Para conectarte a boada: “ssh -X [par1311@boada.ac.upc.edu](mailto:par1311@boada.ac.upc.edu)" passwd: Noo3awoth4mu

boada-6 a boada-9 INTERACTIVO

boada-11 a boada-14 EJECUCIÓN EN COLA

“source ~/environment.bash” SIEMPRE QUE SE INICIA

EN lab1/arch : sbatch submit-arch.sh

This script will execute the lscpu and lstopo commands in order to obtain information about the hardware in one of the nodes of execution queue (boada-11 to 14). pag5

* Queueing a job for execution: "sbatch [-p partition] ./submit-xxxx.sh" .

Use "squeue" to ask the system about the status of your job submission. You can use "scancel" followed by the job identifier to remove a job from the queueing system.

* Interactive execution: “./run-xxxx.sh”

Hacer los makefiles de los archivos que quieras, por ejemplo pi\_seq.c

run-seq.sh INTERACTIVA

submit-seq.sh

COMANDO more SI QUIERES MÁS INFORMACIÓN

1 sesion lab1

* Conectar a boada -> "ssh -X [parXXYY@boada.ac.upc.edu](mailto:parXXYY@boada.ac.upc.edu)"
* Copiar sesion de lab -> $ cp /scratch/nas/1/par0/sessions/lab1.tar.gz .
* Descomprimir -> "tar -zxvf lab1.tar.gz"
* Crear .bashrc -> $ nano .bashrc -> “source environment.bash” -> $ source .bashrc
* Copiar de boada a maquina local -> "scp parXXYY@boada.ac.upc.edu:lab1/pi/pi seq.c ." -> current directory, represented with the "."
* Consultar información de la arquitectura de boada -> sbatch submit-arch.sh command in directory lab1/arch -> ejecuta lscpu y lstopo -> genera 3 ficheros -> 1) lscpu-boada-number, 2) lstopo-boada-number, 3) map-boada-number.fig -> xfig map-boada-number.fig para visualizar la arquitectura de boada
* Poner en cola un trabajo para su ejecución -> "sbatch [-p partition] ./submit-xxxx.sh" -> usar “squeue” to ask the system about the status of your job submission -> usar "scancel" followed by the job identifier to remove a job from the queueing system
* Ejecución interactiva -> ./run-xxxx.sh
* Mirar el tiempo transcurrido al ejecutar en cola un script -> Mirar el archivo generado despues de la ejecución “time-pi seq-boada-Y”, being Y the node where the execution happened -> si es paralelizable (OpenMP)
* Comandos para analizar la escalabilidad -> para strong scalability “submit-strong-omp.sh” y para weak scalability “submit-weak-omp.sh” -> Use the ghostscript gs command to visualise the Postscript file generated

Sockets = NUMAnode

—-------------------------------

3 sesion lab1

geany 3dfft\_omp.c

sbatch submit-strong-extrae.sh 3dfft\_omp

cd 3dfft\_omp-strong-extrae/

mv 3dfft\_omp-strong-extrae nombre\_a\_renombrar\_v0

xpdf modelfactor-tables.pdf

wxparaver