# Introduction to the HPC Cluster Environment

A High Performance Computing (HPC) cluster is, at its core, a mechanism for providing shared access to a collection of computing, storage and network resources. The term “High Throughput Computing” (HTC) is also often used to describe this type of cluster environment when the architecture is oriented toward less tightly coupled workloads.

There are numerous ways to accomplish sharing of resources, for instance, any time you use a cloud server or a service hosted in The Cloud(tm) you are using a shared resource with a scheduler that assigns customer tasks to a specific real resource. There are several differences,however, that are specific to HPC/HTC which, in effect, make it HPC/HTC (even if configured to run on top of a Cloud environment).

1. A common Operating System image, providing a consistent kernel, driver and software environment on all resources.
2. A high speed interconnect dedicated to internode trusted communication and for accessing storage.
3. A scheduler, accepting requests and managing the resources, starting and stopping processes, etc.
4. A common storage namespace on all nodes, no confusion about which node has what data or storage mounts.
5. A centrally maintained software stack, with installs of a wide range of software ready and easy to access and use, and consistent across all the cluster nodes.
6. A set of login nodes, providing an interface for users to use command line tools, submit jobs interactively or in batch and access/manage data.
7. A web interface providing a graphical interface to GUI tools and software like Jupyter Notebooks, RStudio, etc.

This is a very simple, but powerful, paradigm for accessing and using computational resources requiring nothing more than a simple script or command passed to the scheduler to be executed on the next available resource. Additionally, it’s possible to do simple and non-intensive work directly on the login nodes without the need for involving the scheduling layer until ready to scale a task up or out across the resources. At the other end of the spectrum it’s possible to use workflow tools to architect complex pipelines and processes, all in the same shared, multi-user environment. .