 This is part of **Family API** which allow to create dual-os version of program runs under OS/2 and DOS

Note: This is legacy API call. It is recommended to use 32-bit equivalent

2021/09/17 04:47 · prokushev · 0 Comments
2021/08/20 03:18 · prokushev · 0 Comments

MouSetPtrShape

This call allows a process to set the pointer shape and size to be used as the mouse device driver pointer image for all applications in a session.

Syntax

```
MouSetPtrShape (PtrBuffer, PtrDefRec, DeviceHandle)
```

Parameters

- PtrBuffer (PBYTE) - input: Address of a buffer containing the bit image used by the mouse device driver as the pointer shape for that session. The buffer consists of AND and XOR pointer masks in a format meaningful to the pointer draw device driver.

For CGA compatible text modes (0, 1, 2, and 3) the following describes the AND and XOR pointer mask bit definitions for each character cell of the masks. Bit values are:

Bit	Description
15	Blinking
14-12	Background colour
11	Intensity
10-8	Foreground colour
7-0	Character

- PtrDefRec (PPTRSHAPE) - input: Address of the structure where the application stores the necessary data for the pointer draw device driver to build a row-by-column image for each bit plane for the current display mode.

Programming Note: For other custom displays and for the extended modes of the EGA attachment, it is possible to set the display to modes that require multiple bit planes. In these cases, the area sized by the row and column limits must be repeated for each bit plane supported in that mode. Consequently, the calling process must supply enough data to allow the mouse device driver to draw the pointer shape on all currently supported bit planes in that session. For text modes, row and column offset must equal 0.

- DeviceHandle (HMOU) - input: Contains the handle of the mouse device obtained from a previous MouOpen.

Return Code

rc (USHORT) - return:Return code descriptions are:

- 0 NO_ERROR
- 385 ERROR_MOUSE_NO_DEVICE
- 387 ERROR_MOUSE_INV_PARMS
- 466 ERROR_MOU_DETACHED
- 501 ERROR_MOUSE_NO_CONSOLE
- 505 ERROR_MOU_EXTENDED_SG

Remarks

An application passes a data image to the mouse device driver that the mouse driver applies to the screen whenever the logical pointer position is not located in the application-defined collision area. The application synchronizes use of the screen with the mouse driver by way of MouRemovePtr and MouDrawPtr.

The pointer shape is dependent on the display device driver used to support the display device. OS/2 supports text and graphics modes. These modes are restricted to modes 0 through 7, depending on the display device. Character modes (modes 0, 1, 2, 3, and 7) support the pointer cursor only as a reverse block character. This reverse block character has a character height and width equal to 1.

The pointer shape is mapped by the Pointer Draw Device Driver and determined completely by the application. The height and width may vary from 1 through the pel size of the display screen. For restrictions concerning the Pointer Draw Device Driver, see IBM Operating System/2 Version 1.2 I/O Subsystems And Device Support Volume 1.

Bindings

C

```
typedef struct _PTRSHAPE { /* moups */
  USHORT cb;               /* total length necessary to build image */
  USHORT col;              /* # of columns in mouse shape */
  USHORT row;              /* number of rows in mouse shape */
  USHORT colHot;           /* column coordinate of pointer image hotspot */
  USHORT rowHot;           /* row coordinate of pointer image hotspot */
} PTRSHAPE;

#define INCL_MOU

USHORT  rc = MouSetPtrShape(PtrBuffer, PtrDefRec, DeviceHandle);
```

```
PBYTE      PtrBuffer;     /* Pointer shape buffer */
PPTRSHAPE  PtrDefRec;     /* Pointer definition record */
HMOU       DeviceHandle;  /* Mouse device handle */

USHORT     rc;            /* return code */
```

MASM

```
PTRSHAPE struc
  moups_cb      dw  ? ;total length necessary to build image
  moups_col     dw  ? ;# of columns in mouse shape
  moups_row     dw  ? ;number of rows in mouse shape
  moups_colHot  dw  ? ;column coordinate of pointer image hotspot
  moups_rowHot  dw  ? ;row coordinate of pointer image hotspot
PTRSHAPE ends

EXTRN  MouSetPtrShape:FAR
INCL_MOU            EQU 1

PUSH@  OTHER   PtrBuffer     ;Pointer shape buffer
PUSH@  OTHER   PtrDefRec     ;Pointer definition record
PUSH   WORD    DeviceHandle  ;Mouse device handle
CALL   MouSetPtrShape

Returns WORD
```

Family API		
DOS	Process Manager	DosBeep DosExit DosSleep DosExecPgm
	File Manager	DosChDir DosChgFilePtr DosClose DosDelete DosDupHandle DosMkDir DosMove DosQCurDir DosQCurDisk DosSetFileMode DosOpen DosQFileInfo DosRead DosQFileMode DosQFSInfo DosQVerify DosRmDir DosSelectDisk DosFindClose DosFindFirst DosFindNext DosSetFileInfo DosSetVerify DosWrite DosFileLocks DosSetFHandState DosNewSize DosBufReset DosQFHandState DosSetFSinfo DosShutdown
	Memory Manager	DosFreeSeg DosSubAlloc DosSubFree DosSubSet DosAllocHuge DosAllocSeg DosReallocHuge DosReallocSeg DosGetHugeShift DosCreateCSAlias
	NLS	DosCaseMap DosGetCtryInfo DosGetDBCSEv DosSetCtryCode DosGetCollate DosGetMessage DosInsMessage DosPutMessage
	Date and Time	DosSetDateTime DosGetDateTime
	Devices	DosDevConfig DosDevIOCtl DosDevIOCtl2
	Signals	DosHoldSignal DosSetSigHandler
	Misc	BadDynLink DosGetEnv DosGetMachineMode DosGetVersion DosError DosErrClass DosSetVec
KBD		KbdCharIn KbdFlushBuffer KbdGetStatus KbdSetStatus KbdStringIn KbdPeek

Family API	
VIO	VioGetBuf VioGetConfig VioGetCurPos VioGetCurType VioGetPhysBuf VioReadCellStr VioReadCharStr VioScrollUp VioScrollDn VioScrollLf VioScrollRt VioScrUnLock VioSetCurPos VioSetCurType VioSetMode VioGetMode VioShowBuf VioWrtCellStr VioWrtCharStr VioWrtCharStrAtt VioWrtNAttr VioWrtNCell VioWrtNChar VioWrtTTY VioScrLock VioPopUp
Tools	BIND
Modules	DOSCALLS.DLL VIOCALLS.DLL KBDCALLS.DLL MSG.DLL
Libraries	API.LIB OS2386.LIB FAPI.LIB DOSCALLS.LIB SUBCALLS.LIB

2018/08/25 15:05 · prokushev · 0 Comments

From:
http://osfree.org/doku/ - **osFree wiki**

Permanent link:
http://osfree.org/doku/doku.php?id=en:docs:fapi:mousetptrshape

Last update: **2021/11/04 13:11**