
Subject: Re: how does /no_copy work???

Posted by [John Persing](#) on Thu, 03 Jun 1999 07:00:00 GMT

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

Peter Mason <menakkis@my-deja.com> wrote in message
news:7j5a1n\$8d1\$1@nnrp1.deja.com...

- > Further to what David has written, there is a way to capture the
- > "spirit" of NO_COPY, in general - wherever there's some kind of
- > assignment going on. Use the TEMPORARY() function. e.g., If you do
- > A=B then A is set up with a copy of B's stuff (B is left intact). If
- > you do A=TEMPORARY(B) then B's stuff is essentially "switched over" to A
- > (B is deleted).
- > This technique is only worthwhile in cases where the amount of data
- > concerned is *large* (e.g., large arrays), or in cases where the amount
- > of data is not insignificant and the operation is done very frequently.

But let me ask, how can this be possible when deal with a variable that "starts" on the stack and "ends up" on the heap? If B is an ordinary array and A is property of an object, then this is what will occur. The heap and stack are entirely different memory locations.

It seems that what you say will be slick if B is a pointer to an array, then the assignment to the object will be fast. Of course, there is hardly the need for TEMPORARY for such a small assignment.

--

}3 John Persing }3

<http://www.frii.com/~persing> persing@frii.com

Half of all Americans earn less than the median income!!!!!!
