Instructor guide

1. The runbatch script is used to submit jobs but it can be repurposed for other jobs.
2. Watch out for the number of MPI ranks assigned per compute node. It is a common mistake to assign all MPI ranks to a single node.
3. Take the time to go over the VMD tutorial : https://www.ks.uiuc.edu/Training/Tutorials/vmd/tutorial-html/
4. Students should be asked to complete the tutorial before the module.
5. Since this is a 25 minute session make sure that students ask for a small number of nodes for a short time. You may consider requesting a reservation for the activity, you will need to contact the sys-admins at TACC.
6. You can make the tarball available to students before the activity so students can look at the files and load them in VMD.
7. Explain to students the role that a distributed file system plays in analysing data and particularly how it alleviates I/O for large-scale analysis.

Common pitfalls

1. Using the 'module vmd' installed in Stampede2. This version of VMD is not MPI-enabled and won't work with the exercises present in the module. VMD must be compiled with MPI support in Stampede2, binary access is provided in the directory : /work/06295/fabiogon/stampede2/VMD_MPI/vmd.
2. Using mpirun instead of ibrun.
3. A typical error is to run more than one MPI-rank per node. Typical analysis workloads are limited by I/O, therefore for best performance it is best to use 1 MPI rank per node.
4. VMD incorporates several analysis subroutines that are parallelized for multi-core architectures. Thread control is not regulated by OpenMP environment variables. VMD uses environmental variables to control the number of threads.
5. Both the structure file (parmtop) and trajectory file (dcd) should load on the same molecule VMD otherwise will display a blank screen.
6. A distributed system (LUSTRE) is required for the analysis presented in the module to scale properly with the number of nodes. The best place to put the files for analysis is in $SCRATCH.