

2º curso / 2º cuatr.

Grado en  
Ing. Informática

# Arquitectura de Computadores

## Seminario 0. Entorno de programación: atcgrid y gestor de colas

Material elaborado por los profesores responsables de la asignatura:

Mancia Anguita – Julio Ortega

*Licencia Creative Commons*



ugr

Universidad  
de Granada

ETSIIT

Escuela Técnica Superior  
de Ingenierías Informática  
y de Telecomunicación



ATC

Departamento de Arquitectura  
y Tecnología de Computadores  
UNIVERSIDAD DE GRANADA



# Contenidos

- Cluster de prácticas (atcgrid)
- Sistema de colas
- Ejemplo de script

# Contenidos

- Cluster de prácticas (atcgrid)
  - Componentes
  - Placa madre
  - Chip de procesamiento (procesador)
  - Acceso
- Sistema de colas
- Ejemplo de script

# Cluster de prácticas (atcgrid): componentes



Cables



**Nodos de cómputo (tres):** Servidores rack  
SuperMicro SuperServer 6016T-T

<http://www.supermicro.com/products/system/1U/6016/SYS-6016T-T.cfm>

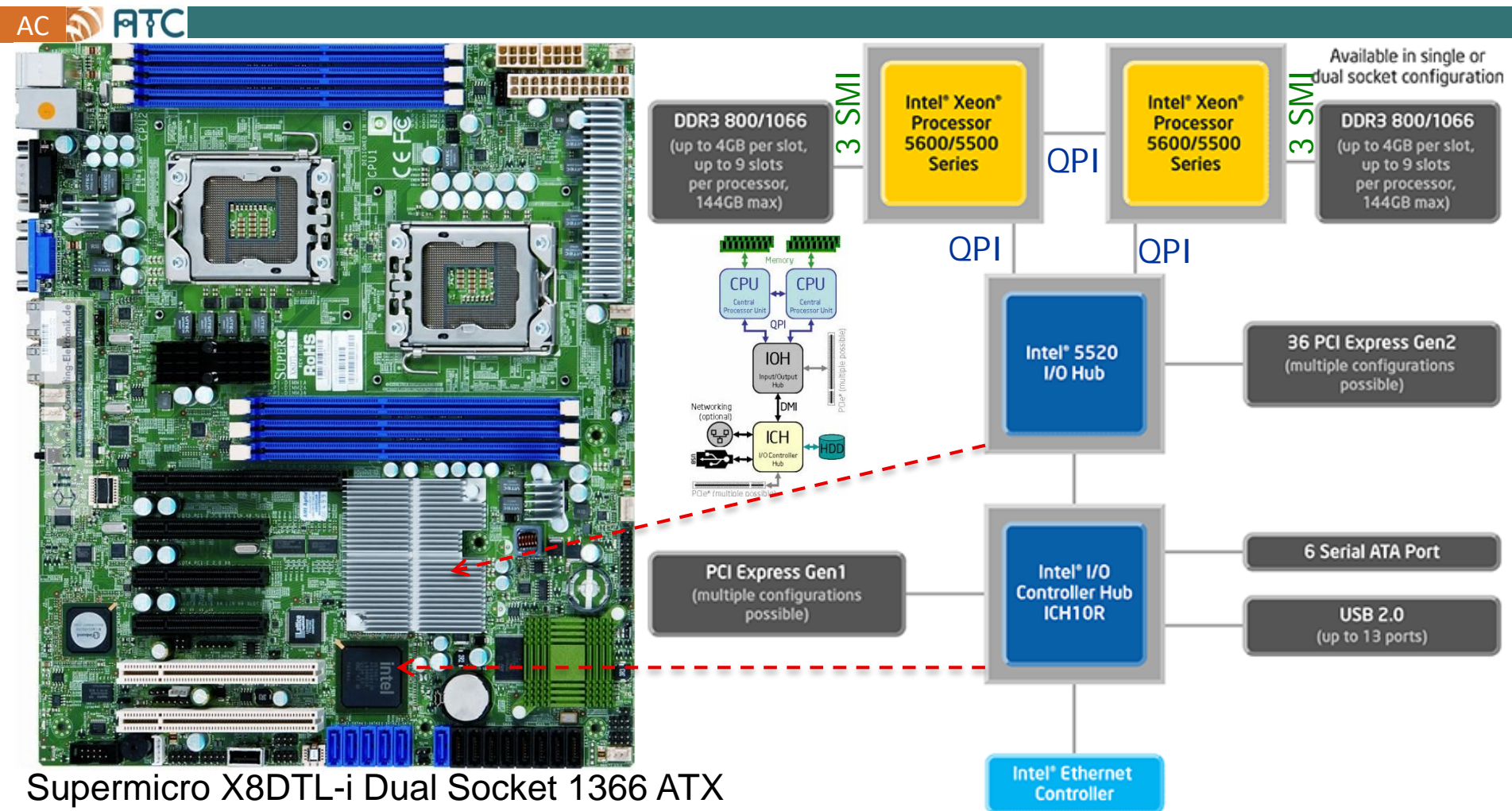


Switch



*Nodo front-end  
(host, master)*

# Cluster de prácticas (atcgrid): placa madre de un nodo



Supermicro X8DTL-i Dual Socket 1366 ATX

Server Mainboard Intel 5520 chipset

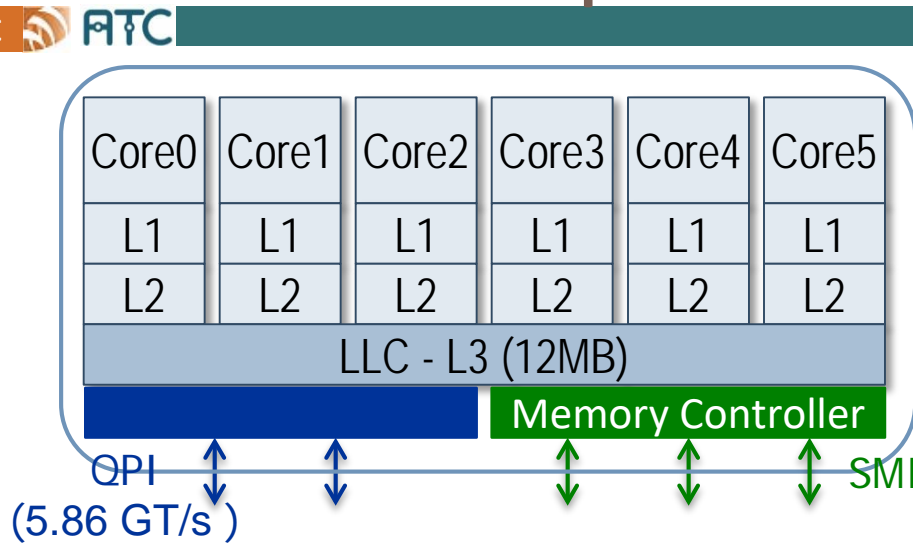
<http://www.supermicro.com/products/motherboard/QPI/5500/X8DTL-i.cfm>

Intel® 5520 Chipset

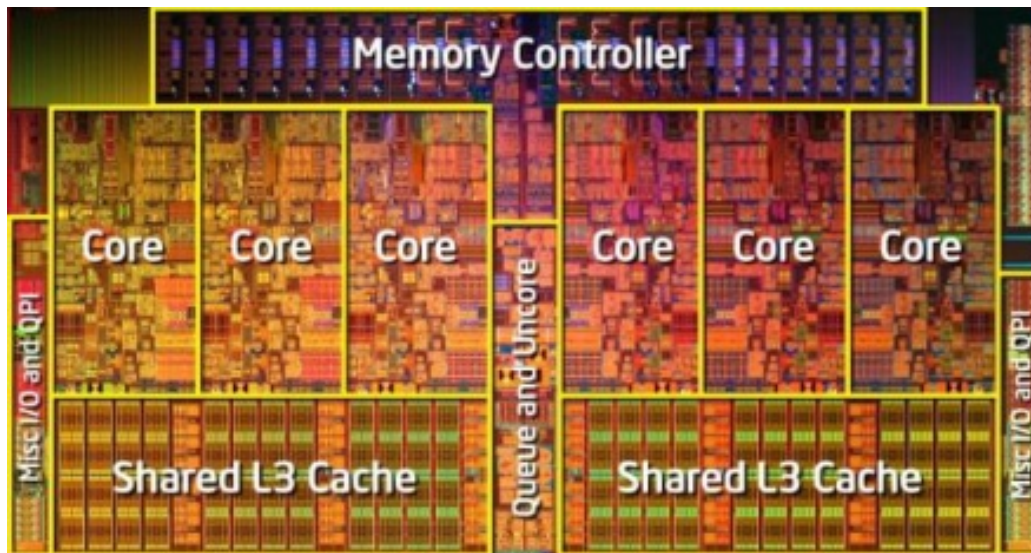
[http://www.intel.com/p/en\\_US/embedded/hwsw/hardware/xeon-5600-5500/overview](http://www.intel.com/p/en_US/embedded/hwsw/hardware/xeon-5600-5500/overview)



# Cluster de prácticas (atcgrid): chip de procesamiento



6 cores  
hyperthreading  
2.4 GHz (12  
threads)



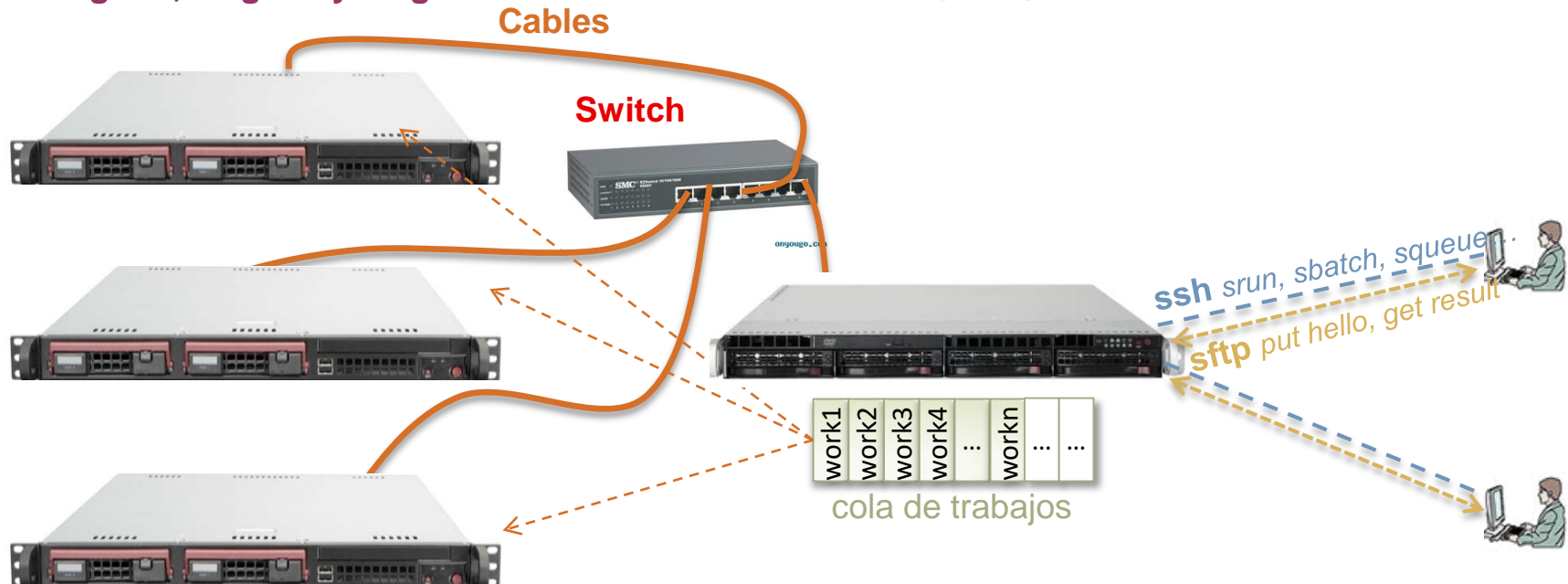
Intel Xeon E5645 (6 cores/12 threads, 12M L3 Cache compartida, 2.40 GHz cada core, 5.86 GT/s Intel® QPI)  
[http://ark.intel.com/products/48768?wapkw=\(E5645\)](http://ark.intel.com/products/48768?wapkw=(E5645))

# Cluster de prácticas (atcgrid): acceso

**Nodos de cómputo:**  
atcgrid1, atcgrid2 y atcgrid3

**Nodo front-end:**  
atcgrid.ugr.es

**usuarios**



- Cada **usuario** tiene un home en el nodo **front-end** del **clúster atcgrid**. Se puede acceder al home:
  - Para ejecutar comandos (srun, sbatch, squeue...), con un cliente **ssh** (**secure shell**):
    - Linux: \$ `ssh -X username@atcgrid.ugr.es` (pide *password* del usuario “username”)
  - Para cargar y descargar ficheros (put hello, get slurm-9.out, ...), con un cliente **sftp** (**secure file transfer protocol**):
    - Linux: \$ `sftp username@atcgrid.ugr.es` (pide *password* del usuario “username”)

# Contenidos

- Cluster de prácticas (atcgrid)
- Sistema de colas
- Ejemplo de script



# Ejemplos con comandos slurm

Ejemplo	Explicación
<pre><b>srun</b> -p ac ./hello <b>srun</b> -p ac lscpu</pre>	srun envía a ejecutar un trabajo (en los ejemplos, el ejecutable hello, y lscpu) a través de una cola slurm (se ha especificado la cola con -p).
<pre><b>sbatch</b> -p ac script.sh <b>sbatch</b> -p ac --wrap "echo Hola" <b>sbatch</b> -p ac --wrap "./hello" <b>sbatch</b> -p ac --wrap "echo Hola ; ./hello"</pre>	sbatch envía a ejecutar un <i>script</i> (en este caso script.sh, "echo Hola", "./hello" y "echo Hola ; ./hello") a través de una cola slurm. La salida se devuelve en un fichero. La ejecución con srun es <i>interactiva</i> , con sbatch es en <i>segundo plano</i> . Se recomienda usar sbatch
<pre><b>squeue</b></pre>	Muestra todos los trabajos en ejecución y los que están encolados
<pre><b>scancel</b> jobid</pre>	Elimina el trabajo con identificador "jobid"
<pre><b>sinfo</b></pre>	Lista información de las colas (particiones) y de los nodos

Tabla resumen: <https://slurm.schedmd.com/pdfs/summary.pdf>

Páginas de manual: [https://slurm.schedmd.com/man\\_index.html](https://slurm.schedmd.com/man_index.html)

# Contenidos

- Cluster de prácticas (atcgrid)
- Sistema de colas
- Ejemplo de script

# Ejemplo hello OpenMP

## HelloOMP.c

- Cada *thread* imprime su identificador
- El identificador se obtiene con la función OpenMP `omp_get_thread_num( )`

```
/* Compilar con:
gcc -O2 -fopenmp -o HelloOMP HelloOMP.c
*/
#include <stdio.h>
#include <omp.h>

int main(void) {

#pragma omp parallel
    printf("( %d:!!!Hello world!!!)",
           omp_get_thread_num());

return(0);

}
```

# Script para la ejecución del ejemplo HelloOMP en atcgrid

## script\_helloomp.sh

```
#!/bin/bash
#Órdenes para el sistema de colas:
#1. Asigna al trabajo un nombre
#SBATCH --job-name=helloOMP
#2. Asignar el trabajo a una cola (partición)
#SBATCH --partition=ac
#2. Asignar el trabajo a un account
#SBATCH --account=ac

#Obtener información de las variables del entorno del sistema de colas:
echo "Id. usuario del trabajo: $SLURM_JOB_USER"
echo "Id. del trabajo: $SLURM_JOBID"
echo "Nombre del trabajo especificado por usuario: $SLURM_JOB_NAME"
echo "Directorio de trabajo (en el que se ejecuta el script):"
$SLURM_SUBMIT_DIR"
echo "Cola: $SLURM_JOB_PARTITION"
echo "Nodo que ejecuta este trabajo:$SLURM_SUBMIT_HOST"
echo "Nº de nodos asignados al trabajo: $SLURM_JOB_NUM_NODES"
echo "Nodos asignados al trabajo: $SLURM_JOB_NODELIST"
echo "CPUs por nodo: $SLURM_JOB_CPUS_PER_NODE"
#Instrucciones del script para ejecutar código:
echo -e "\n 1. Ejecución helloOMP una vez sin cambiar nº de threads (valor por defecto):\n"
srun ./HelloOMP
echo -e "\n 2. Ejecución helloOMP varias veces con distinto nº de threads:\n"
for ((P=12;P>0;P=P/2))
do
    export OMP_NUM_THREADS=$P
    echo -e "\n - Para $P threads:"
    srun ./HelloOMP
done
```

Órdenes para  
sistema de  
colas

Para imprimir  
variables del  
sistema de colas

Instrucciones del  
script

No olvidar poner **srun** delante del ejecutable

# Utilidades

- Formateo de código a insertar en los cuadernos:  
<http://qbnz.com/highlighter/index.php>