

# Instalación de MPE

(*MPI Parallel Environment*)

*Programación Distribuida y Paralela*



Universidad de Granada



Departamento del LSI

Por:

Daniel Guerrero Martínez  
Sergio Rodríguez Lumley

# Índice de contenido

Introducción.....	3
Requerimientos previos a la instalación.....	3
Instalación de MPE2.....	4
Uso de MPE y Herramientas de visualización.....	5



## Introducción

MPE (Mpi Parallel Environment) es un visualizador de la ejecución de un programa basado en MPI. Este programa genera la información necesaria durante la ejecución para después poder leerla e interpretarla como texto o gráficamente. **No es capaz de compilar ni ejecutar código C++.**

La página principal de documentación, como referencia es [www.mcs.anl.gov](http://www.mcs.anl.gov). En la cual podréis encontrar (poca) información y los enlaces de descarga pertinentes (detallados en este documento también, así que no debería ser necesaria la ayuda de la página mencionada).

La herramienta MPE viene integrada con la implementación de MPI de MPICH (versiones 1 y 2). Nosotros vamos a estudiar la herramienta MPE2, que es su versión más moderna.

## Requerimientos previos a la instalación

- Será necesario tener instalada una implementación de MPI, en concreto, para esta asignatura utilizaremos OpenMPI.
- Tener instalada una máquina virtual de Java versión 6 o superior (Entorno de ejecución “jre” y herramientas de desarrollo “jdk”).

## Instalación de MPE2

El proceso de instalación es a través de consola, requiere un uso mínimo de consola, aunque pondremos todos los comandos para facilitar la instalación.

1. Descargar MPE2 del enlace <ftp://ftp.mcs.anl.gov/pub/mpi/mpe/mpe2.tar.gz>.
2. Descomprimir el archivo.

```
tar -xvf mpe2.tar.gz
```

3. Ingresamos en la carpeta que se habrá creado

```
cd mpe2*
```

4. Una vez llegamos a este punto hay ciertos datos que debemos recopilar:
  - a) El ejecutable de MPICC (Compilador para MPI), si instalamos desde repositorios la ruta será `/usr/bin/mpicc`, sino será `directorio_de_instalacion/mpicc`.
  - b) La carpeta de inclusión de cabeceras de MPI, si instalamos desde repositorios la ruta por defecto es `/usr/include/openmpi/`, sino, será `directorio_de_instalacion/include/`.
  - c) La carpeta de librerías de MPI, si instalamos desde repositorios la ruta por defecto será `/usr/lib/openmpi/lib/`, en otro caso `directorio_de_instalacion/lib/`.
  - d) El directorio de instalación de Java, si lo hemos hecho desde repositorios y hemos instalado la máquina de "Sun Microsystems (Oracle)", el directorio por defecto es `/usr/lib/jvm/java-6-sun-1.6.x.x/` donde X son las versiones, recordamos que debe ser una versión 6 o superior.
5. Si ya hemos recopilado la información necesaria, ejecutaremos la configuración con los siguientes parámetros:
  - a) `--enable-f77=no --with-mpicc="/directorio_de_instalacion/mpicc"`
  - b) `--with-mpiinc="-I/directorio_de_instalacion/openmpi/include"`
  - c) `--with-mpilibs="-L/directorio_de_instalacion/lib/"`
  - d) `--with-java="directorio_de_lib_java/java-6-sun-1.6.x.x/"`
    - Si instalamos todo a través de repositorios (Es necesario especificar correctamente la carpeta de instalación de java)

```
./configure --enable-f77=no --with-mpicc="/usr/bin/mpicc" --with-mpiinc="-I/usr/include/openmpi/" --with-mpilibs="-L/usr/lib/openmpi/lib/" --with-java="/usr/lib/jvm/java-6-sun-1.6.0.15/"
```

- Si hicimos la instalación manual, siguiendo nuestro manual de instalación (con ruta de instalación `/home/$USER/.openmpi/`), hay que especificar la ruta de instalación de java.

```
./configure --enable-f77=no --with-mpicc="/home/$USER/.openmpi/bin/mpicc" --with-mpiinc="-I/home/$USER/.openmpi/include" --with-mpilibs="-L/home/$USER/.openmpi/lib/" --with-java="directorio_de_lib_java/java-6-sun-1.6.x.x/"
```

- Entonces preparamos la compilación y compilamos, haciendo uso del superusuario (sudo, en este caso).

```
sudo make  
sudo make install
```

Con esto ya tenemos instalada la herramienta.

## Uso de MPE y Herramientas de visualización.

Ahora vamos a aprender a utilizar brevemente la herramienta.

- Compilamos el código que vamos a analizar con MPE.

```
mpecc -mpilog p1.c -o ejecutable
```

- Una vez compilado, lo ejecutamos de forma normal.

```
mpirun -np 4 ejecutable
```

- Al ejecutarlo, nos habrá creado un fichero de salida con el mismo nombre del ejecutable pero con la extensión añadida (".clog2"), que es el formato que lee la herramienta de visualización jumpshot. La abrimos con esta.

```
jumpshot ejecutable.clog2
```

Nos pedirá convertir el fichero a un formato legible, aceptamos y observamos su salida.