

OpenMPI al clúster de PS3

Joan V. Puertos

juanvi.puertos@gmail.com

Index

Introducció.....	1
Obtenció del software i desempaquetament.....	1
Configuració, compilació e instal·lació.....	2
Modificació de l'entor d'usuari.....	3
Afegir ruta a variable PATH.....	3
Accés SSH sense password.....	3
Execució d'un procés distribuït al cluster PS3 amb OpenMPI.....	3
Treball futur, alternatives.....	3
Referències.....	3

Introducció

Al present document es descriu un procediment relativament senzill per a compilar e instal·lar el *framework* [OpenMPI](#) (versió 1.4.4) per al [clúster](#) de PlayStation3 al departament d'informàtica de la Universitat de València. La compilació es realitza de forma nativa, i els binaris resultants s'instal·len a l'espai de disc del alumne, es presenta també un exemple d'execució d'un programa distribuït amb repartiment dinàmic de les tasques que calcula el conjunt de Mandelbrot.

Obtenció del software i desempaquetat

El primer pas que hem de fer, es accedir remotament a alguna de les PS3 del departament, una vegada fet açò hem de descarregar les fonts des de el servidor oficial, afortunadament no cal instal·lar-hi cap dependència extra. Per a descarregar l'última versió d'OpenMPI i desempaquetar-la, tan sols cal executar els següents comandaments:

```
$ mkdir OPMI
$ cd OPMI
$ wget http://www.open-mpi.org/software/ompi/v1.4/downloads/openmpi-1.4.4.tar.bz2
$ tar -xvjf openmpi-1.4.4.tar.bz2
```

Configuració, compilació e instal·lació

Una vegada desempaquetades les fonts, podem configurar el procés de compilació, afortunadament OpenMPI es basa en [autotools](#) per a la seva construcció, el que facilita i oferix gran flexibilitat. Arribats a aquest punt, hem d'estar ben segurs que estem connectats a alguna PS3, ja que anem a realitzar la compilació nativa del paquet OpenMPI, es podria fer compilació creuada si ho desitjarem, ja que *autotools* ens ho permet però no es l'objectiu del present document explicar la compilació creuada. Ara detallem primerament cada paràmetre que cal que passem al script de configuració de la compilació:

--prefix=/home/alu/ii/johndoe/OMPI/ompi_build_nodebug : Indiquem al script de configuració, que volem instal·lar els binaris al directori que li passem com argument. Si no indicàrem açò, l'ordre `make install`, intentaria instal·lar-hi al directori arrel, però com no tenim permisos per fer això, cal que indiquem on volem instal·lar-hi el software. Com podeu veure, **hem creat** el directori `ompi_build_nodebug`. No utilitzeu mai rutes relatives per a indicar el *prefix*, i empreu el vostre nom d'usuari en lloc del de *johndoe*.

--enable-mpirun-prefix-by-default: Indica a les ferramentes d'OpenMPI que deuen buscar els binaris tant de la llibreria, com de la resta de ferramentes, al directori indicat per la variable de compilació *prefix*. D'aquesta forma evitem que busquen a `/usr/lib` o d'altres rutes que podrien contindre un altra versió d'OpenMPI o bé uns binaris d'algun altra arquitectura (per exemple x86).

--enable-picky: Permet al compilador escollir les optimitzacions que considere adients.

--disable-mpi-f77: No volem els bindings per Fortran77.

--disable-mpi-f90: Tampoc volem els bindings per Fortran90.

Aleshores, el comandament per a llançar el procés de configuració quedarà com:

```
$ ./configure --prefix=/home/alu/ii/johndoe/OMPI/ompi_build_nodebug --enable-mpirun-prefix-by-default --enable-picky --disable-mpi-f77 --disable-mpi-f90
```

Quan el procediment acabe després d'uns minuts, podem passar a compilar. Com el processador *PowerPC* de la PS3 disposa de doble nucli, podem compilar indicant al *make* que pot realitzar dos tasques en paral·lel (paràmetre `-j <num>`).

```
$ make -j 2
```

Finalment, instal·lem els binaris:

```
$ make install
```

Modificació de l'entorn d'usuari

Després d'haver instal·lat OpenMPI, cal modificar lleugerament l'entorn d'usuari per a permetre que el sistema hi trobe les ferramentes d'OpenMPI (com el compilador `mpicc`, per exemple). Per altra banda cal afegir una llista de *hosts* autoritzats per a poder accedir a aquestos sense haver-hi d'introduir cap password.

Afegir ruta a variable PATH

Per a que el sistema pugui trobar les ferramentes d'OpenMPI que tenim instal·lades al nostre espai en disc, tan sols em d'afegir la ruta a la variable PATH, al nostre fitxer `~/.profile` de forma que quede com:

```
PATH=$PATH:$HOME/bin:$HOME/OMPI/ompi_build_nodebug/bin
```

Com podeu veure, la ruta que hem afegit es l'última a la llista de la variable d'entorn PATH, i per tant serà l'últim lloc on el sistema busque els binaris d'OpenMPI, d'aquesta forma si connectem a alguna màquina amb diferent arquitectura a PowerPC que tinga alguna instal·lació d'OpenMPI no hi interferirem.

Accés SSH sense password

Com que OpenMPI necessita accedir per SSH per a poder llançar els processos, cal tindre accés a les màquines, i també cal poder fer-ho sense haver-hi d'introduir la password.

A l'entorn LDAP del campus, podem simplement accedir a cada una de les PS3 des de algun host com ara be **slabii** per a que s'afegisquen al fitxer `~/.ssh/known_hosts`

Si teniu altres necessitats relacionades amb l'acces ssh sense password, podeu pegar una ullada al següent article: http://www.csua.berkeley.edu/~ranga/notes/ssh_nopass.html

Execució d'un procés distribuït al clúster PS3 amb OpenMPI

Finalment gaudim d'un entorn de desenvolupament per a MPI al clúster de PS3. Podem invocar al compilador `mpicc` des de qualsevol PS3 per a compilar qualsevol software que necessite MPI.

Llista de hosts

Hi ha moltes maneres de gestionar la distribució dels processos al clúster, però nosaltres hem optat per simplement indicar a `mpirun` una llista de màquines a les que podem accedir (indicant el número de núclids dels quals disposa cadascuna d'elles), per al clúster PS3 del departament em creat un fitxer anomenat `ps3.list`:

```
ps3inf00 slots=2
ps3inf01 slots=2
ps3inf02 slots=2
ps3inf12 slots=2
ps3inf04 slots=2
ps3inf05 slots=2
ps3inf06 slots=2
ps3inf07 slots=2
ps3inf08 slots=2
ps3inf09 slots=2
ps3inf10 slots=2
ps3inf11 slots=2
ps3inf12 slots=2
```

Hem executat el programa que calcula en conjunt de Mandelbrot en paral·lel amb repartiment dinàmic de les tasques:

```
$ mpirun -np 26 m -2 2 8001 8001 0.0005 fractal_XXL.pnm
```

Treball futur, alternatives

Entre altres coses, considerem interessant dur a terme les següents tasques:

1. Escriure un script per a realitzar tot el procés automàticament.
2. Empaquetar els binaris resultants per a poder distribuir-los entre usuaris.
3. Instal·lar OpenMPI a algun lloc comú a disc.
4. Instal·lació d'algun gestor de treballs.

Bibliografia

<http://www.open-mpi.org/faq/>

<http://www.netlib.org/netlib/utk/people/JackDongarra/PAPERS/scop3.pdf>

http://www.csua.berkeley.edu/~ranga/notes/ssh_nopass.html