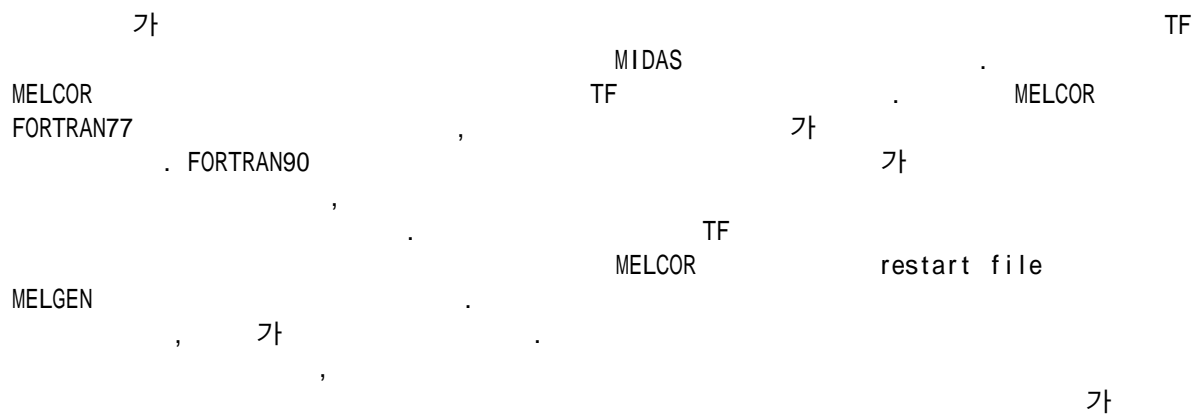


(MIDAS) TF

A Restructuring of TF package for MIDAS Computer Code

150



Abstract

TF package which defines some interpolation and extrapolation condition through user defined table has been restructured in MIDAS computer code. To do this, data transferring methods of current MELCOR code are modified and adopted into TF package. The data structure of the current MELCOR code using FORTRAN77 causes a difficult grasping of the meaning of the variables as well as waste of memory. New features of FORTRAN90 make it possible to allocate the storage dynamically and to use the user-defined data type, which lead to an efficient memory treatment and an easy understanding of the code. Restructuring of TF package addressed in this paper does module development and subroutine modification, and treats MELGEN which is making restart file as well as MELCOR which is processing calculation. The validation has been done by comparing the results of the modified code with those from the existing code, and it is confirmed that the results are the same. It hints that the similar approach could be extended to the entire code package. It is expected that code restructuring will accelerate the code domestication thanks to direct understanding of each variable and easy implementation of modified or newly developed models.

1.

MELCOR 가
MIDAS MELCOR
MELCOR 가
가 , 가
가 가
가 FORTRAN90
가 data type
[1,2,3]
(readability) (DMM)
subroutine 가 (derived type variables)
[4,5].
[6,7],
TF (table extrapolation
) restart file , MELCOR
MELGEN

2.

MELCOR code 3 restart file
MELGEN, restart file , log file plot file
MELCOR, PLOT
COR, HT, SPR, TF 20 가
data subroutine
message subroutine, code subroutine
subroutine [8].

2.1 Restart file

data MELCOR RESTART
file read write subroutine MXXRS MXXRSW , read
write , subroutine xyzPRS
1, 2 , subroutine xyzR10
coefficient , Real, integer, logical, character
4 array . (xyz :)

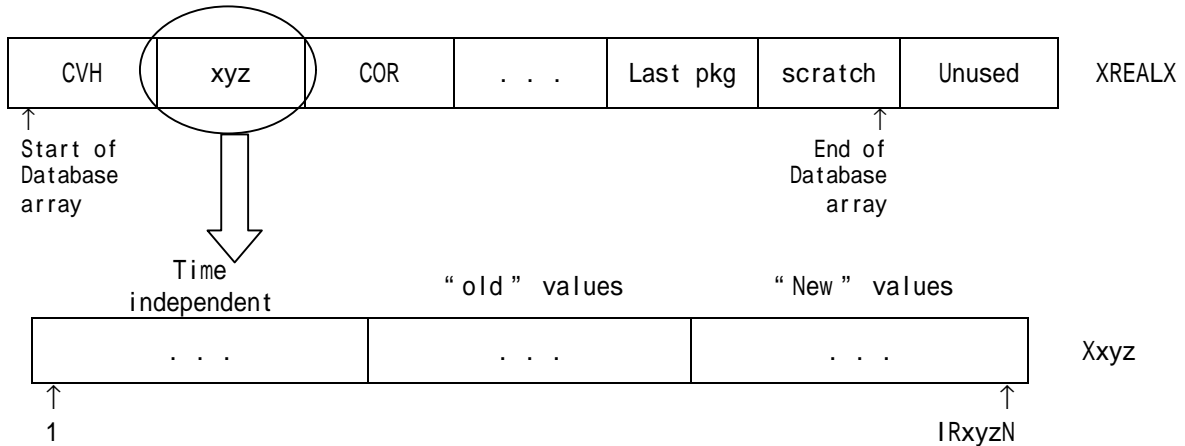
2.2 Database

MELCOR data 4가

- XREALX : floating point variables (500,000 array)

- INTEGE : integer variables (50,000 array)
- LOGICA : logical variables (5,000 array)
- CHARAC : character variables (30,000 array)

(1),



1.

Database array subroutine 274
 . Database array argument 1
 2 .

```

SUBROUTINE xyzDBC
*- INCLUDE BLANK
COMMON /DBREAL/ NEDREL, IRELCS, IRELES
COMMON / / DREALX(NUMREL/2)
DOUBLE PRECISION DREALX C
DIMENSION XREALX(NUMREL)
EQUIVALENCE (XREALX(1), DREALX(1))
DOUBLE PRECISION VREALX(NUMREL/2)
EQUIVALENCE (VREALX(1), DREALX(1))
*- INCLUDE xyzDB
COMMON /xyzDB/ IRxyzF , IRxyzN , IxyzF , IxyzN ,
1 ILxyzF , ILxyzN , ICxyzF , ICxyzN
CALL xyzDBD(
1 IRxyzN , XREALX(IRxyzF) ,
2 IxyzN , INTEGE(IxyzF) ,
3 ILxyzN , LOGICA(ILxyzF) ,
4 ICxyzN , CHARAC(ICxyzF) , . . . )
C
RETURN
END

```

2. 1 database subroutine

Database 2
array argument , subroutine 3 .

```

SUBROUTINE xyzDBD (
1 NxyzR , Rxyz ,
2 NxyzI , Ixyz ,
3 NxyzL, Lxyz,
4 NxyzC, Cxyz, . . . )
C
DIMENSION Rxyz(NxyzR), Ixyz(NxyzI), . . .
C
*- INCLUDE xyzPNT
COMMON /xyzPNT/ Nvar1, Nvar2, . . .
C
CALL xyzRUN (Nvar1, Rxyz(Ivar1), Nvar2,Ixyz(Ivar2), . . . )
C
RETURN
END

```

3. 2 database subroutine

Database argument
subroutine 4 .

```

SUBROUTINE xyzRUN(
1 Nvar1, var1, Nvar2, var2, . . . )
C
DIMENSION var1(Nvar1), . . .
C
FROM HERE ON, SIMPLY USE var1 AND var2.
C FORGETTING ALL DETAILS OF WHERE AND HW THEY ARE STORED.
C
CALL xyz. . .
C
RETURN
END

```

4. database subroutine

2.3

subroutine xyzDBD subroutines database subroutine xyzDBC
comment argument pass

(1) 1

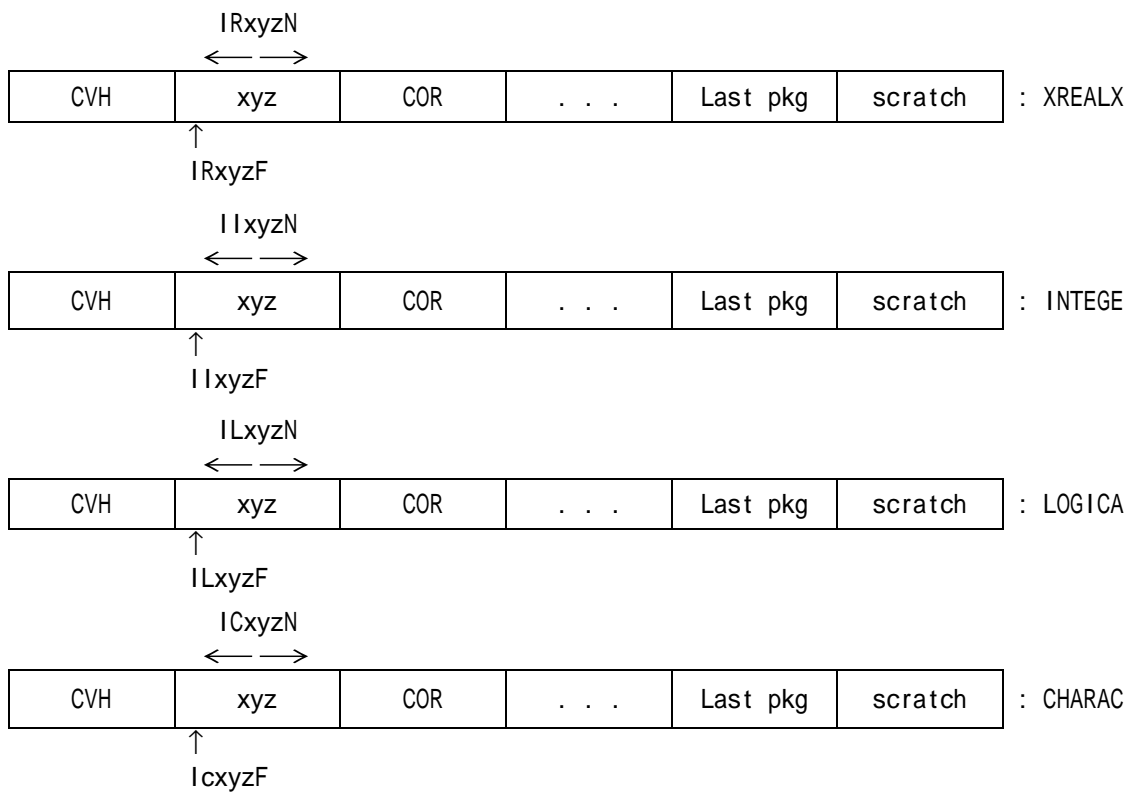
subroutine ' xyzDB ' common block (5),
4 data type 2 . 1
(6).

```

*- INCLUDE xyzDB
C    POINTERS TO xyz PACKAGE DATABASE
COMMON /xyzDB/ IRxyzF, IRxyzN, IIxyzF, IIxyzN,
*                    ILxyzF, ILxyzN, ICxyzF, ICxyzN
C

```

5. 1



6. 1

(2) 2

subroutine 'xyzPNT' common block
 , 4 data type database 가 TF . 2
 7 . TF

```
COMMON /TFPNT/ KTFNUM, KTFPAR, KTFSCCL, KTFADC, KTFBDL, KTFBDU,
1 KTFNMS, KPTPAR, KTFNAM, KTFX , KTFY , NUMTF , NMTF ,
2 MXTFPR, NUNPAR, NMPAR , KTFHSH, NTFHSH
```

7. TF 2

TF 가 subroutine TF data TF
 , TF . TF
 TF MELCOR 가

3.

restart file
 , restart file
 MELCOR 1.8.4

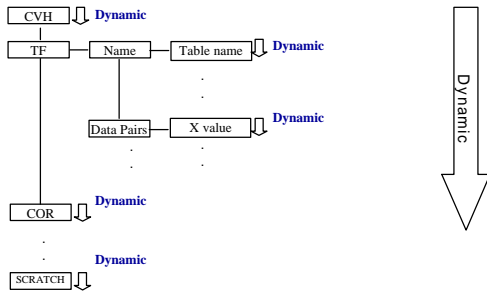
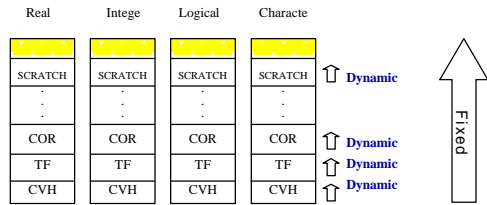
MELGEN

MELCOR
 FORTRAN90
 가

MELCOR
 (derived data type)

FORTRAN90

8



8.

extrapolation

가 TF(table , TF)

MELCOR

subroutine TF (MELtoMID)

33

subroutine

subroutine

[4,5].

3.1

Subroutine TFNDBA
 subroutine TFNDBC

subroutine TFNDBB

(local variable)

2

TF

9

3.2 Subroutine

Subroutine

FORTRAN77

MELCOR

FORTRAN90

TF

subroutine , argument가

[9].

subroutine 9 ,

subroutine

subroutine local variable
 dimension

argument

array member variable

ITFNUM(MMTF) ' ,

array

' TF_NM(NMTF)%ITFNUM '
 2 3

1 ,

subroutine

```

!!!!!!!!!!!!!! TF (Tabular Function) package data !!!!!!!!!!!!!!!
!
!   IMPLICIT NONE
!   MODULE TF_MDL
!
!   ***** TABULAR FUNCTION GLOBAL DATA *****
!   INTEGER :: NUMTF, NMTF, MXTFPR, NUMPAR, NMPAR, NTFHSH, NTF_FLAG
!
!   ***** TF_NM *****
!   TYPE TF_N ; SEQUENCE
!   INTEGER :: ITFNUM, NTFPAR
!   REAL :: TFSCAL, TFADCN
!   INTEGER :: NTFBDL, NTFBDU, ITFNMS, IPTPAR
!   CHARACTER(LEN=16) :: TFNAMEE
!   END TYPE
!   TYPE (TF_N), ALLOCATABLE :: TF_NM(:)
!
!   ***** TF_PAR *****
!   TYPE TF_P ; SEQUENCE
!   REAL :: TFX, TFY
!   END TYPE
!   TYPE (TF_P), ALLOCATABLE :: TF_PAR(:)
!
!   ***** TF_HSH *****
!   TYPE TF_H ; SEQUENCE
!   INTEGER :: IHASH
!   END TYPE
!   TYPE (TF_H), ALLOCATABLE :: TF_HSH(:)
!
!   END MODULE TF_MDL
!
!!!!!!!!!!!!!!

```

9. TF

1. Array

ITFNUM(MMTF)	TF_NM(NMTF)%ITFNUM
NTFPAR(MMTF)	TF_NM(NMTF)%NTFPAR
TFSCAL(MMTF)	TF_NM(NMTF)%TFSCAL
TFADCN(MMTF)	TF_NM(NMTF)%TFADCN
NTFBDL(MMTF)	TF_NM(NMTF)%NTFBDL
NTFBDU(MMTF)	TF_NM(NMTF)%NTFBDU
ITFNMS(MMTF)	TF_NM(NMTF)%ITFNMS
IPTPAR(MMTF)	TF_NM(NMTF)%IPTPAR
TFNAME(MMTF)	TF_NM(NMTF)%TFNAMEE
TFX(NMPAR)	TF_PAR(NMPAR)%TFX
TFY(NMPAR)	TF_PAR(NMPAR)%TFY
IHASH(NTFHSH)	TF_HSH(NTFHSH)%IHASH

2. Subroutine TFVDBG

```

Subroutine TFVDBG ( )
SUBROUTINE TFVDBG (
1 NMTF , NUMTF , NTFHSH, IHASH,
2 IPTPAR , NTFPAR ,
3 TFSCAL , TFADCN ,
4 NTFBDL , NTFBDU ,
2 NMPAR , X , Y ,
1 NDIM , NUMBTF , XVALUE ,
C OUTPUT
2 YVALUE , IERR , IGERR )
. . . . .

IF ( NTFBDL( NUM ) .EQ. 0 ) THEN
C HOLD VALUE CONSTANT AT LAST DEFINED POINT
C VALUE OF YY DEFINED BY TFBFND
ELSE IF ( NTFBDL(NUM) .EQ. 1 ) THEN
C LINEAR EXTRAPOLATION
YY = Y(IPTPAR(NUM) )+
1 (XVALUE(N) -X(IPTPAR(NUM) ))*
2 (Y(IPTPAR(NUM))-Y(IPTPAR(NUM)+1))/
3 (X(IPTPAR(NUM))-X(IPTPAR(NUM)+1))
ELSE IF ( NTFBDL(NUM) .EQ. 2 ) THEN
C ERROR - RUNNING OFF BOTTOM OF TABLE PROHIBITED
IERR(N) = 1
IGERR = 1
CHMESS(1)='ERROR IN TABULAR FUNCTION PACKAGE'
CHMESS(2)='RAN OFF BOTTOM OF TABLE'
CHMESS(3)=' '
30 WRITE(CHMESS(2)(25:32),'(18)',ERR=30) NUMBTF(N)
INMESS=0
CALL MESERB
ENDIF
. . . . .

Subroutine TFVDBG ( )
SUBROUTINE TFVDBG ( NDIM, NUMBTF, XVALUE, YVALUE, IERR, IGERR)
. . . . .

USE TF_MDL
. . . . .

IF (TF_NM(NUM)%NTFBDL .EQ.0) THEN
! HOLD VALUE CONSTANT AT LAST DEFINED POINT
! VALUE OF YY DEFINED BY TFBFND
!! NOTICE: Converted by MELtoMID
ELSEIF (TF_NM(NUM)%NTFBDL .EQ.1) THEN
! LINEAR EXTRAPOLATION
!! NOTICE: Converted by MELtoMID
YY = Y (TF_NM(NUM)%IPTPAR ) + ( XVALUE (N) - &
X (TF_NM(NUM)%IPTPAR ) ) * ( Y (TF_NM(NUM)%IPTPAR ) - &
Y (TF_NM(NUM)%IPTPAR + 1) ) / ( X (TF_NM(NUM)%IPTPAR ) - &
X (TF_NM(NUM)%IPTPAR + 1) ) )
!! NOTICE: Converted by MELtoMID
ELSEIF (TF_NM(NUM)%NTFBDL .EQ.2) THEN
! ERROR - RUNNING OFF BOTTOM OF TABLE PROHIBITED
IERR (N) = 1
IGERR = 1
CHMESS (1) = 'ERROR IN TABULAR FUNCTION PACKAGE'
CHMESS (2) = 'RAN OFF BOTTOM OF TABLE'
CHMESS (3) = ' '
30 WRITE (CHMESS (2) (25:32), '(18)', ERR = 30) NUMBTF (N)
INMESS = 0
CALL MESERB
ENDIF
. . . . .

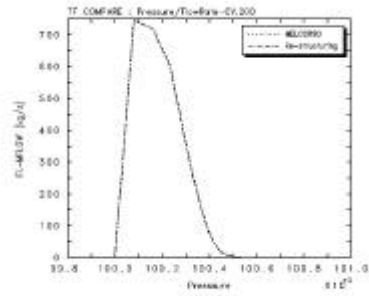
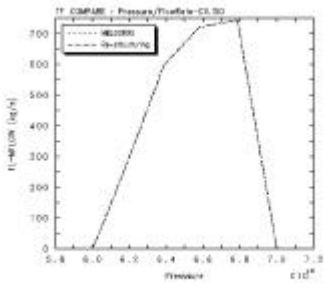
```


3. Subroutine TFDBZ

Subroutine TFDBZ ()	
C	<pre> SUBROUTINE TFDBZ (INPUT 1 MAXREL, RSTORE, NUSREL, NSCREL, NSEREL, 2 MAXINT, INTEGR, NUSINT, NSCINT, NSEINT, 3 MAXCHR, CHSTOR, NUSCHR, NSCCHR, NSECHR, 4 MXSCRL, MXSCIN, MXSCCH, C OUTPUT 4 IGOOD, IACTV) 200 IF (NTF .LT. NUMTF) THEN IF (NTF .EQ. 0) REWIND ISTORE READ (ISTORE) 1 INTEGR(KTFNUM+NTF), CHSTOR(KTFNAM+2*NTF), * CHSTOR(KTFNAM+2*NTF+1), INTEGR(KTFPAR+NTF), * RSTORE(KTFSCAL+NTF), RSTORE(KTFADC+NTF), 2 INTEGR(KTFBDL+NTF), INTEGR(KTFBDU+NTF), 3 (RSTORE(KTFX+NNN+MMM), MMM=0, MAX(0, INTEGR(KTFPAR+NTF)-1)), 4 (RSTORE(KTFY+NNN+MMM), MMM=0, MAX(0, INTEGR(KTFPAR+NTF)-1)) INTEGR(KTFNMS+NTF) = NTF+1 NNN = NNN + INTEGR(KTFPAR+NTF) IF (NTF .EQ. 0) THEN INTEGR(KPTPAR+NTF) = 1 ELSE INTEGR(KPTPAR+NTF) = 1 INTEGR(KPTPAR+NTF-1)+INTEGR(KTFPAR+NTF-1) ENDIF NTF = NTF + 1 GO TO 200 ENDIF </pre>
Subroutine TFVDBG ()	
	<pre> SUBROUTINE TFDBZ (MAXREL, NUSREL, NSCREL, NSEREL, MAXINT, & NUSINT, NSCINT, NSEINT, MAXCHR, NUSCHR, NSCCHR, & NSECHR, MXSCRL, MXSCIN, MXSCCH, IGOOD, IACTV) USE TF_MDL 200 IF (NTF.LT.NUMTF) THEN IF (NTF.EQ.0) REWIND ISTORE READ (ISTOR) TF_NM(NTF+1)%ITFNUM, TF_NM(NTF+1)%TFNAMEE, & TF_NM(NTF+1)%NTFPAR, TF_NM(NTF+1)%TFSCAL, TF_NM(NTF+1)%TFADCN, & TF_NM(NTF+1)%NTFBDL, TF_NM(NTF+1)%NTFBDU, & (TF_PAR(NNN+MMM+1)%TFX, MMM=0, MAX(0, TF_NM(NTF+1)%NTFPAR-1)), & (TF_PAR(NNN+MMM+1)%TFY, MMM=0, MAX(0, TF_NM(NTF+1)%NTFPAR-1)) ! !! NOTICE: Converted manually TF_NM(NTF+1)%ITFNMS = NTF + 1 ! INTEGR (KTFNMS + NTF) = NTF + 1 ! NNN = NNN + TF_NM(NTF+1)%NTFPAR ! NNN = NNN + INTEGR (KTFPAR + NTF) IF (NTF.EQ.0) THEN ! TF_NM(NTF+1)%IPTPAR = 1 ! INTEGR(KPTPAR+NTF) = 1 ELSE ! TF_NM(NTF+1)%IPTPAR = TF_NM(NTF)%IPTPAR + TF_NM(NTF)%NTFPAR ! INTEGR(KPTPAR+NTF) = INTEGR(KPTPAR+NTF-1) + INTEGR(KTFPAR+NTF-1) ENDIF !- NTF = NTF + 1 GOTO 200 ENDIF </pre>

4.

FORTRAN90 , TF
 FORTRAN77 library execution file MELCOR FORTRAN90
 MELCOR TFMELCOR TF FORTRAN90
 SRV
 , control function SRV()가 hysteresis
 , table function 가 ,
 FORTRAN90 MELCOR TFMELCOR TF
 (10).



10. TF

5.

MIDAS , TF MELCOR TF
 TF subrou t ine , TF
 가 , TF
 TF 가
 [10], 가 가

- [1] A Multi-Dimensional Thermal-Hydraulic System Analysis Code, MARS 1.3.1, Vol.31, Number 3, pp.344-363, June 1998.
- [2] Realistic Thermal-Hydraulic System Code Development Workshop, '98 Fall KNS Conference, 1998.
- [3] 3rd MARS Users Group Meeting, '99 Dec. 2 KAIST Conference, 1999.
- [4] Development of a Computer Program for Automatic Variable Conversion in MELCOR Code, , 2000
- [5] MIDAS , KAERI/TR-2220/2002, Jun, 2002.
- [6] A Restructuring Proposal for MIDAS, , 2000.
- [7] Development of Restructuring Template for MELCOR, 5th PSAM Conference, Japan, 2000.11.26 2000.12.2
- [8] MELCOR , KAERI/TR-1543/00, June, 2000.
- [9] MELCOR , KAERI/TR-1536/2000, March, 2000
- [10] , KAERI/RR-2216/2001, pp 302-321, May, 2002.