

23

Distribution:

- WM File  109.9
- WMEG
- NMSS
- REBrowning
- MJBell
- LHBarrett
- TCJohnson
- JVoglewede
- MRKnapp
- JOBunting
- LBHigginbotham

DEC 8 1984

- 1 -

JVSMITH2

MEMORANDUM FOR: J. Barry Badini  
 System Support Branch  
 Division of Automated Information Services, RM

FROM: John C. Voglewede  
 Engineering Branch  
 Division of Waste Management, NMSS

SUBJECT: DATA GENERAL FORTRAN 77 COMPILER WARNING MESSAGE

Under certain conditions, the most recent version of the Data General MV/8000 FORTRAN compiler (AOS/VS F77 Rev 02.21.00.00) generates the following warning message:

Warning 244 severity 1 beginning on line 1  
 This statement potentially may redefine the current DO loop index.

This warning message occurs whenever a DO loop index appears as an argument in a CALL parameter list.

In my opinion, this warning is inappropriate. This opinion is based on the following observations:

1. This warning was not generated by the previous version of the F77 compiler used on our MV/8000 computer. Strict compatibility has not been maintained.
2. Although ANSI FORTRAN 77 (Ref. 1) does not permit modification of a DO loop index within a loop (which the warning is intended to prevent), I do believe the Standard allows the index to be used in a CALL parameter list (but not modified on the subprogram). The warning, therefore, guards against something permitted by the Standard.
3. Other commonly used FORTRAN compilers (e.g., CDC FTN5/ANSI) do not flag this usage - even with ANSI FORTRAN 77 option evoked.
4. Based upon examination of computer programs used in the regulatory process, this usage is common and the warning message gets very old very fast.

WM Record File 109.9 WM Project 1  
 Docket No. \_\_\_\_\_  
 PDR   
 LPDR \_\_\_\_\_

8502280259 841203  
 PDR WASTE PDR  
 UM-1

Distribution:

DFC :WMEG <i>JCW</i> :WMEG <i>J</i> :	(PDR - 16 WM, 623-SS)	:	:	<i>CZ</i>	:
NAME :JVoglewede:gh:TCJohnson :	:	:	:	:	:
DATE :12/03/84 :	:12/3/84 :	:	:	:	:

DEC 3 1984

JVSMITH2

- 2 -

- 5. There is no way to selectively disable a particular warning message from the Data General F77 compiler (unless one wishes to ignore all warning messages). This makes it difficult to spot a different warning in a whole pile of Warning 244's.
- 6. Discussions with Roy Glixon of your staff (see Attachment 1) have resulted in a reprogramming suggestion to eliminate message. This fix involves parenthetical use of the DO loop index - forcing a call-by-value, rather than call-by-name, reference to the index. Although this fix does work, it requires program modifications which, in the case of computer codes under NRC review, may obscure real misuse of the index within a subprogram. I am therefore reluctant to modify computer programs to eliminate the warning.

With your concurrence, I suggest that Data General be informed of this negative opinion of the 244 Warning. Perhaps it could be eliminated in future revisions of the F77 compiler. Should you require additional information in order to take action on this subject, please call me at 42-74275.

*original signed by*

John C. Voglewede  
 Materials Section  
 Engineering Branch  
 Division of Waste Management

cc: J. Shields  
 James P. Smith  
 R. Glixon  
 E. Robinson

Enclosure:  
 As stated

---

DFC	:WMEG	:WMEG	:	:	:	:	:
NAME	:JVoglewede:gh:	TCJohnson	:	:	:	:	:
DATE	:11/ /84	:11/ /84	:	:	:	:	:

---

DEC 3 1984

JVSMITH2

- 3 -

REFERENCES

1. "American National Standard Programming Language FORTRAN," ANSI X3.9-1978 FORTRAN 77, American National Standards Institute, Inc.

---

DFC	:WMEG	:WMEG	:	:	:	:	:
NAME	:JVoglewede:gh:TCJohnson	:	:	:	:	:	:
DATE	:11/ /84	:11/ /84	:	:	:	:	:

---

-----  
ELECTRONIC MAIL FACILITY  
-----

To: voslewede.Jc  
From: slixon.re  
Date: 26-SEP-84  
Time: 07:39:18  
Subject: F77 compiler warnings  
-----

I finally got your "Index in Do-loop may be redefined" . I agree with your "It gets old" comment. You can avoid this error with the following work-around.

```
      Do 100 I=1,100  
          Call Sub(temp1, temp2, (I), Temp3)  
100      continue
```

Note that the variable "I" is enclosed with parenthesis in the subroutine call. This instructs Fortran to put "I" into a temporary variable location (on the stack) so that its value will not get inadvertently changed.

This feature is probably geared more toward the inexperienced programmer. I've had this type of problem trying to define a common block with mixed data types (like INTEGER and Character types).

I feel there should be a switch available to over-ride the reporting of these types of errors.

You may Print, Delete, or Keep this mail  
Enter the first letter only.

(P)rint, (D)elete, (K)eep ? D