Linking Libraries with Overlays

Overview

- Extracting the modules
- Preparing the modules
- Linking in the modules
- Example

Extracting the Modules

- Identify target modules
  - Psylib /l
- Separate target modules
  - Psylib /x

Preparing the Modules

- Use *prefsect.exe*
  - Renames object file
  - Assigns a prefix to all sections of file
  - Can create a group and place sections in it
  - Can be used in a makefile

Linking the Modules

- Assign each new section to the appropriate group
- Add an *include* statement for each new object
- Remove the *inclib* statement for the pertinent library

Example-Libpress

- Why libpress?
  - It’s fairly large, particularly the encoding / decoding tables
  - Unless you’re streaming compressed data, you don’t want or need it in RAM all of the time, so it’s easy to relegate to an overlay
  - The number of modules is very small, so the extraction / linking process is easier
  - Everyone asked about it last time!

Psylib

- Identify the modules
  - *psylib /l libpress.lib*
  - Who needs ENCSPU?
- Extract the modules
  - *psylib /x libpress.lib libpress vlc table*

Prefsect

- Give each a unique prefix
  - *prefsect libpress.obj mypress.obj mypress*
  - *prefsect vlc.obj myvlc.obj myvlc*
  - *prefsect table.obj mytable.obj mytable*
- In a makefile:

```
mypress.obj : libpress.obj
        prefsect libpress.obj mypress.obj mypress
```

The linker file

- For each module, assign all of the new sections to the overlay group
    - Each module may not have the full assortment of sections

```
section    mypress.rdata,movie
section    mypress.text,movie
section    mypress.data,movie
section    myvlc.text,movie
section    mytable.data,movie
```

The linker file

- For each module, insert an *include* statement with the new object name

```
include    mypress.obj
include    myvlc.obj
include    mytable.obj
```

The linker file

- Delete the *inclib* line that refers to libpress

```
...
inclib      “c:\ps\psx\lib\libgte.lib”
inclib      “c:\ps\psx\lib\libgpu.lib”
~~inclib      “c:\ps\psx\lib\libpress.lib”~~
inclib      “c:\ps\psx\lib\libcard.lib”
...
```

What happens now?

- The good news:
  - Main code is about 70k smaller
- The bad news:
  - If your movie code was already your largest overlay, you haven’t really gained anything

“Non-Overlay” overlays

What are they?

- Segments of code that are not always in memory
- When they *are* in memory, they reside in an area not normally used for code--the heap
- As such, they technically are *not* overlays, but the process of implementing them is similar

Why use them?

- Take advantage of temporarily unused memory
- Leave all of your main program in RAM
- Useful for small pieces of code where the size/importance does not warrant a regular overlay

How are they made?

- Compiler switches
  - Same as for normal overlays
- Linker file
  - Put the group after the main program’s bss group
  - Or use the *org* directive to place the group in a manner to allow partial use of the heap

```
extra     group     file(extra.bin),org($80140000)
```

How are they loaded?

- Loaded same as a normal overlay
- Address file not really needed
  - Can use *__heapbase*, which is set to end of main bss section
  - Use the addresss specified with the*org* directive