

Exploring files in Lustre

Description

Simple changes to how we find and list files on Lustre filesystems can have a massive impact on performance. This largely relates to what information is stored on the metadata servers vs what is stored where the data is.

For interactive tasks involving directories containing tens or hundreds of files, this is not usually an issue. For directories with thousands of files, or for scripted activities that loop over many files, it is important to follow these guidelines.

- Avoid outputting colour with ls. Use 'ls --color=never'. This is the default when standard output is connected to a terminal, but this might have been overridden by the LS_COLORS environment variable.
- Avoid obtaining more information than you need. In most scripted activities the modification times and user permissions are not required. 'ls -l' will likely provide much more information than you need, and involve querying the object storage targets rather than just the metadata server. Avoid using 'ls -l'.
- Avoid sorting ls output if not required.
- With the 'find' command, avoid using '-type' or any of the timestamp, permission or ownership based tests.

To loop over files, here are some options.

- Loop over the output of 'ls -1 --color=never'. This will result in each line of output containing one file. 'ls -1 -r --color=never' to do this recursively. Note that in this form it will output all files and directories, so you may need to pipe through grep to get only the files you are interested, for example those ending in '.out'.
- Use find rather than ls. E.g. 'find . -name "*.out"'. The example is in double quotes so the asterisk is not evaluated by bash and is instead passed to find.

Related articles

- [Deleting large numbers of files on scratch and group](#)
- [Exploring files in Lustre](#)
- [Use of tmp by Running Jobs](#)
- [Python 2 End of Life](#)
- [Jobscript Reproducibility](#)
- [Setting File Permissions](#)
- [Shell Initialisation Scripts](#)
- [Where to install software on supercomputers](#)
- [Defining wallclock time in the jobscript](#)
- [Installing Python modules on supercomputers](#)