

# Sistemas Operativos

## sesión 6: compilación (extendido)

Grado en Ingeniería Informática  
Universidad Carlos III de Madrid

# Contenidos



- Proceso de compilación
- Inspección de un ejecutable/proceso
- Bibliotecas estáticas y dinámicas

# Contenidos



- **Proceso de compilación**
- Inspección de un ejecutable/proceso
- Bibliotecas estáticas y dinámicas

# Motivación

- ¿Qué fases hay en la compilación y ejecución de un programa?

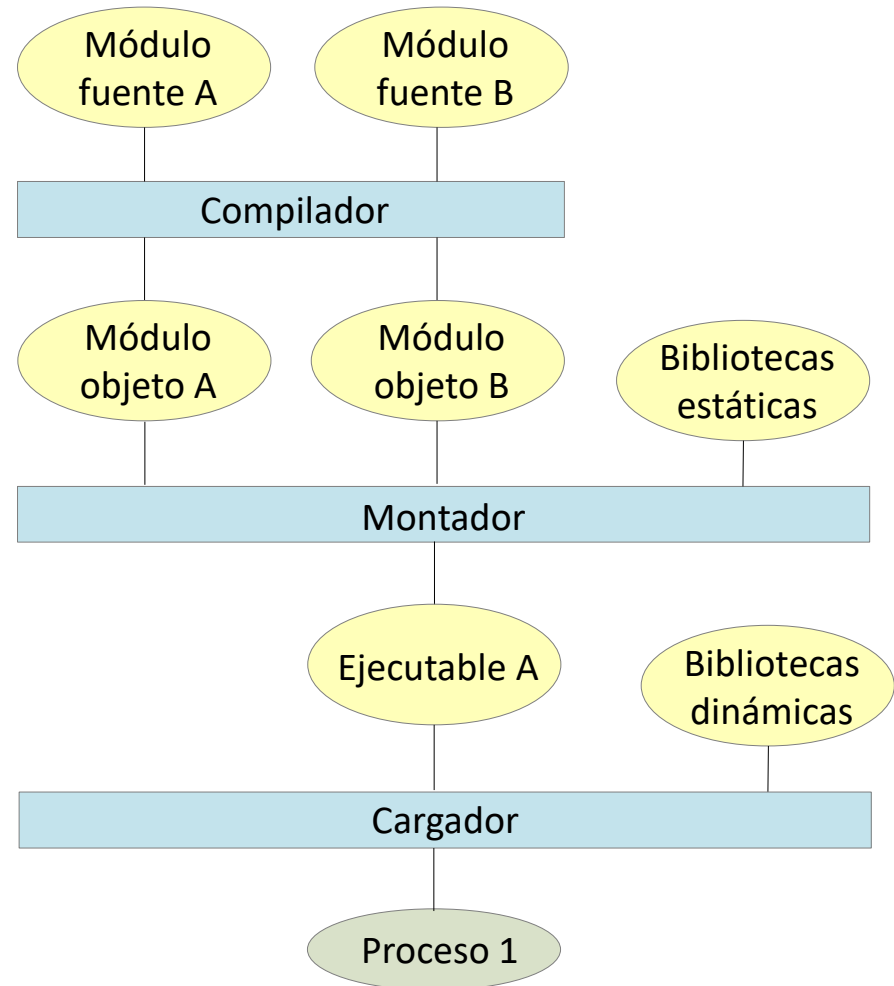
```
acaldero@phoenix:~/work$ gcc -o programa fuente.c  
acaldero@phoenix:~/work$ ./programa
```

Cargando...

...

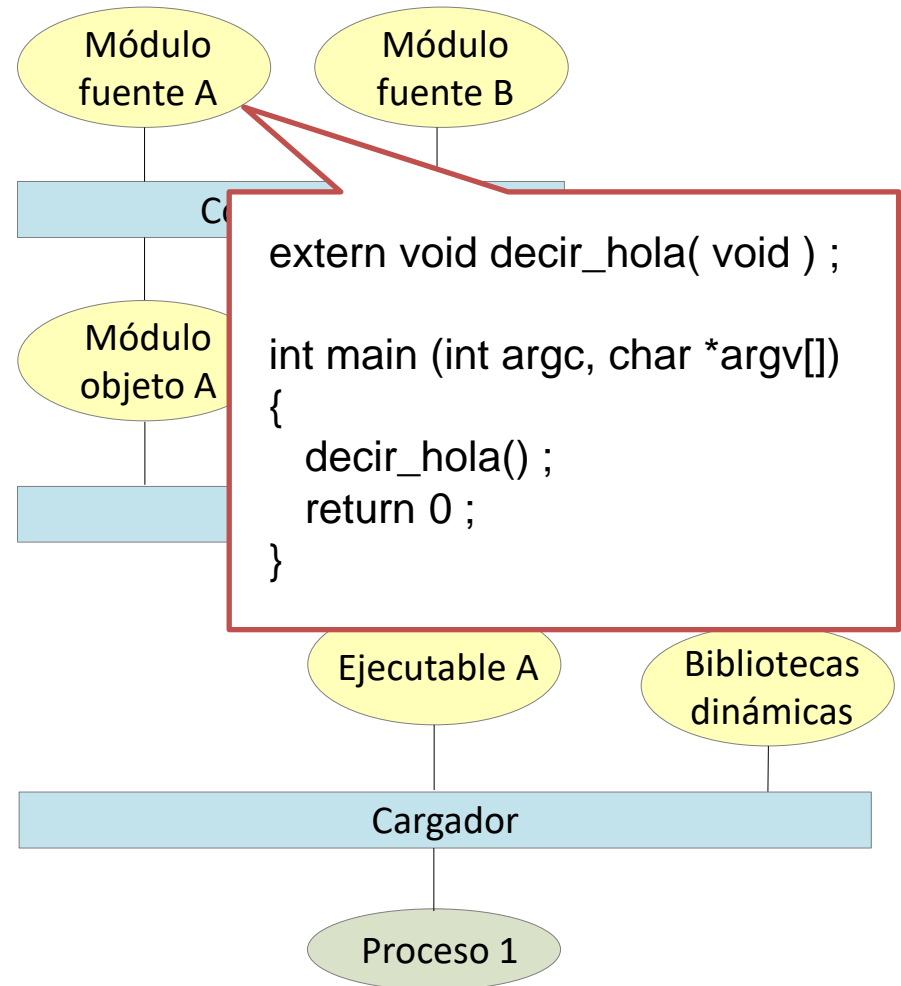
# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - Compilación
  - Montaje
  - Enlazado dinámico
  - Ejecución



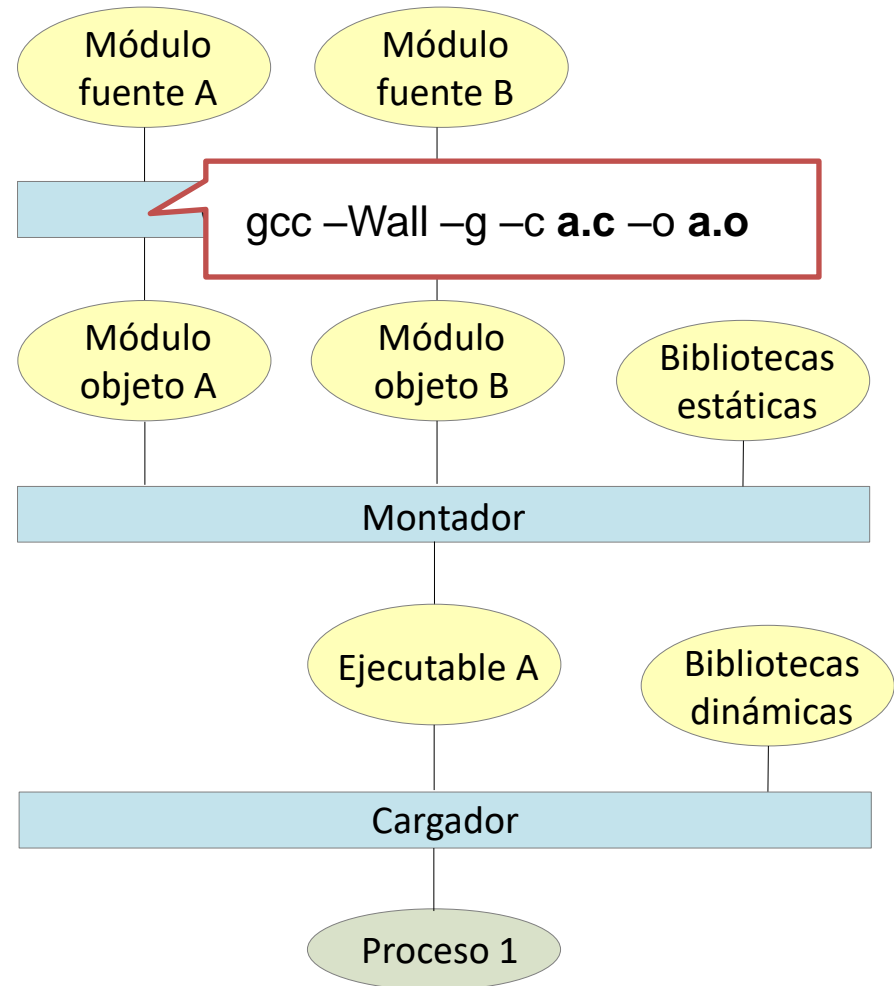
# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - **Compilación**
  - Montaje
  - Enlazado dinámico
  - Ejecución



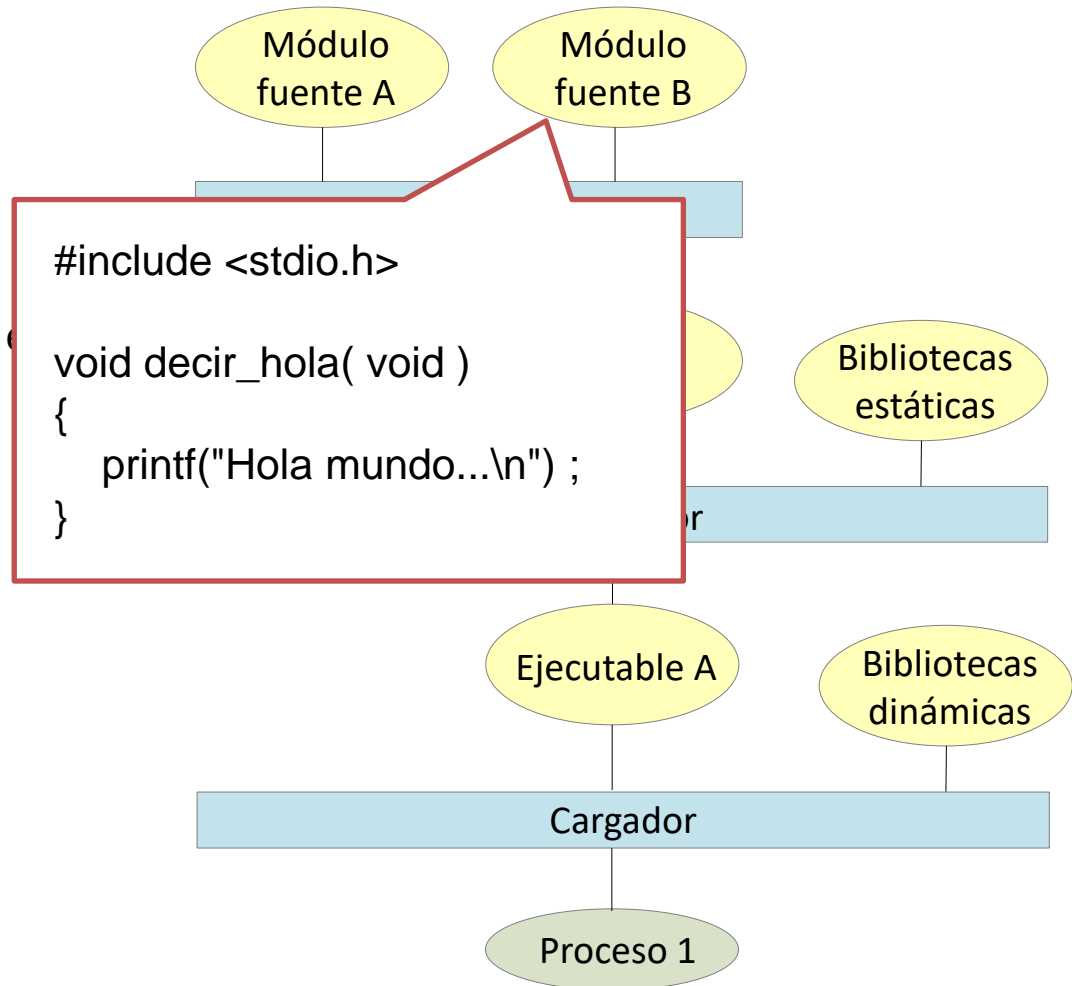
# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - **Compilación**
  - Montaje
  - Enlazado dinámico
  - Ejecución



# Generación y ejecución de programas

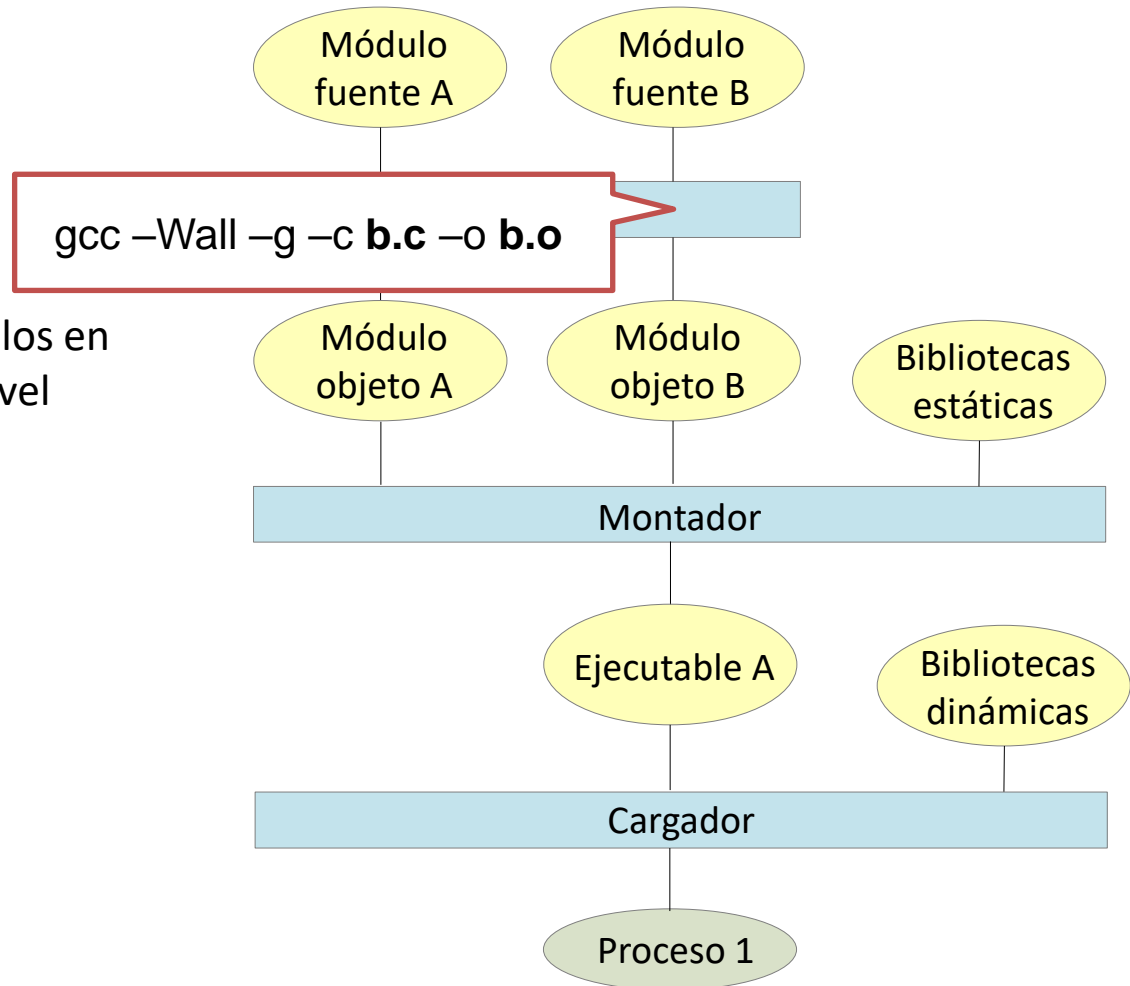
- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - **Compilación**
  - Montaje
  - Enlazado dinámico
  - Ejecución





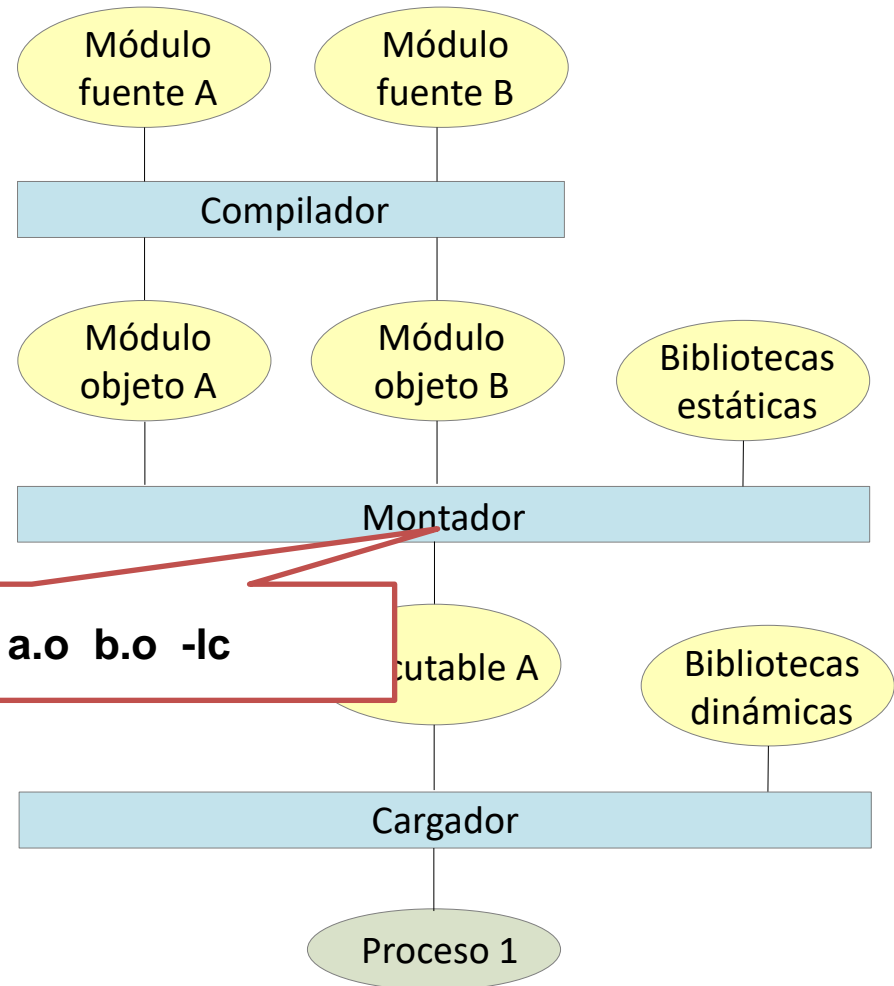
# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - **Compilación**
  - Montaje
  - Enlazado dinámico
  - Ejecución



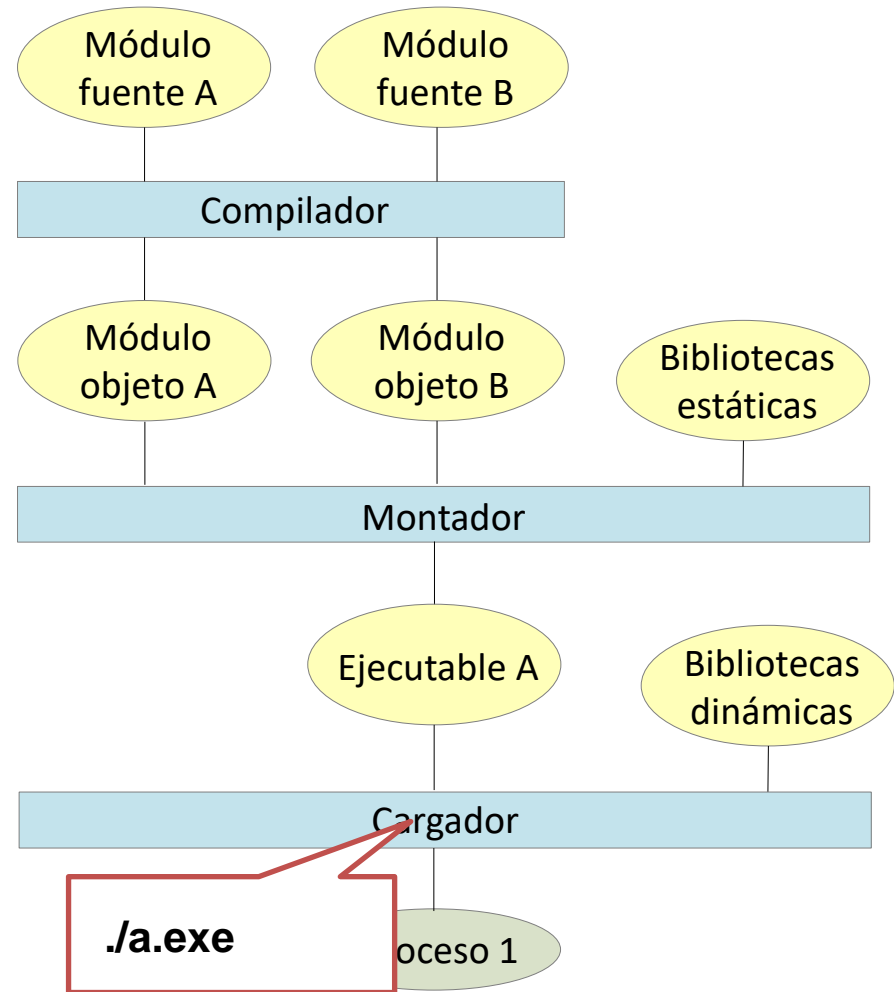
# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - Compilación
  - Montaje
  - Enlazado
  - Ejecución



# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - Compilación
  - Montaje
  - Enlazado dinámico
  - Ejecución



# Contenidos



- Proceso de compilación
- **Inspección de un ejecutable/proceso**
- Bibliotecas estáticas y dinámicas

# Motivación

- ¿Cómo saber qué llamadas al sistema hace un programa?
- ¿Qué bibliotecas usa?

```
acaldero@phoenix:~/work$ ./programa
```

```
Cargando...
```

```
...
```

# Inspeccionar un ejecutable

- Dependencias de un ejecutable (lib. dinámicas):

```
acaldero@phoenix:~/infodso/$ ldd main.exe
linux-gate.so.1 => (0xb7797000)
libdinamica.so.1 => not found
libc.so.6 => /lib/libc.so.6 (0xb761c000)
/lib/ld-linux.so.2 (0xb7798000)
```

- Símbolos de un ejecutable:

```
acaldero@phoenix:~/infodso/$ nm main.exe
08049f20 d _DYNAMIC
08049ff4 d _GLOBAL_OFFSET_TABLE_
0804856c R _IO_stdin_used
          w _Jv_RegisterClasses
08049f10 d __CTOR_END__
08049f0c d __CTOR_LIST__
...
```

# Inspeccionar un proceso

- Detalles de las secciones de un proceso:

```
acaldero@phoenix:~/infodso/$ cat /proc/1/maps
b7688000-b7692000 r-xp 00000000 08:02 1491      /lib/libnss_files-2.12.1.so
b7692000-b7693000 r--p 00009000 08:02 1491      /lib/libnss_files-2.12.1.so
b7693000-b7694000 rw-p 0000a000 08:02 1491      /lib/libnss_files-2.12.1.so
b7694000-b769d000 r-xp 00000000 08:02 3380      /lib/libnss_nis-2.12.1.so
b769d000-b769e000 r--p 00008000 08:02 3380      /lib/libnss_nis-2.12.1.so
b769e000-b769f000 rw-p 00009000 08:02 3380      /lib/libnss_nis-2.12.1.so
b769f000-b76b2000 r-xp 00000000 08:02 1414      /lib/libnsl-2.12.1.so
b76b2000-b76b3000 r--p 00012000 08:02 1414      /lib/libnsl-2.12.1.so
b76b3000-b76b4000 rw-p 00013000 08:02 1414      /lib/libnsl-2.12.1.so
b76b4000-b76b6000 rw-p 00000000 00:00 0
..
b78b7000-b78b8000 r-xp 00000000 00:00 0          [vdso]
b78b8000-b78d4000 r-xp 00000000 08:02 811      /lib/ld-2.12.1.so
b78d4000-b78d5000 r--p 0001b000 08:02 811      /lib/ld-2.12.1.so
b78d5000-b78d6000 rw-p 0001c000 08:02 811      /lib/ld-2.12.1.so
b78d6000-b78ef000 r-xp 00000000 08:02 1699      /sbin/init
b78ef000-b78f0000 r--p 00019000 08:02 1699      /sbin/init
b78f0000-b78f1000 rw-p 0001a000 08:02 1699      /sbin/init
b81e5000-b8247000 rw-p 00000000 00:00 0          [heap]
bf851000-bf872000 rw-p 00000000 00:00 0          [stack]
```

# Inspeccionar un proceso

- Llamadas al sistema realizadas por un proceso:

```
acaldero@phoenix:~/infodso/$ strace ls -las
```

```
execve("/bin/ls", ["ls", "-las"], [/ * 20 vars *]) = 0
brk(0)                                = 0x8bb7000
access("/etc/ld.so.nohwcap", F_OK)     = -1 ENOENT (No such file or directory)
mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb78c3000
access("/etc/ld.so.preload", R_OK)     = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY)     = 3
fstat64(3, {st_mode=S_IFREG|0644, st_size=123227, ...}) = 0
mmap2(NULL, 123227, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb78a4000
close(3)                               = 0
access("/etc/ld.so.nohwcap", F_OK)     = -1 ENOENT (No such file or directory)
open("/lib/i386-linux-gnu/libselinux.so.1", O_RDONLY) = 3
read(3, "\177ELF\1\1\1\0\0\0\0\0\0\0\0\0\3\0\3\0\1\0\0\0\300>\0\0004\0\0\0"... , 512) = 512
fstat64(3, {st_mode=S_IFREG|0644, st_size=104116, ...}) = 0
mmap2(NULL, 109440, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0xb7889000
mmap2(0xb78a2000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x18) = 0xb78a2000
close(3)                               = 0
access("/etc/ld.so.nohwcap", F_OK)     = -1 ENOENT (No such file or directory)
open("/lib/i386-linux-gnu/librt.so.1", O_RDONLY) = 3
read(3, "\177ELF\1\1\1\0\0\0\0\0\0\0\0\0\3\0\3\0\1\0\0\0\300\30\0\0004\0\0\0"... , 512) = 512
...
```



# Contenidos



- Proceso de compilación
- Inspección de un ejecutable/proceso
- **Bibliotecas estáticas y dinámicas**

# Motivación

- ¿Qué es una biblioteca estática?
- ¿Qué es una biblioteca dinámica?
- ¿Cómo se utilizan?

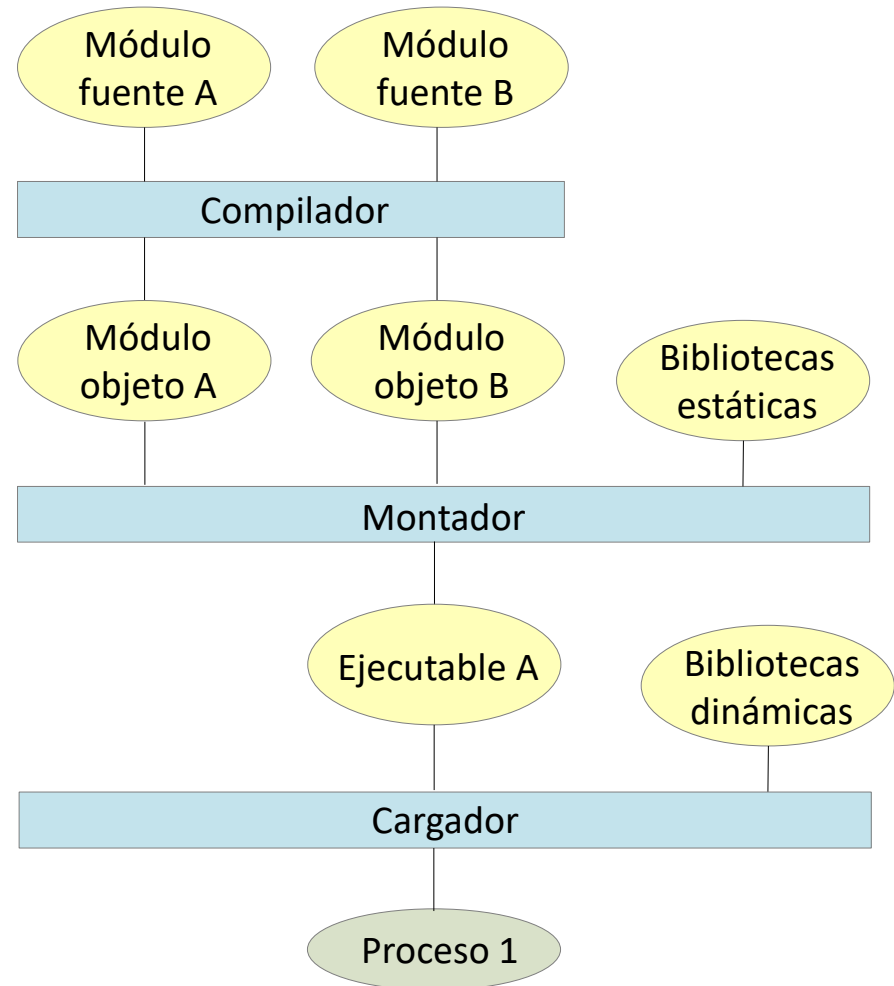
```
acaldero@phoenix:~/work$ gcc -o programa fuente.c libestatica1.a  
acaldero@phoenix:~/work$ ./programa
```

Cargando...

...

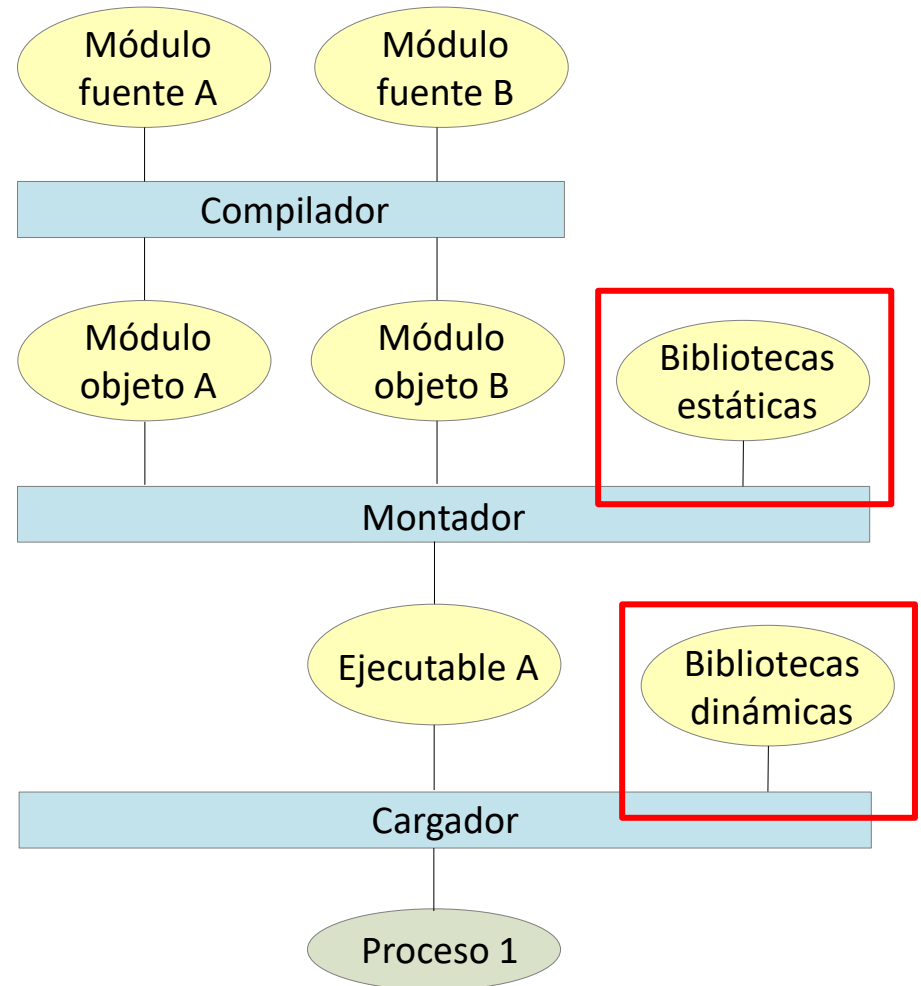
# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - Compilación
  - Montaje
  - Enlazado dinámico
  - Ejecución



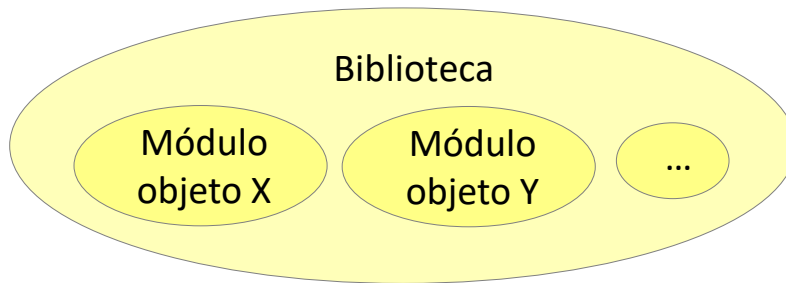
# Generación y ejecución de programas

- Aplicación
  - Conjunto de módulos en lenguaje de alto nivel
- Fases:
  - Compilación
  - Montaje
  - Enlazado dinámico
  - Ejecución



# Bibliotecas de objetos

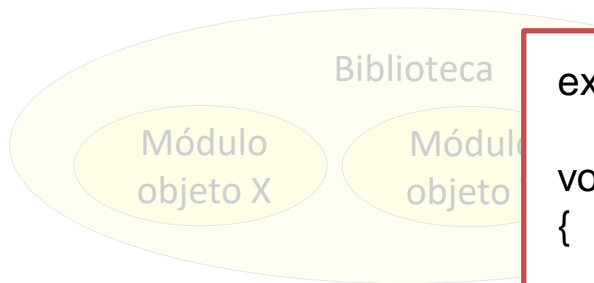
- Biblioteca
  - Colección de módulos objetos relacionados



- Biblioteca **estática**
  - Carga y montaje en tiempo de compilación
- Biblioteca **dinámica**
  - Carga y montaje en tiempo de ejecución
  - Se indica al montar qué biblioteca usar, carga y montaje posterior

# Bibliotecas de objetos

- Biblioteca
  - Colección de módulos objetos relacionados



**B.C**

```
#include <stdio.h>

void decir ( char * str )
{
    printf("%s",str) ;
}
```

**A.C**

```
extern void decir ( char * str ) ;

void decir_hola( void )
{
    decir("Hola mundo...\n") ;
}
```

**MAIN.C**

```
extern void decir_hola( void ) ;

int main (int argc, char *argv[])
{
    decir_hola() ;
    return 0 ;
}
```

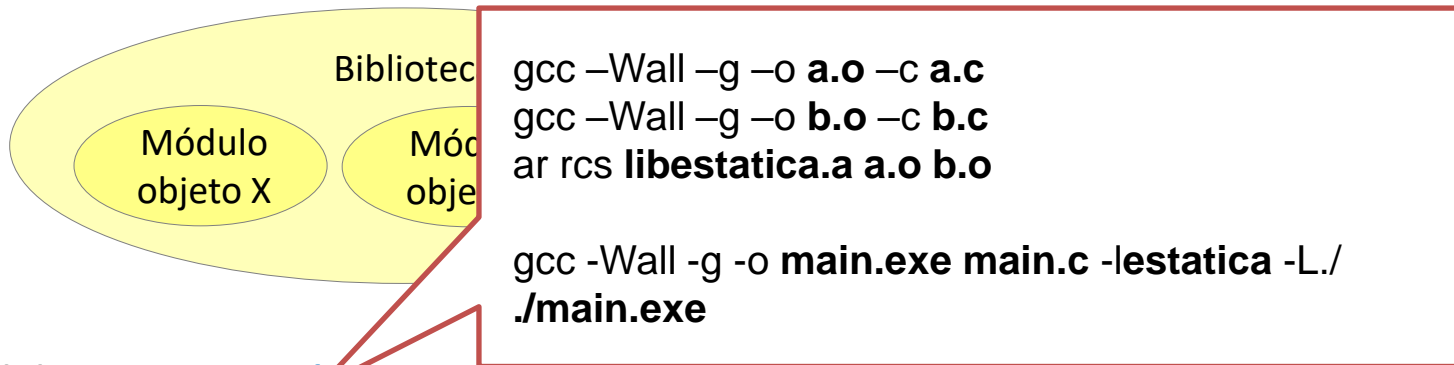
tiempo de compilación

tiempo de ejecución

se marca al montar qué biblioteca usar, carga y montaje posterior

# Bibliotecas de objetos

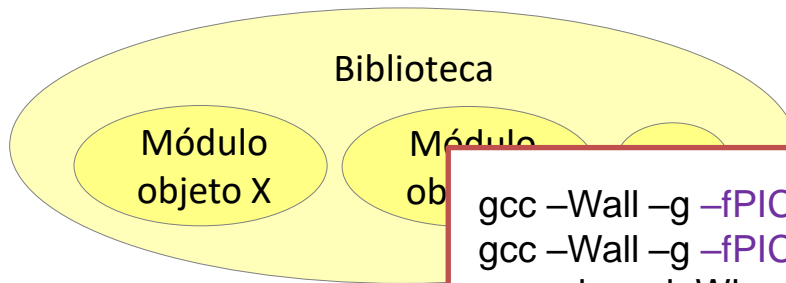
- Biblioteca
  - Colección de módulos objetos relacionados



- Biblioteca **estática**
  - Carga y montaje en tiempo de compilación
- Biblioteca **dinámica**
  - Carga y montaje en tiempo de ejecución
  - Se indica al montar qué biblioteca usar, carga y montaje posterior

# Bibliotecas de objetos

- Biblioteca
  - Colección de módulos objetos relacionados



- Biblioteca **estática**

- Carga y montaje

- Biblioteca **dinámica**

- Carga y montaje en tiempo de ejecución
  - Se indica al montar qué biblioteca usar, carga y montaje posterior

```
gcc -Wall -g -fPIC -o a.o -c a.c
gcc -Wall -g -fPIC -o b.o -c b.c
gcc -shared -Wl,-soname,libdinamica.so \
    -o libdinamica.so.1.0 a.o b.o
ln -s libdinamica.so.1.0 libdinamica.so

gcc -Wall -g -o main.exe main.c -ldinamica -L./
env LD_LIBRARY_PATH=$LD_LIBRARY_PATH:./main.exe
```



# Sistemas Operativos

## sesión 6: compilación (extendido)

Grado en Ingeniería Informática  
Universidad Carlos III de Madrid