TEXAS ADVANCED COMPUTING CENTER

WWW.TACC.UTEXAS.EDU

TEXAS
The University of Texas at Austin

Introduction to HPC

Interacting with High Performance Computing Systems

PRESENTED BY:

Virginia Trueheart, MSIS

Texas Advanced Computing Center

vtrueheart@tacc.utexas.edu

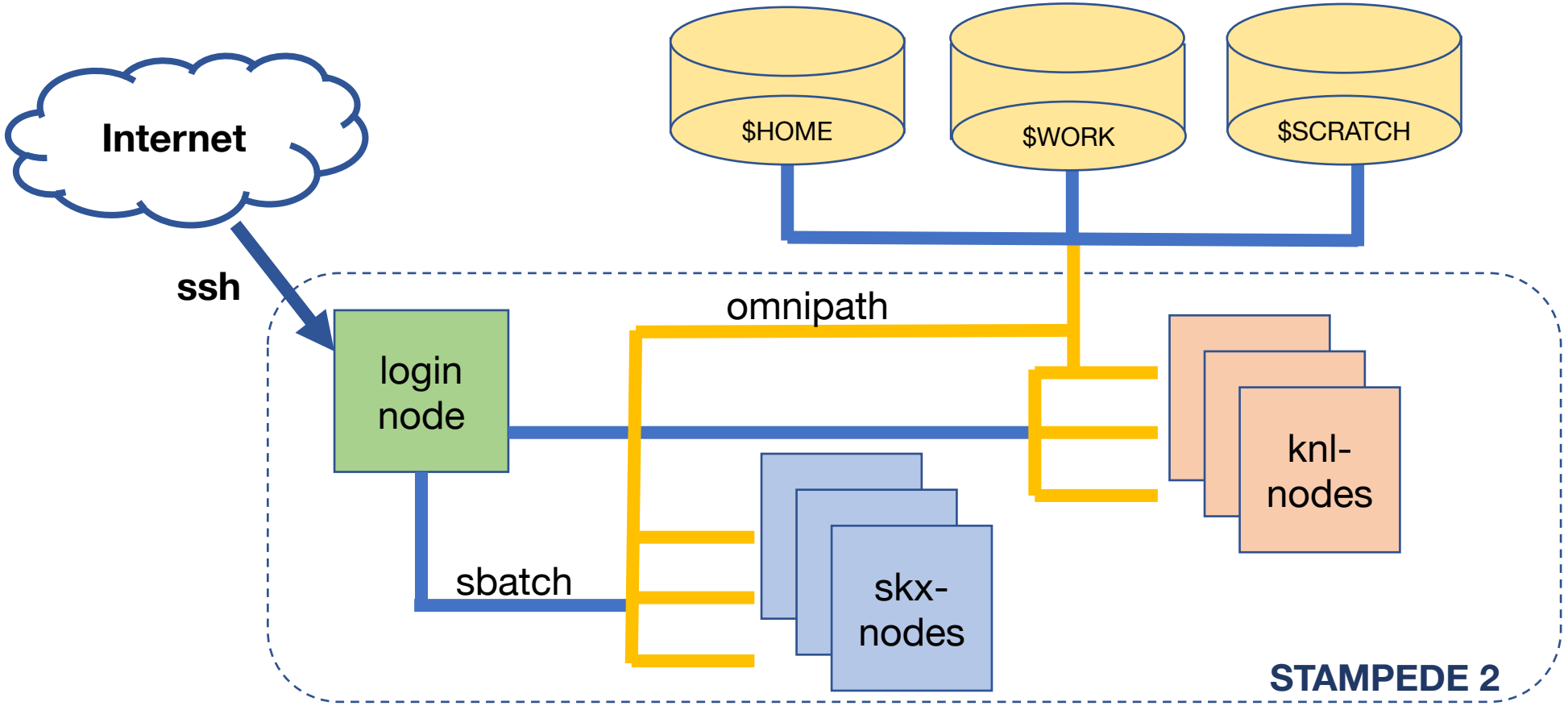
An Overview of HPC: What is it?

- High Performance Computing
  - Parallel processing for advanced computation
  - A “Supercomputer”, “Large Scale System”, or “Cluster”
- The same parts as your laptop
  - Processors, coprocessors, memory, operating system, etc.
  - Specialized for scale & efficiency
- Scale and Speed
  - Thousands of nodes
  - High bandwidth, low latency network for large scale I/O

An Overview of HPC: Stampede2

- Peak performance: 18 PF, rank 12 in Top 500 (2017)
- 4,200 68-core Knights Landing (KNL) nodes
- 1,736 48-core Skylake (SKX) nodes
- 368,928 cores and 736,512GB memory in total
- Interconnect: Intel's Omni-Path Fabric Network
- Three Lustre Filesystems
- Funded by NSF through grant #ACI-1134872

An Overview of HPC: Architecture

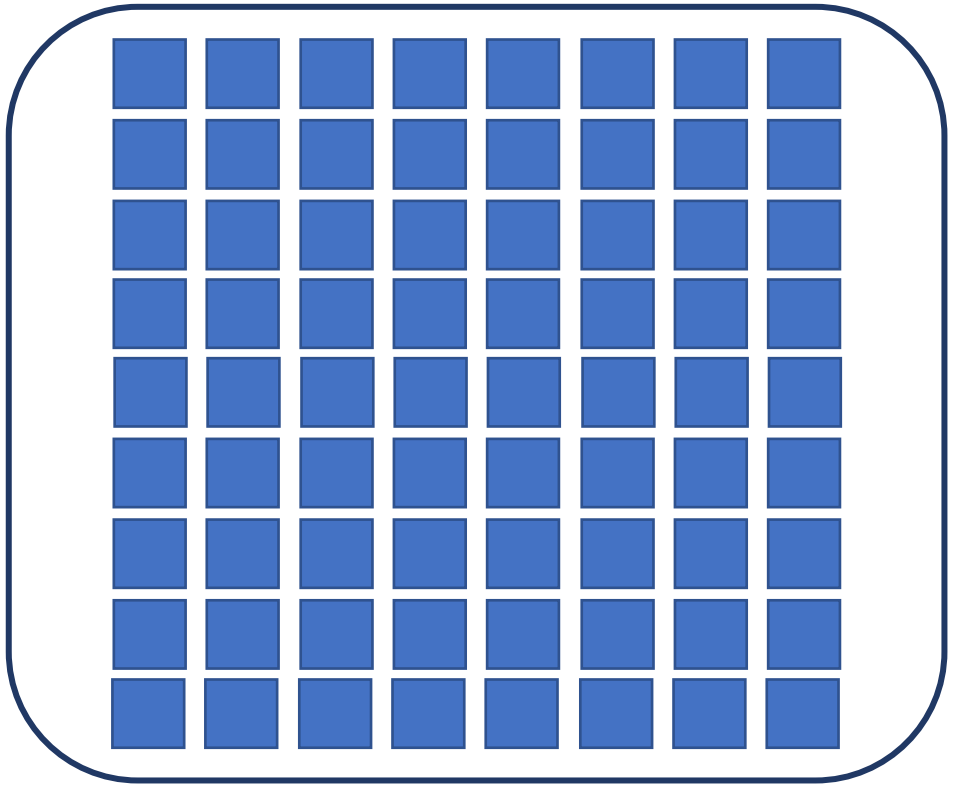


Ex: SKX Compute Node

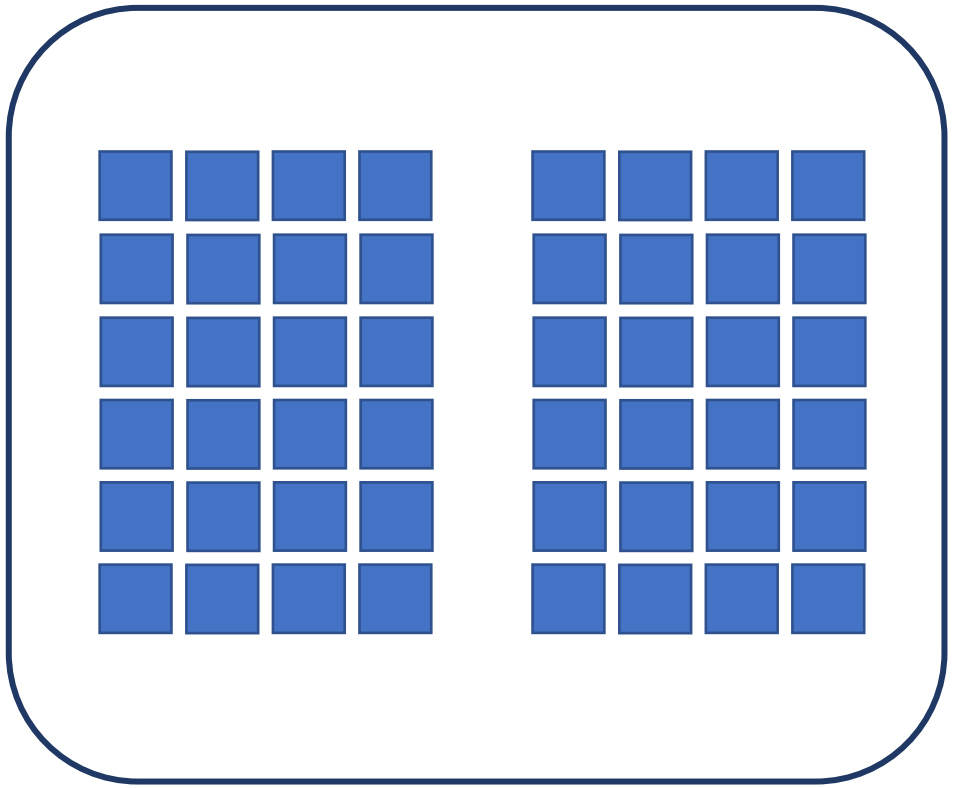
Model	Intel Xeon Platinum 8160 ("Skylake")
Cores Per Node	48 cores on two sockets (24 cores/socket)
Hardware Threads per Core	2
Hardware Threads per Node	96
Clock Rate	2.1Ghz
RAM	192GB
Cache	57MB per socket
Local Storage	144GB /tmp partition

Ex: Physical Layout

KNL Node (68 cores per node)



SKX Node (24 cores/socket * 2)



An Overview of HPC: Using a System

- Why would I use HPC resources?
  - Large scale problems
  - Parallelization and Efficiency
  - Collaboration
- How do I find an HPC resource?
  - Check with your institution
  - Check with national scientific groups (NSF in the US)

Overview: What Can I Find on a System

- Modules and Software
  - Basic compilers and libraries
  - Popular packages
  - Licensed software

- Build Your Own!
  - github or other direct sources
  - pip, wget, curl, etc
  - You won’t have sudo access

What Do I Need to Get Started?

- User Account
- Allocation & Project
- Two Factor Authentication

SSH Protocols

- Secure Shell
  - Encrypted network protocol to access a secure system over an unsecured network
  - automatically generated public-private key pairs
  - Your Wi-Fi → Stampede2 or other secure machine
  - File transfers (scp & rsync)
- Options
  - .ssh/config
  - Host & Username
  - Make connecting easier
  - Passwordless Login

Where am I?

- Login Nodes
  - Manage files
  - Build software
  - Submit, monitor and manage jobs
- Compute Nodes
  - Running jobs
  - Testing applications

Allocations

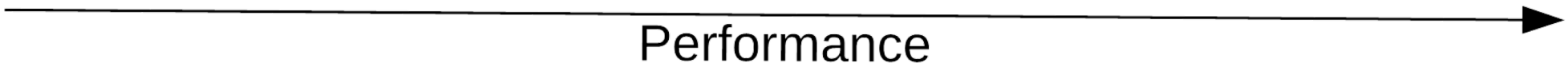
- Active Project with a Project Instructor attached
- Service Units (SUs)
  - SUs billed (node-hrs) = (# nodes) x (wall clock hours) x (charge rate per node-hour)
- Shared Systems
  - Be a good citizen

Filesystems

- Division of Labor
  - Linux Cluster (Lustre) system that look like a single hard disk space
  - Small I/O is hard on the system
  - Striping large data (OST, MDS)

- Partitions
  - $HOME: 10GB, $WORK: 1TB, $SCRATCH: Unlimited
  - Shared system

$HOME Smallest Filesystem 6 Servers	$WORK Large Filesystem 64 Servers	$SCRATCH Large Filesystem 70 Servers
• Small therefore possible to backup • Low performance	• Shared center-wide therefore not all servers provide to Stampede2 • Convenient • Persistent	• Most performance • No space limitations • Admins purge the least valuable files periodically



Has Quota Is backed up	Has Quota Not backed up	No Quota Not backed up

Filesystems: Cont.

- Where am I?
  - `pwd` – print working directory
  - `cd` – change directory
  - `cd ..` – move up one directory
- New Files
  - `mkdir` – make directory
  - Editors – vi(m), nano, emacs
  - `mv` – move a file to another location

Types of Code

- Serial Code
  - Albeit a very, very small one
  - Single tasks, one after the other
  - Single node/single core
- Parallel Code
  - Array or “embarrassingly parallel” jobs
  - Many node/many core
  - Uses MPI
  - Hybrid codes

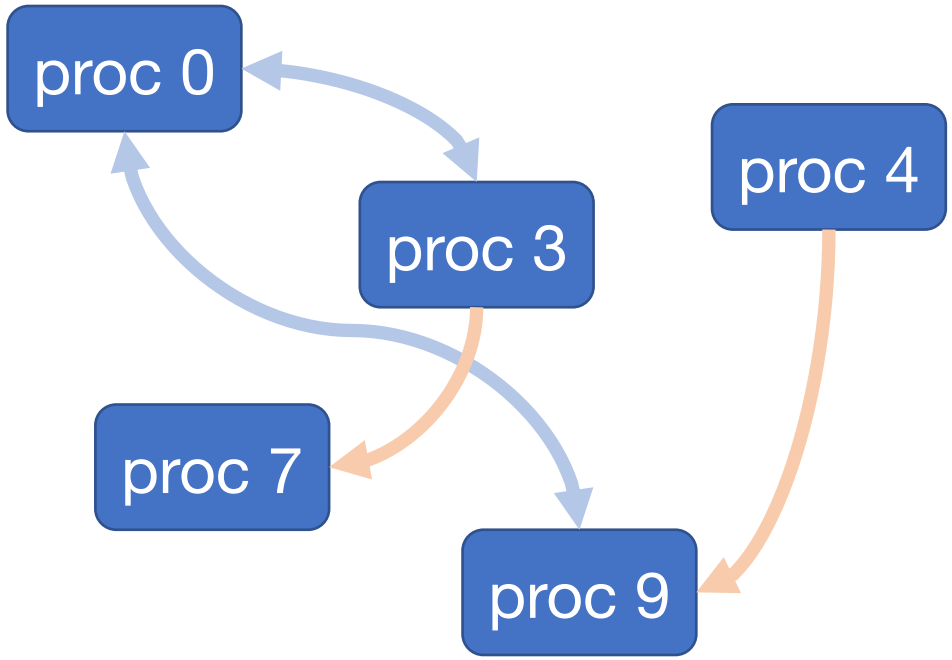
Message Passing Interface

- `ibrun` is TACC specific
  - "Wrapper" for `mpirun`
  - Execute serial and parallel jobs across the entire node

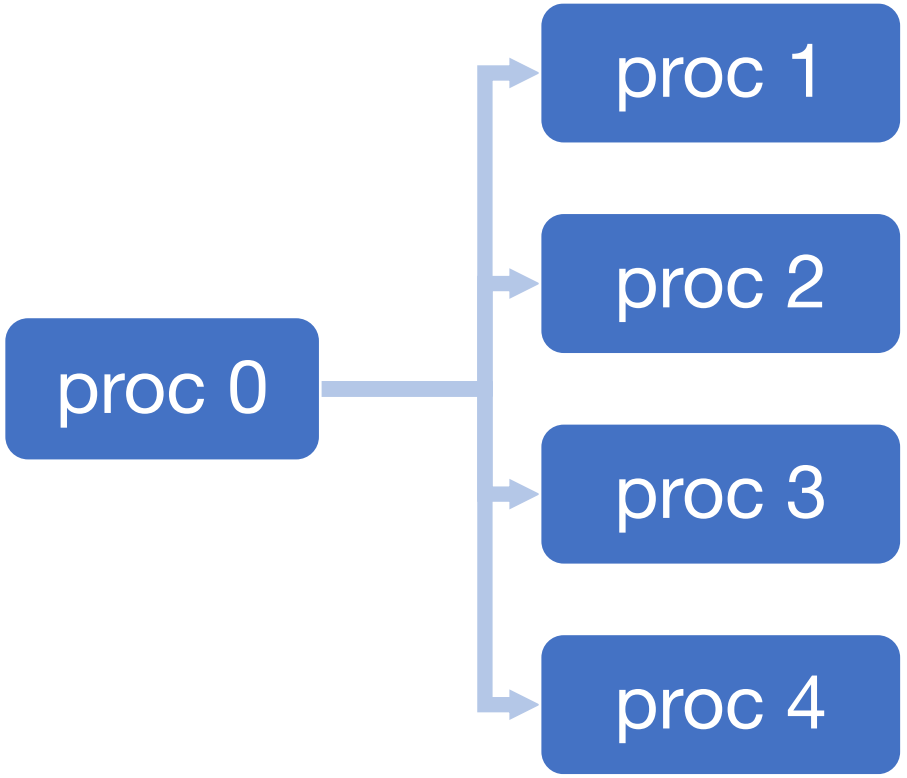
- MPI Functions
  - Allows communication between all cores and all nodes
  - Move data between parts of the job that need it
  - Point-to-Point or Collective Communication

Ex: MPI Communication

Point-to-Point



Collective



Submitting a Job

- Why submit?
  - Larger jobs, more nodes
  - You don’t have to watch it in real time
  - Run multiple jobs simultaneously
- Queues
  - Pick the queues that suit your needs
  - Don’t request more resources than you need
  - Remember this is a shared resource

- Never run on a login node!

Queue Name	Node Type	Max Nodes per Job	Max Duration	Max Jobs in Queue	Charge Rate
development	KNL cache-quad	16 nodes	2hrs	1	1SU
normal	KNL cache-quad	256 nodes	48hrs	50	1SU
large**	KNL cache-quad	2048 nodes	48hrs	5	1SU
long	KNL cache-quad	32 nodes	96hrs	2	1SU
flat-quadrant	KNL flat-quad	24 nodes	48hrs	2	1SU
skx-dev	SKX	4 nodes	2hrs	1	1SU
skx-normal	SKX	128 nodes	48hrs	25	1SU
skx-large**	SKX	868 nodes	48hrs	3	1SU

Submitting a Job cont.

- `sbatch`
  - Simple Linux Utility for Resource Management (SLURM)
  - Linux/Unix workload manager
  - Allocates resources
  - Executes and monitors jobs
  - Evaluates and manages pending jobs
- Using a Scheduler
  - Gets you off of the login nodes (shared resource)
  - Means you can walk away and do other things

Submission Options

Option	Argument	Comments
`-p`	queue_name	Submits to queue (partition) designated by *queue_name*
`-J`	job_name	Job Name
`-N`	total_nodes	Required. Define the resources you need by specifying either: (1) "`-N`" and "`-n`"; or (2) "`-N`" and "`--ntasks-per-node`".
`-n`	total_tasks	This is total MPI tasks in this job. When using this option in a non-MPI job, it is usually best to set it to the same value as "`-N`".
`-t`	hh:mm:ss	Required. Wall clock time for job.
`-o`	output_file	Direct job standard output to *output_file* (without -e option error goes to this file)
`-e`	error_file	Direct job error output to *error_file*
`-d=`	afterok:jobid	Dependency: this run will start only after the specified job successfully finishes
`-A`	projectnumber	Charge job to the specified project/allocation number.

Managing Your Jobs

- `qlimits` – all queues restrictions
- `sinfo` – monitor queues in real time
- `squeue` – monitor jobs in real time
- `showq` – similar output to `squeue`
- `scancel` – manually cancel a job
- `scontrol` – detailed information about the configuration of a job
- `sacct` – accounting data about your jobs

```
staff.stampede2(1009)$ squeue -u vtrue
    JOBID   PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
    1604426 development idv20717    vtrue  R      16:57      1 c455-001

staff.stampede2(1010)$ scontrol show job=1604426
JobId=1604426 JobName=idv20717
   UserId=vtrue(829572) GroupId=G-815499(815499) MCS_label=N/A
   Priority=400 Nice=0 Account=A-ccsc QOS=normal
   JobState=RUNNING Reason=None Dependency=(null)
   Requeue=0 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   RunTime=00:18:08 TimeLimit=00:30:00 TimeMin=N/A
   SubmitTime=2018-06-09T21:27:33 EligibleTime=2018-06-09T21:27:33
   StartTime=2018-06-09T21:27:36 EndTime=2018-06-09T21:57:36 Deadline=N/A
   PreemptTime=None SuspendTime=None SecsPreSuspend=0
   LastSchedEval=2018-06-09T21:27:36
  ...
```

On Node Monitoring

- `cat /proc/cpuinfo`
  - Follow a read out of the cpu info on the node
- `top`
  - See all of the processes running and which are consuming the most resources
- `free –g`
  - Basic print out of memory consumption
- Remora
  - This is an open source tool developed by TACC that can help you track memory, cpu usage, I/O activity, and other options

Modules

- Modules
  - TACC uses a tool called Lmod
  - Add, remove, and swap software packages
  - Saves you from having to build your own
- Commands
  - `module spider <package>` - search for a package
  - `module list` – see currently loaded modules
  - `module avail` – list all available packages
  - `module load <package>` - load a specific package or version

Q&A

Questions Answered and Demonstrations Provided

TACC
TEXAS ADVANCED COMPUTING CENTER
WWW.TACC.UTEXAS.EDU

TEXAS
The University of Texas at Austin

Further Information

Main Website: www.tacc.utexas.edu

User Support: www.portal.tacc.utexas.edu

Email: info@tacc.utexas.edu