Instructions to Use the Test Rocky Linux 9 Cluster

As part of the upcoming upgrade to Rocky Linux 9 on our main compute cluster, we have set up a small test cluster to facilitate a smooth transition and enable checking compatibility of your environments and applications to adapt your workflow in time before we make the switch. We strongly encourage to start testing as soon as possible.

Overview of the Test Cluster

- **Login Node**: `login04.cluster.uni-hannover.de`
- **Compute Nodes**: 12 nodes accessible via the SLURM partition `taurus_rl`
- **GPU Computing**: 1 node with GPUs via the SLURM partition `gpu_rl`
- **Storage Filesystems**: All [storage filesystems](#) available on the main cluster are also accessible on the test cluster, allowing you to access your files and directories in the same way

Slurm Partition Restrictions

To ensure fair access to the ressources, the following restrictions are in place:

- `taurus_rl` Partition:
    - **Jobs per Account**: 2 running job and 1 additional job in the queue
    - **Max Walltime per Job**: 12 hours
    - **Max Nodes**: 2 nodes per job
    - **Max CPUs**: 8 CPUs per job

- `gpu_rl` Partition:
    - **Jobs per Account**: 1 running job and 1 additional job in the queue
    - **Max Walltime**: 12 hours
    - **Max CPUs**: 12 CPUs per job
    - **Max GPU**: 1 GPU slot per job

Please plan your job submissions accordingly to adhere to these limits and, if possible, stay low-profile to give everybody a chance to test.

Accessing the Login Node

You can access the head node `login04.cluster.uni-hannover.de` using the same credentials as normal.

There are three methods to connect:

1. **SSH (Secure Shell)**
    - Open a terminal on your local machine
    - Connect to the head node using the following command: `ssh [your_username]@login04.cluster.uni-hannover.de`

- replace [your_username] with your actual cluster username

2. **X2Go**
   - Ensure X2Go is installed properly on your local machine (see our hints in [the cluster documentation](#))
   - Set up a new session in X2Go using the following settings:
     - **Host**: login04.cluster.uni-hannover.de
     - **Login**: Your username
     - **Session Type**: XFCE
     - Connect using your normal credentials

3. **OOD (Open OnDemand) Web Platform**
   - Open your web browser and navigate to the OOD portal: https://login04.cluster.uni-hannover.de
   - Log in with your cluster credentials
   - Use the web-based interface to access the head node, submit jobs, manage files and start configured applications

Software Modules

The list of software provided via modules (Lmod) has been updated. Older versions of some software have been removed, and newer versions have been installed.

- **Loading Modules**
  - For information on how to find available software modules and how to load them, please refer to the [dedicated page](#)
  - If you do not find a specific software module that you need, please contact us at cluster-help@luis.uni-hannover.de with an installation request. In accordance with our own limited ressources, and to keep the module system from growing chaotic, we will usually answer to requests for software that **a**) at least one institute needs to use, and **b**) whose license permits usage for all cluster users. If you are individually using a software, please install it for yourself. In-between these two cases, we will decide on a per-request basis whether we can invest the ressources to install a particular package.

Environment Variable Change

Please note the following important change to environment variables:

- **Old Variable**: $ARCH
- **New Variable**: $LUIS_CPU_ARCH

The $LUIS_CPU_ARCH variable is used to identify the CPU type of compute and head nodes, replacing the old $ARCH variable to avoid conflicts. The new possible values are:

- sse (replaces nehalem)
- avx (replaces sandybridge)
- avx2 (replaces haswell)
- avx512 (replaces skylake)

Ensure that scripts or environment setups using the $ARCH variable are updated to use

$LUIS_CPU_ARCH and the corresponding values.

Submitting Jobs to Compute Nodes

You can submit jobs to the compute nodes using the SLURM workload manager. There are two partitions available:

- **Standard compute nodes in migration partition `taurus_rl`**
    - Use this partition to test your jobs on the standard compute nodes
    - Example of a simple job submission script:

    my_job_script.sh

    ```
    #!/bin/bash -l
    #SBATCH --job-name=test_job
    #SBATCH --partition=taurus_rl
    #SBATCH --ntasks=8
    #SBATCH --nodes=2
    #SBATCH --time=01:00:00


    srun ./my_application
    ```

    - Submit the job with the following command: `sbatch my_job_script.sh`

- **GPU compute server in migration partition `gpu_rl`**
    - Use this partition to test jobs that require GPU resources
    - Example of a job submission script for a GPU job:

    my_gpu_job_script.sh

    ```
    #!/bin/bash
    #SBATCH --job-name=gpu_test
    #SBATCH --partition=gpu_rl
    #SBATCH --ntasks=8
    #SBATCH --gres=gpu:1
    #SBATCH --time=02:00:00


    srun ./my_gpu_application
    ```

    - Submit the job with the following command: `sbatch my_gpu_job_script.sh`

Best Practices for Testing

- **Environment Setup**: Recreate your environment on the test cluster using the same modules, virtual environments, or container images you use on the main cluster
- **Job Testing**: Submit a variety of interactive jobs using `salloc`, including those with different resource requirements, to ensure compatibility of your environment with Rocky Linux 9. After

successfully testing your setup interactively, proceed to the next step and submit batch jobs
- **Feedback**: Report any issues or discrepancies you encounter to cluster-help@luis.uni-hannover.de

From:
https://docs.cluster.uni-hannover.de/ - **Cluster Docs**

Permanent link:
https://docs.cluster.uni-hannover.de/doku.php/guide/test_rl_cluster

Last update: **2024/09/10 18:14**