

Cómo instalar ANTLR4 en Linux for n00bs

Por David Campos Rodríguez

1. Java

Comprobad que teneis java correctamente instalado y configurado. Podéis comprobarlo poniendo:

```
java --version
```

Si no lo tenéis instalado, es muy sencillo, podeis seguir las instrucciones en la página de java.

2. Antlr4 complete

Vais a <https://antlr.org> y allí encontráis un Quick Start para Linux que os copio aquí (puede que la última versión haya cambiado según cuando estéis leyendo esto):

```
$ cd /usr/local/lib
$ wget http://www.antlr.org/download/antlr-4.7.1-complete.jar
$ export CLASSPATH=".:usr/local/lib/antlr-4.7.1-complete.jar:$CLASSPATH"
$ alias antlr4='java -jar /usr/local/lib/antlr-4.7.1-complete.jar'
$ alias grun='java org.antlr.v4.gui.TestRig'
```

Es posible que necesiteis usar *sudo* con algunos de ellos para que funcione bien. Los alias son opcionales, si quereis tenerlos siempre (pues se borrarán al cerrar la consola) podéis crear/editar el fichero `~/.bash_profile` y añadir ahí los dos comandos. Para que tenga efecto en la consola deberéis reiniciar o, alternativamente, podéis usar `source ~/.bash_profile` para poder hacerlo funcionar en la consola actual sin necesidad de reiniciar.

3. Descargar Antlr4 Cpp target

En <https://antlr.org/download.html> encontraréis el Antlr4 Cpp Target para descargar, aseguraos de descargar la versión para Linux. Si no cambia, al momento de escribir esto el link es el siguiente: <http://www.antlr.org/download/antlr4-cpp-runtime-4.7.1-source.zip>

Lo extraéis donde prefiráis. Dentro tiene un README.md que explica como compilarlo, pero antes de ponernos a ello vamos con dos cosas que se necesitan extra.

4. Antes de compilar

Para compilar la librería necesitamos cmake y uuid. Los podeis obtener simplemente con:

```
$ sudo apt-get install cmake
$ sudo apt-get install uuid-dev
```

5. Compilando Antlr4

Para compilar la librería os ponéis donde está el *README.md*. Escribís los siguientes comandos:

```
$ mkdir build && mkdir run && cd build
$ cmake .. -DANTLR_JAR_LOCATION=/usr/local/lib/antlr-4.7.1-complete.jar -DWITH_DEMO=True
$ make
$ DESTDIR=<antlr4-dir>/run make install
```

Ojo a la parte que os marco en rojo, si la última versión que habéis descargado en el paso 2 no era la 4.7.1, entonces el nombre de archivo será distinto, claro. Lo que he marcado en violeta lo cambias por la ruta donde está el *README.md*. Por ejemplo, yo había extraído el zip en *Documentos/antlr* así que ahí puse *DESTDIR=~/.Documentos/antlr/run make install*.

6. Copiando archivos

En la carpeta *run* se os habrá generado una carpeta *usr* y dentro todos los archivos necesarios. Los copiais a */usr/local/*, como es lógico (supongo que si no indicais el *DESTDIR* así, este paso sería evitable, pero yo lo pongo exactamente como yo lo hice). Una vez los hayáis copiado, ya tendréis todo disponible, pero hay una cosa que solucionar con respecto a los archivos *Makefile* de clase.

7. El “hack”

Los *Makefile* de clase buscan antlr4 en */usr/local/bin/antlr4*, que en vuestro sistema no existirá. Yo lo he solucionado creando un archivo *antlr4* en */usr/local/bin/* con el siguiente contenido:

```
java -jar /usr/local/lib/antlr-4.7.1-complete.jar “$@”
```

A continuación le dais permisos de ejecución.

```
$ chmod 777 /usr/local/bin/antlr4
```

8. Final

Y listo. Podéis ir al primero de los ejemplos de clase y ejecutar:

```
$ make antlr
$ make
$ ./main t1.expr
```

Si todo ha ido bien, debería salir el output correcto. Si recibís un error con respecto a la librería dinámica, eso es porque no habéis actualizado las librerías dinámicas. Para ello simplemente haced:

```
$ sudo ldconfig
```