

---

Subject: Re: Zeiger in IDL -> call by reference in IDL  
Posted by [peter](#) on Mon, 08 May 1995 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Karl Krieger (kak@ipp-garching.mpg.de) wrote:  
: css@ipp-garching.mpg.de writes:

: >Weiss jemand, ob in IDL eine Pointeruebergabe moeglich ist,  
: >(aehnlich wie in C) ? Ansonsten sind ja staendig riesige  
: >Commonbloecke vonnoeten.

: After reading this guys question (if it was possible to pass variables  
: by reference in IDL) I read the fine manual on that subject which leads  
: me to another question:  
: They state that if you pass a variable to a procedure, it is saved in a  
: temporary location. When returning from the procedure, the new parameter  
: values are copied back into the corresponding variables.  
: Does this really mean that IDL allocates new memory for each parameter  
: variable even when calling by refernce i.e. passing a pointer?

It says that in the manual, but it isn't true. If you allocate big  
arrays till you run out of memory, then try passing that big array to a  
subroutine which, say, prints a single element, the call will succeed,  
indicating that the array was not copied.

I guess the manual is a convoluted attempt to explain 'pass by  
reference' to newbies.

Peter

-----  
Peter Webb, HP Labs Medical Dept  
E-Mail: [peter\\_webb@hpl.hp.com](mailto:peter_webb@hpl.hp.com)  
Phone: (415) 813-3756

---

---

Subject: Re: Zeiger in IDL -> call by reference in IDL  
Posted by [kak](#) on Mon, 08 May 1995 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

css@ipp-garching.mpg.de writes:

> Weiss jemand, ob in IDL eine Pointeruebergabe moeglich ist,  
> (aehnlich wie in C) ? Ansonsten sind ja staendig riesige  
> Commonbloecke vonnoeten.

After reading this guys question (if it was possible to pass variables  
by reference in IDL) I read the fine manual on that subject which leads

me to another question:

They state that if you pass a variable to a procedure, it is saved in a temporary location. When returning from the procedure, the new parameter values are copied back into the corresponding variables.

Does this really mean that IDL allocates new memory for each parameter variable even when calling by reference i.e. passing a pointer?

If this is true, does it help to use only COMMON blocks for large arrays in order to conserve memory? I guess in that case you really have only one instance of your variable in memory all the time.

Karl Krieger

--

IPP, PO Box 1533 | Phone: +49-89-3299-1655 | E-Mail:

D-85740 Garching | FAX : +49-89-3299-2591 | [kak@ipp-garching.mpg.de](mailto:kak@ipp-garching.mpg.de)

---