
Subject: Re: How to get numbers into passed structure elements
(pass-by-value/reference problem).

Posted by [Paul Van Delst\[1\]](#) on Wed, 27 Feb 2008 15:56:22 GMT

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

David Fanning wrote:

> Paul van Delst writes:

>

>> So, I realise this is one of those pass-by-reference or pass-by-value things, but how does

>> one get around it? Do I:

>> a) make the Type component of the Cloud structure a pointer? (Yuk!)

>> b) change the way I pass the Cloud structure into the CRTM_Read_Cloud_Record() fn?

>> i.e. not reference the cloud structure array via the index [n].

>>

>> I would much prefer (b), but will that entail copying entire structures? The number of

>> "Clouds" associated with any particular "Atm[m]" profile is variable.

>

> Too much misdirection for me to follow this early in the morning,

> but if you pass a structure array in by reference, what are you

> copying?

Well I meant if I do this:

```
FOR n = 0, Atm.n_Clouds-1 DO BEGIN
  ; Need to store data into (*Atm.Cloud)[n]
  result = CRTM_Read_Cloud_Record( FileID, Cloud, DEBUG=Debug )
  ...Copy Cloud into (*Atm.Cloud)[n]
ENDFOR
```

> I think you are only "copying" if you pass something in

> by value. I would go with (b).

I just completed an experiment where I passed the entire pointer array and looped over clouds in my reader,

```
result = CRTM_Read_Cloud_Record( FileID, *Atm.Cloud, DEBUG=Debug )
```

since it was passing (*Atm.Cloud)[n] that was bugging me up.

I find it a fundamental flaw of IDL that the argument passing mechanism (reference or value) is so exposed to the user that they have to tailor their code to prevent data loss. I understand why it's there, but even Fortran has moved beyond it (for about 18(!) years now).

cheers,

paulv
