A Few Words About Dask at LXPLUS

Massimiliano Galli

13/02/22

Context

▷ Since last year there has been an ongoing effort to give users the possibility to run Dask on LXPLUS

▷ As a result of this, the Batch team at CERN wrote a preliminary version of the dask-lxplus package, which extends Dask-Jobqueue's `HTCondorCluster` wiith the configuration necessary to run on LXPLUS `Gitlab repo`

▷ Within $H \to \gamma\gamma$ we are working on a Run3 framework based on Coffea

▷ Since part of the group usually works on LXPLUS, we are interested in testing the possibility to use Coffea+Dask there

CernCluster

```python
from dask_lxplus import CernCluster
import socket

n_port = 8786
wi = "/cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/batch-team/dask-lxplus/lxdask-cc7:
    latest"

cluster = CernCluster(
    cores=1,
    memory="20GB",
    disk="20GB",
    image_type="singularity",
    worker_image=wi,
    death_timeout="3600",
    scheduler_options={"port": n_port, "host": socket.gethostname()},
    job_extra={
        "log": "dask_job_output.log",
        "output": "dask_job_output.out",
        "error": "dask_job_output.err",
        "should_transfer_files": "Yes",
        "when_to_transfer_output": "ON_EXIT",
        "+JobFlavour": '"longlunch"'
    },
)
```

Test

Processor: we select diphoton pairs and perform the core operations (Chained Quantile Regression to correct Shower Shapes and Isolations, PhotonID MVA, diphoton preselection) code

Samples:

▷ DAS

▷ more than 87M events (116.7 GB)

▷ files not local to LXPlus site (located at PSI's tier3)

Results and Considerations
To be taken with a grain of salt

Good:

▷ dataset was processed on the `longlunch` queue around 3/4 times on different days, always taking **between 40 and 55 minutes** to complete

Not so good:

▷ a huge difference was found (as expected, probably) when instead of `longlunch` we used `workday`: the time required to get the workers up and running was much longer, leading to an overall time of about 4 hours

▷ even during the successful tests with the `longlunch` queue, the amount of jobs running at the same time was always about half the minimum required (i.e., the first number passed to `cluster.adapt`) - makes me think there is still room for improvement

▷ the port number constraint seem to allow only one cluster running at once - could be difficult to run small tests and debug stuff while having some other processes submitted