Cluster Workshop

Introduction

"If you were plowing a field, which would you rather use: Two strong oxen or 1024 chickens?"

–Seymour Cray

Introduction

- What is a cluster?

Introduction

- What is a cluster?
  - ⇒ A collection of machines that work together

Introduction

- What is a cluster?
  - ⇒ A collection of machines that work together
- Why use a cluster?

Introduction

- What is a cluster?
  - ⇒ A collection of machines that work together
- Why use a cluster?
  - ⇒ Chickens are cheaper than oxen

Introduction

- What is a cluster?
  - ⇒ A collection of machines that work together
- Why use a cluster?
  - ⇒ Chickens are cheaper than oxen
  - ⇒ (and easier to feed)

Introduction

- What is a cluster?
  - ⇒ A collection of machines that work together
- Why use a cluster?
  - ⇒ Chickens are cheaper than oxen
  - ⇒ (and easier to feed)
  - ⇒ ...but try making 1024 chickens move in the same direction

Cluster Terminology

- **Core (CPU)**

Cluster Terminology

- **Core (CPU)**
- **Hyperthreaded CPU**

Cluster Terminology

- **Core (CPU)**
- **Hyperthreaded CPU**
- **Socket**

Cluster Terminology

- Core (CPU)
- Hyperthreaded CPU
- Socket
- Core (memory)

Cluster Terminology

- Core (CPU)
- Hyperthreaded CPU
- Socket
- Core (memory)
- Swap

Cluster Terminology

- Core (CPU)
- Hyperthreaded CPU
- Socket
- Core (memory)
- Swap
- Thrash

Cluster Terminology

- **Node** An individual computer in the cluster

Cluster Terminology

- **Node** An individual computer in the cluster
- **Head Node** The main node. Coordinates scheduling jobs among the nodes.

Cluster Terminology

- **Node** An individual computer in the cluster
- **Head Node** The main node. Coordinates scheduling jobs among the nodes.
- **Login Node** The node users log in to and use to submit jobs. May be the same as the head node.

Cluster Terminology

- **Node** An individual computer in the cluster
- **Head Node** The main node. Coordinates scheduling jobs among the nodes.
- **Login Node** The node users log in to and use to submit jobs. May be the same as the head node.
- **Interconnect** The network or networks that connects the nodes together

Types of Jobs

- **interactive** A single-part job to be run immediately.

Types of Jobs

- **interactive** A single-part job to be run immediately.
- **batch (serial)** A single-part job to run in the background

Types of Jobs

- **interactive** A single-part job to be run immediately.
- **batch (serial)** A single-part job to run in the background
- **parallel** A job that has been split into multiple parts to run on more than one processor.

Types of Jobs

Parallel cluster jobs can be categorized according to the amount of communication required between parts of the job:

- **fine-grained parallel** Substantial communication required.

Types of Jobs

Parallel cluster jobs can be categorized according to the amount of communication required between parts of the job:

- **fine-grained parallel** Substantial communication required.
- **course-grained parallel** Occasional communication required.

Types of Jobs

Parallel cluster jobs can be categorized according to the amount of communication required between parts of the job:

- **fine-grained parallel** Substantial communication required.
- **course-grained parallel** Occasional communication required.
- **embarrassingly parallel** Almost no communication required.

Types of Jobs

Even serial jobs that cannot be split up to run in parallel can still benefit from a cluster environment, for example, by running with different sets of input files or parameters simultaneously.

Cluster Storage

- Home directory

Cluster Storage

- Home directory
- Shared cluster space

Cluster Storage

- Home directory
- Shared cluster space
- Local storage

Storage: Home directory

- Network mounted

Storage: Home directory

- Network mounted
- Geared towards wide availability, not high performance

Storage: Home directory

- Network mounted
- Geared towards wide availability, not high performance
- Good place for your code, final results

Storage: shared

- Network shared, but only within the cluster

Storage: shared

- Network shared, but only within the cluster
- Medium availability, medium performance

Storage: shared

- Network shared, but only within the cluster
- Medium availability, medium performance
- Good place for dataset that all nodes need to access

Storage: shared

- Network shared, but only within the cluster
- Medium availability, medium performance
- Good place for dataset that all nodes need to access
- Performance can suffer with many simultaneous accesses

Storage: local

- Not network shared, only available to a single node

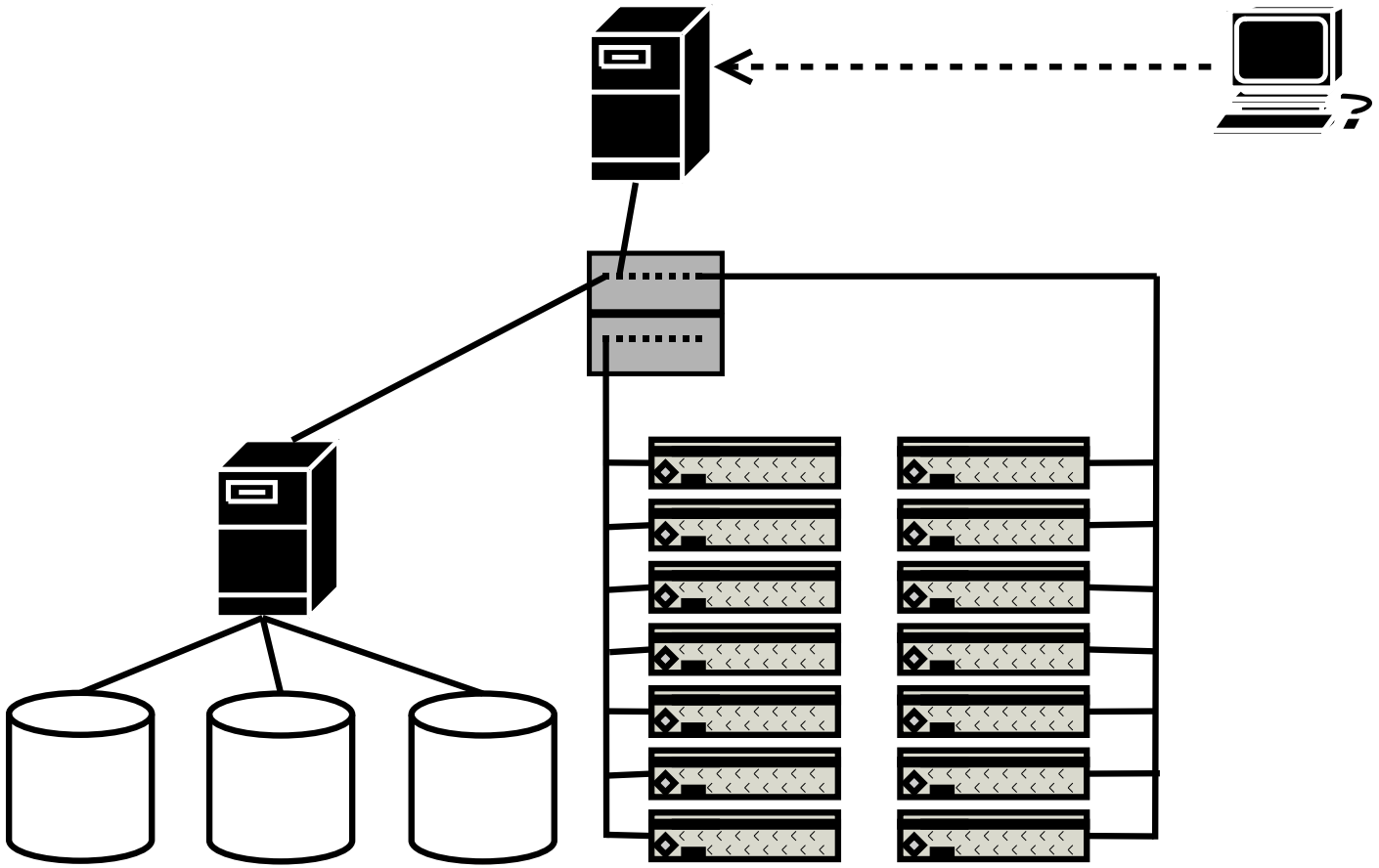
Storage: local

- Not network shared, only available to a single node
- Highest performance, but least convenient

Storage: local

- Not network shared, only available to a single node
- Highest performance, but least convenient
- Good place for working set

Anatomy of a Cluster



Sun Grid Engine

- What is a batch queue?

Sun Grid Engine

- What is a batch queue?
  - ⇒ Manage cluster resources

Sun Grid Engine

- What is a batch queue?
  - ⇒ Manage cluster resources
- Why use the batch queue?

Sun Grid Engine

- What is a batch queue?
  - ⇒ Manage cluster resources
- Why use the batch queue?
  - ⇒ Share cluster resources

Sun Grid Engine

- What is a batch queue?
  - ⇒ Manage cluster resources
- Why use the batch queue?
  - ⇒ Share cluster resources
  - ⇒ You don’t need to worry about when/where your jobs run

Sun Grid Engine

- What is a batch queue?
  - ⇒ Manage cluster resources
- Why use the batch queue?
  - ⇒ Share cluster resources
  - ⇒ You don’t need to worry about when/where your jobs run
  - ⇒ Submit a whole bunch of jobs and go home!

The Cluster Café

You can think of the cluster as a restaurant:

- **Jobs** are parties coming to eat

The Cluster Café

You can think of the cluster as a restaurant:

- **Jobs** are parties coming to eat
- **Tables** are the nodes of the cluster

The Cluster Café

You can think of the cluster as a restaurant:

- **Jobs** are parties coming to eat
- **Tables** are the nodes of the cluster
- Usually the scheduler will try to place each job at its own table, but if it’s a busy day, you might have to share a table with someone you don’t know.

Slots

A **slot** is a placeholder where a job or part of a job can run. In the Cluster Café, it is a chair at a table.

- A resource allocated to your job.

Slots

A **slot** is a placeholder where a job or part of a job can run. In the Cluster Café, it is a chair at a table.

- A resource allocated to your job.
- We define one slot per CPU core

Slots

A **slot** is a placeholder where a job or part of a job can run. In the Cluster Café, it is a chair at a table.

- A resource allocated to your job.
- We define one slot per CPU core
- Request number of slots when you submit job

Slots

A **slot** is a placeholder where a job or part of a job can run. In the Cluster Café, it is a chair at a table.

- A resource allocated to your job.
- We define one slot per CPU core
- Request number of slots when you submit job
- Allocating a slot is only advisory: the scheduler has reserved the requested number of slots.

Resources

It is possible to request other resources when you submit your job

Resources

It is possible to request other resources when you submit your job

$\Rightarrow$“I’d like a table by the window”

Resources

It is possible to request other resources when you submit your job

$\Rightarrow$“I’d like a table by the window”

Or even to request a specific node

Resources

It is possible to request other resources when you submit your job

$\Rightarrow$“I’d like a table by the window”

Or even to request a specific node

$\Rightarrow$“I want to sit at table 2. I’ll wait.”

Resources

It is possible to request other resources when you submit your job

$\Rightarrow$"I'd like a table by the window"

Or even to request a specific node

$\Rightarrow$"I want to sit at table 2. I'll wait."

The scheduler will wait to run your job until it can fulfill your requirements.

Parallel Environment

A **Parallel Environment** sets up the resources required to run a multi-node job.

- Need to use a PE when you want more than one slot

Parallel Environment

A **Parallel Environment** sets up the resources required to run a multi-node job.

- Need to use a PE when you want more than one slot
- Request desired PE when you submit your job

Parallel Environment

A **Parallel Environment** sets up the resources required to run a multi-node job.

- Need to use a PE when you want more than one slot
- Request desired PE when you submit your job

Tells the scheduler how you'd like the table set

Parallel Environment

- mpi Sets up environment for distributed jobs using MPI

Parallel Environment

- mpi Sets up environment for distributed jobs using MPI
- serial/threaded Makes sure all slots are on the same node, does not set up any inter-node communication environment.

Task Arrays

- Run the same job multiple times

Task Arrays

- Run the same job multiple times
- submit/manage as a single job

Task Arrays

- Run the same job multiple times
- submit/manage as a single job
- Ideal for running the same program repeatedly with different input files or parameters

Resources

Resources represent the hardware and software configuration of a node. They can represent things like memory, CPU architecture, or software licenses.
They come in two basic flavors:

- **non-consumable** - they don’t go away when requested

Resources

Resources represent the hardware and software configuration of a node. They can represent things like memory, CPU architecture, or software licenses.
They come in two basic flavors:

- **non-consumable** - they don’t go away when requested
- **consumable** - when requested, the resource is marked as used until the job requesting them is finished.

SGE Commands

- qsub/qlogin/qsh: submit jobs

SGE Commands

- qsub/qlogin/qsh: submit jobs
- qstat: get job status

SGE Commands

- qsub/qlogin/qsh: submit jobs
- qstat: get job status
- qdel: remove a job

SGE Commands

- qsub/qlogin/qsh: submit jobs
- qstat: get job status
- qdel: remove a job
- qlogin: interactive login

SGE Commands

- qsub/qlogin/qsh: submit jobs
- qstat: get job status
- qdel: remove a job
- qlogin: interactive login
- qalter: change a job after submission

SGE Commands

- qsub/qlogin/qsh: submit jobs
- qstat: get job status
- qdel: remove a job
- qlogin: interactive login
- qalter: change a job after submission
- qacct: view accounting information

Job submission

- **qsub** - submit your job in the background

Job submission

- **qsub** - submit your job in the background
- **qlogin** - interactive login

Job submission

- **qsub** - submit your job in the background
- **qlogin** - interactive login
- **qsh** - interactive login with X

Things to do

- Use the scheduler!

Things to do

- Use the scheduler!
- Checkpoint your job

Things to do

- Use the scheduler!
- Checkpoint your job
- Make use of local storage on the nodes for intermediate results

Things NOT to do

- Run jobs on the head node

Things NOT to do

- Run jobs on the head node
- Many simultaneous writes to network filesystem

Things NOT to do

- Run jobs on the head node
- Many simultaneous writes to network filesystem
- Go around scheduler and run directly on the nodes