

Re: Once and for all! Fastest way to load large jpeg

## Re: Once and for all! Fastest way to load large jpeg

---

*Source:* <http://coding.derkeiler.com/Archive/Delphi/alt.comp.lang.borland-delphi/2006-12/msg00214.html>

---

- *From:* [erewhon@xxxxxxxxxxx](mailto:erewhon@xxxxxxxxxxx) (J French)
  - *Date:* Thu, 14 Dec 2006 09:44:29 GMT
- 

On 13 Dec 2006 21:42:50 -0800, jimbo@xxxxxxxxxxx wrote:

<snip>

How about kicking off a separate App that generates the thumbnails in the back ground, that way you'll be using the processor while the user is twiddling with the mouse.

The thumbs are no problem.

<snip>

User chooses folder from directorylistbox populating a filelistbox.  
I create an object for each entry with path and other relevant info and add it to an array with a common index to the filelistbox. So I use whichever is convenient.

Where exactly are those JPEGs coming from

CD USB or any external drive.

How long do you want to keep the JPEGs

Re: Once and for all! Fastest way to load large jpeg

In practice only long enough to assign it to the larger  
TImage. I free  
it immediately. See original example.

<snip>

Could you explain the 4mb  
– do you mean that a 4mb file is slow to load ?

Yes from external media. I originally tested with 2MB files from various  
cameras. When I started testing with larger 4MB+ files the speed is  
borderline acceptable. I need to improve loading the image from disk  
somehow in order to properly support large files.

If that is the case, you could try reading the entire file into a  
memory stream and then use LoadFromStream  
– I'm not sure, but it could cut down on physical disk reads

I found this. Would this help?

```
var
bm2: TBitmap;
st: TMemoryStream;
begin
bm2:= TBitmap.Create;

bm2.PixelFormat := pf24bit;
bm2.Width := 256 ;
bm2.Height := 256 ;

st := TMemoryStream.Create;
bm2.SaveToStream (st);
st.Seek( - (256*256*3), soFromEnd);
st.WriteBuffer( ColorArray[ScrollPos,0,0,1] , 256*256*3);
st.Seek(0, soFromBeginning );
bm2.LoadFromStream(st);
Image1.Picture.Bitmap := bm2;
bm2.Destroy ;
st.Destroy;

end;
```

## Re: Once and for all! Fastest way to load large jpeg

The above code is just creating a sort of emptyish Bitmap and shoving it into an Image.

What I would do is to take one of your large Bitmaps, load it into a TMemoryStream then load the image from the TMemoryStream

If you have FileMon, you could check how many file reads are made on the jpg on the CD – it might well be that there are hundreds of them

Things like this are normally slow because of excessive disk activity.

You also might try looking for a TBufferedStream component, I wrote one some time ago, but being an idiot did not descend it from an abstract TStream. I'm too ashamed to post or distribute it.

The improvement in reading and writing speeds was incredible.

.