

Note: This API calls are shared between DOS and Win16 personality.

DPMI is a shared interface for DOS applications to access Intel 80286+ CPUs services. DOS DMPI host provides core services for protected mode applications. Multitasking OS with DOS support also provides DMPI in most cases. Windows standard and extended mode kernel is a DPMI client app. Standard and extended mode kernel differs minimally and shares common codebase. Standard Windows kernel works under DOSX extender. DOSX is a specialized version of 16-bit DPMI Extender (but it is standard DPMI host). Standard mode is just DPMI client, exnhanced mode is DPMI client running under Virtual Machime Manager (really, multitasker which allow to run many DOS sessions). Both modes shares DPMI interface for kernel communication. The OS/2 virtual DOS Protected Mode Interface (VDPMI) device driver provides Version 0.9 DPMI support for virtual DOS machines. Win16 (up to Windows ME) provides Version 0.9 DPMI support. Windows in Standard Mode provides DPMI services only for Windows Applications, not DOS sessions.

DPMI host often merged with DPMI extender. Usually DPMI extender provide DPMI host standard services and DOS translation or True DPMI services.

2021/08/05 10:15 · prokushev · 0 Comments

Int 31H, AH=05H, AL=05H

Version

1.0

Brief

Resize Linear Memory Block

Input

AX = 0505H ESI = memory block handle ECX = new block size (bytes, must be nonzero) EDX = flags

Bit	Significance
0	0 = create uncommitted pages 1 = create committed pages
1	0 = do not update segment descriptors 1 = segment descriptor update required
2-31	reserved, must be zero

```
and, if bit 1 of EDX is set (1):
ES:EBX = selector:offset of a buffer containing an array of selectors, 1
word (16 bits) per selector
```

```
EDI = count of selectors in array
```

Return

```
if function successful
Carry flag = clear
EBX = new linear base address of memory block
ESI = new handle for memory block

if function unsuccessful
Carry flag = set
AX = error code
8001H   unsupported function (16-bit host)
8012H   linear memory unavailable
8013H   physical memory unavailable
8014H   backing store unavailable
8016H   handle unavailable
8021H   invalid value (ECX=0)
8023H   invalid handle (in ESI)
```

Notes

Changes the size of a memory block that was previously allocated with the Allocate Linear Memory Block function (Int 31H Function 0504H).

A DPMI 1.0 host that is 16-bit only will not support this function.

A 16-bit client of a 32-bit DPMI 1.0 host can use this function.

After this function returns, the previous handle for the memory block is invalid and should not be used.

If this function fails, the block's size and base address are always unmodified.

If the size of the block is increased, the new pages are committed or uncommitted according to the value of bit 0 of EDX, and the block's linear base address may change. If the size of the block is decreased, pages at the end of the block are freed, and the block's base address is always unchanged.

If the block's linear base address is changed by this function, and the function was called with bit 1 of EDX set (1), the DPMI host updates the descriptors f or each of the segments in the update list which fall within the memory block. Descriptors for segments which do not fall within the memory block are not modified. Expand-up segments fall within the memory block if the segment base is within the block. Expand-down segments fall within the memory block if the (segment base + the limit - 1) is within the block. In either case, the segment base is modified by the distance the block moves, and the segment limit is not changed. The moving of the memory block and the updating of descriptors is performed atomically; i.e. the host will not deliver any hardware interrupts to the client during the update.

Int 31H Function 0503H, which also resizes linear memory blocks, does not necessarily page-align blocks and cannot create uncommitted pages or update descriptors.

See also

Note

Text based on http://www.delorie.com/djgpp/doc/dpmi/

DPMI	
Process manager	**INT 2FH** 1680H, 1687H
Signals	
Memory manager	
Misc	**INT 2FH** 1686H, 168AH
Devices	

2021/08/13 14:23 · prokushev · 0 Comments

From:
https://cocorico.osfree.org/doku/ - **osFree wiki**

Permanent link:
https://cocorico.osfree.org/doku/doku.php?id=en:docs:dpmi:api:int31:05:05

Last update: **2021/08/27 04:24**