Using ntuplizer

Reynier Cruz Torres

October 19, 2020

Contents

1 Running jobs with ntuplizer

Jobs are run on lxplus, then copied to and merged and stored on Cori. Lastly, the output files should be deleted from the grid.

1.1 Running jobs on lxplus

- ssh into lxplus: `ssh rcruztor@lxplus.cern.ch`
- clone ntuplizer (only once): `https://github.com/alwina/ntuple-gj.git`
- execute the following two commands (possibly add them to the bashrc file):
  - `source /cvmfs/alice.cern.ch/etc/login.sh`
  - `alienv enter VO_ALICE@AliPhysics::vAN-20200629_ROOT6-1`
- cd into the ntuplizer main directory
- run the ntuplizer `./macros/runNTGJ.C config/embed_19i3c2_pthat1.yaml full`
- check the status of your jobs by going to alimonitor.cern.ch, login, and go to the 'My jobs' tab

1.2 Copying, merging, and storing jobs on Cori

- `ssh -Y reynier@cori.nersc.gov`
- from lxplus (only once):

  `scp -r ~/.globus reynier@cori.nersc.gov:/global/homes/r/reynier/`
- add the following to your .bashrc file (only once):

```
#ALICE
if [ -d /cvmfs ];then
        export alien_CLOSE_SE="ALICE::LBL::EOS"
        source /cvmfs/alice.cern.ch/etc/login.sh
        module use /cvmfs/alice.cern.ch/x86_64-2.6-gnu-4.1.2/Modules/modulefiles
        alias alish="eval \$(alienv load AliPhysics)"
fi
```

- run the .bashrc file
- `alish` and if that does not work, do `module load AliPhysics` instead.
- `cd /project/projectdirs/alice/reynier`

- clone ntuplizer (only once): `https://github.com/alwina/ntuple-gj.git`
- cd into the ntuplizer main directory
- `alien-token-init rcruztor`
- To copy (on cori - note the 8 at the end of the second line):

  ```
  ./macros/alien_fastcopy \
  /alice/cern.ch/user/r/rcruztor/workingdir/outputdir_19i3c2_pthat1 \
  /global/project/projectdirs/alice/reynier/19i3c2/pthat1 8
  ```

- To merge (on cori - note the "" and /*):

  ```
  ./macros/MergeNtuple.C \
  "/global/project/projectdirs/alice/reynier/19i3c2/pthat1/*/*/AnalysisResults.root" \
  /global/project/projectdirs/alice/reynier/19i3c2/18q_embed_19i3c2_pthat1.root
  ```

  * Replace /global/project/projectdirs/alice/reynier/19i3c2/pthat1 as you like

  * Be careful in the final merging step. You will overwrite any existing file of the same name. One option is to create the merged ntuple somewhere else and then move it into NTuples/embed when it's done.

1.3 Deleting copied files from the grid

From lxplus, depending on exactly which version of AliPhysics you have, you either do `aliensh` or `alien.py` (from anywhere). Then that'll take you into a shell from your grid home directory From there, you should be able to see workingdir, and inside there, the outputdir_XXX directories if you used aliensh, you have to use rmdir to remove a directory if you used alien.py, you just use rm -r like a normal shell from alimonitor.cern.ch, you can click "My home dir" in the top toolbar to navigate that from a web browser, but I have yet to figure out how to remove files through that. So instead we go through alien.py/aliensh

1.4 Checking output files

- `root -l filename.root`
- `_file0->cd("AliAnalysisTaskNTGJ")`
- `_tree_event->GetEntries()`

2 Email from Alwina (07/15/2020)

Hi Rey,

This email is basically, start-to-finish, my understanding of how to produce ntuples. As such, it's rather long, but it includes pretty much everything I could think of.

Generally speaking, to run the ntuplizer, you have to have the ALICE software framework somewhere and a grid certificate and then to submit jobs through that. You also need access to cori to pull the outputs from the grid and merge them into a single file.

Personally, I built the ALICE software using alidock. The general "tutorial" for building the software is here: https://alice-doc.github.io/alice-analysis-tutorial/building/. I have had few if any problems building with alidock running ubuntu 16; the hard part was figuring out what flags and tags and such I wanted to use.

I know that Fernando had some issues trying to use alidock. He has a Mac, which I think caused some problems when the ntuplizer was built on top of ROOT 5. However, with the new updates, it runs with ROOT 6, so it's quite possible that he and anyone else using MacOS will be able to build it now.

Yue Shi has some build scripts for building this on NERSC. In particular, I think we would want to get this to run on cori, but as I mentioned, I have not been thinking about this very much. The scripts are here: https://github.com/yslai/build-nersc.

I do know that there is some version of AliPhysics already on cori. I have been able to do 'module load AliPhysics' and get some things to run. However, I haven't been able to submit jobs to the grid, and I haven't tried very hard to figure out why.

Speaking of the grid, you will need a grid certificate to submit jobs and interact with the grid in general. I think Fernando has more detailed notes, but the "official" (and, unfortunately, sometimes self-contradictory) instructions are here:

https://alice-doc.github.io/alice-analysis-tutorial/start/cert.html

I am unclear about what your status is with ALICE, so there may be something you have to do there first.

Once you've built the ALICE software and have a grid certificate, you can run the ntuplizer itself. Doing so is fairly straightforward; once you've pulled the repo (https://github.com/yslai/ntuple-gj), you just do something like

```
./macros/runNTGJ.C config/15o.yaml full
```

The first argument is the configuration file you have to use; I think the ones in the repo are out of date, but we're still figuring out what configuration settings we need, so probably nothing works at the moment. I think I have a sense of what all the settings do, so feel free to ask about those.

The second argument can be either "test" or "full"; I believe "test" pulls files from the grid but uses your machine and resources to run the job, while "full" submits the job to the grid, which

then gets distributed and such. As you might guess, "test" is only really used for testing, and "full" is what we use to actually produce the ntuples.

To monitor the job status, you need to have a grid certificate loaded into your browser and to go here: https://alimonitor.cern.ch/users/jobs.jsp

Once the jobs are done, the next step is to copy. This, I highly suggest you do on cori unless you have lots of network bandwidth and disk space. The copy command looks something like this:

```
./macros/alien_fastcopy /alice/cern.ch/user/a/alwina/workingdir/15o/outputdir /global/project/projectdirs/alice/alwina/15o 8
```

The first argument is the location you're copying from. The second argument is the location you're copying to. The third argument is the number of parallel processes; I've always just used 8. It's smart about copying; I think it's more like rsync than scp, but in any case, it won't copy a file that already exists. The upshot is that you can pause or interrupt the copying process in the middle and it'll pick up where it left off.

Finally, once things are copied, you want to merge all the outputs into a single ntuple. This is done with something like

```
./macros/MergeNTGJ.C "/global/project/projectdirs/alice/alwina/15o/*/*/AnalysisResults.root" /global/project/projectdirs/alice/NTuples/PbPb/15o.root
```

Note that the first argument is the same as the second argument in the copy command, only with /*/*/AnalysisResults.root appended (this is always the same) and within quotation marks (this is important). The second argument is the name of the output. Unlike copying, this has to be done all at once and cannot be parallelized or interrupted. There's also nothing to really monitor the status of the merging, except to look at the size of the output file and watch it grow.

As a side note, you may notice that on cori, I always use directories within:

```
/global/project/projectdirs/alice/
```

This is because there is more available space here than in the individual home directory. However, that space is not infinite and is shared amongst ALICE people (as you might imagine), so it's generally nice to delete things you don't need.

Also, there's some level of attrition at each level. There's always some portion of the subjobs that fail. You can try resubmitting those through the job monitoring site, but often it's not worth it. There's also some portion of the outputs that don't get copied for one reason or another. You can try to get some of those by rerunning the copy command, but there's always at least a few that don't make it. Over time, I have come to expect something like a 90

I think that's it. If you have any questions, feel free to ask.

Best, Alwina

3 Meeting with Dhruv (07/15/2020)

Levels of data filtering in Alice:

- Raw
- ESD
- AOD

With more skimming as we go down this list.

The main codes to read are `AliAnalysisTaskNTGJ.cxx` and `AliAnalysisTaskNTGJ.h` in the ntuplizer repo https://github.com/yslai/ntuple-gj. These two codes are the body of the ntuplizer.

The ntuplizer is run with `macros/runNTGj.C`.

`run_mode = "test"`

`run_mode = "full"`

+ config file

Run locally. Need to install Alibuild. Run with python 3.7 or higher.