
Subject: Re: Strange memory problem

Posted by [thompson](#) on Wed, 22 Dec 1999 08:00:00 GMT

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

"Mark D. Williams" <markw-xxxnospamxxx@resource-eng.com> writes:

> "R. Kyle Justice" wrote:

>>

>> When I do the following:

>>

>> temp=bytarr(1000,1000,25)

>> temp(*)=10

>>

>> my computer (a Sparc20 with 128MB of *available* ram) grinds

>> to a halt. Actually it starts using swap. Don't tell me

>> PV-Wave is making 5 or 6 copies of the array just to do this

>> simple process!

>>

>> Has anyone else seen this curious behavior on PV-Wave? IDL? I have

>> verified it on a PC, and using different versions of PV-Wave(6.1,7.0).

>>

>> I have played around with the z-value and 25 (i.e. 25MB) is roughly

>> the cutoff for going into swap for a 128MB system.

>>

>> Kyle

> I don't have access to a Sparc, but I tried this on PV-WAVE 7.0 on

> both Windows NT 4.0 and RedHat Linux 6.0 on a system with 256 Mb of

> physical RAM and didn't experience any problem: i.e., no swapping,

> returned to the prompt within 3-4 seconds.

> FWIW, if you want to save time and memory, a faster way to do the above

> is as follows:

> WAVE> temp = BYTARR(1000,1000,25, /NoZero) + 10B

~~~~~

Should this /NOZERO be here? I don't know about PVWAVE, but in IDL, this causes the array to be created without initializing the memory. You end up with an array, but filled with nonsense values. I think what you meant was

WAVE> temp = BYTARR(1000,1000,25) + 10B

> Note, was it your intent to end up with temp being an INTARR? By

> assigning

> 10 instead of 10B, that is what you're ending up with.

Again, I don't know anything about PVWAVE, but in IDL the statement

```
>> temp(*)=10
```

does not change the type of the array temp. Since temp was created as a byte array, the integer value 10 is converted to type byte before storing into the array.

I'm not sure about what's happening internally when a command like temp(\*)=10 is used, but it may be that a temporary index array is created to store the positions of all the referenced points. It would have to be a long array, so it would use up four times as much space as the byte array. That's just my guess.

As stated before, the best way in IDL to do this would be

```
temp = replicate(10b, 1000,1000,25)
```

William Thompson

---