
Subject: Tutorial for analysis macros on the GRID
Posted by [StefanoSpataro](#) on Thu, 23 Jun 2011 16:07:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,
in order to run analysis on the GRID for tracking TDR, I have written some sample files, to use it as tutorial.

First of all, you have to create a collection with the .zip files containing the root file you want to analyze. You can run at present a maximum amount of 60 jobs in parallel, thus a maximum of 60 folders.

In order to create a collection "coll_702_a" with all the sig_output_run702.zip files from runid 702 (pipi), all the sub-folders in 10*:

```
find -c /panda/user/s/spataro/tdr11/coll/coll_702_a  
/panda/user/p/pbarprod/tdr11/output/sig/run702/10*/ sig_output_run702.zip
```

Now we have to create the script doing the job. This should stay in the /panda/user/x/xxx/bin folder (i.e., in my case, /panda/user/s/spataro/bin). I have done a copy of the test one in pbarprod account, that you can use:

```
cp /panda/user/p/pbarprod/bin/test-ana.sh /panda/user/s/spataro/bin/test-ana.sh
```

This script, if you read it, is unzipping the .zip file (containing the root files), renaming the files into the standard evt_XXX_stt.root, and executing the macro run_ana_pipi.C with root. If the output file "finalroot_piplus.root" is not existing then it gives an exit statement (error!).

Toggle Spoiler

```
[pgdb2.gla.ac.uk:3307] /panda/user/s/spataro/tdr11/ > cat  
/panda/user/s/spataro/bin/test-ana.sh
```

```
#!/bin/bash
```

```
#
```

```
# Dependencies
```

```
# -----
```

```
# pbarprod@pandaroot::may11
```

```
# pbarprod@panda_extern::may11
```

```
#
```

```
# input parameters:
```

```
# -----
```

```
#
```

```
# Output:
```

```
# -----
```

```
# finalroot_piplus.root
```

```
#
```

```
# Ste, June. 2011
```

```
#
```

```
#
```

```
# Run the scripts
```

```
#
```

```
export ranseed=`date +%N`
```

```
echo "This is a script to run analysis"
```

```

echo "from evtgen events for STT"
echo
echo "Starting now the analysis:"
echo

source $VMCWORKDIR/./build/config.sh

echo "----- "
echo "starting analysis"
echo "run_ana_pipi.C"

echo "unzipping..." >> stdout 2>&1
unzip *.zip >> stdout 2>&1

echo "renaming files" >> stdout 2>&1
mv evt_points* evt_points_stt.root
mv evt_pid* evt_pid_stt.root

ls -ltr >> stdout 2>&1

root -b -q "run_ana_pipi.C" >> stdout 2>&1

echo ""
if [ -s "finalroot_piplus.root" ]; then
  echo "run_ana_pipi.C finished successfully"
else
  echo "ERROR: run_ana_pipi.C finished without producing root file!"
  exit 12
fi

ls ./*.root

echo ""
echo "----- YUHUU -----"

```

The question now is: how to set the input file, the output folder, which script to execute and from where taking the macro. For this, you have to create your jdl, or copy the one I have created for you:

```
cp /panda/user/p/pbarprod/tdr11/jdl/test-ana.jdl /panda/user/s/spataro/tdr11/test-ana.jdl
```

Here you set your script, the input data collection, the "input file" (i.e. the macros that will be copied), the output archive (the root file will be archived into one .zip stored in 2 disks, and also the log files will be stored in 1 disk). Split directory means that a subjob will be sent for each directory inside your collection. Finally the output files will be stored into Outputdir. You have to modify it (using "vi" or just "getting" the file in your local pc, updating it and "adding" it again into the grid) because you do not want to retrieve files and store output inside the pbarprod, but in your account, and you want to use your new collection. At the end the jdl could look like:

Toggle Spoiler

```
[pgdb2.gla.ac.uk:3307] /panda/user/s/spataro/tdr11/ > cat test-ana.jdl
```

```

#
# JDL for macro
# by spataro@to.infn.it - June 09, 2011
#
# Usage: submit test-ana.jdl
#
Executable="/panda/user/s/spataro/bin/test-ana.sh";
Packages={
  "pbarprod@panda_extern::may11",
  "pbarprod@pandaroot::may11"
};
TTL = "20000";
InputDataCollection = "LF:/panda/user/s/spataro/tdr11/coll/coll_702_a";
InputFile={
  "LF:/panda/user/s/spataro/tdr11/run_ana_pipi.C"
};
OutputArchive={
  "ana_output.zip:finalroot_piplus*.root@disk=2",
  "log_files_ana.zip:stdout,stderr,resources,*.log@disk=1"
};
Split="directory";
Outputdir="/panda/user/s/spataro/tdr11/ana/#alien_counter#";
#ValidationCommand="/panda/user/s/spataro/bin/validatemacro-tdr11.sh";

```

And you have only to submit it:

```

[pgdb2.gla.ac.uk:3307] /panda/user/s/spataro/tdr11/ > submit test-ana.jdl
Jun 23 17:50:11 info Submitting job '/panda/user/s/spataro/bin/test-ana.sh' ...
Jun 23 17:50:11 info There is no price defined for this job in the jdl. Putting the default '1.0'
Jun 23 17:50:11 info *** calling PackMan with arguments list -silent -all
Jun 23 17:50:11 info Calling directly getListPackages (list -silent -all)
Jun 23 17:50:11 info Checking the input collection
LF:/panda/user/s/spataro/tdr11/coll/coll_702_a
Jun 23 17:50:11 info Input Box: {run_ana_pipi.C}

```

ATTENTION. You just submitted a JDL containing the tag 'OutputArchive'. The OutputFile and OutputArchive tags will be dropped in future versions of AliEn. For the moment the old tags work as usual, but please update your JDLs in the near future to utilize the 'Output' tag:

The syntax of the actual entries is still the same, but now you can just mixup files and archives, as e.g.:

```
Output = { "fileA,fileB,*.abc" , "myArchive:fileC,fileD,*.xyz" } ;
```

Thanks a lot!

```

Jun 23 17:50:15 info OK, all right!
Jun 23 17:50:15 info Command submitted (job 1171481)!!

```

Job ID is 1171481 - 0

You can see how your jobs are going with masterJob command:

```
[pgdb2.gla.ac.uk:3307] /panda/user/s/spataro/tdr11/ > masterJob 1171481
Jun 23 17:56:21 info Checking the masterjob of 1171481
Jun 23 17:56:21 info The job 1171481 is in status: SPLIT
It has the following subjobs:
  Subjobs in DONE: 10
  Subjobs in STARTED: 1
```

In total, there are 11 subjobs

Once all the jobs are DONE (you can check it also here), you have to do a collection with your root files, storing your ntuples, get the collection into your local pc, merging the histogram and show nice results!

I hope this guide is clear enough.
