

# Rutas Relativas y Absolutas

Tutorial extraído del blog [C para Operativos](#) (autor: Matías García Isaia)

Más de una vez hemos puesto rutas absolutas en nuestro código, porque es común no entender cómo funcionan las rutas absolutas dentro de un programa.

## Current working directory

Cuando ejecutamos un proceso, éste se ejecuta con un directorio de trabajo asociado (el current working directory, que le llaman). La idea es marcar un directorio como el "actual" en que se encuentra ese proceso. Desde el programa podemos acceder a ese valor usando la función `getcwd()` :

cwd.c

```
1  #include <stdio.h>
2  #include <unistd.h>
3  #include <stdlib.h>
4
5  int main(void) {
6      char *current_dir = getcwd(NULL, 0);
7      printf("El directorio actual es %s\n", current_dir);
8      free(current_dir);
9      return 0;
10 }
```

Guardemos el código en nuestro directorio home ( `/home/utnso` ), compilemos y ejecutemos:

```
utnso@utnso40:~$ pwd
/home/utnso
utnso@utnso40:~$ gcc -Wall cwd.c -o cwd
utnso@utnso40:~$ ./cwd
El directorio actual es /home/utnso
utnso@utnso40:~$ cd /etc/
utnso@utnso40:/etc$ ./cwd
El directorio actual es /etc
utnso@utnso40:/etc$
```

Por defecto, al iniciar un proceso desde la consola, el directorio de trabajo actual es el directorio en que estábamos cuando ejecutamos el proceso. Existe, también, `chdir()` , para cambiar el directorio de trabajo actual, pero se va del scope de este post su uso.

## Rutas relativas

Y si el CWD nos interesa, eso es porque los paths relativos se resuelven a partir del directorio actual del proceso. De `man path_resolution` :

If the pathname does not start with the '/' character, the starting lookup directory of the resolution process is the current working directory of the process. (This is also inherited from the parent. It can be changed by use of the `chdir(2)` system call.)

Pathnames starting with a '/' character are called absolute pathnames. Pathnames not starting with a '/' are called relative pathnames.

Entonces, si, por ejemplo, a una syscall como `open()` le paso el path `archivo.log`, `open` va a abrir el archivo llamado `archivo.log` ubicado en el directorio actual de nuestro proceso. Este no necesariamente es el mismo directorio desde el que ejecutamos el proceso (porque pudo haber hecho un `chdir()`), pero por lo general sí lo va a ser.

## Eclipse

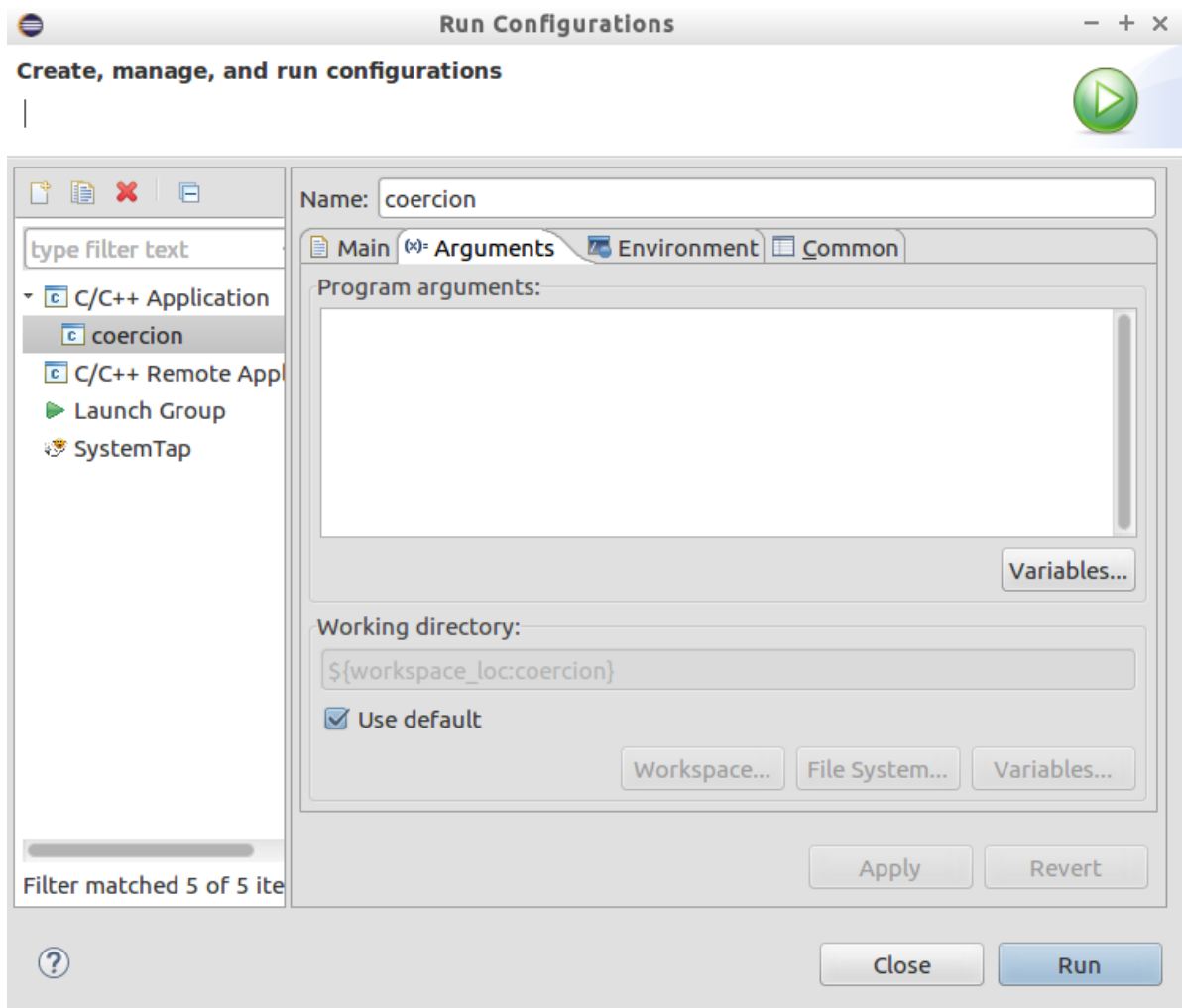
---

Pero más de una vez nos pasa que, trabajando con Eclipse, los path relativos se resuelven bien ejecutando nuestro proceso desde Eclipse, pero falla cuando lo ejecutamos desde la consola.

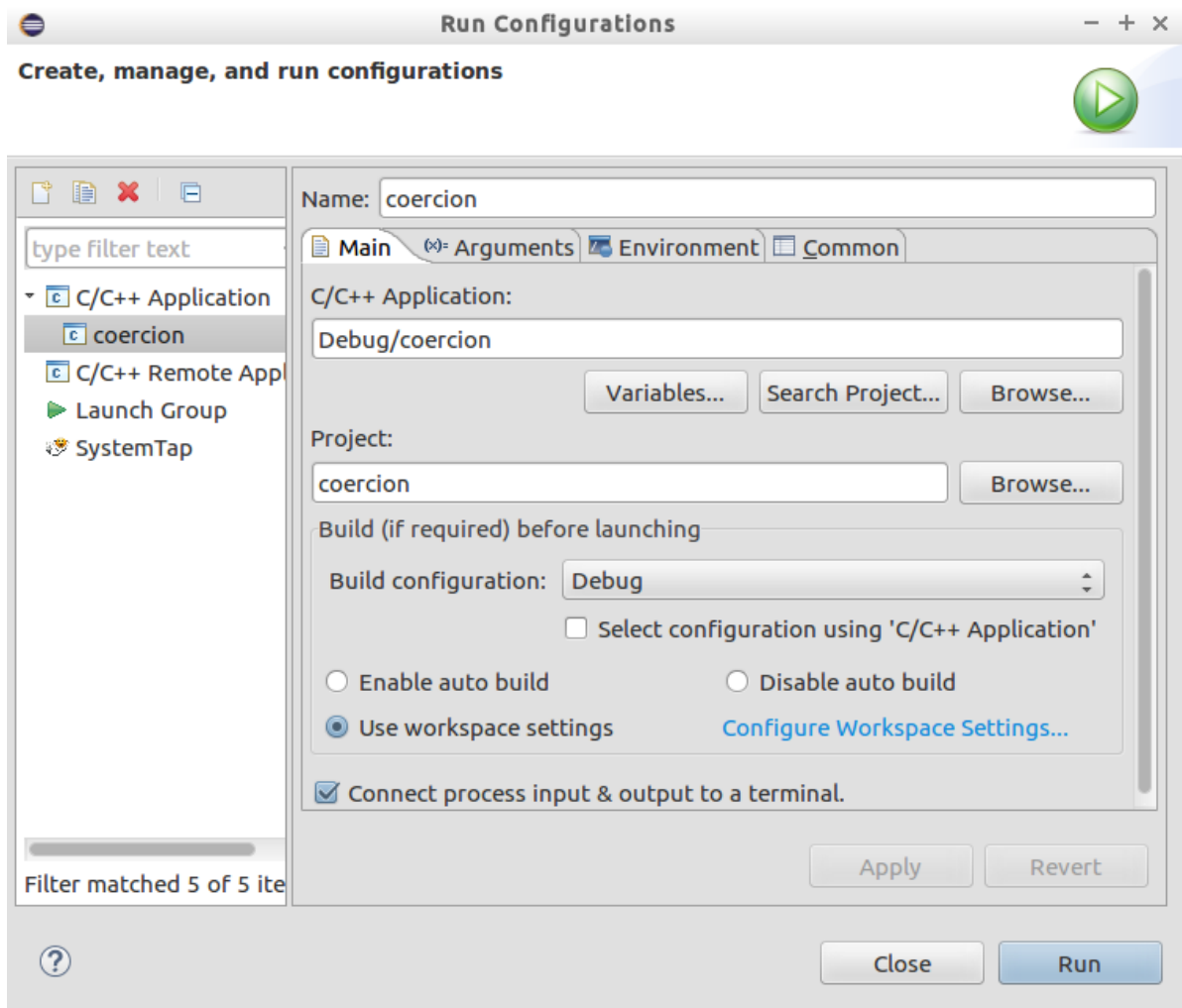
El problema, acá, es el CWD en cada caso.

Cuando vamos a ejecutar desde la consola, por lo general vamos al directorio `Debug/` que contiene nuestro ejecutable, y le damos `./miEjecutable` para correrlo. En ese caso, el CWD es nuestro directorio `Debug/`, y todas las rutas relativas se resuelven a partir de allí.

Pero eclipse no hace eso, o al menos no por defecto. Si vamos a las Run Configurations de eclipse ( `Run > Run configurations...` ) y seleccionamos la de algún proyecto que ya hayamos ejecutado, vamos a encontrar que la pestaña de "Arguments" es algo así:



¡Ahí está! Abajo de todo dice que el working directory de nuestro proceso va a ser `${workspace_loc:NUESTRO_PROYECTO}` , que es la forma horrenda que tiene eclipse para decir "el directorio raíz del proyecto". De hecho, si miramos la pestaña "Main" de las `Run Configurations` , podemos observar algo raro arriba de todo:



Nuestra aplicación no es coercion (el nombre del proyecto al que le saqué la captura), si no Debug/coercion. Ese es el comando que eclipse "va a ejecutar en la consola". Digamos, eclipse va a hacer una especie de `cd` al directorio que marca como "Working directory", y desde ahí va a ejecutar lo que diga en el campo `C/C++ Application` :

```
cd coercion
./Debug/coercion
```

sh

Entonces, como nosotros desde la consola solemos entrar a Debug en lugar de ejecutar desde el directorio raíz del proyecto, ahí es donde aparecen las diferencias con las rutas relativas.

## Del otro lado del espejo

Si quisieras seguir jugando, podrías ponerte a probar qué pasa si cambiás el comando de esa línea. ¿Podés ejecutar, por ejemplo, `ls` ? ¿Podés ejecutar `cd` ? ¿Podés pasarle los parámetros ahí? ¿Qué pasa cuando cambiás el working directory?