# 杭州电子科技大学高性能计算集群使用培训

2024-03-13



第二:美国的超算,只可以用于地质勘探,不 可以用于其他领域。

第三: 机房的钥匙由美国人掌握, 每次使用, 都必须向美国人报批,美国人同意以后才可以 用。并且使用的时候要有美国人在边上监督。



每秒230万亿次浮点运算 首次突破百万亿次



·中国首台实测性能超千万亿次 高性能计算机 每秒1270万亿次浮点运算

下一代"E级"超级

全浸没式液冷计算机

每秒超亿万次浮点运算不远













## 集群系统设置



- 平台配置管理节点2台,分别配置2颗Intel(R) Xeon(R) Silver 4314 CPU @ 2.40GHz处理器,管理节点一(tc6000)是 用户登入集群的门户;管理节点二(tc6001),部署宁畅SIMS 5.1集群管理系统,用户可通过IP地址实现 Web访问;
- 平台配置CPU计算系统有60台,配置2颗Intel(R) Xeon(R) Platinum 8358P CPU @ 2.60GHz 32核处理器,256GB 内存;节点号为node1~node60;
- 平台配置GPU计算系统有5台,节点号node191~node193配置2颗Intel(R) Xeon(R) Platinum 8358P CPU @ 2.60GHz 32核处理器,1TB内存;每台节点配置了8块NVIDIA NVLink A100 40GB GPU卡;节点号 node194~node195配置2颗Intel(R) Xeon(R) Platinum 6248R CPU @ 3GHz 24核处理器,256GB内存;每台节点 配置了8块NVIDIA NVLink RTX 4090 24GB GPU卡。
- 平台配置一套**曙光Parastor**分布式存储系统,由5台ostor节点组成,采用4+2:1容错机制,可提供约906TB 的存储空间,挂载至每个节点/public目录,供整个集群使用。
- 所有节点使用2段网络进行连接,其中监控管理与IPMI管理网采用1Gb以太网互连,计算与存储网采用 200Gb HDR Infiniband互连;
- 平台采用CentOS Linux release 7.8.2003版本操作系统,配置intel\GNU的C与fortran编译器、MPI并行环境、Python、java、数学库等软件。

## 集群系统设置-管理、计算、存储资源



计算子系统	节点名	节点描述	节点数	作业队列
管理节点1	TC6000	登录方式: Shell	1	-
管理节点2	TC6001	登录方式: Shell\WEB	1	-
CPU计算节点	node1-68	2颗Intel 8358P 2.60GHz 32核处理器, 256GB内存	68	normal
GPU计算节点	Node191-195	2颗Intel 8358P 2.60GHz 32核处理器, 1TB内存, 8块NVLink A100 40GB GPU卡	5	gpu
存储		/public共享目录,所有节点均共同挂载		

## 集群系统设置-软件部署路径约束



- 用户家目录/public/home/username,用户只对属于自己的家目录具有完全的读写权限,用 户间家目录禁止互相访问,私有软件及个性化环境变量,可以部署在自己的家目录下。
- 软件安装目录/public/software/, 集群共享软件部署路径。
- 注:安装在/public/software/下在的软件以软件名加版本号来区分。
- 环境变量目录/public/software/profile.d/集群中所有部署软件的环境变量文件均放置在该目 录下,可按需调用。
- 注:每个软件对应1个环境变量配置文件,变量文件名以软件名加版本号来区分。 普通用户自已安装在自己家目录中的软件,环境变量可以写在自已家目录中的.bashrc文件中。
- 软件源码目录/public/sourcecode,存放安装程序源代码











# 集群系统登录方式



WEB浏览器登录地址: <u>https://manage-sc.hdu.edu.cn/sso/login</u> 作业提交、计算请使用普通账号, root账号仅用于服务器配置、软件安装使用。 登录工具: putty命令行登录工具, (推荐Xmanager Enterprise)

VNC图形桌面登录: VNC Viewer;

Linux to Windows文件传输工具: winscp (推荐xftp)

Linux to Linux文件传输: scp





□SSH Secure Shell Client, PuTTY, SecureCRT、xshell等SSH客户端软件登录。

□Linux客户端可以直接在命令行终端中执行ssh命令进行登录:

\$>ssh username@登录节点IP地址



tegory:		
Session	Basic options for your PuTT	Ysession
Logging     Terminal     Keyboard     Bell     Features     Window     Appearance     Behaviour     Translation     Selection     Colours     Connection     Data     Proxy     Telnet     Rlogin     SSH     Serial	Specify the destination you want to conduct Name (or IP address)  1、此处输入服务器IP地址 Connection type: <ul> <li>Raw</li> <li>Telnet</li> <li>Rlogin</li> </ul> <li>Load, save or delete a stored session Saved Sessions <ul> <li>第一次使用可以在此输入会计</li> </ul> </li> <li>Default Settings <ul> <li>点SAVE保存会以后只要双击会话可连接服务器</li> </ul> </li> <li>Close window on exit: <ul> <li>Always</li> <li>Never</li> <li>Only on</li> </ul></li>	onnect to Port 22 2、端口号22 SSH Serial 话名称 正ad E名称 正ad Delete Delete
About	5、 点open) Open	王女版方話 Cancel

## 集群登录方式二: SIMS WEB登录



#### Web登录地址: <a href="https://manage-sc.hdu.edu.cn/sso/login">https://manage-sc.hdu.edu.cn/sso/login</a> 兼容浏览器: <a href="https://chanage-sc.hdu.edu.cn/sso/login">chanage-sc.hdu.edu.cn/sso/login</a> 兼容浏览器: <a href="https://chanage-sc.hdu.edu.cn/sso/login">chanage-sc.hdu.edu.cn/sso/login</a>

		促父 作业官理 数据官理	節令行 [E-Snell]						
	9 您的团队账户u42840_test已欠费,请及即	① 您的团队账户u42840_test已欠费,请及时充值,以免影响提交作业。 2							
账号登录 	快捷入口	快捷入口							
142840 test	模板提交 BASE	>_	图形提交						
······	基础模板	命令行	Linux桌面						
	← G https://manage-so 命令行 (E-Shell ) 文件管	c.hdu.edu.cn/ac/console3/index.ht ]理(E-File)	ml#/space/shell						
登录 找回密码	Last login: Wed Mar 6 10:20	2:38 2024							

# 集群登录方式三: VNC桌面登录



Shell登录集群,执行vncserver -geometry 1366x768,即开 启一个分辨率为1366x768的VNC桌面(注意桌面ID)。



Creating default startup script /public/home/user00/.vnc/xstartup Starting applications specified in /public/home/user00/.vnc/xstartup Log file is /public/home/user00/.vnc/tc6000:2.log

#### 在VNC Viewer客户端中输入IP:ID, 登录VNC桌面



### Applications Places Terminal - 🗆 X user00@tc6000:~ File Edit View Search Terminal Help [user00@tc6000 ~]\$ Home 3 Trash

在VNC桌面桌面中可以使用终端或图形化执行任务。

## VNC工具使用



### 启用VNC步骤:

- 1、以自己的用户名命令行登录系统
- 2、输入vncserver (注意, 第一次运行会提示设置密码)
- 运行完后,注意生成的ID号,然后即可以用VNC viewer客户端工具连接。

高级操作:

- 1、vncserver -kill:ID (杀掉某个ID的VNC桌面)
- 2、示例使用: vncserver -geometry 1920x1080 24 :5 指定分辨率1920x1080,颜色位深度为24,连接端口ID为5;
- 3、改图形化登录密码: vncpasswd
- 注:为了安全,建议图形VNC登录密码不要与命令行登录密码设置为同一个。

# 上传下载数据: WinSCP简介



### ■ Windows与集群进行文件传输可使用WinSCP或SFTP等软件

- defaultsttp - SSH Secure F	File Transfer			
File Edit View Operation	<u>W</u> indow <u>H</u> elp			
<b>     # #</b>   <b># #</b>   <b># #</b>	🕽   🦠   J 🕆   🖻	≞	22年 🖸 🥔 🍂	
👔 Quick Connect  📄 Profile	25			
🔁 🛛 🚳 📾 🗢 🛛 📸 🗙 🗍			🔽 🛛 Add 🛛 🎦 🏄 📾 🕫 🕌 🗙 🗍	▼ Add
ocal Name 🔼	Size Type	Modified	Remote Name Size Type	Modified Attributes
■ 库 ■ 家庭组 ■ liujiahui ■ 计算机 ■ 内格 ■ 控制面板 ■ 空制面板 ■ SSH Secure File Transfe ■ SSH Secure Shell Client ■ SSH Secure Sh	<ul> <li>系统文件,</li> <li>系统文件,</li> <li>系统文件,</li> <li>系统文件,</li> <li>系统文件,</li> <li>系统文件,</li> <li>系统次文件,</li> <li>系统次次,</li> <li>系统次次,</li> <li>系统次次,</li> <li>系统次次,</li> <li>系统次次,</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li> <li>系统公式、人、</li></ul>		Connect to Remote Host       Wost Name:     Connect       User Name:     Cancel       Fort     22       Authentication     Profile Settings>	
Transfer Queue				
		cton	Destination Directory Size Status	- Coood Time

Linux操作系统间复制文件,直接使用命令
 scp filename test@ip:/home/test













软件	安装目录
编译器	Intel2021、GCC-7.3.1、Cmake-3.20.1; /public/software/compiler/
数学库	Fftw-3.3.9、Gsl-2.6、Hdf5-1.12.0、Jasper-1.900.1、Lapack-3.9.1、Netcdf-4.7.4、Petsc-3.15.0; /public/software/mathlib/
MPI	Intelmpi-2021.3.0、Openmpi-4.0.3; /public/software/mpi/intelmpi/2021.3.0
anaconda3	/public/software/apps/anaconda3/2021.0
ansys	/public/software/apps/ansys_inc/v211
OpenFOAM6	/public/software/apps/OpenFOAM-6/





软件名称		软件信息
	软件版本	4.8.5
	安装路径	/usr
GNU编译器	调用方式	C: gcc C++: g++ F77: g77 F90: gfortran
	软件版本	2021
	安装路径	/public/software/compiler/intel-compiler/2021.3.0/
Intel编译器	环境变量配置文件	/public/software/profile.d/compiler_intel-compiler-2021.3.0.sh
	调用方式	C: icc C++: icpc F77: ifort F90: ifort



软件名称		软件信息				
	说明	OpenMPI,与Intel编译器关联,支持以太网和InfiniBand。				
	软件版本	openmpi-4.0.3				
OpenMDI	安装路径	/public/software/mpi/openmpi/intel/4.0.3				
OpenMPI (与Intel编	环境变量配置文件	/public/software/profile.d/mpi_openmpi-intel-4.0.3.sh				
(年 話 大 味 )	调用方式	C: mpicc C++: mpicxx F77: mpif77 F90: mpif90 mpirun -np <n> -machinefile <machinefile></machinefile></n>				





软件名称		软件信息			
	说明	单精度版FFTW3数学库			
FFTW3-	软件版本	fftw-3.3.9			
float	安装路径	/public/software/mathlib/fftw/intelmpi/3.3.9_single/			
	调用方式	<编译器> -I/public/software/profile.d/mathlib_fftw-intelmpi- 3.3.9_single.sh			
	说明	双精度版FFTW3数学库			
FFTW3-	软件版本	fftw-3.3.9			
uouble	安装路径	/public/software/mathlib/fftw/intelmpi/3.3.9_double/			
	调用方式	<编译器> -I/public/software/profile.d/mathlib_fftw-intelmpi- 3.3.9_double.sh			





软件名称	软件信息								
Intel MKL	说明	Intel MKL, 包含BLAS、LAPACK、FFT、ScaLAPACK、BLACS等							
	软件版本	Intel Compiler 2021自带版本							
	安装路径	/public/software/compiler/intel-compiler/2021.3.0/MKL							
	环境变量配置文件	/public/software/profile.d/compiler_intel-compiler- 2021.3.0.sh							
	调用方式	ifort/icc –L\${MKLROOT}/lib/em64t -lmkl_intel_lp64 - lmkl_sequential -lmkl_core							

## anaconda3



软件名称		软件信息						
	python	3.8						
anaconda3	软件环境	<pre># conda environments: base * /public/software/apps/anaconda3/2021.05 pytorch-cpu /public/software/apps/anaconda3/2021.05/envs/pytorch-cpu tf-cpu /public/software/apps/anaconda3/2021.05/envs/tf- cpu tf-gpu /public/software/apps/anaconda3/2021.05/envs/tf- gpu</pre>						
	安装路径	/public/software/apps/anaconda3						
	调用软件方法	conda activate tf-gpu						











## 集群环境变量文件路径



#### 安装软件的环境变量设置脚本路径(/public/software/profile.d)

[root@tc6001 profile.d]# pwd /public/software/profile.d [root@tc6001 profile.d]# ls apps anaconda3-2021.05.sh apps ansys194.sh apps\_ansys211.sh apps openfoam6.sh benchmark hpl-intelmpi-2.2.sh benchmark htop-gnu-2.1.0.sh benchmark iozone-gnu-3.430.sh compiler cmake-compiler-3.20.1.sh compiler gcc-7.3.1.sh compiler intel-compiler-2017u8-env.sh compiler\_intel-compiler-2021.3.0.sh cuda-10.1-env.sh cuda-11.4.3-env.sh [root@tc6001 profile.d]#

mathlib\_fftw-intelmpi-3.3.9\_double.sh
mathlib\_fftw-intelmpi-3.3.9\_single.sh
mathlib\_gsl-gnu-2.6.sh
mathlib\_df5-gnu-1.12.0.sh
mathlib\_hdf5-gnu-1.12.0.sh
mathlib\_hdf5-intel-1.12.0.sh
mathlib\_netcdf-gnu-4.7.4.sh
mpi\_intelmpi-2021.3.0.sh
mpi\_openmpi-4.1.2-intel2021-env.sh
mpi\_openmpi-gnu-3.1.3.sh
mpi\_openmpi-gnu-4.0.3.sh

环境变量文件中记录了软件安装的路径等信息,通过加载不同软件的环境变量,操作系统才可以 识别并使用相关软件。

## 用户自定义调用默认环境变量



#### • 各用户在.bashrc中选择需要的进行设置

```
.bashrc
#
# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi
# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD PAGER=
# User specific aliases and functions
source /public/software/profile.d/mpi openmpi-intel-4.0.3.sh
```

# 临时调用所需的软件环境变量



/public/software/profile.d/

├── apps\_ansys211.sh

. . . . . . . . .

├── apps\_anaconda3-2021.05.sh

L---- mpi\_openmpi-intel-4.0.3.sh

使用参考: source /public/software/profile.d/mpi\_openmpi-intel-4.0.3.sh ##openmpi 环境变量激活

## 运行计算程序



#### ■运行一个串行程序a.out

需要注意该程序是否为单进程多线程程序,即openmp并行程序,如高斯软件

[test@tc6000 test]\$ ./a.out

### ■运行一个标准的mpi并行程序a.out

[test@tc6000 test]\$ mpirun -np 180 -machinefile ./nodelist ./a.out

### ■运行一个商业版并行程序fluent

[test@tc6000 test]\$ fluent 2d -t240 -p -cnf=./nodelist -ssh -mpi=intel -g -i Input >out.log



# 注: 单进程串行程序运行特点



top - 11:42:3	30 up	147	days, 2:	14, 1 us	er, l	oad a	verage:	28.81,	28.7	3, 28	.76			
Tasks: <b>516</b> to	otal,	29	running,	487 sleep	ing,	0 st	opped,	0 zom	oie					
%Cpu(s): <b>99.</b>	6 us,	0.4	sy, 0.0	ni, <b>0.</b> 0	id,	0.0 w	a, 0.0	) hi, 0	0 si	, 0.	0 st			
KiB Mem : 394	468774	+tot	al, 36435	603+free,	20222	<b>688</b> u	sed, 10	108992	ouff/	cache				
KiB Swap: <b>16</b> 7	777212	tot	al, 16400	576 free,	376	<b>636</b> u	sed. 36	916816+a	avail	Mem				
PID USER	F	PR N	I VIRT	RES	SHR	S %C	PU %MEM	IIT I	1E+ C	omman	D			
97802 jingq	_LJ 2	20	0 620604	380932	7136	R 100	.0 0.1	40:59	.02 S	UBSPA	CE_PB_D	IA		
99185 jingq	_LJ 2	20	0 459884	271024	7116	R 100	.0 0.1	. 33:45	90 S	UBSPA	CE_PB_D	IA		
99994 jingq	_LJ 2	20	0 453884	234624	7072	R 100	.0 0.1	. 30:07	.19 S	UBSPA	CE_PB_D	IA		
101082 jingq	_LJ 2	20	0 479264	252884	8632	R 100	.0 0.1	22:16	81 S	UBSPA	CE_PB_D	IA		
101310 jingq_	_LJ 2	20	0 453744	235036	8880	R 100	.0 0.1	20:35	79 S	UBSPA	CE_PB_D	IA		
101415 jingq	_LJ 2	20	0 497548	273504	8808	R 100	.0 0.1	20:31	47 S	UBSPA	CE_PB_D	IA		
102706 jingq_	_LJ 2	20	0 382800	169512	8800	R 100	1 1						0.00.1	
102934 jingq	_LJ 2	20	0 467872	244968	8556	R 100			00.03 00.0%	1 0			0.03	15 [         100.0%] 22 [        100.0%]
97252 jingq	_LJ 2	20	0 495388	234168	7192	R 99	3		00.0%	] 1 ] 1	0 I	10         10		<b>17</b> [          100.0%] <b>23</b> [          100.0%]
98362 jingq	_LJ 2	20	0 581924	353736	7152	R 99	4		00.0%	1 1	1	1111110	0.0%	18 [
99307 jingq	_LJ 2	20	0 554260	319788	7008	R 99	5 1		00.0%	i 1	2		0.0%	19 11111111100.0% 26 111111111100.0%
100239 jingq	_LJ 2	20	0 494168	267944	6868	R 99	6		00.0%	j	.3 [	10	0.0%	20 [       100.0%] 27 [       100.0%]
100938 jingq	_LJ 2	20	0 346816	155136	8556	R 99	7 []		00.0%	] 1	.4 [	10	0.0%	21 [        100.0%] 28 [        100.0%]
101573 jingq	LJ 2	20	0 441012	223792	8796	R 99	Mem					23.2G/	376G	Tasks: 127, 56 thr; 30 running
101978 jingq	LJ 2	20	0 422520	207260	8868	R 99	Swp					368M/1	.6.0G	Load average: 29.23 28.86 28.81
102136 jingq	LJ 2	20	0 369516	163264	8804	R 99								Uptime: <b>147 days(!), 02:15:20</b>
102262 jingq	LJ 2	20	0 422748	205772	8852	R 99			DDT					
102408 jingq	LJ 2	20	0 451208	227244	8792	R 99	PID U				TM 1CEN	5 SHK	5 CPU%	MEM% IIME+ COMMAND
102531 jingq	LJ 2	20	0 474460	246368	8868	R 99	02136 J.		20	0 30	6M 173N	1 8556	R 101.	0.0 15:02.71/SUBSPACE_PB_DIAG_P0P9_X_V
						L	98136 i	ingq_LJ	20	0 66	5M 427N	1 0350 1 7152	R 99 9	0.1 40.20 07 /SUBSPACE PB DIAG POP9 X V
							00239	ingq_LJ	20	0 48	2M 261	6868	R 99.9	0.1 28:25.15/SUBSPACE PB DIAG POP9 X V
							02828	ingg LJ	20	0 36	1M 158M	8784	R 99.9	0.0 6:19.74/SUBSPACE PB DIAG POP9 X V
							99185 j	ingq LJ	20	0 57	0M 346M	7116	R 99.9	0.1 34:37.69/SUBSPACE PB DIAG POP9 X V
						-	03325 j	ingq_LJ	20	0 38	3M 180N	<mark>1 9</mark> 704	R 99.9	0.0 1:41.34/SUBSPACE_PB_DIAG_POP9_X_V
							99994 j	ingq_LJ	20	0 44	3M 229N	7072	R 99.9	0.1 30:58.97/SUBSPACE_PB_DIAG_POP9_X_V
							00712 j	ingq_LJ	20	0 37	0M 181	8544	R 99.9	0.0 25:29.09/SUBSPACE_PB_DIAG_POP9_X_V
						-	01188 j	ingq_LJ	20	0 41	7M 207N	1 8544	R 99.9	0.1 22:50.26/SUBSPACE_PB_DIAG_POP9_X_V
							02/06 ]: 07252 -	ingq_LJ	20	0 37	9M 153N	1 8800	R 99.9	0.0 6:55.53/SUBSPACE_PB_DIAG_P0P9_X_V
							1 <mark>Help</mark>	Engq_LJ E2 <mark>Setup</mark> E	20 3 <mark>502 r</mark>		1 tor 504		r 99.9	Nice = E8Nice + E9Kill = E10Ouit
							петр	Zsecup r	Sear			ree ros	огсбуг/	NICE - FONICE + FONICE FIOQUIL

#### 单进程多线程程序运行特点 注:

top - 11:41:03 up 147 days, 2:05, 1 user, load average: 29.67, 29.15, 28.94 Tasks: 423 total, 2 running, 421 sleeping, 0 stopped, 0 zombie %Cpu(s): 99.9 us, 0.1 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st KiB Mem : 13175065+total, 11128040+free, 11066500 used, 9403752 buff/cache KiB Swap: 15624188 total, 15624188 free, 0 used. 11757093+avail Mem

PID	USER	PR	NI	VIRT	RES	SHR S	%CPU	%MEM	TIME+ C	COMMAND
18515	ZCY_HG	20	0	59.367g	2.442g	8916 R	2796	1.9	42140:0 <mark>9</mark> l	.502.exe
1768	root	20	Θ	851172	51736	1712 S	1.3	0.0	1878:35 c	Mgcd
2267	root	20	Θ	113328	49928	11884 S	0.3	0.0	321:28.91 p	obs_mom
5298	root	20	0	157980	2520	1544 R	0.3	0.0	0:00.04 t	ор
1	root	20	Θ	192892	5908	2504 S				
2	root	20	Θ	Θ	Θ	0 S	1		100.0%	] 8 [
3	root	20	Θ	Θ	Θ	0 S	2	[]]]	100.0%	] 9 [
5	root	Θ	-20	Θ	0	0 S	3	[	100.0%	] 10 [
8	root	rt	Θ	Θ	0	0 S	4		100.0%	
9	root	20	Θ	Θ	0	0 S	5	Ļ		
10	root	20	0	Θ	0	0 S	: b			] 13 [ ] 14 [
11	root	rt	0	0	0	0 S	/ Me	m		14
10	root	rt	0	0	0	0 5	Sw	n[		

0	321:28.91 pbs 0:00.04 top	_mom			
	100.0%	8	<b>[</b> ] ] ]		10
	100.0%	9	i		110
	100.0%	10	[		j <b>j</b> 10
	100.0%	11			10
	100.0%	12			10
	100.0%	13			10
	100.0%	14			10
				12	.9G/
					0K/1

0%	<b>15</b> [         100.0%]	22 [	100.0%
0%	16 [         100.0%]	23 [	100.0%
0%	17 [         100.0%]	24 [	100.0%
0%	<b>18</b> [          100.0%]	25 [	100.0%
0%	<b>19 [          100.0%]</b>	26 [	100.0%
0%]	20 [         100.0%]	27 [	100.0%
0%	<b>21</b> [         100.0%]	28 [	100.0%
6 <b>G</b>	Tasks: 44, 494 thr; 32	running	
9G	Load average: 29.39 29.	15 28 95	

**Sugon** 

Uptime: 147 days(!), 02:06:05 SHR S CPU% PID USER PRI NI VIRT RES /public/software/g16/g16/l502.exe 18515 ZCY HG 20 0 59.4G 2504M 8916 R 2800 1.9 702h 2063 ZCY HG 0 59.4G 2504M 8916 100. 1.9 10:55.70 20 <mark>8</mark>916 1.9 10:55.71 2089 ZCY HG 20 0 59.4G 2504M 100. 2064 ZCY HG 20 0 59.4G 2504M 8916 100. 1.9 10:54.05 2088 ZCY HG 20 0 59.4G 2504M 8916 100. 1.9 10:55.07 2065 ZCY HG 20 0 59.4G 2504M <mark>8</mark>916 100. 1.9 10:55.59 2074 ZCY HG 0 59.4G 2504M <mark>8</mark>916 100. 1.9 10:55.40 20 <mark>8</mark>916 1.9 10:55.43 2075 ZCY HG 20 0 59.4G 2504M 100. 8916 1.9 10:55.52 2080 ZCY HG 0 59.4G 2504M 100. 20 2090 ZCY HG 0 59.4G 2504M <mark>8</mark>916 100. 1.9 10:55.62 20 2087 ZCY HG 20 0 59.4G 2504M <mark>8</mark>916 100. 1.9 10:55.49 2066 ZCY HG 0 **59.4G** 2504M 8916 R 100. 1.9 10:55.57 20 F1Help F2Setup F3SearchF4FilterF5Tree F6SortByF7Nice -F8Nice +F3Kill



串行程序可以执行taskset -c 0 ./a.out将程序绑定在CPU核Core0上运行,并行程序绑定具体参照各类mpi手册 如openmpi可以使用: mpirun -np <N> -machinefile <machinefile> --bind-to-core --bycore <executable> 理想不送

## 集群作业管理调度系统



HPC计算集群作为公共的平台,当有多个用途或是多个使用者时,必须引入<mark>作业管理调度系统</mark>, 以便对集群进行资源配额,和任务的自动化管理,集群使用者应避免出现以下三种问题:

1、在管理节点上长时间运行计算程序;

2、避免绕开调度系统在计算节点运行程序,这样会因为没有统一分配调度资源,导致某些计算节点执行超负载的计算任务,而部分节点却没有计算任务而资源闲置。

3、OpenMP多线程程序,请合理申请资源,避免单节点超负载,程序运行时,释放多少线程,在作业运行脚本中就申请多少个核心的资源。

## 作业管理系统的结构





### slurm作业提交步骤



- 1. 准备:编写描述改作业的脚本,包括作业名,需要的资源等。
- 2. <u>提交</u>:使用sbatch命令将该作业提交给slurm服务器
- 3. 排队: 服务器将该任务排入适当的队列
- 4. 调度: 服务器检查各工作节点的状态是否符合该作业的要求,并进行调度。
- 5. <u>执行</u>:当条件满足时,作业被发给相应的执行服务器执行。程序运行时执行服务器会收集 程序的标准输出和标准 错误流,等程序结束时,将这些信息返回给用户。
- 6. <u>查询和调整</u>:当作业在运行时,用户可以使用squeue进行状态查询。用户发现作业提交错误时,可以使用scancel 删除正在运行的作业。
- 7. 查看结果:使用文本编辑软件vi或者系统命令cat, less等查看输出及错误信息显示。

## slurm作业脚本



□本质是一个SHELL脚本
□注释以"#"开头
□slurm运行参数,以"# SBATCH"开头,此处的"#"不是注释作用,具有实际意义
□可以直接调用SHELL命令和系统命令

#!/bin/sh #SBATCH -J GENERAL\_0331\_080311 #SBATCH -N 5 #SBATCH -n 36 #SBATCH -t 2400:0:0 #SBATCH -p gpu\_v100

cd \$SLURM SUBMIT DIR 进入作业脚本所在的目录,以这个目录为计算目录

申请资源

module load intelmpi/2017.4.239 加载运行所需的环境变量

mpirun -np 180 -machinefile ./ma -rdma ./xhpl\_intel64\_static >\$log.out







#!/bin/bash **#SBATCH - J jobname #SBATCH** -o test.out #SBATCH -p gpu **#SBATCH - N 10** #SBATCH -n 320 **#SBATCH** --ntasks-per-node=32 #SBATCH -t 1:00:00 #SBATCH --gres=gpu:1

# 作业名为 jobname
# 输出文件重定向到 test.out
# 作业提交的队列为 gpu
# 作业申请 1 0个节点
# 总计申请320个核心
# 每节点申请32个核心
# 任务运行的最长时间为 1 小时
# 单个节点使用 1 块 GPU 卡



#SBATCH --qos=debug #SBATCH --cpus-per-task=4 #SBATCH -w node6 #SBATCH -x node6 # 作业使用的 QoS 为 debug
# 单任务使用的 CPU 核心数为 4
# 指定运行作业的节点是node6,若不填写系统自动分配节点
#计算时排除node6节点

## slurm作业提交基本命令



□ 在作业调度系统中,用户使用 sbatch命令提交用户程序。用户运行程序的 命令及slurm环境变量设置组成slurm作业脚本,作业脚本使用如下格式提交 到slurm系统运行:

sbatch < slurm作业脚本>

sbatch -J wrf --nodes=N -p batch <slurm作业脚本>





变量名	<mark>说 明</mark>
登陆SHELL继承来的变量	包括\$HOME, \$LANG, \$LOGNAME, \$PATH, \$MAIL, \$SHELL和\$TZ。
\$SLURM_SUBMIT_HOST	sbatch提交的节点名称
\$SLURM_SUBMIT_DIR	sbatch提交的作业的绝对路径
\$SLURM_JOBID	作业被slurm系统指定的作业号
\$SLURM_NPROCS	申请的总核心





# 这是一个串行作业脚本的例子 #SBATCH -J test #SBATCH -N 1 #SBATCH -n 1 cd \$SLURM\_SUBMIT\_DIR ./a.out

# 这是一个并行作业脚本的例子 #SBATCH -J wrf #SBATCH -N 2 #SBATCH -n 20 #SBATCH -p test cd \$SLURM\_SUBMIT\_DIR source /public/software/profile.d/mpi-openmpi-intel.sh srun hostname -s | sort -n > slurm.hosts mpirun -np \$SLURM\_NPROCS -machinefile slurm.hosts ./a.out







作业提交后,会生成一个作业号,如: [dawning@tc6000~]\$ **sbatch test.pbs** Submitted batch job 89

查看集群作业运行状态: [test@tc6000 hpl-6230]\$ squeue TIME JOBID PARTITION NAME USER ST NODES NODELIST(REASON) v100 5 89 GENERAL node[17-21] R 0:03 test

作业状态说明:

E: 退出 Q: 排队 H: 挂起 R: 运行 C: 结束

## 命令行查询作业状态(续)



查询作业命令 squeue [参数],其中参数可为:

- -p:列出系统队列信息
- -a:列出队列中的所有作业及其分配的节点
- -t R:列出正在运行的作业
- -j jobid:列出指定作业的信息

## 命令行查看删除作业



[test@tc6000 hpl-6230]\$ squeue JOBID PARTITION NAME USER ST TIME NODES NODELIST(REASON) 89 v100 GENERAL\_test R 10:09 5 node[17-21

[test@tc6000 hpl-6230]\$ scancel 89

注: 用户只能删除自己的作业, 管理员可以删除所有用户作业

# 集群信息查看命令

- •存储可用资源
  - df –h

[root@tc6000 ~]# df -h					
Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	126G	Θ	126G	0%	/dev
tmpfs	126G	11G	116G	9%	/dev/shm
tmpfs	126G	1.4G	125G	2%	/run
tmpfs	126G	0	126G	0%	/sys/fs/cgrou
/dev/mapper/centos-root	376G	53G	324G	14%	/
/dev/sda2	1.9G	173M	1.7G	10%	/boot
/dev/sda1	200M	12M	189M	6%	/boot/efi
tmnfc	266	52K	266	1%	/run/user/A
cloudstor_public	906T	30T	877T	4%	/public

- 集群计算资源
  - pestat

							-	ф	<b><b><b></b>  曙光</b></b>
							S		
[root@tc600	α #1∽ 00	esta	t						
HOSTNAME	NODE	CPU	CORE	CPU LOAD	MEMSIZE	FREE MEM	JOB LIST		
	State	Use	/All	NUMBER	(MB)	(MB)	JobId User		
node1	idle	0	64	0.14	257436	236906			
node2	idle	Θ	64	20.78*	257436	229133			
node3	idle	0	64	0.11	257436	237930			
node4	idle	0	64	0.18	257436	239663			
node5	idle	0	64	0.17	257436	239816			
node6	idle	Θ	64	0.10	257436	239725			
node7	idle	Θ	64	0.16	257436	239794			
node8	idle	0	64	0.11	257436	239809			
node9	idle	0	64	0.17	257436	239660			
node10	idle	0	64	0.22	257436	239792			
node11	idle	0	64	0.10	257436	238033			
node12	idle	Θ	64	0.14	257436	239330			
node13	idle	Θ	64	0.16	257436	244586			
node14	idle	0	64	0.13	257436	239533			
node15	idle	0	64	0.06	257436	239693			
node16	idle	0	64	0.15	257436	239627			
node17	idle	0	64	0.16	257436	239673			_
node18	drain*	Θ	64	<u>293.90*</u>	257436	176690			
node19	idle	0	64	0.15	257436	239988			
node20	idle	0	64	0.09	257436	239562			
node21	idle	0	64	0.19	257436	239964			
node22	idle	0	64	0.19	257436	239992			
node23	idle	0	64	0.12	257436	239920			
node24	idle	0	64	0.21	257436	239858			
node25	1dle	0	64	0.15	257436	239856			
node26	1dle	0	64	0.15	257436	239342			
node27	1dle	0	64	0.14	257430	239832			
node28	idle	0	64	0.18	257430	239832			
node29	Tare	0	64	0.15	257430	239342			
node30	1016	0	64	0.18	257430	239270			
node31	idle	0	64	0.22	257430	239347			
nodesz	idle	0	64	0.20	257430	232070			
nouess	Tare	U	04	0.10	237430	239330			





← C ↔ https://manage-sc.hdu.edu.cn/ac/console3,	/index.html#/job-list/curre	nt			
◎ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★	数据管理 命令行「E-S	Shell]			
当前作业历史作业					
全部集群 > 展示组成员作业					作业ID > 搜索.
作业ID 作业名	应用	队列	集群	开始时间 💠	运行时长 🗢
				暂无数据	





数字经济基础设施的领导者 数据中国百城百行的发起者 中科院产业化联盟的推动者 信息技术应用创新的践行者