

---

Subject: Re: Creating pointer in structure

Posted by [John-David T. Smith](#) on Thu, 01 Nov 2001 22:49:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

"Pavel A. Romashkin" wrote:

>  
> JD Smith wrote:  
>>  
>> One other case where having true null pointers is useful: when saving  
>> objects or structures, it's sometimes useful to kull out useless bits  
>> (like widget states and id's), which aren't really appropriate for  
>> keeping on disk.  
>  
> Let's see... A widget ID costs us 4 bytes. To fill up one kilobyte on  
> your 80 Gb disk, it'll take... wait... 250 widgets. I have never been  
> able to make even that many. My screen isn't big enough. If I stored a  
> droplist widget with its value array, 100 characters wide by 1000  
> entries long (which is crazy by itself), that would take up 100 Kb,  
> roughly, right?  
>  
> The time needed to weed out GUI elements before saving seems to be worth  
> more than the space savings. I just re-fill my restored, useless GUI IDs  
> with new IDs when I restore older legacy structures, and for the newer  
> code, I drop GUI altogether for saving, creting it from scratch on  
> restore by the Restore method.  
>  
> I can appreciate it though if the reason not to store widget hierarchies  
> is not so much storage space driven as it contradicts JD's aesthetic code  
> of IDL programming :-)  
>

A more useful reason (for non-aesthetes) to avoid saving unnecessary data: it may contain implicit definitions of structures or object classes which you don't want to maintain as an external type, but keep malleable and flexible. See the copious postings on the pitfalls of restoring saved objects for more info on this subtle but important problem.

JD

---