

# Introducción al uso de la infraestructura del NLHPC



## Objetivos

- Accediendo al cluster Guacolda-Leftraru
  - SSH
  - Infraestructura y recursos
- Uso de Slurm
  - Parámetros
  - Uso interactivo y encolado de tareas
  - Uso de comandos informativos
  - Monitoreo de las tareas
- Uso de Módulos
  - Consultando por software y sus versiones
  - Cargando módulos
- Escalamiento
- Ejercicios prácticos



## Participación general

Nuestra metodología busca ser dinámica y participativa

- Realizar consultas durante la presentación
- Se realizarán ejercicios en grupo
- En cada ejercicio los usuarios deberán:
  - Participar en la realización de los ejercicios
  - Compartir pantalla de manera grupal
  - Explicar los resultados de los ejercicios
- Se asignarán distintos usuarios para la realización de cada ejercicio



## Infraestructura

#### Nodo Login/debug (gn)



#### Partición Debug

- 4 Nodos
  - Intel(R) Xeon(R) CPU E5-2660
  - 20 CPUs
  - **59 GB RAM**
- Límites de tiempo de ejecución a 30 minutos
- Orientado a acceso y pruebas de compilación
- Total partición:
- 80 CPUs
- 236 GB RAM



#### Infraestructura - Leftraru

#### Nodo Slims (cn)



#### **Partición Slims**

- 132 Nodos
  - Intel(R) Xeon E5-2660 v2
  - 20 CPUs
  - 46 GB RAM
- Límites de tiempo de ejecución: 30 días
- Partición por defecto
- Total partición:
- 2.640 CPUs
- 6.072 GB RAM



#### Infraestructura - Leftraru

#### Nodo General (sn)



#### Partición General

- 48 nodos
  - Intel(R) Xeon Gold 6152
  - 44 cores
  - 187 GB RAM DIMM DDR4
- Límites de tiempo de ejecución: 30 días
- Total partición:
- 2.112 CPUs
- 8.976 GB RAM



#### Infraestructura - Leftraru

#### Nodo Largemem (fn)



#### Partición Largemem

- 9 nodos
  - Intel(R) Xeon Gold 6152
  - 44 cores
  - **765 GB RAM**
- Destinado para tareas de 192G+ RAM
- Límites de tiempo de ejecución: 30 días
- Total partición:
- 396 CPUs
- 6.885 GB RAM



#### Infraestructura - Guacolda

#### Nodo GPU (gn)



#### Partición GPU

- 2 Nodos
  - Intel(R) Xeon Gold 6152
  - 44 cores
  - 187 GB RAM
  - 2 NVIDIA Volta V100 cada nodo
    - 16GB
    - 5120 CUDA cores cada una
- Destinado a tareas que requieran uso de GPUs
- Límites de tiempo de ejecución: 30 días
- Total partición:
- 88 CPUs
- 374 GB RAM
- 20.480 CUDA cores



#### Infraestructura NLHPC



- 266 TFlops
- 5236 cores
- 191 nodos
- 4 PB almacenamiento IBM Spectrum Scale
- Red LAN Infiniband FDR 56Gbps

https://wiki.nlhpc.cl/Hardware\_Disponible



#### Accediendo al Cluster



#### Accediendo al Cluster

#### Protocolo: SSH Host: leftraru.nlhpc.cl

Linux, macOS: ssh usuario@leftraru.nlhpc.cl

Last login: Mon May 24 11:07:02 2	2021 from pc-62-	250-239-201.cm.vtr.net
**************************************	иммимимимимимими	ммммммммммммммммммммммммммм
★ BIENVENID@ A ★ MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM	4MMMNmhhhyhmMMMM	MMMMMMMMMMM NLHPC MMMMMMM
* GUACOLDA * MMMMMMMMMMMMMMMMM	My/−``````.oNM	ммммммммммммммммммммммммммм
* LEFTRARU * MMMMMMMMMMMMMMMMM	1y::::+M	ммммммммммммммммммммммммммм
**************** MMMMMMMMMMMMMMMMMMMMM	do://:+o:///+::y	ммммммммммммммммммммммммммм
мммммммммммммммммммммммммммммммм	+-/://:0/+/:+`::	ONMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
мимимимимимимимимимимимимимимимини.	`/+ssso:-`	улимимимимимимимимимимими
мммммммммммммммммммммммммммм```	`.ososss+o+s+-	. NMMMMMMMMMMMMMMMMMMMMM
мммммммммммммммммммммммммм.``'	:-sssososss	оммимимимимимимимими
мммммммммммммммммммммму+`	-/o-ss+000s-/0/	: odNMMMMMMMMMMMMMMM
MMMMMMMMMMMMMMMMMMMMMMMMM :	` .oso+/` ```	. omMMMMMMMMMMMMM
MMMMMMMMMMMMMMMMMMMMMMMM	`-/+so/+s/:.`/	+/o:``SNMMMMMMMMMMMM
ммммммммммммммммммммммммммо.+`:``	``-//+:/:+	ssos:` . `dMMMMMMMMMMM
ММММММММММММММММММММУ++/::/:-:	::-:-::::::::///	ooooss+o hMMMMMMMMMMM
МММММММММММММММММММО//////::::	:::::+s	sssssss: -+MMMMMMMMMMMMM
МММММММММММММММММММО///////:	/yho. `+s	ssosssss: :NMMMMMMMMMMMMM
MMMMMMMMMMMMMMMMm::::::::	-:o/-+yhhh++ys+-	+s/ssssss.oMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
MMMMMMMMMMMMMMMMMN :	:d:/yss+o-:/o+	.::++//:::+MMMMMMMMMMMMMMMM
МММММММММММММММS``	+ /:os:/- `	.:::::::::
MMMMMMMMMMMMMMMN:/osss-```	`+oss+so+o	`hd:NMMMMMMMMMMMMM
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM	` // .+` +.	yMMy:/oMMMMMMMMMMMMM
MMMMMMMMMMMMMMMM+sosss/::	/: .s o.	mMMMh:ossssodMMMMMMMMMMM
MMMMMMMMMMMMMMMysssss-`-:///`	+:s o-	hMMMMNoossss+mMMMMMMMMMM
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM		`/sNMMMMy/osssoNMMMMMMMMM
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM	```+` `o /:	.: `+yNMMN+sssssyMMMMMMMMM
MMMMMMMMMMMMMMhssssss:::		./+:.+ohM+ssssooNMMMMMMMMM
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM		:+++/0/sso+:-0MMMMMMMM
M*KID*MMMMMMMMm+ssss/NMMNo	`:osos+::-	./++++/:MMMMMMMM
MMMMMMMMMMMMd+//://+MMMMMNh.`	`/ Guacolda/	-:/++++:``:MMMMMMMM
MMMMMMMMMMMMy/+oo/-sMMMMMMo``	-:/:/+//:/+s`	:::++++/`:MMMMMMMM
MMMMMMMMMMMMy+dmmmMMMMMhhhhhhhhh	hyhhdhhh:`yhhdmm	dhyyssyhdddmmmmddmMmmNMMMM
MMMd:yMMMMMMN mMMMMd``mMMMMMMMMMN	. mMMMMMM ` mMMh:.	``:sMMMMMmo-`.` sMMM
MMM- /mMMMMh -MMMMs `NMMMMMMMMS	-MMMMMd .MMMdh/	/MMMMMo /MMMd: :yNy``hMMM
MMd /y.`oNMMo sMMMM: +MMMMMMMMMM:	YMMMMM+ +MMMMN`	`mMMdosNMMo -dMMMNdNMMMM
MM: dMNs`.hM: mMMMN` dMMMMMMMMm	` hMMMMy	.+-`-odMMMMs -NMMMMMMMMNoy
Md -MMMMm: o`.MMMMy -MMMMMMMMMMo :	dddmmm NMMMM+	.ohNMMMMMMMN` hMMMMMMMd+.+N
M+ oMMMMMMo :MMMM: oNdys++/sM- s	SMMMMMN mMMMM+	/MMMMMMMMMN. yMMMmy+sNMM
M/ yMMMMMMMdohMMMM- `.:/osssmM/.c	MMMMMMM/:mMMMMMy	-MMMMMMMMMMd: ``+ymMMMMMM
ммиммимимимимимимимимимимимимими	IMMMMMMMMMMMMMmm	NMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
Laboratorio Nacional de Computac	ion de Alto Rend	limiento (NLHPC)

Laboratorio Nacional de Computacion de Alto Rendimiento (NLHPC Centro de Nodelamiento Matematico (CMM) Jniversidad de Chile

#### Windows:

- Session - Logging - Terminal - Keyboard - Bell - Features - Window - Appearance - Behaviour - Translation - Selection - Colours - Connection - Co	Basic options for your PuTTY se	ssion
	Specify the destination you want to conne Host <u>Name</u> (or IP address)	ct to Port
	[leftraru.nlhpc.cl] Connection type: ○ Raw_ ○ <u>T</u> elnet ○ Rlogin ● <u>S</u> SH	22
	Load, save or delete a stored session Sav <u>e</u> d Sessions	
	Default Settings .61	Load
Data	1000.017	Sa <u>v</u> e
Proxy Telnet Rlogin		<u>D</u> elete
i∰⊶ SSH I Serial	Close window on exit: Always Never Only on cl	ean exit
About	<u>O</u> pen	<u>C</u> ancel



## ¿Qué es SLURM

- Gestor de recursos
- Administra los recursos de las particiones de Leftraru y Guacolda.
- Gestiona las tareas en ejecución y en espera en el cluster.
- Reserva recursos compartidos.
- Permite la ejecución de tareas hasta por 30 días





#### Obteniendo información de las particiones

sinfo: ver estado de las particiones 

PARTITION AVAIL TIMELIMIT NODES STATE NODELIST

[root@master2 ~]# sinfo

# workload manager

up	infinite	1	drain	cn037
up	infinite	12	mix	cn[023-024,026,045,072-073,079,087,096,107,129,131]
up	infinite	63	alloc	cn[019-020,038-044,108-128,130]
up	infinite	55	idle	cn[001-018,021-022,082-083,086,088-090,132]
up	infinite	9	mix	sn[002,006,014-016,021,028,030-031]
up	infinite	39	alloc	sn[001,003-005,007-013,032-048]
up	infinite	2	mix	fn[001,007]
up	infinite	2	alloc	fn[002,004]
up	infinite	5	idle	fn[003,005-006,008-009]
up	infinite	2	mix	gn[001-002]
up	infinite	4	idle	leftraru[1-4]
	up up up up up up up up	<pre>up infinite up infinite</pre>	upinfinite1upinfinite63upinfinite55upinfinite9upinfinite39upinfinite2upinfinite5upinfinite2upinfinite5upinfinite4	upinfinite1drainupinfinite12mixupinfinite63allocupinfinite55idleupinfinite9mixupinfinite39allocupinfinite2mixupinfinite2allocupinfinite2mixupinfinite4idle





36 Nodos libres Nodos mantención

## squeue: listado de tareas en ejecución y pendientes

#### Monitorear desde la consola:

[usuario@leftraru1 ~]\$ squeue



JOBID	PARTITION	NAME	USER	ST	TIME	NODES NODE	LIST(REASON)
4400799	slims	example	usuario	R	0:00	1	cn042



#### sacct: estados de tareas ejecutadas



[usuario@leftraru1 ~]\$ sacct -X

JobID	JobName	Partition	Account	AllocCPUS	State	ExitCode
24118136	14131-DIA+	slims	users	2	RUNNING	0:0
24118147	14132-DIA+	slims	users	2	RUNNING	0:0
24118148	14133-DIA+	slims	users	2	COMPLETED	0:0
24118154	14137-DIA+	slims	users	2	COMPLETED	0:0



## Información detallada de job

[root@leftraru1 ~]# scontrol -dd show job 9160565

```
JobId=9160565 JobName=w.fepc-f-cnt-oo.m2
UserId=workxwz(11942) GroupId=fisica_cmm(11222) MCS_label=N/A
Priority=109951 Nice=0 Account=unab QOS=120-30-std
JobState=RUNNING Reason=None Dependency=(null)
Requeue=0 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
RunTime=04:28:03 TimeLimit=3-00:00:00 TimeMin=N/A
SubmitTime=2017-10-09T11:25:36 EligibleTime=2017-10-09T11:25:36
StartTime=2017-10-09T15:04:40 EndTime=2017-10-12T15:04:40 Deadline=N/A
PreemptTime=None SuspendTime=None SecsPreSuspend=0
Partition=slims AllocNode:Sid=leftraru4:52235
ReqNodeList=(null) ExcNodeList=(null)
```



#### Enviar trabajos en SLURM





Submitted batch job 9142401

[usuario@leftraru1 ~]\$ squeue

JOBIDPARTITIONNAMEUSERSTTIMENODESNODELIST(REASON)9142401slimspi\_levqurootR0:081cn105

[usuario@leftraru1 ~]\$ scancel 9142401

Cancelar trabajo



## Parámetros en SLURM

Parámetro	Uso	Acción	
-J	-J mi-tarea	Asigna nombre a la tarea	`, <b>;;;;;;;;;;;;</b> ;;;;;;;;;;;;;;;;;;;;;;;;
-р	-p slims	Indica partición a utilizar	SIUrm
-n	-n 1	Nº de procesos	WUI KIUdu II Idi Idyel
-C	-c 20	CPUs por proceso	
ntasks-per-node	ntasks-per-node=20	Procesos por nodo	
mem-per-cpu	mem-per-cpu=2300	Memoria por CPUs	
-0	-o salida_%j.out	Log de salida	
-е	-e errores_%j.err	Log de salida de errores	
-mail-user	-mail-user=user@abc.xyz	Donde se envia info del JOBs	
-mail-type	-mail-user=ALL	Tipo de información a enviar	



## Ejercicio 1

- Ejecute el comando **hostname** en la partición *Slims* con **srun**:
  - Con un único proceso.
  - Con dos procesos iguales.
  - Con dos procesos en distintos nodos.
  - Lanzando un proceso que tenga dos *hilos*.
- ¿Qué resultados se han obtenido?
- En la partición slims
  - ¿Cuántos cores puedo reservar por proceso? ¿Por qué es ese número?
  - ¿Qué diferencia hay en la partición general?
  - ¿Qué ocurre si reservo más cores de los disponibles?
- ¿Qué ocurre si no especifico la partición en la que quiero ejecutar mi comando?





## Ejemplo de script básico SBATCH

Utilizar su editor por consola preferido: nvim, vim, vi, nano

#!/bin/bash #SBATCH -J ejemplo #SBATCH -p slims #SBATCH -n 1 #SBATCH -c 1 #SBATCH -o archivo\_%j.out #SBATCH -e archivo\_%j.err #SBATCH --mail-user=foo@example.org #SBATCH --mail-type=ALL sleep 10

Y ejecutar el script:

sbatch test.sh



## Ejercicio 2

- Crea un *script* de ejecución para lanzarlo con **sbatch**, con las siguientes consideraciones:
  - Utilizar la partición *slims*.
  - Reserva un único core.
  - Ejecuta el comando stress -c 1
- El comando stress sirve para poner a prueba los distintos componentes de un computador. En este ejemplo estamos pidiendo usar una *CPU* (al 100%) durante un tiempo ilimitado. Ya que no se le ha especificado al comando un tiempo de término, en principio, la tarea no debiera terminar nunca. En relación a esto:
  - ¿Cuánto tiempo estará la tarea en ejecución?
  - ¿Qué comando puedo utilizar para obtener información acerca de la tarea en ejecución?
  - ¿Cómo puedo cancelar mi tarea?
- Para poder lanzar tareas, debo conocer el uso del Cluster ¿Qué comando me permite conocer el estado de las particiones?





#### Monitorear Job - htop

Puede ingresar a través de ssh a un nodo en donde tenga una tarea en ejecución y ejecutar **htop** 

1 2 3 4 5 Mem Swp			100. 0. 100. 0.	. 0%] . 0%] . 0%] . 0%]	6 [ 7 [ 8 [ 9 [ 10 [		1.56G, 0K,	0.0% 0.0% 0.7% 0.0% /62.7G /62.5G	%]       11 [       0.0%]       16 [       0.0%]         %]       12 [       0.0%]       17 [       0.0%]         %]       13 [       0.0%]       18 [       0.0%]         %]       14 [       0.0%]       19 [       0.0%]         %]       15 [       0.0%]       20 [       0.0%]         %]       15 [       0.0%]       20 [       0.0%]         %]       15 [       0.0%]       20 [       0.0%]         %]       15 [       0.0%]       20 [       0.0%]         %]       15 [       0.0%]       20 [       0.0%]         %]       15 [       0.0%]       20 [       0.0%]         %]       15 [       0.0%]       20 [       0.0%]         %]       12 [       0.0%]       20 [       0.0%]         %]       13 [       0.0%]       20 [       0.0%]         %]       14 [       0.0%]       20 [       0.0%]         %]       15 [       0.0%]       2.0%]       0.0%]         %]       14 [       0.0%]       0.0%]       0.0%]         %]       14 [       0.0%]       0.0%]       0.0%]
PID	USER	PRI	NI	VIRT	RES		CPU%		TIME+ Command
18305	nperinet	20	0	19020	3768	892 F	100.	0.0	2h52:34 ./LL_RK4.x
18335	nperinet	20	Θ	17104	1584	892 F	100.	Θ.Θ	2h47:30 ./LL_RK4.x
18605	root	20	Θ	121M	2432	1476 F	8 0.0	Θ.Θ	0:00.05 htop
1	root	20	Θ	185M	5068	2384 5	5 0.0	Θ.Θ	0:11.06 /usr/lib/systemd/systemdswitched-rootsystemde
669	root	20	0	112M	2000	1564 S	5 0.0	Θ.Θ	0:00.00 /bin/bash
671	root	20	Θ	36816	7236	6908 S	5 0.0	Θ.Θ	0:00.78 /usr/lib/systemd/systemd-journald
726	root	20	Θ	43808	2428	1268 5	5 0.0	Θ.Θ	0:00.83 /usr/lib/systemd/systemd-udevd
1012	root	16		51188	1620	1236 5	5 0.0	Θ.Θ	0:00.05 /sbin/auditd -n
1002	root	16		51188	1620	1236 5	5 0.0	Θ.Θ	0:00.25 /sbin/auditd -n
1206	root	20	0	448M	10956	6968 S	5 0.0	Θ.Θ	0:00.00 /usr/sbin/NetworkManagerno-daemon
1209	root	20	Θ	448M	10956	6968 S	5 0.0	0.0	0:00.09 /usr/sbin/NetworkManagerno-daemon
1139	root	20	Θ	448M	10956	6968 S	5 0.0	Θ.Θ	0:02.34 /usr/sbin/NetworkManagerno-daemon
1144	avahi	20	Θ	30220	1564	1300 5	5 0.0	Θ.Θ	0:00.44 avahi-daemon: running [cnf004.local]
1148	dbus	20	Θ	28824	1772	1352 5	5 0.0	0.0	0:00.44 /bin/dbus-daemonsystemaddress=systemd:nofork
1166	root	20	0	198M	1236	776 5	5 0.0	0.0	0:00.00 /usr/sbin/gssproxy -D
1167	root	20	0	198M	1236	776 5	5 0.0	0.0	0:00.00 /usr/sbin/gssproxy -D
1168	root	20	0	198M	1236	776 S	0.0	Θ.Θ	0:00.00 /usr/sbin/gssproxy -D
1169	root	20	Θ	198M	1236	776 S	5 0.0	Θ.Θ	0:00.00 /usr/sbin/gssproxy -D
1170	root	20	Θ	198M	1236	776 5	0.0	Θ.Θ	0:00.00 /usr/sbin/gssproxy -D
1164 F1 <mark>Help</mark>	F2Setup	20 F3 <mark>Se</mark>	0 arch	198M F4 <mark>Fil</mark>	1236 ter <mark>F5</mark> Tr	776 S	6 0.0	0.0 F7 <mark>Nice</mark>	0:00.30 /usr/sbin/gssproxy -D e -F8Nice +F9Kill F10Duit



#### Generador de scripts SBATCH



En el siguiente link podrá crear de forma simple sus scripts SBATCH para ser ejecutados en el cluster.

https://wiki.nlhpc.cl/Generador\_Scripts



## Ejercicio 3

- Crea un *script* para lanzarlo con **sbatch**, con las siguientes consideraciones:
  - La partición a lanzar es *slims*.
  - Reserva un nodo completo.
  - Ejecuta el comando **stress** -c 40 -t 10m.
- Verifica en qué nodo se está ejecutando tu tarea, accede mediante ssh al nodo y ejecuta htop.
- ¿Cuántos procesos se están ejecutando?
- ¿Cuál es el porcentaje de uso de cada proceso?
- ¿Cómo sería la manera correcta de lanzar la tarea con el fin de que cada proceso se ejecute al 100%?
- Compara y explica el uso de CPU entre los puntos 3 y 4.





#### Reserva Memoria RAM

Límite en la memoria RAM:

- Por defecto se reserva 1GB de RAM por core reservado
- Si se excede se obtendrá el error: "Exceeded job memory limit"
- Reservar RAM por *core* usado: **#SBATCH** --mem-per-cpu=2300

#### # systemd-cgtop -m | grep job\_id

Path	Tasks	%CPU	Метогу	Input/s	Output/s
/	485	4356.4	23.4G		-
/slurm			14.3G		_
/slurm/uid_2398			14.3G		-
/slurm/u8/job_24117526			14.3G		-
/system.slice		4354.2	8.4G		-



## Ejercicio 4

- Crea un *script* para lanzarlo con **sbatch**, con las siguientes consideraciones:
  - La partición a lanzar es *slims*.
  - No asignar memoria RAM
  - Reserva un único core.
  - Debe enviar un correo electrónico cuando la tarea cambie de estado.
  - Ejecutar stress -m 1 --vm-bytes 2048M -t 15m
- Revisa los log de salida
- ¿Qué ocurre con la ejecución? ¿Cuál es la razón?
- Modifica el script para ejecutar la tarea.
- ¿Cuántos recursos de RAM está utilizando la tarea?





#### Sistema de Módulos LMOD

- Permite tener diferentes aplicaciones y versiones de estos en un mismo sistema operativo.
- En el NLHPC usamos Lmod (<u>https://github.com/TACC/Lmod</u>).
- El actual sistema de módulos está disponible para las distintas arquitecturas de procesador (AVX512, AVX, SSE4.2)



## Lmod: Buscar módulo

eguerra@leftraru1:/home/eguerra\$ ml spider Python
Python:
Description: Python is a programming language that lets you work more quickly and integrate your systems more effectively.
Versions: Python/2.7.15 Python/3.7.2 Python/3.7.3 Other possible modules matches: Biopython IPython protobuf-python
To find other possible module matches execute:
\$ module -r spider '.*Python.*'
For detailed information about a specific "Python" module (including how to load the modules) use the module's full name. For example: \$ module spider Python/3.7.3

#### Lmod: Cargar distintas versiones

equerra@leftraru1:/home/equerra\$ ml Pvthon/3.7.3 equerra@leftraru1:/home/equerra\$ ml Currently Loaded Modules: 1) GCCcore/8.2.0 4) impi/2019.2.187 7) intel/2019b 10) libreadline/8.0 13) SOLite/3.27.1 16) libffi/3.2.1 2) icc/2019.2.187-GCC-8.2.0-2.31.1 5) imkl/2019.2.187 8) bzip2/1.0.6 11) ncurses/6.1 14) XZ/5.2.4 17) Pvthon/3.7.3 3) ifort/2019.2.187-GCC-8.2.0-2.31.1 6) binutils/2.32 9) zlib/1.2.11 12) Tcl/8.6.9 15) GMP/6.1.2 eguerra@leftraru1:/home/eguerra\$ python -V Pvthon 3.7.3 eguerra@leftraru1:/home/eguerra\$ ml Python/2.7.15 The following have been reloaded with a version change: 1) Pvthon/3.7.3 => Pvthon/2.7.15 equerra@leftraru1:/home/equerra\$ ml Currently Loaded Modules: 1) GCCcore/8.2.0 4) impi/2019.2.187 7) intel/2019b 10) libreadline/8.0 13) SOLite/3.27.1 16) libffi/3.2.1 2) icc/2019.2.187-GCC-8.2.0-2.31.1 5) imkl/2019.2.187 8) bzip2/1.0.6 11) ncurses/6.1 14) XZ/5.2.4 17) Python/2.7.15 3) ifort/2019.2.187-GCC-8.2.0-2.31.1 6) binutils/2.32 9) zlib/1.2.11 12) Tcl/8.6.9 15) GMP/6.1.2 eguerra@leftraru1:/home/eguerra\$ python -V Pvthon 2.7.15 lational Laboratory for High Performance

> Computing Chile

## Ejercicio 5

- Descarga el siguiente código Python con wget: <u>n-queens-problem-3.py</u> en tu directorio de trabajo.
- Crea un *script* de para lanzarlo con **sbatch**, con las siguientes consideraciones:
  - Utilizar la partición *slims*.
  - Cada trabajo reserva un único core.
  - Supondremos que cada trabajo reserva 2300 Mb de RAM.
  - Ejecuta el código con la versión de Python/3.9.5



#### Ver cuota de disco

[\$USER@leftraru1 ~]# usoDisco

Uso de disco del usuario: \$USER

Cuota = 200G

Utilizado = 148.95G

% de utilización = 74.5%



#### Eficiencia Computacional

- Comportamiento de un programa al ejecutarlo de manera paralela (más de una CPU)
- Un programa puede escalar en un rango de procesadores [1..n]
- Se logra la eficiencia cuando la medición se mantiene constante sobre un factor de 0,5
- Esto es importante, ya que un programa no se ejecutará en la mitad de tiempo si se ejecuta en un doble de procesadores.
  - Buscamos hacer un uso óptimo y eficiente de los recursos



#### Eficiencia Computacional - Speedup y Eficiencia

- **SpeedUp** es la métrica que nos indica la ganancia mediante la paralelización:
  - SpeedUp = Tiempo Original / Tiempo Mejora
- Eficiencia es la métrica del uso de los recursos computacionales
  - Eficiencia = SpeedUp / Número de CPU

Procesadores	Tiempo de Ejecución	Speedup	Eficiencia
1	1:00:27	1,0	1,0
2	0:33:47	1,8	0,9
4	0:18:02	3,4	0,8
8	0:09:13	6,6	0,8
16	0:05:06	11,9	0,7
32	0:02:31	24,0	0,8
64	0:01:18	46,5	0,7
128	0:00:48	75,6	0,6
256	0:00:34	106,7	0,4
500	0:00:31	117,0	0,2
600	0:00:35	103,6	0,2
700	0:00:35	103,6	0,1

https://wiki.nlhpc.cl/Escalamiento



#### Eficiencia Computacional - Speedup y Eficiencia - Gráfico

#### Speedup y Eficiencia





## Uso de partición GPUS

- ¿Cómo ejecutar un programa mediante Slurm que necesite utilizar la partición gpus?
  - ¿Qué opciones existen actualmente en el Cluster?
  - ¿Qué módulos se deben cargar para utilizar la partición gpus?

```
#!/bin/bash
#SBATCH -J gpu-example
#SBATCH -p gpus
#SBATCH -n 1
#SBATCH -c 1
#SBATCH --gres=gpu:1
#SBATCH --mem-per-cpu=4250
ml purge
ml fosscuda/2019b
```

ml NAMD/3.0alpha9



## Enlaces de interés

Los invitamos a visitar nuestra página web:

www.nlhpc.cl

También tenemos una wiki pública con información útil:

https://wiki.nlhpc.cl/Bienvenida\_NLHPC

Solicitud de cuentas de usuarios

https://solicitudes.nlhpc.cl/

Dashboard

https://dashboard.nlhpc.cl/

En caso de dudas, pueden escribirnos a soporte@nlhpc.cl





# ¡Gracias por participar! :)

www.nlhpc.cl 2022

