Subject: Re: Empty arrays?
Posted by Spon on Mon, 10 Nov 2008 22:35:31 GMT
View Forum Message <> Reply to Message

On Nov 10, 10:07 pm, Demitri wrote:
> Hello,
>
> I'm starting to dig into IDL programming, and one thing I've found
> particularly annoying is the lack of support for empty arrays. Although
> I'm using IDL's OO capabilities, I find that I'm spending a lot of time
> testing for return results where an empty array would be much more
> suitable.
>
> If I'm building an array that will have zero or more entries, I create
> a variable, "insert" a dummy value, go through the code adding entries
> based on whatever logic, then pop the top dummy value. But even then I
> have to check that more than one entry is there so I dont throw an
> error with a
>
> return, a[1:*]
>
> This is very basic functionality. Does anyone have a particularly
> elegant... well, I'd say solution, but anything is obviously a
> workaround. I don't want to fight the language, but I get the feeling
> that IDL's philosophy is "yeah, you do it yourself. I can't be
> bothered."
>
> Cheers,
>
> Demitri

Hi Demitri,

For these sorts of cases, I tend to use empty pointers. Here's an example using pointers to provide the functionality I think you're looking for:

```
var = ptr_new(/allocate_heap)
help, *var
IDL> <PtrHeapVar1>   UNDEFINED = <Undefined>
```

This article by JD Smith on David Fanning's website should help get you started on pointers:
http://www.dfanning.com/misc_tips/pointers.html

Good luck!
Chris

Subject: Re: Empty arrays?
Posted by Michael Galloy on Mon, 10 Nov 2008 22:56:56 GMT
View Forum Message <> Reply to Message

On Nov 10, 3:35 pm, Spon <christoph.b...@gmail.com> wrote:
> For these sorts of cases, I tend to use empty pointers. Here's an
> example using pointers to provide the functionality I think you're
> looking for:
>
> var = ptr_new(/allocate_heap)
> help, *var
> IDL> <PtrHeapVar1> UNDEFINED = <Undefined>
>
> This article by JD Smith on David Fanning's website should help get
> you started on pointers:http://www.dfanning.com/misc_tips/pointers.html

Yes, empty pointers work well for these type of things, but there are still two issues with them in this situation:

1. you still have to check to see if *var is undefined to determine if you are appending to or creating an array.

2. repeatedly appending to an array is inefficient

A better solution is to create a reasonable sized array to begin with. Then fill in values and keep track of how many are filled in. If the array fills up, create a new bigger one and copy the values over. Of course, this is even more of a hassle with bookkeeping, so I made an object that does this for me, see:

http://michaelgalloy.com/2006/04/24/collection-package-mgarr aylist.html

Of course, it would be nicer if IDL just allowed empty arrays, but there would be a lot of backward compatibilities if that were the case.

Mike
--
www.michaelgalloy.com
Tech-X Corporation
Associate Research Scientist