

\* **sobre llamadas al sistema para E/S:** ver Libro de Kerrisk pags. 69-70 ("The Linux programming interface"; Kerrisk, M.). En esas páginas se puede ver una descripción de las llamadas al sistema para E/S en una primera aproximación; en las páginas sucesivas a partir de la 72 en adelante puede verse una descripción detallada de dichas llamadas.

\* **ejemplo de macro en C**

```
#define MITAD(X)    (X/2)

...
main (int argc, char *argv[])
{ ....printf ("mitad de 8 es %d\n", MITAD(8)); }
```

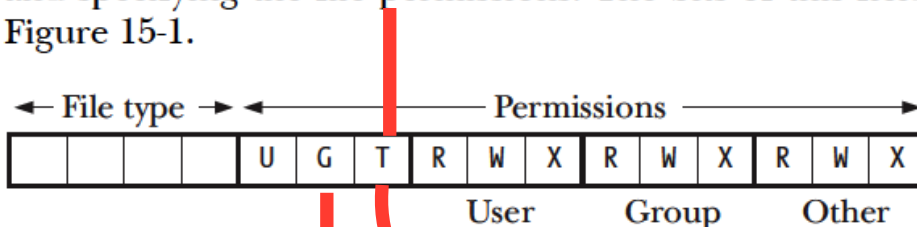
\* **Sobre los atributos de un archivo; llamada al sistema `stat`**

```
// cualquier elemento del árbol de archivos o directorios, sea del tipo que sea
struct stat atributos;

....
if ( stat("datos/d2014",&atributos) < 0) {
    perror("\nError en lstat");
    exit(-1)
}
```

**Estructura `stat`: campo `st_mode`:** (página 281 del libro de Kerrisk)

The *st\_mode* field is a bit mask serving the dual purpose of identifying the file type and specifying the file permissions. The bits of this field are laid out as shown in Figure 15-1.



**Figure 15-1:** Layout of *st\_mode* bit mask

The file type can be extracted from this field by ANDing (&) with the constant `S_IFMT`. (On Linux, 4 bits are used for the file-type component of the *st\_mode* field.

```
if ((atributos.st_mode & 0170000) == 0100000) /// archivo regular
/////////
                        S_IFMT      S_IFREG
.....
```

Because this is a common operation, standard macros are provided to simplify the above to the following:

```
if (S_ISREG(statbuf.st_mode))
    printf("regular file\n");
```

En `st_mode`, además de los permisos de usuario, grupo y otros, están los siguientes bits, denotados en la figura anterior como U, G y T:

**set-user-ID (bit 04000)**

cuando un archivo ejecutable tiene este bit establecido, al ejecutarse, el proceso resultante tiene como **usuario efectivo** al propietario del archivo ejecutable (en lugar del usuario que lanza el proceso como es lo usual)

**set-group-ID (bit 02000)**

cuando un archivo ejecutable tiene este bit establecido, al ejecutarse, el proceso resultante tiene como **grupo efectivo** al grupo propietario del archivo ejecutable (en lugar del grupo al que pertenece el usuario que lanza el proceso como es lo usual)

**sticky (bit 01000)**

(Ver libro de Kerrisk pag. 300)