

1. An Example

Following is an example to run a MPI benchmark mdtest with Slurm on 2 TIERS™ system.

1.1 Upload application to AWS Login machine

Since login node in the CloudyCluster™ system has a public IP, you can get an application code from github on the Login node. Use github to get the application (mdtest in this example) on the Login node:

```
$ git clone https://github.com/MDTEST-LANL/mdtest
```

1.2 Build and copy binary on AWS Login node machines

After the source code is downloaded, login in to the terminal of Login node build the application binary and copy the binary to OrangeFS. For mdtest, you can use following command to compile and build:

```
/opt/mpich/bin/mpicc -Wall -Dlinux -D_FILE_OFFSET_BITS=64 -D_LARGEFILE_SOURCE=1
-D_USE_LARGEFILE64=1 -g -o mdtest mdtest.c -lm
```

You should have a shared directory /mnt/orangefs on login node and all the compute nodes. Now copy your application binary (mdtest in our example) to /mnt/orangefs/mdtest.

1.3 Prepare Slurm script file; mdtest.slurm

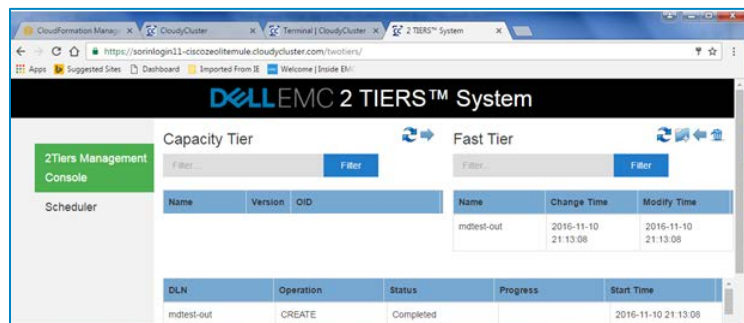
Following is an example of a Slurm script file to run mdtest, save it as a file in your local machine.

```
#!/bin/bash
#SBATCH --workdir=/mnt/orangefs/mdtest-slurm-md
#SBATCH -n 16
#SBATCH -N 8
#SBATCH --bb="size=1G"
#BB stage_in DLN=mdtest-out
#BB stage_out DLN=mdtest-out
/opt/mpich/bin/mpirun /mnt/orangefs/mdtest -n 400 -F -w 1024 -d /mnt/orangefs/mdtest-out
```

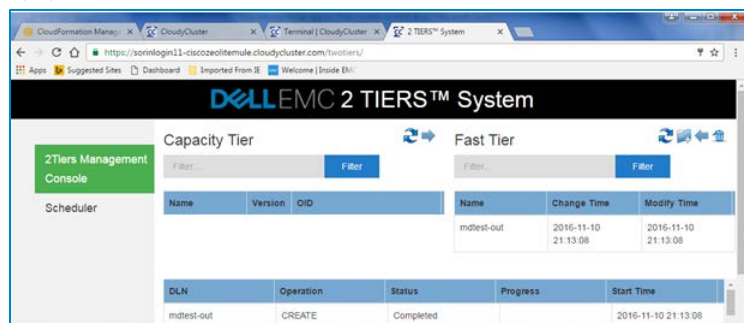
Before running the script create the Slurm working directory /mnt/orangefs/mdtest-slurm-md. The output files with the results of the run will be written to the Slurm working directory.

1.4 Create DLN with WebGUI

Before using WebGUI to run the Slurm job, the DLN should be created. Use step 4.2 to create a DLN named mdtest-out.



After this unload the DLN to CT (s3).

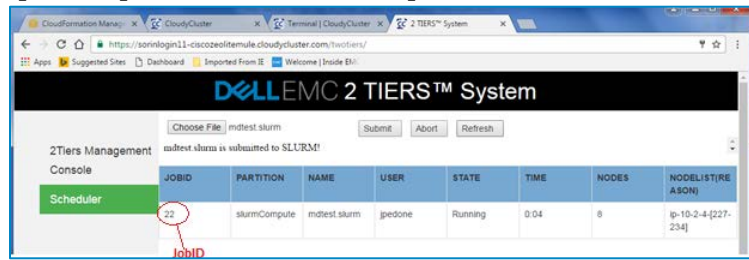


1.5 Run script from WebGUI

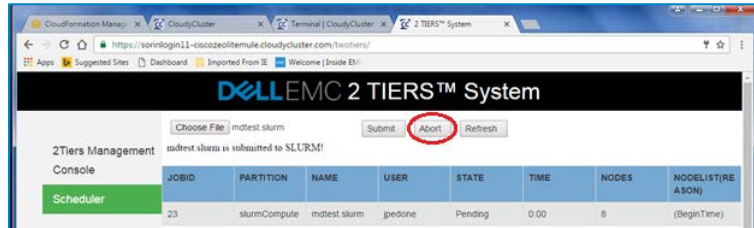
After creating DLN, now you can start to run the Slurm job. Click “Scheduler” item at the left side on 2 TIERS™ system web console, click “Choose File” button to select the script file saved at step 5.3, and then click the “Submit” button. The mdtest job will be shown in the job list. During the job run check availability of compute resources before starting the run using slurm *sinfo* command. Check that the resources are allocated correctly during the job running:

Schedule and Running an Application on a 2 TIERS™ System on AWS

```
$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
slurmCompute* up infinite 8 alloc ip-10-2-4-[227-234]
```

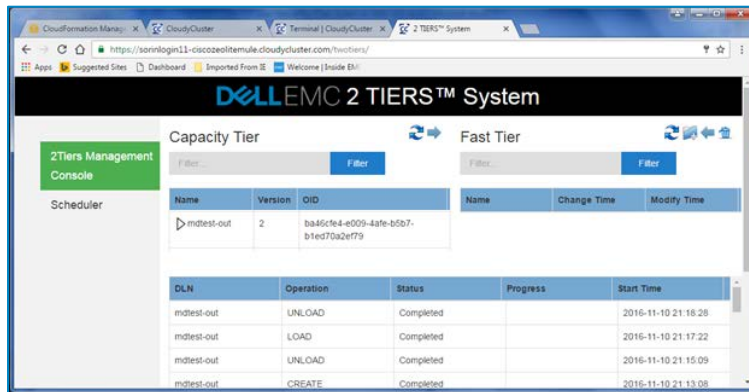


In case that the resources allocated do not match the script you can abort the job by highlighting the running job line and pressing Abort button.



1.6 Check the progress of Slurm job by refreshing the display

After the job finishes, you can login to login node terminal to check the result. In the WebUI you will now see a new version of the DLN in the CT marked as version 2.

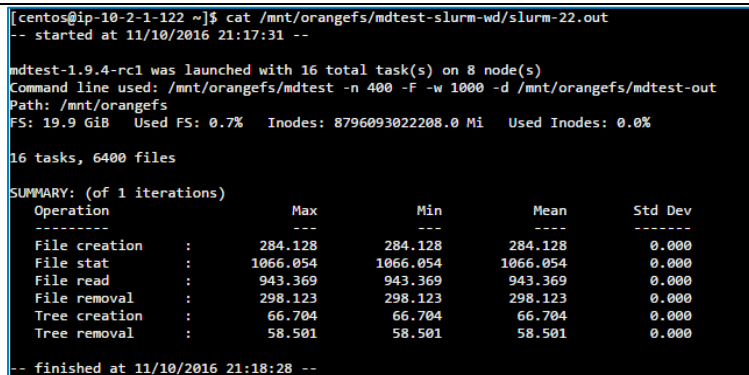


Note that all the DLN operations; CREATE, UNLOAD, LOAD, UNLOAD have completed and there is no DLN left in the FT.

1.7 Check out the result file

The results of the run will be saved in the file called: slurm-xx.out, where xx is the job ID of the scheduled job, 22 in the example of 5.5. You can check the output file in the Slurm working directory:

```
$ ls /mnt/orangefs/mdtest-slurm-wd/slurm-* -la
-rw-r--r--. 1 jpedone jpedone 1040 Nov 10 21:18 /mnt/orangefs/mdtest-slurm-wd/slurm-22.out
```



When you finished the run and the experiment do not forget to pause all the instances to reduce cost incurred.