SetThreadExecutionState function

Enables an application to inform the system that it is in use, thereby preventing the system from entering sleep or turning off the display while the application is running.

Syntax

C++

```
EXECUTION_STATE WINAPI SetThreadExecutionState(
  _In_ EXECUTION_STATE esFlags
);
```

Parameters

esFlags [in]

The thread's execution requirements. This parameter can be one or more of the following values.

Value	Meaning
ES_AWAYMODE_REQUIRED	Enables away mode. This value must be specified with **ES_CONTINUOUS**.

0x00000040	Away mode should be used only by media-recording and media-distribution applications that must perform critical background processing on desktop computers while the computer appears to be sleeping. See Remarks.
ES_CONTINUOUS 0x80000000	Informs the system that the state being set should remain in effect until the next call that uses **ES_CONTINUOUS** and one of the other state flags is cleared.
ES_DISPLAY_REQUIRED 0x00000002	Forces the display to be on by resetting the display idle timer.
ES_SYSTEM_REQUIRED 0x00000001	Forces the system to be in the working state by resetting the system idle timer.
ES_USER_PRESENT 0x00000004	This value is not supported. If **ES_USER_PRESENT** is combined with other *esFlags* values, the call will fail and none of the specified states will be set.

Return value

If the function succeeds, the return value is the previous thread execution state.

If the function fails, the return value is **NULL**.

Remarks

The system automatically detects activities such as local keyboard or mouse input, server activity, and changing window focus. Activities that are not automatically detected include disk or CPU activity and video display.

Calling **SetThreadExecutionState** without **ES_CONTINUOUS** simply resets the idle timer; to keep the display or system in the working state, the thread must call **SetThreadExecutionState** periodically.

To run properly on a power-managed computer, applications such as fax servers, answering machines, backup agents, and network management applications must use both **ES_SYSTEM_REQUIRED** and **ES_CONTINUOUS** when they process events. Multimedia applications, such as video players and presentation applications, must use **ES_DISPLAY_REQUIRED** when they display video for long periods of time without user input. Applications such as word processors, spreadsheets, browsers, and games do not need to call **SetThreadExecutionState**.

The **ES_AWAYMODE_REQUIRED** value should be used only when absolutely necessary by media applications that require the system to perform background tasks such as recording television content or streaming media to other devices while the system appears to be sleeping. Applications that do not require critical background processing or that run on portable computers should not enable away mode because it prevents the system from conserving power by entering true sleep.

To enable away mode, an application uses both **ES_AWAYMODE_REQUIRED** and **ES_CONTINUOUS**; to disable away mode, an application calls **SetThreadExecutionState** with **ES_CONTINUOUS** and clears **ES_AWAYMODE_REQUIRED**. When away mode is enabled, any operation that would put the computer to sleep puts it in away mode instead. The computer appears to be sleeping while the system continues to perform tasks that do not require user input. Away mode does not affect the sleep idle timer; to prevent the system from entering sleep when the timer expires, an application must also set the **ES_SYSTEM_REQUIRED** value.

The **SetThreadExecutionState** function cannot be used to prevent the user from putting the computer to sleep. Applications should respect that the user expects a certain behavior when they close the lid on their laptop or press the power button.

This function does not stop the screen saver from executing.

Examples

C++

```cpp
// Television recording is beginning. Enable away mode and prevent
// the sleep idle time-out.
//
SetThreadExecutionState(ES_CONTINUOUS | ES_SYSTEM_REQUIRED | ES_AWAYMODE_REQUIRED);
```

```
//
// Wait until recording is complete...
//

//
// Clear EXECUTION_STATE flags to disable away mode and allow the system to idle to sleep normally.
//
SetThreadExecutionState(ES_CONTINUOUS);
```

Requirements

Minimum supported client	Windows XP [desktop apps only]
Minimum supported server	Windows Server 2003 [desktop apps only]
Header	Winbase.h (include Windows.h)
Library	Kernel32.lib
DLL	Kernel32.dll

See also

Power Management Functions
SetSuspendState
SetSystemPowerState
WM_POWERBROADCAST