UCCS INCLINE Documentation

Brandon Runnels

May 18, 2021

CONTENTS:

CHAPTER

ONE

GETTING STARTED ON INCLINE

1.1 Create an account

Todo: Document account creation

Section must cover

- link to account creation page
- Instructions for UCCS users
- Instructions for non-UCCS users

1.2 Log in to INCLINE

Todo: Login information

Must cover

- Login node names
- Instructions for linux and mac users
- Instructions for windows users

1.3 Requesting an allocation

Todo: Allocation request information

1.4 HPC Etiquette

CHAPTER

TWO

PREPARING YOUR CODE

2.1 Compiling your code using environment modules

Todo: Compile code with env modules

If you need new software, please use the software request page

2.2 Using Docker and Singularity

Todo: Docker and Singularity

2.3 Commercial Software

The use of commercial software (such as matlab) is *not* supported on INCLINE. This does not mean that you cannot run your commercial code; however, we cannot provide the type of support specific to commercial codes (such as setting up license servers).

CHAPTER

THREE

RUNNING JOBS ON INCLINE

CHAPTER

FOUR

DATA TRANSFER ON INCLINE

4.1 Transferring files with SCP using Data Transfer Nodes

4.2 GLOBUS

CHAPTER

FIVE

SYSTEM OVERVIEW

5.1 Computational Nodes

INCLINE contains 30 dedicated nodes for computation. The majority are `COMPUTE` nodes, which are CPU only and 2GB/core, but INCLINE also contains two `HIGH MEMORY` nodes (16GB/core) and two `GPU` nodes, each with two NVIDIA A100 GPUs. Specifics are outlined in the table below.

Type	QTY	CPU Type	Cores/Threads	Memory	GPU
`COMPUTE`	26	2x AMD EPYC 7662	128C/256T	256GB	None
`HIGH MEMORY`	2	2x AMD EPYC 7662	128C/256T	2048GB	None
`GPU`	2	2x AMD 7452	64C/128T	1024GB	2x NVIDIA A100

5.2 Other Nodes

5.3 High Speed File System

5.4 Normal File System

5.5 High Speed Interconnect

5.6 Networking

CHAPTER

SIX

INCLINE RESOURCES

6.1 Allocations

Allocation Type	Description
`REAS Account`	Allocation for activities related to research projects described here. Granted on a quarterly basis based on competitive proposal process.
`CRSE Account`	Allocation for course-related activities. Allocations granted on a quarterly basis based on competitive proposal.
`MISC Account`	Limited "seed" accounts given on a first-come-first-served basis. Limits predetermined by committee based on demand and resources.
`EXTR Account`	Accounts for all off-campus and RMACC users. Granted by competitive proposal. Limits determined by committee.

6.2 Queues

Queue Name	**Description**	
`compute`	Standard workhorse queue for jobs that use the compute nodes.	
	12 Hours	10 Jobs
`compute-dev`	High-priority queue with tight limits on job length and queue number. Use for testing code but not for production runs.	
	1 Hour	1 Job
`compute-long`	Low-priority queue with unlimited job length. Only one job allowed in the queue at a time	
	Unlimited	1 Job
`gpu`	Queue for running jobs on the GPU nodes	
`hm`	Queue for running jobs on the high memory nodes	
`hetero`	Special queue for jobs that use any combination of compute, GPU, and high membory nodes.	

6.3 Calculating Computing Units

INCLINE resources are described in terms of **CPU Hours** (CPUH). One standard CPUH is defined as one hour of allocated use by a `compute` node. **Important**: the CPUH will be billed to your account regardless of whether the node is fully utilized.

Time allocated on specialty nodes are denoted differently. One hour of allocated time by a `gpu` node is a GPU-CPUH; an hour of allocated time by a `high memory` node is a HM-CPUH. Resources for CPUH, GPU-CPUH, and HM-CPUH are billed independently, and are non-transferrable.

Example 1

Alice needs to run 10 large CFD simulations. She estimates that, running with approximately 2000 MPI processes (no special memory requirements), each simulation will complete in four hours.

To run a concurrent job with 2000 MPI processes will require 8 nodes. (8 x 256 threads = 2048). Each job will then require (4 hours) x (8 nodes) = 32CPUH. Therefore, Alice should request at least 320CPUH.

Example 2

Bob is working on a fictitious machine learning project that will require extensive data processing. He estimates that training and testing his model will take 2 days on a single A100 GPU. (There is no way to split the job across multiple GPUs; it must be done serially.) Afterwards, the data analysis will take about 30 hours on a high memory node.

Even though Bob will only be using one of the two GPUs on a `gpu` node, he must request time on the entire node. Because he also needs to use a `high memory` node, he must specify that allocation independently from the `gpu` node. Therefore, he should request **48 CPU-CPUH** and **30 HM-CPUH**.

CHAPTER

SEVEN

TUTORIALS