

# Re: How to manually copy managed texture from systemem to real vide

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.directx.graphics/2005-07/ms>

---

- *From:* "Robert Dunlop [MS MVP]" <[rdunlop@xxxxxxxx](mailto:rdunlop@xxxxxxxx)>
  - *Date:* Sun, 10 Jul 2005 05:40:57 -0700
- 

"Igor Kokarev" <[IgorKokarev@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:IgorKokarev@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)> wrote in message [news:74648F16-66C4-405D-9364-2F44B0E04B3C@xxxxxxxxxxxxxxxxxxxx](mailto:news:74648F16-66C4-405D-9364-2F44B0E04B3C@xxxxxxxxxxxxxxxxxxxx)

> However it doesn't fully solve the problem.  
> I have working 3D scene.  
> In another separated thread (even with more lower priority) I call PreLoad  
> for my new future texture (1024x768). And I watch that in moment of this  
> call, objects in my scene jerk for a moment.  
>  
> I completed another test. I devided my large texture to 8 small textures.  
> Because of this I call PreLoad 8 times for each texture.  
> And I have SAME RESULT with jerking.  
>  
> After this I added Sleep(5) after each PreLoad call and now loading of  
> textures don't affect to 3D scene – it draws OK.  
>  
> I thoughted maybe my main thread stops during PreLoad? I added counter and  
> it shown me that it works without any skips. So I do summary, PreLoad  
> method  
> or first show of textured quad (if you don't make PreLoad call) add delay  
> and  
> skip several frames.

If you are doing this from a separate thread, I suspect you are running into a situation where the current frame is unable to complete and subsequently be presented until all of the texture transfers have been completed. If loading textures to video memory in a separate thread, I would consider using a mutex or critical section to interlock these operations so that PreLoad() always occurs outside of the frame rendering loop:

```
HANDLE hMutex=CreateMutex(...
```

```
....Rendering code...  
WaitForSingleObject(hMutex...  
Clear()  
BeginScene()  
....render  
EndScene()
```

Re: How to manually copy managed texture from systemem to real vide

Present()

ReleaseMutex(...

(Note: may want to release mutex before present, would have to test to see what works best...)

....Loader code...

```
for (i=0;i<8;i++) {  
WaitForSingleObject(hMutex...  
texture[i]->PreLoad(...  
ReleaseMutex(...  
Sleep(0);  
}
```

Note that using Sleep() with a timeout of 0 allows any threads of equal or greater priority that are ready to run to do so, or returns immediately if there are no pending threads. This would allow your main rendering thread to render the next frame if it is ready to do so.

No guarantees that this will solve the problem, but from past experience I think this may help. I've found it best to isolate access to device resources to the main thread, and if it is beneficial to do otherwise then keeping such operations from overlapping with the rendering of the scene can avoid some nasty conflicts.

Also, are you using the D3DCREATE\_MULTITHREADED flag when creating your device?

---

Robert Dunlop

The X-Zone

<http://www.directxzone.com/>

Microsoft DirectX MVP

-----  
The opinions expressed in this message are my own personal views and do not reflect the official views of the Microsoft Corporation.

The MVP program does not constitute employment or contractual obligation with Microsoft.

.

---

• *Follow-Ups:*

◆ **Re: How to manually copy managed texture from systemem to real vide**

◇ From: Igor Kokarev

• *References:*

◆ **How to manually copy managed texture from systemem to real video me**

◇ From: Igor Kokarev

◆ **Re: How to manually copy managed texture from systemem to real video me**

◇ From: Gabest

Re: How to manually copy managed texture from systemem to real vide

◆ **[Re: How to manually copy managed texture from systemem to real vide](#)**

◇ From: Igor Kokarev

◆ **[Re: How to manually copy managed texture from systemem to real vide](#)**

◇ From: Igor Kokarev

- Prev by Date: **[Re: box from frustum](#)**
- Next by Date: **[Re: Memory leak from D3DXCreateTextureFromFile call](#)**
- Previous by thread: **[Re: How to manually copy managed texture from systemem to real vide](#)**
- Next by thread: **[Re: How to manually copy managed texture from systemem to real vide](#)**
- Index(es):
  - ◆ **[Date](#)**
  - ◆ **[Thread](#)**