E815 CLUBS/FNALU User's Guide Part II

Version 1.3

H. Schellman

June 26, 1997

1 Things you need to setup to run batch jobs, read and write tapes

`setup fbatch` - set up the batch system

`setup nt` - set up the needfile tape access system

`setup -d fmss` - set up the fmss robot access system

`setup mssout` - set up the mssout robot → tape dump system

2 e815_analysis conventions on clubs

In your job script, you should have the following line:

```
ln -s $FERMINT_DPOOL_DIR scratch_dir
```

This will make the temporary work space available as a subdirectory of the job you are running the job from.

3 General Code for reading a tape from clubs

For an 8mm tape, our convention is to call the files on the tape: `NNNNNN.nn` where `NNNNNN` is the tape name and `nn` is the file number. The code examples below are the e815_analysis convention and assume that the files are called

```
scratch_dir/NNNNNN.nn
```

The splitter code uses a similar implementation.

To read the tape, you need to stage the file to disk. When you are done, you need to remove the file from the temporary disk. It is often efficient to prestage the next file while you are using the present file. The routine

process.f has code to perform this function but you may with do so in a different way.

3.1 Pre-staging the file

This code will issue a command to the system to stage the file but will not wait until the file is staged to continue. The option `-nw` says not to wait.

```
* parse filename

      TAPE_VOLUME = filename(13:18)
      TAPE_FILE = filename(20:21)

* build command

      CLUBS_PRE = 'needfile -vsn '//TAPE_VOLUME//
     *' -lb '//TAPE_VOLUME//'.'//TAPE_FILE//' -tape 8mmd '//
     *' -file '//TAPE_FILE//' -lrecl 7200  -block 7200 -nw'

* execute command

      call cl_callsys(CLUBS_PRE)
```

3.2 Staging the file

This code stages the file to disk and waits until the stage has been successful. You should always do this before trying to access the file even if you have issued a prestage command previously.

```
* parse filename

     TAPE_VOLUME = filename(13:18)
     TAPE_FILE = filename(20:21)
```

```
* build command
      CLUBS_PRE = 'needfile -vsn '//TAPE_VOLUME//
     *' -lb '//TAPE_VOLUME//'.'//TAPE_FILE//' -tape 8mmd '//
     *' -file '//TAPE_FILE//' -lrecl 7200  -block 7200 '

* execute command

      call cl_callsys(CLUBS_PRE)
```

3.3 Clearing the file

This command should be issued after the file has been processed to clear up space for the next file.

```
* clear the  file

      TAPE_VOLUME = filename(13:18)
      TAPE_FILE = filename(20:21)

      CLUBS_CLR = 'needfile -clear '//
     *TAPE_VOLUME//'.'//TAPE_FILE

      call cl_callsys(CLUBS_CLR)
```

This code is currently installed in e815_analysis and is turned on by the clubs option:

```
make clubs
```

4 Writing output into the robot

Writing output to the robot is simple. You write a disk file and then, when you close a file, you issue a system command which copies the file into the robot. It is often a good idea to delete the disk file after you have copied it to save space.

5 Command syntax

`fmss mkdir mss:/e815/newdir` will make a new directory. Everyone in the e815 group has access to the /e815/

`fmss -R disk:myfile.dat mss:/e815/newdir/` will copy the file `myfile.dat` into the directory `/e815/newdir`. If you use wild cards, you have to enclose the whole file specification in double quotes `"`.

`fmss ls mss:/e815/newdir/` will list the contents of the directory

`fmss rm mss:/e815/newdir/myfile.dat` will remove the file `myfile.dat` from the directory.

You can do these from any machine on fnalu/clubs and, in future, from the e815 cluster.

Warning, the present system is accessible to all e815 collaborators. However, you must make your own directory with your username and keep your files there. Any files floating in the general area will be deleted. In addition, space in the robot is a limited resource, individual users should not keep more than 5GB of files. Anything larger should be discussed with the group and made into a production project.

6 e815_analysis implementation

In e815_analysis there will soon be two utility routines `TCHINI` and `TCHDMP` which set up and then dump a disk file into the robot.

`TCHINI` takes an output file name `NAME` and constructs a file name `NAME1 = /scratch_dir/NAME` for later use. The normal e815 server code can be used to open this output device.

`TCHDMP` copies the output file `NAME1` into the directory users robot directory. You should then delete the disk file from `scratch_dir`.

As of June 5, this code is not yet fully installed as it requires some CVS work. In future the `clubs` option will turn it on. You should specify `TYPE OCLB XXXXXX` to get the output you want.

7 Split code implementation

The directory ~e815prd1/splitter contains what you need to submit split jobs.

7.1 GETTING the splitter READY

It expects to have the executable in:

`splitter.exe`

and the file

`splitter.inp` as input. `splitter.inp` has a special format with the input tape name replaced by the string `XXXXXX` and the number of input files replaced by the string `YYYYYY`. Do not blindly replace this file!

You need to prestage the tape first to find out how many files are on it.

The syntax is `prestage XXXXXX`

`ndflist XXXXXX` will tell you when the tape is prestaged.

7.2 RUNNING a JOB

To submit a job type

`submit_split XXXXXX 4hr`

where `XXXXXX` is the tape you want to split and `4hr` is the queue.

This will create a subdirectoy `spXXXXXX` which the job will run from.

e815prd1 gets an email when the job completes.

You can monitor the job by doing `fbatch_jobs`

`fbatch_peek` is a more advanced command you might want to check out.

7.3 CHECKING the job

the files `spXXXXX.out` and `spXXXXXX.err` will appear in the `spXXXXXX` directory when the job is done. Look at these to confirm that the job ran ok.

Then go to the splitter directory and type:

```
mv spXXXXXX success
```

that will move the job into the success directory.

type

```
fmss ls "mss:/e815/split/"
```

to see the files in the robot.

7.4 OUTPUTTING a tape

the command

```
make_output OUTVSN STREAM INVSN1 INVSN2 INVSN3 ....
```

will take the outputs for stream `STREAM` coming from tapes `INVSN...` and create a job which will output them to tape `OUTVSN`.

After this creation you will find a file `STREAM_OUTVSN.mssout`

look at this file, make certain it has the files you want in it and then type

```
mssout STREAM_OUTVSN.mssout
```

this will submit a tape dump request.

you will receive email when it finishes.