

# MPI i OpenMP

---

Autor: Meriem Abjil Bajja

2018

# INDEX

## Contingut

MPI i OpenMPI .....	3
Instal·lació MPI i OpenMPI en Ubuntu .....	4
Instal·lació MPI.....	4
Instal·lació OpenMPI.....	5
Compilar i executar .....	7

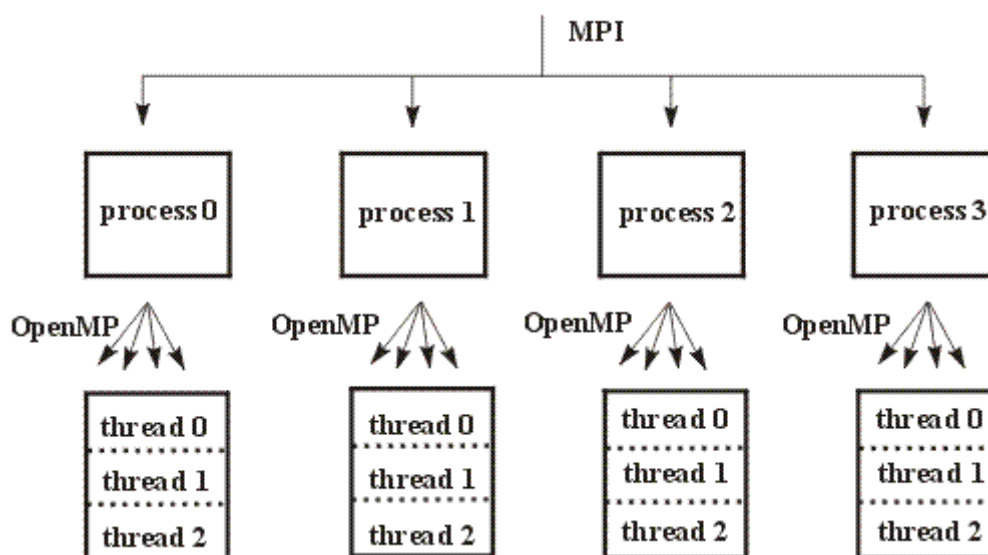
### MPI i OpenMP

**MPI (Message Passing Interface)** és un estàndard que defineix la sintaxi i la semàntica de les funcions contingudes en una biblioteca de pas de missatges dissenyada per ser usada en programes que explotin l'existència de múltiples processadors.

El pas de missatges és una tècnica empleada en programació concurrent per aportar sincronització entre processos i permetre l'exclusió mútua. La seva principal característica és que no precisa de memòria compartida, per la qual cosa és molt important a la programació de sistemes distribuïts.

**OpenMP (Open Multi-Processing)** és una interfície de programació d'aplicacions (API) per a la programació multiprocés de memòria compartida en múltiples plataformes. També podria definir-se com un model de programació portable i escalable que proporciona als programadors una interfície simple i flexible per al desenvolupament d'aplicacions paral·leles per a les plataformes que van des de les computadores d'escriptori fins als supercomputadors.

L'ús de **programes híbrids amb MPI i OpenMP** tracten de fer un ús més eficaç i eficient d'un Cluster d'alt rendiment, el primer que s'ha d'observar és que el compilador de suport tant a OpenMP com a MPI. A partir de GCC 4.2 es dona suport a OpenMP pel que alguns Clusters hauran d'escalar la seva versió del compilador.



La imatge mostra que MPI s'utilitza per a la passada de missatges entre els nodes del Cluster i OpenMP utilitza la memòria compartida dels processadors amb l'ús de fils, el numero de fils va d'acord al numero de processadors de determinat node.

# Instal·lació MPI i OpenMPI en Ubuntu

---


## Instal·lació MPI


1. `sudo apt-get update`
2. `sudo apt-get upgrade`
3. Instal·larem el gestor de paquets Synaptic amb la següent comanda:  
  
**`sudo apt-get install synaptic`**
4. Obrirem el gestor de paquets Synaptic
5. A partir del gestor de paquets Synaptic, instal·lem els següents paquets.

 Binutils

 Automake

 Autoconf

 g++


 gcc

 m4

 libtool

 console

 kdevelop

 libopenmpi1.6-dbg

6. Cal tenir en compte que pot ser necessari instal·lar altres paquets necessaris per instal·lar els paquets descrits anteriorment.
7. `sudo apt-get install libopenmpi-dev openmpi-bin openmpi-doc`
8. `sudo apt-get install ssh`
9. **`ssh-keygen -t dsa`** A l'executar la funció ens indicarà que posem el nom del fitxer (id\_dsa)

10. En situarem a la ruta on hàgim posat el nom del fitxer

```
cd ~/.ssh
```

11. cat id\_dsa.pub

12. cat id\_dsa.pub >> authorized\_keys

## Instal·lació OpenMPI

1. Descarrega l'última versió de OpenMPI des de la web oficial:

<http://www.open-mpi.org/software/ompi>

2. Aquí podrem descarregar la versió com a tar.gz, tar.bz2 o rp. Ens descarregarem el tar.gz

3. Descomprimir l'arxiu descarregat (hauria de ser cridat a alguna cosa com openmpi-x.x.x.tar.xxx, canviant x.x.x per la versió descarregada):

```
tar -xvf openmpi-*
```

4. Anem a dintre de la carpeta creada de la descompressió:

```
cd openmpi-*
```

5. Configura el fitxer d'instal·lació (fes servir els privilegis de superuser en el teu sistema) (Aquesta configuració pot tardar bastants minuts.)

És necessari afegir en el prefix el directori d'instal·lació que volem fer servir per OpenMPI. Lo normal per fer seria seleccionar el següent directori «/home/<user>/.openmpi».

```
./configure --prefix="/home/$USER/.openmpi"
```

6. Ara procedirem a instal·lar-ho i ho farem a partir de l'eina make

```
make
```

```
sudo make install
```

7. Tot el que queda per fer és incloure el path al nostre path environment i a la biblioteca de variables d'entorn, les llibrerias de la instal·lació realitada:

```
export PATH="$PATH:/home/$USER/.openmpi/bin"
```

## Computació i altres prestacions

---

```
export LD_LIBRARY_PATH="$LD_LIBRARY_PATH:/home/$USER/.openmpi/lib/"
```

8. Si vols que l'exportació pugui ser eficaç per les pròximes sessions i terminals, hauràs d'escriure les exportacions en l'arxiu de la variable d'entorn . Per defecte, hauria de ser `"/home /<user>/.bashrc"` Per bash usuaris.

```
echo export PATH="$PATH:/home/$USER/.openmpi/bin" >> /home/$USER/.bashrc
```

```
echo export LD_LIBRARY_PATH="$LD_LIBRARY_PATH:/home/$USER/.openmpi/lib/" >>  
/home/$USER/.bashrc
```



### Compilar i executar

---

El programa que realitzem tant podria ser un programa amb MPI, un programa amb OpenMPI o un programa híbrid amb MPI i OpenMPI. Segons el que fem servir, haurem de compilar i executar d'una manera o d'un altre.

- ✚ Per compilar un programa amb MPI i OpenMPI, cal executar la següent comanda:

```
mpicc -fopenmp -o <executable> <font>
```

on **executable** és el nombre del fitxer de sortida i **font** és el programa en C a compilar.

- ✚ Per executar un programa amb MPI i OpenMPI, cal executar la següent comanda:

```
mpirun -np <#processos> <executable>
```

on **#processos** és el número decimal dels processos que vol executar l'**executable** és l'arxiu de sortida al compilar-ho amb mpicc.