

# Gestor de Proyectos Merlin

## Introducción

Merlin es un sistema que gestiona el proceso de compilación y enlace del código fuente que forma parte nuestros proyectos de C/C++. Otros gestores de proyectos como make o ant necesitan utilizar ficheros scripts para describir las dependencias entre los fuentes y las tareas que debemos realizar para el conjunto del código. Merlin se basa en la información intrínseca del código (`#include`) para establecer las dependencias entre los diversos módulos que componen nuestro proyecto. De esta manera, reducimos los ficheros que debemos mantener para el proceso de compilación/enlace y disminuimos los posibles errores que podamos realizar al establecer dependencias entre los diversos ficheros.

## Organización Del Código

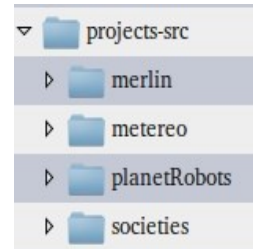
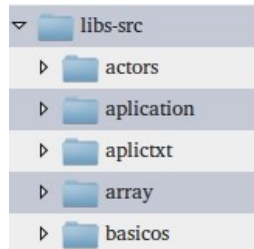
Para que Merlin pueda analizar nuestros proyectos debemos organizar nuestro código de una forma concreta. El análisis se basa en dos directorios principales, `libs-src` y `projects-src` y pueden contener ficheros `.c`, `.cpp`, `.h`, `.hpp` principalmente. Además deben cumplir varias propiedades:

1. En `libs-src` debe estar el código común a todos los proyectos. Puede contener varias carpetas, cada una de las cuales va a representar una librería. Si queremos que otra librería use código de otra, sólo debemos incluir las cabeceras que necesitamos usar. Esto lleva inherentemente que entonces la librerías que usa la cabecera dependa de la librería que contiene la misma.
2. No pueden existir sub-carpetas dentro de las carpetas que representan librerías. Es decir, dentro de `libs-src` sólo existe un nivel de carpetas.
3. No se podrá incluir dentro de fuentes otros fuentes. Sólo se considerará relaciones con cabeceras.
4. Para todos los fuentes dentro `libs-src` debe existir su respectiva cabecera. Si no existe la cabecera el código no se estará considerando.
5. Tampoco se considerará el código cuyas cabeceras no se incluyan en ningún fichero de código fuente.
6. En `projects-src` tendremos todas las carpetas que representen proyectos. Un



proyecto está compuesto por uno o varios ejecutables.

7. En las carpetas de projects-src pueden existir fuentes sin cabeceras que Merlin identificará como ejecutables. Dentro de estos fuentes debe existir la función 'main'.
8. En projects-src pueden existir ficheros cabecera, pero en ningún caso una cabecera de un proyecto debe estar incluida en otro proyecto y sólo pueden ser utilizados por los proyectos.



## Ejecución De Merlin

Existen varios modos de ejecución de Merlin dependiendo de que demandemos de él. Para saber todo lo que puede hacer Merlin debemos ejecutar:

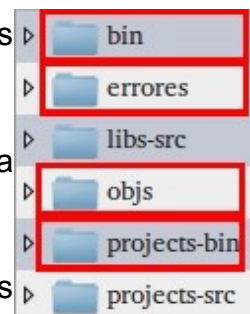
*merlin -help*

La opción principal y la que más se usará será:

*merlin -hacer:[rel,deb] directorio\_de\_proyectos*

Con esta opción Merlin procesará los fuentes, compilará los ficheros necesarios (aquellos cuyos fuentes tengan una fecha mayor que su objeto compilado o que no lo tengan) y recopilará los objetos necesarios para construir todos los ejecutables que estén en los proyectos. Con esta opción Merlin creará varios directorios:

1. El directorio projects-bin donde generará los ejecutables de los proyectos.
2. El directorio errores donde generará los mensajes de error de la compilación y linkado.
3. El directorio objs donde se generará los objetos de los ficheros fuentes.



Merlin puede generar dos versiones de los proyectos. La versión release (rel) será una versión con todas las optimizaciones de código ejecutable y sin información de debug. Es la versión que podemos distribuir. La versión debug (deb) es la versión compilada para la depuración. En esta versión se desactivarán todas las optimizaciones y se generará la información necesaria que utiliza el depurador para poder depurar el código del proyecto.

Existen otras opciones de interés como por ejemplo:

*merlin -make:nombre\_proyecto directorio\_de\_proyectos*

Esta opción nos generará un directorio dentro del directorio make con el nombre del proyecto donde estará todo el código necesario para compilar el proyecto y con el fichero makefile necesario para compilar el proyecto. Esta opción sirve sobre todo si queremos distribuir nuestro proyecto sin necesidad de distribuir el ejecutable Merlin. nombre\_proyecto debe ser un proyecto que se tenga dentro de la carpeta projects-src.

Merlin también nos puede decir que dependencias tiene un fuente. Para ello se debe ejecutar la opción:

*merlin -incluido:nombre\_fichero directorio\_de\_proyectos*

Existe otra opción que nos ayuda a saber si las dependencias que tienen nuestros proyectos son correctos con determinadas reglas que se han ido estipulando para que un proyecto esté diseñando correctamente. En ese caso debemos ejecutar la opción:

*merlin -integridad directorio\_de\_proyectos*

Como se puede ver, Merlin siempre necesita del directorio\_de\_proyectos para poder saber donde están las carpetas de fuentes. Podemos eliminar este parámetro si definimos como variable del sistema MERLIN\_PROJECTS=directorio\_de\_proyectos. De esta forma una opción como la compilación queda como:

*merlin -hacer:rel*

## **Conclusión.**

Lo que hemos pretendido con este documento es aproximar al usuario a la gestión de compilación y enlazado de proyectos. La herramienta Merlin nos ayuda a que esta gestión sea muy fácil y libre de errores. En el futuro se desea desarrollar más opciones para que la gestión de proyectos grandes sea más cómoda.