run 12766: PBS job 2322.e614db.Triumf.CA
Processing run 12767: PBS job 2323.e614db.Triumf.CA
Processing run 12768: PBS job 2324.e614db.Triumf.CA
Processing run 12769: PBS job 2325.e614db.Triumf.CA
Processing run 12770: PBS job 2326.e614db.Triumf.CA
Processing run 12771: PBS job 2327.e614db.Triumf.CA
Processing run 12772: PBS job 2328.e614db.Triumf.CA
Processing run 12773: PBS job 2329.e614db.Triumf.CA
Processing run 12774: PBS job 2330.e614db.Triumf.CA
Processing run 12775: PBS job 2331.e614db.Triumf.CA
rach $
```

Now you can use standard PBS commands to monitor your jobs.

To do

Some of the possible developments of the system are:

1. Handle GEANT runs. This requires extensive modifications to the Run Database to handle multiple data types. If we need to write MC files to tape this should be done anyway. The batch system part is simple.

2. The system as is has a limitation of using a single file system to store the output (the `--outdir` argument). If it turns out to be an issue (multiple parallel jobs might overload a NFS file server), it might be important to allow for multiple output directories.

3. Add disk space checks. The system can be extended to accept an estimate of required disk space per job (e.g. one more argument to `submit_mofia.pl`). If there is not enough space on the output file system, the scheduler should not start the job. Disk requirements for multiple jobs running in parallel can be accounted for through creating per job disk space reservations in the database. Nice, but probably very low priority.