Propagating environment settings to a PBS job

10/27/2021 – Beginning today, users are able to submit jobs to run on the Cheyenne and Casper systems from either of those systems with the introduction of the new peer-scheduling capability. Following is a description of a related change that was made to facilitate the implementation of peer scheduling and what, if anything, users need to do.

Page contents

Change in PBS job behavior

Previously on Cheyenne, the environment variables and modules loaded in the shell at submission time were propagated to the job environment. On Casper, no environment settings are propagated to the job. The default propagation behavior on Cheyenne has been changed to match that of Casper, so environment settings are not propagated by default for any PBS jobs.

What to do

While the change has no impact on many workflows, some users may wish to set environment variables in their login environment that can be temporarily used for multiple batch jobs without modifying the job script. This practice can be particularly useful during iterative development and debugging work.

PBS has two approaches to propagation:

1. specific variables can be forwarded to the job upon request
   or
2. the entire environment can be forwarded to the job.

In general, the first approach is preferred because the second may have unintended consequences.

These settings are controlled by **qsub** arguments that may be used at the command line or as directives within job scripts. Here are examples of both approaches:

```
# Selectively forward runtime variables to the job (lower-case v)
qsub -v DEBUG=true,CASE_NAME job.pbs
```

When you use the selective option (lower-case v), you can either specify only the variable name to propagate the current value (as in CASE_NAME in the example), or you can explicitly set it to a given value at submission time (as in DEBUG).

```
# Forward the entire environment to the job (upper-case V)
qsub -V job.pbs
```

Warning: *Do not use full propagation when peer-scheduling jobs. Doing so will cause libraries and binaries to be inherited via variables like PATH and LD_LIBRARY_PATH. These inherited settings WILL cause applications to break, and may render the job completely unusable.*