11falsenonelisttrue

Overview

This documentation describes compute group and queue designations and their uses.

For users (established and new) that only use the general queue, this information isn't needed to use the Compute Platform.

This documentation is for those that have a condo or subscription tier.

Designations of Groups and Queues

The following is a list of the group name examples and queue names associated with the various subscriptions.

Subscription	Group Name (-G)	Queue Name (-q)	Availability
General	`${compute-group}`	`general`	Available
Condo	`${compute-group}`	`${condo-name}`	Available
Tier 1 Subscription	`${compute-subscription-group}`	`subscription`	Available
Tier 2 Subscription	`${compute-subscription-group}`	`subscription`	Available
Tier 3 Subscription	`${compute-subscription-group}`	`subscription`	Available

`${compute-group}` and `${compute-subscription-group}` should be replaced with the designation associated with the lab or research group the subscription is associated with, commonly this will be a PI washukey.

Example: `${compute-group}` becomes `compute-ris` for the RIS group

More information on the different subscriptions available can be found here: https://ris.wustl.edu/services/compute/resources/

General Queue

The general queue is the base subscription. All active compute users have access to this queue.

There are no guaranteed resources associated with a general subscription.

The following example shows how the group name and queue name will look for the general queue.

This queue is used as the example in the rest of the compute documentation examples elsewhere.

You can find the general queue policies here.

`${compute-group}` should be replaced with the name of your compute group

This will be provided at time of compute activation for your group.

Condo Queue

A condo queue is a queue associated with a purchased condo.

More information about condos can be found here: https://ris.wustl.edu/services/compute/compute-condo/

The resources available in this queue are dependent on physical resources purchased as part of the condo.

Below is an example of how to run a job command in a condo queue.

`${condo-name}` should be replaced with the name of your condo queue

`${compute-group}` should be replaced with the name of your compute group

These will be provided at time of activation for your group.

Subscription Tier Queue

A subscription tier is associated with a number of resources that are guaranteed for use based on the tier.

There are currently three subscription tiers.

Tier 1 Resources

25 vCPUs

1 GPU

Tier 2 Resources

50 vCPUs

2 GPU

Tier 3 Resources

100 vCPUs

3 GPU

If you go over on the number of guaranteed vCPUs for a job submitted in this queue type, your job will not be guaranteed to run.

If you go over on the number of guaranteed GPUs for a job submitted in this queue type, the job will stay in pending and never run.

The `-sla` option is required for jobs submitted in this queue type.

You can check out what sla a job used via the following command.

This will list out a lot of information about the job, but the `Service Class` entry will list the sla used.

- , User , Project , Application , Job Group < /elyn/default>, Service Class , User Group , Status , Queue , J ob Priority <50>, Command , Share group charged , Esub]]>

How to run a job using the subscription tier.

Replace `${compute-subscription-group}` with the intended compute subscription group.

Replace `${sla-name}` with sla name associated with your subscription.

These will be provided at time of activation for your group.

Additional Information

The group name and queue must match or the job will not be submitted. An example of what this mismatch and the error it produces looks like is below.

In the example, the condo queue was designated but the group name used was one associated with a tier subscription.

This will affect the potential pipelines of users who had access to the Compute Platform before this change and are moving to a different subscription than the base, which is the general queue.