

# Uso de ambientes en Anaconda en el cluster HPC

Para el presente documento se entiende que se va a usar la última versión de Anaconda instalada a fecha de la creación del documento, la 2024.02-1.

Los cuadros resaltados en gris implican un diálogo con el cluster. En negrita está la parte que el usuario debe introducir, en normal la respuesta del equipo.

Los puntos suspensivos indican que se ha suprimido parte de la salida de los comandos que no tienen interés para el presente documento.

## Qué es un ambiente Python

Un ambiente o entorno virtual es un espacio independiente a la instalación general del cluster para aislar los recursos y librerías.

Las razones por las que deberías usar entornos virtuales son:

- Organización: es conveniente tener controladas las librerías y las versiones con las que estamos trabajando para que no existan problemas de incompatibilidades..
- Estabilidad: puede suceder que, al instalar otro paquete de Python, se actualice la librería que deseamos usar y esta nueva versión sea incompatible con el resto de nuestros programas.

Es recomendable crear un entorno virtual para cada proyecto o al menos uno para cada propósito distinto de programación.

## Cómo crear un ambiente

1. Como con cualquier otro paquete de software en el cluster lo primero que debemos hacer es cargar el módulo correspondiente. En este caso:

```
$ module load Anaconda3/2024.02-1
```

2. Ahora debemos inicializar los ambientes en conda. Para ello usamos estas órdenes:

```
$ __conda_setup="$( '/apps/software/Anaconda3/2024.02-1/bin/conda' 'shell.bash' 'hook' 2> /dev/null )"
$ eval "$__conda_setup"
```

Veremos que nuestro «prompt» pasa de «\$» a «(base) \$»

3. A continuación debemos comprobar qué ambientes están ya creados para no repetirnos. Eso se consigue con el comando:

```
(base) $ conda env list
# conda environments:
#
base                               /apps/software/Anaconda3/2024.02-1
qiime2-amplicon-2024.10            /apps/software/Anaconda3/2024.02-1/envs/qiime2-amplicon-2024.10
qiime2-metagenome-2024.10         /apps/software/Anaconda3/2024.02-1/envs/qiime2-metagenome-2024.10
torchcuda                          /apps/software/Anaconda3/2024.02-1/envs/torchcuda
```

4. Ahora podemos escoger un nombre y crear nuestro ambiente:

```
(base) $ conda create -n miambiente
...
## Package Plan ##

environment location: /home/ai/u32855552/.conda/envs/miambiente

Proceed ([y]/n)? y
...
Executing transaction: done
```

## Cómo usar el ambiente creado

Para usar el ambiente, una vez creado hay que dar tres pasos: cargar el módulo, inicializar los ambientes y activar el nuestro. Para esto último hay que ejecutar el comando:

```
(base) $ conda activate miambiente
(miambiente) $
```

A partir de este momento nuestro proceso tendrá acceso a lo instalado en nuestro ambiente. Los cambios que hagamos, también se referirán a él.

## Instalar paquetes en nuestro ambiente

Si lo que deseamos es instalar un único paquete podemos usar el comando siguiente:

```
(miambiente) $ conda install -n miambiente busco
...
Proceed ([y]/n)? y
...
Executing transaction: done
```

Si son muchos paquetes y disponemos de, o creamos, un fichero de requisitos (requirements.txt), podemos usar el siguiente comando:

```
(miambiente) $ conda install -n miambiente --file requirements.txt
```

## Desactivar el ambiente

Si disponemos de varios ambientes y queremos pasar de uno a otro, antes de activar el nuevo debemos desactivar el anterior:

```
(miambiente) $ conda deactivate  
(base) $
```

## Eliminar el ambiente

Si ya no necesitamos el ambiente y queremos hacer limpieza, hay que eliminarlo con el siguiente comando:

```
(base) $ conda remove -n miambiente --all  
...  
Proceed ([y]/n)? y  
...
```

## Simplificando el inicio

Como para usar el ambiente Anaconda hay que hacer tres cosas, si sólo tenemos un ambiente Anaconda y siempre lo usamos, podemos crear un script para simplificar los pasos a uno sólo.

Primero creamos un fichero (micondainit.sh v.g. en nuestra carpeta HOME) con el siguiente contenido:

```
module load Anaconda3/2024.02-1  
__conda_setup="$('/apps/software/Anaconda3/2024.02-1/bin/conda' 'shell.bash' 'hook' 2> /dev/null)"  
if [ $? -eq 0 ]; then  
    eval "$__conda_setup"  
else  
    if [ -f "/apps/software/Anaconda3/2024.02-1/etc/profile.d/conda.sh" ]; then  
        . "/apps/software/Anaconda3/2024.02-1/etc/profile.d/conda.sh"  
    else  
        export PATH="/apps/software/Anaconda3/2024.02-1/bin:$PATH"  
    fi  
fi  
unset __conda_setup  
conda activate miambiente
```

A partir de ahora sólo hay que dar un comando:

```
$ source ~/micondainit.sh  
(miambiente) $
```

Recuerde añadir dicho comando a sus scripts sbatch.