

C3GAW - GenPipes Error Logs & Pipeline Relaunch

Rola Dali

2019-01-22

C3GAW - GenPipes Error Logs & Pipeline Relaunch

Rola Dali

2019-01-22

Overview:

In this practical, we will take a look at GenPipes error logs and GenPipes relaunch.

GenPipes Error Logs:

When launched, GenPipes will always create a `job_output` folder where it stores the logs and errors from all the jobs. If errors occur, we need to look into the `job_output` folder for the log of the step that failed to see what it last printed before it shut down. This usually helps us understand what potentially happened.

GenPipes will also send you an email once each job succeeds or fails.

When a job finishes successfully, it will create a file with the extension `.done`.

Can you get an overview of the status of each job?

GenPipes Job Status (click here)

Yes, you can. GenPipes creates a job list in the `job_output` folder that can help you do that.

you can type in:

```
#$MUGQIC_PIPELINES_HOME/utils/log_report.pl job_output/{TECHNOLOGY}_job_list_{DATE}T{TIME}
```

#example:

```
#$MUGQIC_PIPELINES_HOME/utils/log_report.pl job_output/RnaSeq_job_list_2018-06-26T12.54.27
```

This returns the status of each job. You can save it as a file and open it in Excel on your laptop.

Where are all the Job Logs and Errors stored?

GenPipes Logs (click here)

They are stored in the `job_output` folder under the appropriate folder for each step.

A note on the Exit Status:

Exit status (click here)

Computer software return “Exit status” codes to indicate the status of the run. **An Exit code of 0 means that the software ran without any errors.**

Different software Exit codes allow software developpers to understand what errors occured. You can take a look at the following page for more details.

The most common Exit codes are:

271 -> Usually means not enough RAM allocated

-11 -> Job was killed for exceeding the allocated walltime

GenPipes Relaunch:

If GenPipes fails, for any reason, you can recreate the commands and relaunch them.

When recreating the commands, GenPipes can detect jobs that have completed successfully and will not rerun them.

That being said, unless you understand why a job failed and fix it, relaunched jobs might fail with the same error.

Relaunch Mini Exercise:

GenPipes Relaunch ([click here](#))

To see how GenPipes skips successfully completed step, go into a folder where a GenPipes run has been completed and recreate the GenPipes commands. If you need a refresher on the command, refer to the GenPipesFiles practical.

Extras:

GenPipes Report:

For most pipelines, GenPipes creates an html report with most of the results in the pipeline. To create the report, you need to rerun the same command you ran to create the GenPipes commands, but add `--report` to it.

Try it!

GenPipes Clean:

GenPipes stores some temporary files that are useful to shorten potential reruns. To delete all these files, you can run the GenPipes command with `--clean`.

This will delete a lot of files that were marked by GenPipe developers as “removable”. If you are interested in temporary files, avoid the `-clean` command.

GenPipes parameters:

In order to keep track of all parameters used, GenPipes will create a final `*.config.trace.ini` file each time it is run. It is a good idea to keep a copy of that file in order to keep track of software versions used when publishing your paper.