Module 9

Linking - Loading

Module Nine

- Linking – Loading - Part Four

- In this presentation, we are going to talk about :

- Linker - Loader Features

Overview

- Previously we talked about:

    Program Loading

    Link-Editor

    Relocation

- Now: Linker - Loader Features

More Linker-Loader features

- Machine Independent Loader Features

  Library Search

  Loader Options

- Design Options

Library Search

- Automatic Library Search

  - Standard Library - input / output routines for example.

  - Programmer uses as if part of the programming language.

  - The linking loader searches to satisfy all the unresolved references in the external symbol table.

Library Search

- Automatic Library Search

  - Programmer can create own versions of library routines by choosing the identical name.

  - May require multiple searches.

  - Library has a directory; name of routine, and pointer to address within the library file.

Loader Options

- Directives for the Loader

    - Command language to specify the options to be used.

    - LIBRARY - Specify library file for loader to search

    - INCLUDE - Loader to read from indicated file.

    - DELETE - Do not load the named control section.

Loader Options

- Directives for the Loader

    - Command language to specify the options to be used.

    - CHANGE - Rename the control section.

    - MAP - Produce a listing report of loaded program

    - NOCALL - Do not resolve these references.

Design Options

- **Linkage Editor** - builds the executable image or load module.

- **Dynamic Linking** - load routine only when or if it is needed.
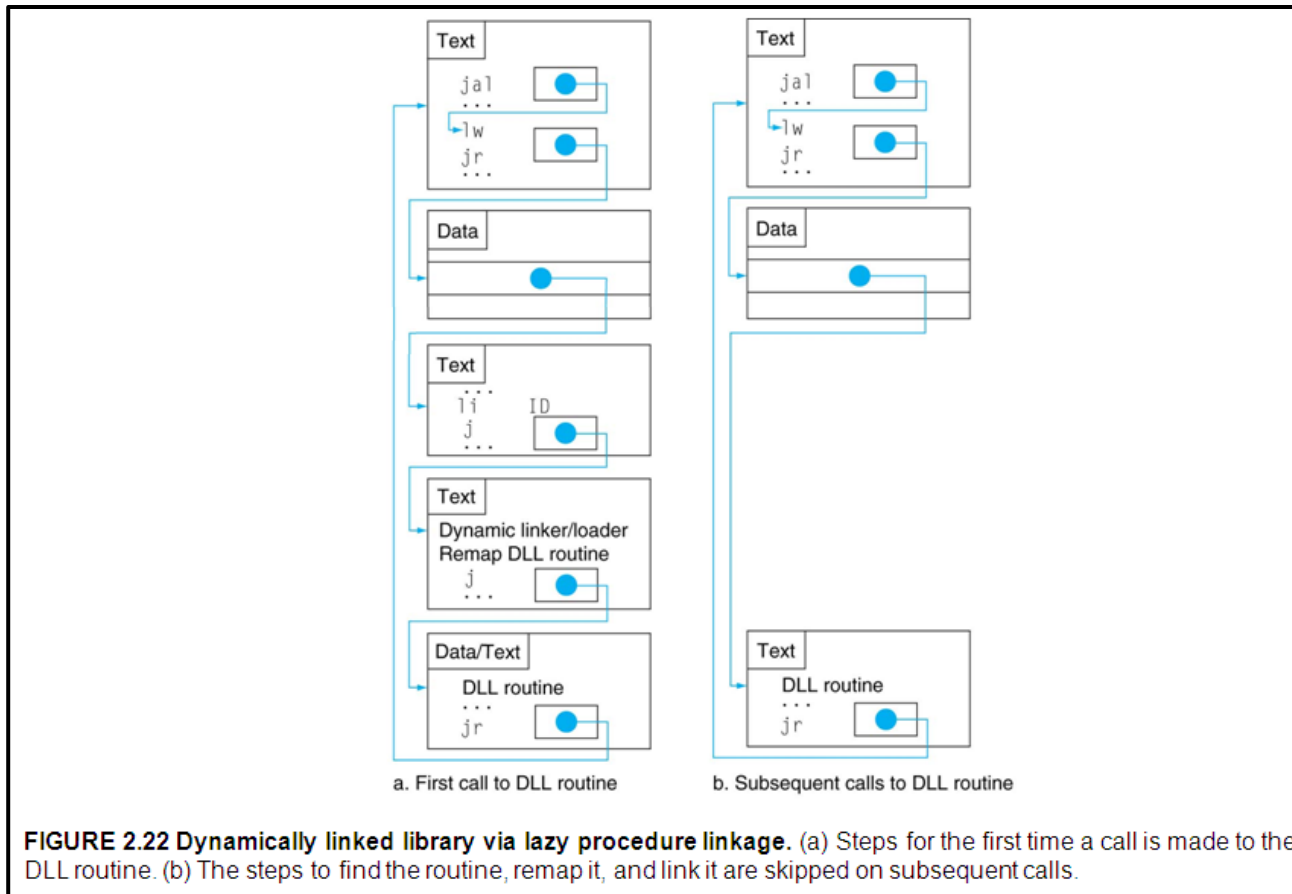
Linkage Editors

- Linking done once.   Stored on the disc.

- Relocation at every load into memory. Uses the Relocating Loader to copy to memory.

- All addresses needing modification are relative to start address. Uses simple one pass to copy into memory.

- Build runtime routine.

- More flexible, more control, more complex than Linking Loader.

Dynamic Linking

- Load the object program (sub-section) as it is needed.

- Operating System service request.

- Allows very flexible use of memory and object modules.

- It is expensive in terms of execution time.

- Dynamic Link Library

- Load few routines from large library

- Load rarely used routines

Dynamic Linking

a. First call to DLL routine

b. Subsequent calls to DLL routine

FIGURE 2.22 Dynamically linked library via lazy procedure linkage. (a) Steps for the first time a call is made to the DLL routine. (b) The steps to find the routine, remap it, and link it are skipped on subsequent calls.

Summary

- Linker - Loader

  - Basic Functions
  - Relocation
  - Linking
  - Algorithm

- Machine Independent Features

  - Library Search
  - Loader Options