



# Module 9

Linking - Loading



# Module Nine

- Linking – Loading - Part Three
- In this presentation, we are going to talk about :
- Program Relocation



# Overview

- Previously we talked about:
  - Program Loading
  - Link-Editor
    - Basic Functions
    - Algorithm
- Now: Relocation

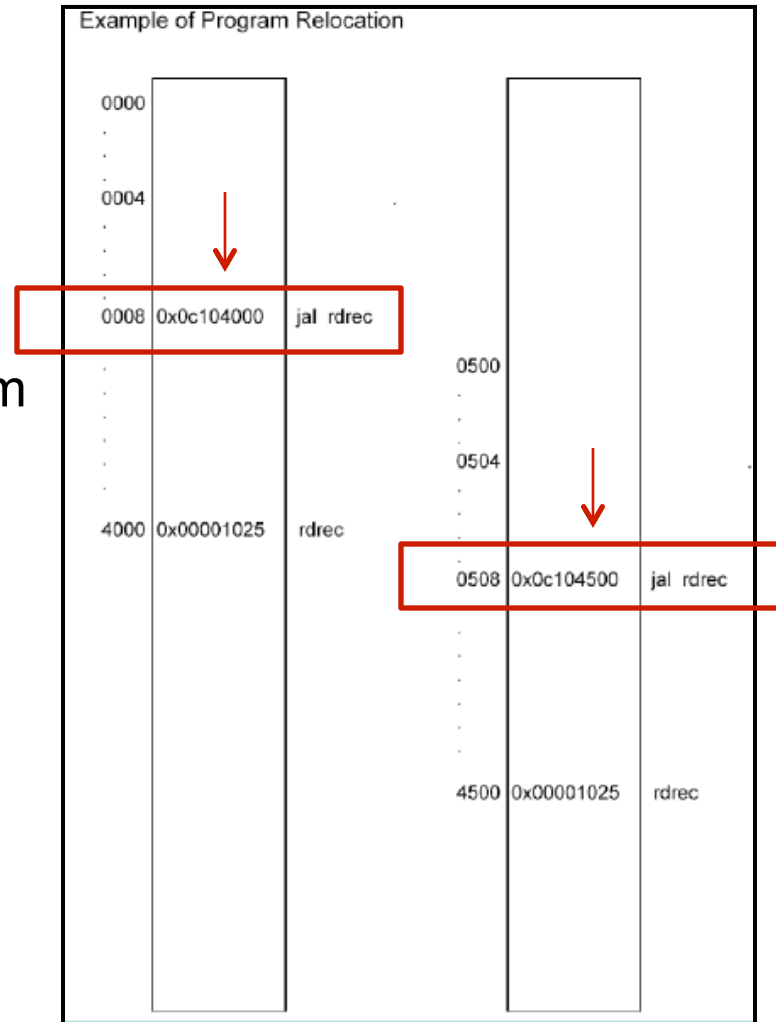


# Relocation

- Here is a definition:
  - Modify the object program so that it can be loaded at any address in memory.
- Relocating Loader or Relative Loader

# Program Relocation

- Need to be able to place a program anywhere in memory
- Small special processors
  - Space for no more than one program
- Systems with large memory
  - Space for several programs
- Need to provide information to loader
- Address issue



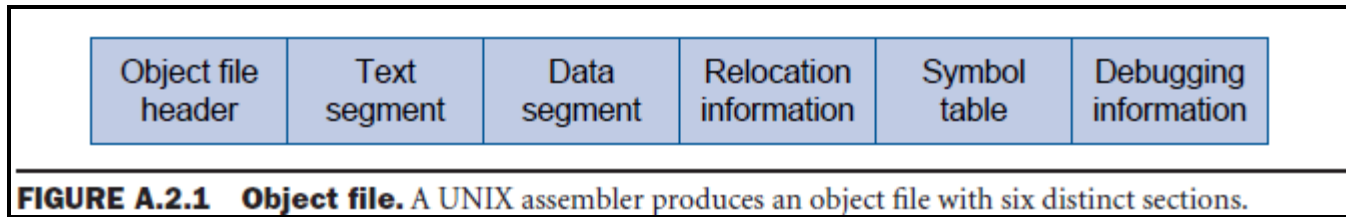


# The Assembler Knows !

- Absolute value - data constants
- Relative value - instruction address, data address
- Relocatable Program - has information needed to change the addresses
- Relocation Record - part of the Object Program file
  - Tells Loader which memory locations need to be modified and by what amount

# Object File

- Make use of the Relocation and Symbol records.



- Text - machine language code for instructions in source file.
- Data - binary representation of the data in source file.
- Relocation - identifies instructions and data that depend on absolute addresses. The references must change when the program is moved in memory.
- Symbol table - external labels defined in source, and unresolved label references.



# Relocation

- Relocating Loader or Relative Loader
  - Modify the object program so that it can be loaded at any address in memory.
- Relocation records

RELO location size + amount of relocation

Relative Address program does not need as many R records

RELO 001036 04 + PROGA





# Algorithm

- Process      PASS ONE - Build the External Symbol Table
    - Get **Program Address** value from Operating system
    - Set **SubPgmAddress** equal to **Program Address**
    - Read Header record
      - Add Name to the **ExtSymTable** with SubPgmAddress value
    - Read Define Symbol record
      - Add name to the **ExtSymTable** with SubPgmAddress value plus relative value from the record
- At end of subroutine update SubPgmAddress with length of subroutine; read next set of records.



# Algorithm

- Process      PASS TWO – Load the program into Memory
  - Get **Program Address** value from Operating system
  - Read Header record
  - Read Text records
    - Copy code values to specified locations in memory
  - Read Relocation records
    - Look-up symbols in **ExtSymTable**
    - Add the symbol value to specified location in memory
  - Read End record
    - Jump to first executable instruction



# Relocation

- Hardware
  - Some machines use hardware base address registers. The program is loaded into memory and the address is copied to the hardware base address register.
  - The program is assembled as if it were loaded at memory address zero. All real addresses are calculated from the software address plus the hardware base address value.



# Relocation

- Software
  - Some machines use virtual memory.
  - The program is assembled as if it were loaded at memory address zero. All real addresses are calculated from the program address plus the virtual memory address.
  - More about virtual memory in the week twelve presentation.



# Summary

- Program Relocation
- Additional Object program record
- Algorithm
- Next: Linker - Loader Features