

# UNIDAD 3. CARGADORES Y LIGADORES

# Sumario

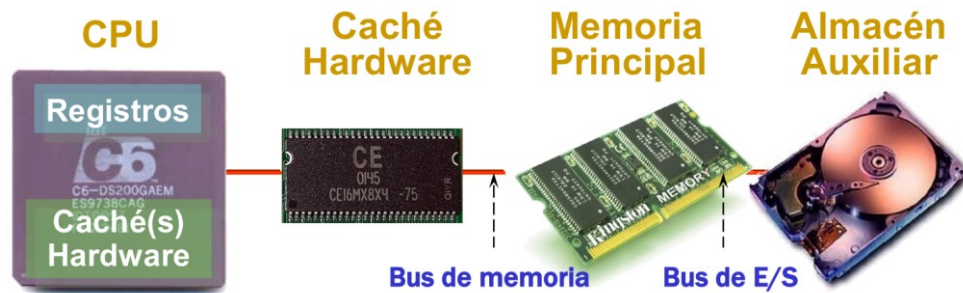
1	Cargadores.....	1
1.1	Funciones de un cargador.....	1
1.2	Tipos de Cargadores.....	2
1.3	Proceso de carga absoluta.....	3
1.4	Proceso de carga mediante ROM.....	3
1.5	Proceso de carga relocizable.....	4
2	Ligadores.....	4
2.1	Funciones del ligador.....	4
2.2	Tipos de ligadores.....	5
2.2.1	Ligador estático.....	5
2.2.2	Ligador dinámico.....	5
3	Referencias.....	7

## Índice de Figuras

Figura 1: Recorrido que sigue un archivo ejecutable almacenado en el HD para poder ejecutarse.....	1
Figura 2: El Sistema Operativo siempre inicia su cargado en la misma región de memoria.....	4
Figura 3: El ligador trabaja con archivos objeto.....	5

# 1 Cargadores

Un cargador es un programa que coloca en la memoria para su ejecución, el programa guardado en algún dispositivo de almacenamiento secundario (Hard Disk, DVD, etc.). La Figura 1 se puede utilizar para imaginar el recorrido que sigue un archivo ejecutable almacenado en el disco duro para llegar a ejecutarse en el procesador.



*Figura 1: Recorrido que sigue un archivo ejecutable almacenado en el HD para poder ejecutarse.*

Un cargador es un programa del sistema que realiza la función de carga, pero muchos cargadores también incluyen relocación y ligado. Algunos sistemas tienen un ligador para realizar las operaciones de enlaces y un cargador separado para manejar la relocación y la carga. Los procesos de ensamblado y carga están íntimamente relacionados.

El cargador consiste en un juego de instrucciones que permiten al dispositivo de entrada (teclado o unidad de cinta) asignar la dirección de inicio de la memoria y asegurar que el computador leerá el programa y lo cargará byte a byte.

En muchos programas el cargador hace la labor del programa de enlace, porque existe solo un “linking loader” y no existe programa de enlace independiente. Es importante señalar que no se necesita un programa de enlace ni un cargador separado para cada traductor en el sistema, ya que estos programas trabajan con el programa objeto, sin importar el lenguaje fuente. Por otro lado es importante que cada compilador o ensamblador produzca el programa objeto usando el mismo formato. Cada programa objeto es un archivo de récord.

## 1.1 Funciones de un cargador

Las funciones mas importantes de un cargador son: colocar un programa objeto en la memoria e iniciar su ejecución. Si tenemos un cargador que no necesita realizar las funciones de ligado y relocación de programas, su operación es simple pues todas las funciones se realizan en un solo paso. Se revisa el registro de encabezamiento para comprobar se ha presentado el programa correcto para la carga (entrando en la memoria disponible).

A medida que lee cada registro de texto, el código objeto que contiene pasa a dirección de la memoria indicada. Cuando se encuentra el registro de fin, el cargador salta a la dirección especificada para iniciar la ejecución del programa cargado. Un programa objeto contiene instrucciones traducidas y valores de datos del programa fuente y especifica direcciones en memoria dónde cargarán estos elementos.

Carga que lleva el programa objeto a la memoria para su ejecución.

- Relocalización que modifica el programa objeto de forma que puede cargarse en una dirección diferente de la localidad especificada originalmente.
- Ligado que combina dos o más programas objeto independiente y proporciona la información necesaria para realizar diferencias entre ellos.
- El cargador es un programa del sistema que realiza la función de carga pero muchos cargadores también incluyen relocalización y ligado. Algunos sistemas tienen un ligador (o editor de ligado) para realizar las operaciones de enlace, y un cargador separado para manejar la relocalización y la carga.
- En la mayoría de los casos todos los traductores de programas (esto es, ensambladores y compiladores) de un sistema en particular producen programas objeto en el mismo formato.

De esta misma forma, puede usarse el cargador o ligador del sistema con independencia del lenguaje de programación fuente original, se suele utilizar el término cargador en lugar de cargador y ligador, los procesos de ensamblado y carga están íntimamente relacionados entre sí.

Se han desarrollado herramientas especiales de software, llamadas cargadores para asistir, al programador en la carga del programa. El cargador es normalmente un programa pequeño que permite al usuario introducir directamente las palabras de instrucción y datos a direcciones concretas de la memoria mediante un dispositivo como un teclado o una cinta magnética.

## 1.2 Tipos de Cargadores

Cargadores Bootstrap: el programa cargador una vez, situado en la memoria del computador, cargará el programa de aplicación y los datos. Pero, previamente, se ha debido cargar el cargador en la memoria. Y esto se puede realizar por los siguientes métodos:

- Entrada manual: mediante el teclado el usuario teclea el cargador BOOTSTRAP. Después de esto, el cargador se carga así mismo en la memoria del computador.
- Entrada por ROM: en algunas computadoras, el cargador de arranque (bootstrap loader) que se utiliza para iniciar la computadora está contenido en la ROM. Esta es programada en la fábrica y no puede modificarse después.

Cargadores Iniciales: indican a la computadora la forma de poner, dentro de la memoria principal unos datos que están guardados en un periférico de memoria externa (cinta, HD, DVD, memoria flash, etc.). Sirven para cargar en la memoria pequeños programas que inician el funcionamiento de una computadora.

Algunas computadoras de carácter general no tienen en memoria ningún programa de forma permanente y cuando se desconectan pierden toda la información de su memoria interna. Al volverlos a conectar no son capaces de controlar ningún periférico. Se hace así para que sea el usuario el que ponga los programas que le interese ejecutar.

Cargadores Absolutos: el programa cargador pone en memoria las instrucciones guardadas en sistemas externos. Independientemente de que sea un cargador inicial, o no sin dichas instrucciones se almacenan siempre en el mismo espacio de memoria (cada vez que se ejecuta el programa cargador) se dice que es un cargador absoluto.

**Cargadores con Reubicación:** en ocasiones un mismo programa necesita ejecutarse en diferentes posiciones de memoria. Para esto la traducción debe estar realizada en forma adecuada, es decir no utilizando referencias absolutas a direcciones en memoria, sino referencias a una dirección especial llamada de reubicación.

**Cargadores Ligadores:** conocidos también como linker. Un linker es un término en inglés que significa montar.

Montar un programa consiste en añadir al programa objeto obtenido a la traducción las rutinas externas a las que hace referencia dicho programa. El ensamblador debe permitir dichas referencias y las rutinas deben estar a su vez en lenguaje máquina guardadas en algún elemento accesible por el montador. Generalmente, dichas rutinas se encuentran guardadas en un fichero especial al que suele denominarse librería porque están almacenadas todas las rutinas externas susceptibles de ser utilizadas por los diferentes programas del usuario. Allí va el programa ligador cuando está realizando el montaje de un programa a buscarlas y las adjunta al programa objeto.

### **1.3 Proceso de carga absoluta**

Para ser ejecutados los programas se deben ubicar en la memoria principal. Una tarea muy importante es asociar instrucciones y datos con localidades específicas de memoria. El trabajo de realizar esta asociación se deja algunas veces al usuario, otras al traductor, otras al programa llamado cargador y otras más al sistema operativo. La asociación de instrucciones con ciertas localidades de memoria se conoce como ligamiento.

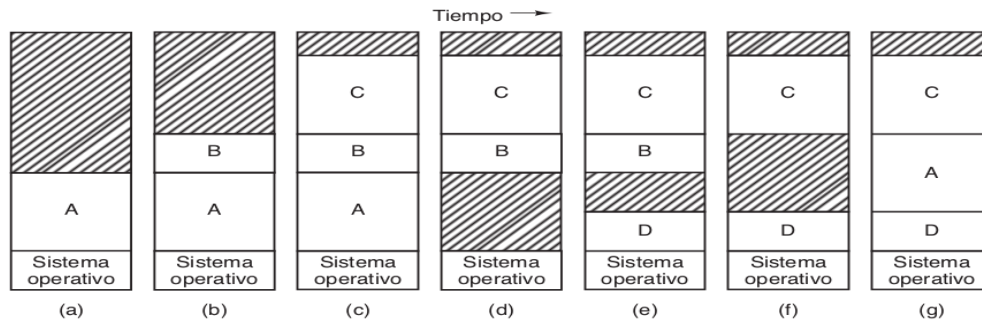
Dentro de la programación el lenguaje máquina, el ligamiento y los sistemas de memoria virtual actuales lo realizan de manera dinámica mientras se ejecuta el programa. El retraso del ligamiento aumenta la flexibilidad para el usuario y el sistema, pero los traductores, ligadores y los sistemas operativos resultan más complejos. Un cargador es un programa que coloca las instrucciones de un programa y sus datos en memoria principal. Un cargador absoluto coloca las instrucciones y los datos en las localidades precisas indicadas por el programa en lenguaje de máquina.

El proceso de carga absoluta consiste en que con la máquina vacía ó inactiva no hay necesidad de hacer relocalización de programas tan solo se puede especificar la dirección absoluta del programa que se cargue en primer lugar. En la mayoría de los casos este programa es el Sistema Operativo que ocupa un lugar predefinido en la memoria. La Figura 2 muestra esta situación, donde A, B, C y D son procesos y el área sombreada es espacio libre. Esto significa que se necesitan algunos medios para realizar las funciones del cargador absoluto. Una opción es que el operador introduzca en la memoria el código objeto de un cargador absoluto, utilizando los interruptores en la consola del computador. Algunos computadores requerían que el operador hiciera exactamente eso. Sin embargo este proceso es demasiado incómodo y propenso a errores para ser una buena solución del problema.

### **1.4 Proceso de carga mediante ROM**

Otra posibilidad es que el programa del cargador absoluto resida permanente en una memoria sólo de lectura (ROM). Cuando se produce una señal de Hardware la máquina empieza a ejecutar este programa de la ROM. En algunos computadores el programa se ejecuta directamente en la ROM; en otros el programa se copia de la ROM a la memoria principal y se ejecuta allí. Sin embargo algunas máquinas no tienen ese almacenamiento sólo de lectura, además puede no ser conveniente cambiar un programa en ROM si es necesario hacer modificaciones en el cargador absoluto.

Una solución inmediata es tener una función de Hardware incorporada que lea un registro de longitud fija de algún dispositivo en alguna localidad fija de la memoria. Una vez completa la operación de lectura, el control se pasa automáticamente a la dirección de la memoria dónde se almaceno el registro. Este registro contiene instrucciones de máquina que cargan el programa absoluto que sigue. Si el proceso de carga requieren de más instrucciones de las que pueden leerse en un solo registro, el primer registro causa la lectura de otros y estos a su vez pueden originar la lectura aún de más registros de ahí el término arranque.



*Figura 2: El Sistema Operativo siempre inicia su cargado en la misma región de memoria.*

La carga absoluta necesita que el módulo de carga ocupe siempre la misma posición de la memoria principal. Así pues, todas las referencias del módulo de carga para el cargador deben ser direcciones específicas ó absolutas en memoria principal.

La asignación de direcciones específicas a la referencia a memoria de un programa pueden ser realizada tanto por el programador como el tiempo de compilación o ensamblaje. Con el primer método se tienen varias desventajas. En primer lugar todos los programadores tendrán que conocer la estrategia de asignación deseada para situar los módulos en memoria principal. En segundo lugar si hace alguna modificación en el programa que suponga inserciones ó borrados en el cuerpo del módulo tendrán que cambiarse todas las direcciones. Por consiguiente es preferible permitir que las referencias a memoria dentro de los programas se expresen simbólicamente y que se resuelvan en el momento de la compilación o el ensamblaje. Así cuando se presente el módulo para la entrada a un cargador absoluto, el ensamblador o el compilador convertirán todas estas referencias direcciones específicas.

## 1.5 Proceso de carga relocizable

En la relocación se establece la correspondencia entre las direcciones usadas en el programa (direcciones virtuales) y direcciones físicas cada vez que se utilizan durante la ejecución del programa. Las direcciones lógicas generadas por el ensamblador y el enlazador no se alteran durante la carga.

## 2 Ligadores

A continuación se describe la importancia, las funciones y los tipos de ligadores.

### 2.1 Funciones del ligador

Un ligador es un programa de sistema que combina dos o mas programas objeto separados y permite que se hagan referencias unos a otros, o sea, que cada uno de estos programas pueda hacer referencia a código ó variables de los otros programas con los que está enlazado. En muchos programas el cargador

hace la labor del programa de enlace, porque existe solo un “linking loader” y no existe programa de enlace independiente. Es importante señalar que no se necesita un programa de enlace ni un cargador separado para cada traductor en el sistema, ya que estos programas trabajan con el programa objeto, sin importar el lenguaje fuente. Por otro lado es importante que cada compilador o ensamblador produzca el programa objeto usando el mismo formato.

Algunos sistemas tienen un ligador para realizar las operaciones de enlaces y un cargador separado para manejar la relocación y la carga.

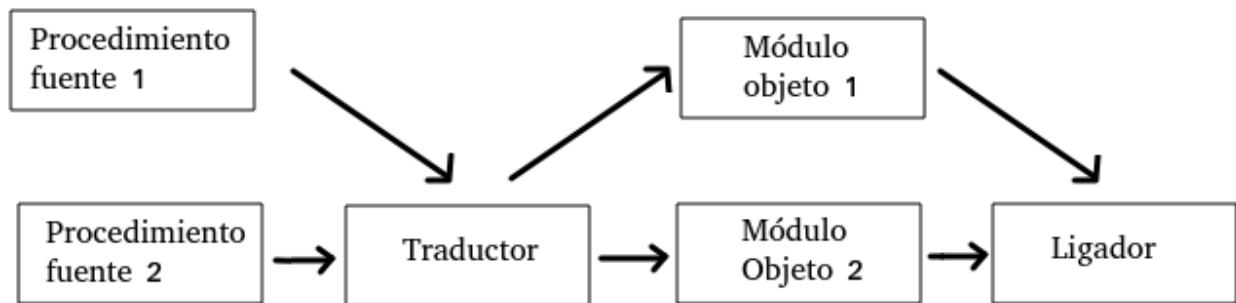


Figura 3: El ligador trabaja con archivos objeto.

## 2.2 Tipos de ligadores

Hay dos tipos de ligadores: ligadores estáticos y ligadores dinámicos. A continuación se describen.

### 2.2.1 Ligador estático

Cuando se utilizan subrutinas en un programa, el código ejecutable de cada una de ellas debe encontrarse en memoria al tiempo de ejecución. Para esto, antes de cargar un programa, debe ligarse su código objeto con los códigos objeto (guardados en uno o más archivos) de cada una de las subrutinas invocadas por él, obteniendo así un programa ejecutable que contiene tanto el código del módulo invocador como el código de los módulos invocados.

En este punto, es posible guardar el resultado del proceso de liga en un archivo que podrá ser utilizado por un cargador, o el mismo programa ligador puede también realizar la tarea de carga. Esto último evita el tener que guardar el código ejecutable en un archivo, con lo que se ahorra espacio en disco.

Este ahorro de espacio en disco se paga con el tiempo gastado al tener que ligar todos los módulos cada vez que se necesite ejecutar el programa.

### 2.2.2 Ligador dinámico

El ligado dinámico ofrece algunas ventajas sobre los otros tipos de ligado. Proporciona la posibilidad de cargar las rutinas sólo cuando y si se necesitan. Si las subrutinas son grandes o tienen muchas referencias externas, se pueden conseguir ahorros considerables de tiempo y espacio de memoria.

De forma similar, supóngase que en cualquier ejecución un programa usa sólo pocas de una gran cantidad de subrutinas posibles, pero el número exacto de rutinas necesarias no puede predecirse hasta que el programa examina su entrada.

Esta situación podría presentarse, con un programa que permita al usuario llamar interactivamente a cualquiera de las subrutinas de una gran biblioteca matemática y estadística. El usuario podría suministrar la entrada de datos desde un terminal de tiempo compartido, y los resultados podrían



exhibirse en el terminal. En este caso podrían ser necesarias todas las subrutinas de la biblioteca, pero en cualquier sesión de terminal solo se usarían unas cuantas.

El ligado dinámico evita la necesidad de cargar la biblioteca completa para cada ejecución. El ligado dinámico puede incluso hacer innecesario que el programa conozca el conjunto de subrutinas que se podría utilizar. El nombre de la subrutina se trataría simplemente como otro elemento de entrada. Para realizar la carga de ligado de una subrutina llamada se puede utilizar varios mecanismos distintos.

En el método que se analiza aquí, las rutinas que se carguen dinámicamente deben llamarse por medio de una solicitud de servicio al sistema operativo. Este método también podría considerarse como una solicitud a una parte del cargador que se mantiene en la memoria durante la ejecución del programa. Cuando se utiliza ligado dinámico, la asociación de una dirección real y el nombre simbólico de la rutina llamada no se hace hasta que se ejecuta la proposición llamada

El Ligado dinámico ofrece algunas ventajas sobre los tipos de ligado. Proporciona la posibilidad de cargar las rutinas sólo cuando si se necesitan. Si las subrutinas son grandes ó tienen muchas referencias externas se pueden conseguir ahorros considerables de tiempo y espacio en memoria. El ligado dinámico evita la necesidad de cargar la biblioteca completa para cada ejecución. Puede incluso hacer innecesario que el programa conozca el conjunto de subrutinas que se podría utilizar.

El nombre de la subrutina se trataría simplemente como otro elemento de entrada. En el método que se utilice aquí las rutinas que se carguen dinámicamente deben llamarse por medio de una solicitud del servicio al sistema operativo. Este método también podría considerarse como una solicitud a una parte del cargador que se mantiene en la memoria durante la ejecución del programa.

Cuando se utiliza ligado dinámico, la asociación de dirección real y el nombre simbólico de la rutina llamada no se hace hasta que se ejecuta la proposición llamada

Ligado Dinámico.

- El ligado se pospone para el tiempo de ejecución del proceso
- Se requiere de un pequeño código para localizar la rutina residente en memoria.
- Este código se modifica así mismo, de tal manera de poder invocar y ejecutar la rutina.
- El SO requiere comprobar si la rutina se encuentra en el espacio de memoria del proceso.
- El ligado dinámico es muy útil para bibliotecas de sistema, también conocidas como bibliotecas compartidas (shared libraries)

### 3 Referencias

<https://programaciondesistemas2015.wordpress.com/cargadores/>

<http://xikotenkaltsb.blogspot.com/2011/04/41-cargadores-y-ligadores.html>

[https://www.slideshare.net/MireyaMoralesHernandez/cargadores-14941212?next\\_slideshow=1](https://www.slideshare.net/MireyaMoralesHernandez/cargadores-14941212?next_slideshow=1)