SOFTWARE RELEASE NOTICE

01. SRN Number: PA-SRN-009	
02. Project Title: Network Flow and Transport in Time-Dependent Velocity Fields, CNWRA Version 1.1	Project No.
03. SRN Title: NEFTRAN II Version 1.1	
04. Originator/Requester: Thomas J. Ratchford	Date: 12/27/93

05. Summary of Actions

- ■ Release of new code admitted to CM System
- ☐ Release of modified code:
  - ☐ Enhancements made
  - ☐ Corrections made
- ☐ Change of access code

06. Persons Authorized Access

Name	RO/RW	A/C/D

07. Element Manager Approval: Date:

08. Remarks:

CNWRA Form TOP-6 (08/93)

SOFTWARE RELEASE NOTICE

01. SRN Number: PA-SRN-009	
02. Project Title: Network Flow and Transport in Time-Dependent Velocity Fields, CNWRA Version 1.1	Project No.
03. SRN Title: NEFTRAN II Version 1.1	
04. Originator/Requester: Thomas J. Ratchford	Date: 12/27/93

05. Summary of Actions

■ Release of new code admitted to CM System

☐ Release of modified code:

☐ Enhancements made

☐ Corrections made

■ Change of access code

06. Persons Authorized Access

Name	RO/RW	A/C/D

07. Element Manager Approval: Date:

08. Remarks:

CNWRA Form TOP-6 (08/93)

SOFTWARE SUMMARY FORM

01. Summary Date: 12/21/93

02. Summary prepared by (Name and Phone): T.J. Ratchford 522-3083

03. Summary Action: New

04. Software Date: 8/15/93

05. Short Title: NEFTRAN II

06. Software Title: NEFTRAN II- Network Flow and Transport in Time-Dependent Velocity Fields.

07. Internal Software ID: NONE

08. Software Type:

- ☐ Automated Data System
- ■ Computer Program
- ☐ Subroutine/Module

09. Processing Mode:

- ☐ Interactive
- ☐ Batch
- ■ Combination

10. APPLICATION AREA

A. General:

- ☐ Scientific/Engineering
- ☐ Auxiliary Analyses
- ☐ Total System PA
- ■ Subsystem PA
- ☐ Other

b. Specific:

11. Submitting Organization and Address: CNWRA, SwRI, San Antonio, Texas

12. Technical Contact(s) and Phone: R. Janetzke, (210) 522-3318

13. Narrative:
The NEFTRAN II computer code simulates radionuclide ground-water transport through saturated porous media, saturated dual-porosity or fractured media, and the added capability to transport radionuclides through time-dependent velocity fields.

14. Computer Platform	15. Computer Operating System:	16. Programming Language(s):	17. Number of Source Program Statements:
CRAY/XMP	UNIX	FORTRAN	7294 lines of code

18. Computer Memory Requirements:	19. Tape Drives:	20. Disk/Drum Units:	21. Graphics:
UNKNOWN	NONE	N/A	UNKNOWN

22. Other Operational Requirements: NONE

23. Software Availability: ■ Available ☐ Limited ☐ In-House ONLY

24. Documentation Availability: ■ Available ☐ Inadequate ☐ In-House ONLY

25. Submission Package Status:

Acceptance Criteria: Met ■ Not Met ☐ Software QA Assessment: Successful ■ Unsuccessful ☐

Code Custodian: Thomas J Ratchford Date: 12/22/93

CNWRA Form TOP-4-1 (08/93)

```
TPA_NEF.NGD.Z    flonef.dat.Z    nef.tty.l.Z      nefdr2.dat.Z     nefrep.out.Z
neftpa.dat.Z     perf.data.Z     sotnef.dat.Z     watrel.in.Z
gemstone.17 ~/tpa/NEFTRAN/hpm0 =>
gemstone.17 ~/tpa/NEFTRAN/hpm0 => cd
gemstone.18 ~ => cd tpa/NEFTRAN/VCS
]gemstone.19 ~/tpa/NEFTRAN/VCS => ls
]ls: Command not found.
gemstone.20 ~/tpa/NEFTRAN/VCS => ls
s.Makefile.Z*    s.TPA_NEF.NGD*   s.adjb.F.Z*      s.band.F.Z*
s.branch.F.Z*    s.bsolve.F.Z*    s.catch.F.Z*     s.chain.F.Z*
s.chkpth.F.Z*    s.coeff.F.Z*     s.dtupdt.F.Z*    s.dxdt.F.Z*
s.et.F.Z*        s.facer.F.Z*     s.flonef.dat.Z*  s.flowin.F.Z*
s.getfld.F.Z*    s.getrv.F.Z*     s.git.F.Z*       s.inidr2.F.Z*
s.inprin.F.Z*    s.intg.F.Z*      s.lechmod.F.Z*   s.lhsoooo.out*
s.method.F.Z*    s.mxcll.F.Z*     s.nef.F.Z*       s.nefdit.F.Z*
s.nefmks.cpp.Z*  s.nefmks.pre.Z*  s.opnfil.F.Z*    s.prp.F.Z*
s.pthlen.F.Z*    s.ratio.F.Z*     s.rdsmpl.F.Z*    s.setdis.F.Z*
s.setup.F.Z*     s.sift.F.Z*      s.sizes.H.Z*     s.skpfld.F.Z*
s.skpspc.F.Z*    s.sotnef.dat.Z*  s.source.F.Z*    s.split.F.Z*
s.srcin.F.Z*     s.stopper.F.Z*   s.timer.F.Z*     s.timval.F.Z*
s.tpprt.F.Z*     s.tracer.F.Z*    s.trnspt.F.Z*    s.tspfac.F.Z*
s.upcase.F.Z*    s.updcur.F.Z*    s.updr2.F.Z*     s.work.F.Z*
s.wrtdr2.F.Z*    s.x.neftran.co*  s.x.neftran.te*  s.xchng.F.Z*
gemstone.21 ~/tpa/NEFTRAN/VCS =>
                                                    134.20.1.1 09:42:54
```

MR 12/22/93

Revised 3/18/92

Attachment 2 - NEFTRAN2 Modifications

Modification to allow input of and external source term

The current version of NEFTRAN2 does not allow an external source term to be read. A small modification within Subroutine SOURCE will allow this flexibility. The suggested changes are:

1) insert the following at the beginning of SUBROUTINE SOURCE

```
C  ROUTINE TO READ AN EXTERNAL SOURCE BASED ON IOPT(14) .NE. 0
C
      IF (IOPT(14) .NE. 0) THEN
C
C
C   READ THE NUMBER OF ISOTOPES(ITOT) AND TIME STEPS(NUM) TO BE READ
C
      READ(14,*) ITOT,NUM
C
C
C  NOW LOOP OVER THE VALUES WHERE IT IS ASSUMED THAT THE INPUT
C  CONVENTION IS A DUMMY DESCRIPTOR IS READ WHICH CAN NAME THE
C  THE ISOTOPE BEING READ, FOLLOWED BY THE TIMES IN YEARS AND
C  THE AND SOURCES FOR THIS NUCLIDE IN (CI/YR)
C
C
      DO 5 J=1,ITOT
C
      READ(14,'(A80)') (DUMMY STRING VARIABLE)
C
      DO 5 I=1,NUM
C
C  READ THE EXTERNAL TIME (TEXT) AND THE SOURCE (EXTSRC)
C
      READ(14,*) TEXT(I),SRCEXT(I,J)
C
C
C NOW CHANGE (CI/YR) TO WHAT NEFTRAN2 WANTS (ATOMS/YR)
C
      SRCEXT(I,J)=SRCEXT(I,J)*1.6834E18*HLFALL(J)*ACSSFR
C
    5 CONTINUE
C
C
C
C  NOW WRITE THE INFORMATION TO UNIT 24 AS NEFTRAN2 WOULD
C
      WRITE(24) ITOT
C
      DO 20 I=1,20
```

```
C
      WRITE(24) TEXT(I),(SRCEXT(I,J),J=1,ITOT)
C
   20 CONTINUE
C
C  NOW SKIP THE REST OF THE SUBROUTINE
C
      GOTO 3000
C
C
      ENDIF
C
C
```

2) The variable types and array dimension should be declared at the beginning of the subroutine. I would suggest that the "J" index be dimensioned to the "mxiso" the maximum number of isotopes. The "I" index is for the number of time steps used which could be 100 - or a number that the source term people could provide.

Modification to write concentrations to an external file

The current version of NEFTRAN2 has the ability to write concentrations to the standard output (unit 6). However, it would be useful to write the concentrations to an external file for later use by the dose module. The suggested change to NEFTRAN2 is:

insert the following in SUBROUTINE TPPRT after the line which reads (icount=icount + 1):

```
C
C  ROUTINE TO WRITE THE CONCENTRATIONS TO AN EXTERNAL FILE
C  BASED ON IOPT(29) .NE. 0 INDICATING TO READ
C
C
       IF (IOPT(29) .NE. 0) THEN
C
       IF (ICOUNT .EQ. IFREQ) THEN
C
       WRITE(29,9002) T,(CUROUT(J)/QDIS,J=NOISO)
C
       IF (IOPT(10) .EQ. 0) ICOUNT=0
C
       ENDIF
C
       ENDIF
C
C
```

Table 1. Table of variable names, types, and values used in the liquid flow module.

STANDARD

	NAME	TYPE	UNITS	VALUE	RANGE	SOURCE	DESCRIPTION
UNIT 20	INDEX(N)	Integer (array)	--	NA	NA	NA	number of values in the table to distribute flux for the Nth part
	INFINT (N,J,K)	Real (array)	m/yr	NA	NA	NA	interpolation table for the Nth part, with INDEX(N) values where K=1 is the total infiltration value and K=2 is the infiltration amount (assumed to be in ascending order)
UNIT 21	MTINDX(N)	Integer (array)	--	NA	NA	NA	number of values in the table to distribute flux between the fracture and matrix for the Nth layer
	MATINT (N,J,K)	Real (array)	--	NA	NA	NA	interpolation table for the Nth layer, with MTINDX(N) values where K=1 is the ratio of the infiltration to the saturated conductivity and K=2 is the ratio of the matrix flux to the saturated conductivity (assumed to be in ascending order)
INTERNAL	REPLEG	Real	meters	10.	NA	NA	length of repository leg *
	SPASTP	Real	meters	1.0	NA	NA	NEFTRAN2 spatial step size

* does this need to be global ?

GLOBAL (NOT SAMPLED)

NAME	TYPE	UNITS	VALUE	RANGE	SOURCE	DESCRIPTION
ETIME	Real	years	10,000	NA	NA	simulation time
NPART	Integer	--	7	NA	NA	number of parts that make up the repository
NLYERS	Integer	--	9	NA	NA	maximum number of hydrologic layers
NLYERU	Integer	--	9	NA	NA	maximum number of unsaturated hydrologic layers
NCHAIN	Integer	--	2	NA	NA	number of radionuclide chains
NISO	Integer	--	5	NA	NA	total number of radionuclides
MEMPCH(N)	Integer (array)	--	3,2	NA	NA	number of members of Nth chain
NAME(N)	Character (array) A6	--	NA	NA	NA	name of Nth radionuclide
AMASS(N)	Real (array)	--	NA	NA	NA	atomic mass number of Nth radionuclide
PAR1(N)	Integer (array)	--	NA	NA	NA	local index of first parent for the Nth radionuclide
PAR2(N)	Integer (array)	--	NA	NA	NA	local index of second parent for the Nth radionuclide

UNIT 5

UNIT 5

FRCFRM1(N)	Real (array)	--	NA	NA	NA	branching fraction from parent number 1
FRCFRM2(N)	Real (array)	--	NA	NA	NA	branching fraction from parent number 2
INV(N)	Real (array)	curies	NA	NA	NA	intial inventory of Nth radionuclide
(NOTE: THE INVENTORY IS NOT USED WHEN AN EXTERNAL SOURCE TERM IS USED)						
HLIFE(N)	Real (array)	years	NA	NA	NA	half life of Nth radionuclide
EPAWGT(N)	Real (array)	--	NA	NA	NA	weighting factor for calculating EPA limits
AREA(N,J)	Real (array)	m^2	NA	NA	NA	area of the inlet(J=1) and outlet(J=2) for the Nth part
LEGLEN(N,J)	Real (array)	meters	NA	NA	NA	thickness of the Jth unit for the Nth part
SATLEN(N,J)	Real (array)	meters	NA	NA	NA	thickness of the Jth saturated unit for the Nth part
GDENSE(N,J)	Real (array)	kg/m^3	NA	NA	NA	grain density for the Nth unit and matrix(J=1) or fracture(J=2)
INTERNAL ***INFIL(N)***	***Real (array)***	***m/yr***	***NA***	***NA***	***NA***	***output of flow module - flux for the Nth part***

GLOBAL (SAMPLED)

UNIT 5

NAME	TYPE	UNITS	VALUE	RANGE	SOURCE	DESCRIPTION
COND(N,J)	Real (array)	m/yr	NA	NA	NA	matrix(J=1) and fracture(J=2) conductivity for the Nth unit
POR(N,J)	Real (array)	--	NA	NA	NA	matrix(J=1) and fracture(J=2) porosity for the Nth unit
BETA(N,J)	Real (array)	--	NA	NA	NA	matrix(J=1) and fracture(J=2) Beta or power term in the van Genuchten equation for the Nth unit
GRAD(N)	Real (array)	--	NA	NA	NA	gradient for the Nth unit (saturated units)
DISPER	Real	meters	10.			dispersion length
TINFIL	Real	m/yr	1.0	0.0-5.0	?	net recharge
KD(N,J,K)	Real (array)	m^3/kg	NA	NA	NA	Kd for the Nth radionuclide, the Jth unit, for matrix(K=1) and fracture(K=2)

NEFTRAN Fortran Program
Static and Dynamic Analysis

June 29, 1993

Earl S. Marwil
John E. Tolli
Scientific Computing Unit
Idaho National Engineering Laboratory

1. Introduction

This analysis was performed on the Cray version of the software as provided by Southwest Research Institute (SwRI).

One sample problem was supplied along with the source code. The program was analyzed using the Craft (Cross Reference Analysis of Fortran) tool, FORWARN, the Fortran 77 analyzer, and PC-Metric. These tools provide static analysis, coverage analysis, and complexity analysis.

2. References

[1] N.H. Marshall and E.S. Marwil, Cross Reference Analysis of Fortran (CRAFT), EG&G-CATT-9198, EG&G Idaho, Inc., July 1991.
[2] Fortran 77 Analyzer User's Manual, National Bureau of Standards, NBS GCR 81-359, 1981
[3] FORWARN User's Guide, Quibus Enterprises, Inc., July 1991.
[4] PC-Metric User's Guide, SET Laboratories, Inc., 1987.

3. Functions

The NEFTRAN program contains 50 Fortran routines.

NEFTRAN has no alternate entry points.

NEFTRAN has 5 extraneous routines: "rdsmpl", "skpfld", "skpspc", "upcase", and "getfld".

4. Common Block Irregularities

There are 19 common blocks in the NEFTRAN program.

All common block declarations are consistent.

There are 41 instances of a common block being declared in a routine in which none of its elements are otherwise referenced.

5. Interface Irregularities

No exceptions to report.

6. Local Variable Irregularities

There are 367 instances of a parameter not being used in a module in which it is declared. This is mainly due to the parameters being declared via a comdeck.

There are 186 instances of a local variable being undefined and unused in a module in which it is declared.

Integer variable "nubf" is defined but unused in module "facer". Character variable "attrib" is defined but unused in module "opnfil".

Dummy arguments "jtrial" and "vector" are unused in module "getrv". Dummy argument "arenum" is unused in module "inidr2". Dummy arguments "arenum" and "ipass" are unused in module "upddr2".

7. Fortran Extensions

Twenty-five program modules contain some lower case alphabetic characters in their active Fortran.

Program modules "flowin", "getrv", "lechmod", "source", and "stopper" contain entity names which are longer than 6 characters.

Program modules "et", "ratio", and "tspfac" call the non-standard intrinsic function "dfloat".

Program modules "flowin", "skpfld", "skpspc", and "work" contain character assignment statements in which there are potential overlaps.

Program modules "getfld", "work", and "wrtdr2" have format statements which use the "x" descriptor without a repeat count.

Program modules "band", "chain", "dxdt", "facer", "git", "inprin", and "stopper" contain format statements which omit a comma between two consecutive format fields.

8. Optimization

The following table summarizes the performance data gathered from execution of the sample problem. Only those routines exercised by the sample problem are shown (see "Coverage Analysis" for a list of routines not exercised by the sample problem, i.e., coverage = 0%). The table lists all program modules in descending order according to CPU time. To optimize code execution time, emphasis should be placed on those modules which appear highest in the listing.

The performance data show that a high percentage of the overall execution time (80.529%) is spent in the first 5 routines listed (SOURCE, RATIO, SETDIS, ET, TRNSPT). This is due primarily to the following (applies to some or all of the 5 routines):

1) a low percentage of floating point operations which are performed in vector mode (%Vflops is small)

2) a high overhead factor for calls to the routines (IFact > 1)

3) a high level of memory conflicts (MC/MR > 1)

4) a high rate of instruction buffer fetches (IBFR > 1).

A detailed optimization analysis effort should focus on these 4 areas.

PERFORMANCE DATA FOR NEFTRAN

ROUTINE NAME	Time	%ExTime	%AccumT	%Vflops	IFact	MC/MR	IBFR
SOURCE	17.330	33.469	33.469	0.00000	0.00	0.845	0.752
RATIO	13.496	26.066	59.534	0.00000	5.43	3.646	0.964
SETDIS	4.517	8.724	68.258	0.00000	0.00	2.484	1.094
ET	3.262	6.300	74.558	19.73464	5.22	3.914	1.305
TRNSPT	3.092	5.971	80.529	99.80099	0.00	0.387	0.154
SRCIN	2.897	5.596	86.125	32.04083	0.70	1.447	1.184
PRP	2.225	4.298	90.423	51.49160	0.08	0.984	0.761
TIMVAL	1.500	2.897	93.320	0.00000	0.12	1.501	0.013
INPRIN	0.818	1.579	94.899	0.00000	0.00	1.196	0.896
FLOWIN	0.650	1.256	96.155	0.23367	0.00	0.907	0.955
SETUP	0.588	1.135	97.290	58.62715	0.00	1.143	0.768
WRTDR2	0.521	1.006	98.296	0.00000	0.00	1.086	1.027
METHOD	0.214	0.413	98.709	0.00000	0.01	0.162	0.148
TPPRT	0.134	0.258	98.967	15.32202	16.09	3.485	0.834
TSPFAC	0.129	0.249	99.216	22.72657	0.01	0.898	0.567
STOPPER	0.120	0.232	99.447	7.47654	0.05	1.640	0.435
NEFDIT	0.075	0.145	99.592	0.74832	0.00	2.144	1.031
WORK	0.046	0.088	99.680	85.74151	0.00	1.391	0.939
TIMER	0.037	0.071	99.751	98.12270	0.00	0.957	0.299
PTHLEN	0.032	0.061	99.812	26.26593	0.00	1.211	0.851
BRANCH	0.023	0.045	99.857	31.14604	0.06	1.050	0.559
DXDT	0.016	0.030	99.887	47.84756	0.10	1.175	0.733
UPDDR2	0.016	0.030	99.917	87.78932	0.00	1.701	1.161
CHAIN	0.011	0.021	99.939	30.44983	0.14	1.377	0.868
TRACER	0.009	0.018	99.956	34.01477	0.44	2.323	0.449
UPDCUR	0.009	0.017	99.974	90.80559	0.00	1.062	1.049
NEF	0.007	0.014	99.988	0.00000	0.00	1.923	1.148
CATCH	0.003	0.005	99.993	0.00000	0.53	3.984	0.595
OPNFIL	0.001	0.003	99.996	0.00000	0.00	1.016	0.537
GETRV	0.001	0.002	99.998	0.00000	0.01	1.687	1.509
INIDR2	0.000	0.001	99.999	0.00000	0.00	0.480	0.078
CHKPTH	0.000	0.001	100.000	0.00000	0.03	3.547	0.407
SPLIT	0.000	0.000	100.000	0.00000	0.08	13.010	1.227
Totals (All Traced Routines)	51.779	100.000	100.000	67.38434	5.56	0.776	0.844

Key:

%AccumT	- accumulated percentage of total CPU time
%ExTime	- percentage of total CPU time
%Vflops	- percentage of floating point operations due to vector floating point operations
IBFR	- Instruction Buffer Fetch Rate (megafetches/sec)
IFact	- Inline Factor (total calls to routine / average time spent in routine for each call)
MC	- number of memory conflicts
MR	- number of memory references
Time	- total CPU time (sec)

9. Coverage Analysis

NEFTRAN aborts when loaded with a core preset of indefinite.

One sample problem was supplied. A coverage analysis shows that this problem yielded a 39% segment coverage of NEFTRAN. Sample problems provided with simulation programs typically achieve 35% to 50% coverage. A statement of software quality cannot be made for routines that have low coverage, i.e., large portions of the code are untested.

Note that 17 routines have 0% coverage. These routines are not tested with the supplied sample problem.

One routine achieves 1%-19% coverage, 4 routines achieve 20%-39% coverage, 6 routines achieve 40%-59% coverage, 8 routines achieve 60%-79% coverage, 9 routines achieve 80%-99% coverage, and 5 routines achieve 100% coverage.

The following table shows the percent coverage for each routine.

Module Name	Number of Segments in module	Number of Segments Executed	Percent Segment Coverage
NEF	32	19	59.4
ADJB	53	0	0.0
BAND	20	0	0.0
BRANCH	90	56	62.2
BSOLVE	20	0	0.0
CATCH	8	8	100.0
CHAIN	98	28	28.6
CHKPTH	21	10	47.6
COEFF	93	0	0.0
DTUPDT	110	0	0.0
DXDT	178	53	29.8
ET	48	46	95.8
FACER	165	0	0.0
FLOWIN	225	135	60.0
GETFLD	1	0	0.0
GETRV	5	5	100.0
GIT	10	0	0.0
INIDR2	5	5	100.0

Module Name	Number of Segments in module	Number of Segments Executed	Percent Segment Coverage
INPRIN	46	32	69.6
INTG	26	0	0.0
LECHMO	19	0	0.0
METHOD	50	22	44.0
MXCLL	145	0	0.0
NEFDIT	27	25	92.6
OPNFIL	28	10	35.7
PRP	17	12	70.6
PTHLEN	33	20	60.6
RATIO	44	22	50.0
RDSMPL	20	0	0.0
SETDIS	104	91	87.5
SETUP	124	101	81.5
SIFT	11	0	0.0
SKPFLD	3	0	0.0
SKPSPC	4	0	0.0
SOURCE	156	16	10.3
SPLIT	3	3	100.0
SRCIN	39	34	87.2
STOPPE	402	170	42.3
TIMER	56	39	69.6
TIMVAL	13	11	84.6
TPPRT	33	23	69.7
TRACER	21	17	81.0
TRNSPT	91	51	56.0
TSPFAC	28	19	67.9
UPCASE	4	0	0.0
UPDCUR	11	10	90.9
UPDDR2	5	5	100.0
WORK	103	21	20.4
WRTDR2	9	8	88.9
XCHNG	21	0	0.0
Totals	2878	1127	39.2

```
                   0.20      0.40      0.60      0.80      1.00
          ----+----|----+----|----+----|----+----|----+----|
NEF      |******************************         |         |
ADJB     |         |         |         |         |         |
BAND     |         |         |         |         |         |
BRANCH   |*******************************        |         |
BSOLVE   |         |         |         |         |         |
CATCH    |**************************************************
CHAIN    |**************     |         |         |         |
CHKPTH   |************************     |         |         |
COEFF    |         |         |         |         |         |
DTUPDT   |         |         |         |         |         |
DXDT     |***************    |         |         |         |
ET       |************************************************ |
FACER    |         |         |         |         |         |
FLOWIN   |******************************         |         |
GETFLD   |         |         |         |         |         |
GETRV    |**************************************************
GIT      |         |         |         |         |         |
INIDR2   |**************************************************
INPRIN   |***********************************    |         |
INTG     |         |         |         |         |         |
LECHMO   |         |         |         |         |         |
METHOD   |**********************       |         |         |
MXCLL    |         |         |         |         |         |
NEFDIT   |**********************************************   |
OPNFIL   |****************** |         |         |         |
PRP      |***********************************    |         |
PTHLEN   |******************************         |         |
RATIO    |*************************    |         |         |
RDSMPL   |         |         |         |         |         |
SETDIS   |********************************************     |
SETUP    |*****************************************        |
SIFT     |         |         |         |         |         |
SKPFLD   |         |         |         |         |         |
SKPSPC   |         |         |         |         |         |
SOURCE   |*****    |         |         |         |         |
SPLIT    |**************************************************
SRCIN    |********************************************     |
STOPPE   |*********************        |         |         |
TIMER    |***********************************    |         |
TIMVAL   |******************************************       |
TPPRT    |***********************************    |         |
TRACER   |****************************************         |
TRNSPT   |**************************** |         |         |
TSPFAC   |**********************************     |         |
UPCASE   |         |         |         |         |         |
UPDCUR   |*********************************************    |
UPDDR2   |**************************************************
WORK     |**********         |         |         |         |
WRTDR2   |********************************************     |
XCHNG    |         |         |         |         |         |
          ----+----|----+----|----+----|----+----|----+----|
```

```
      coverage = 0.         ADJB    BAND    BSOLVE  COEFF   DTUPDT
                            FACER   GETFLD  GIT     INTG    LECHMO
                            MXCLL   RDSMPL  SIFT    SKPFLD  SKPSPC
                            UPCASE  XCHNG

0.01 <= coverage < 0.20     SOURCE

0.20 <= coverage < 0.40     CHAIN   DXDT    OPNFIL  WORK

0.40 <= coverage < 0.60     NEF     CHKPTH  METHOD  RATIO   STOPPE
                            TRNSPT

0.60 <= coverage < 0.80     BRANCH  FLOWIN  INPRIN  PRP     PTHLEN
                            TIMER   TPPRT   TSPFAC

0.80 <= coverage < 0.85     SETUP   TIMVAL  TRACER

0.85 <= coverage < 0.90     SETDIS  SRCIN   WRTDR2

0.90 <= coverage < 0.95     NEFDIT  UPDCUR

0.95 <= coverage < 1.00     ET

      coverage = 1.00       CATCH   GETRV   INIDR2  SPLIT   UPDDR2

 Program coverage for this run =0.39
```

10. Complexity Analysis

Some key metrics are the number of executable statements (sloc), the number of non-blank comments (ncomt), McCabe's extended cyclomatic complexity (vg2), the number of branching statements (cgoto, ugoto, bIF, and lIF), and Halstead's predicted number of errors in (re)writing the code (bhat). Measures are normalized per 100 executable statements for ease of comparison and are listed in the table below.

The branching measures for this code indicate few unconditional GO TO statements and logical IFs for most program modules. This code appears to be well structured.

Many routines have a good ratio of non-blank comments to source code. However, some routines have a low ratio, e.g., "adjb", "chkpth", "coeff".

McCabe's extended cyclomatic complexity (vg2), normalized per 100 lines of source code, indicates high values. Generally, the routines with the highest complexity are those most likely to have defects. As a guideline, normalized measures of 15 or greater should be considered complex. A software maintenance program should focus on those routines with the highest measures.

Complexity Report by Subprogram for NEFTRAN

Name	loc	sloc	cmnt	ncomt	ncomt /sloc	vg2 /sloc	cgoto	cgoto /sloc	ugoto	ugoto /sloc	bIF	bif /sloc	lIF	lif /sloc	Bhat
NEF	376	75	259	212	282.7	18.7	0	0.0	3	4.0	8	10.7	2	2.7	1
ADJB	139	88	35	16	18.2	29.5	0	0.0	4	4.5	12	13.6	6	6.8	2
BAND	123	48	41	20	41.7	14.6	0	0.0	2	4.2	4	8.3	2	4.2	1
BRANCH	245	166	57	24	14.5	25.3	0	0.0	15	9.0	15	9.0	8	4.8	2
BSOLVE	82	48	19	8	16.7	20.8	0	0.0	2	4.2	4	8.3	0	0.0	1
CATCH	45	12	13	4	33.3	41.7	0	0.0	0	0.0	1	8.3	1	8.3	0
CHAIN	271	174	59	22	12.6	28.7	0	0.0	9	5.2	13	7.5	15	8.6	2
CHKPTH	74	41	10	2	4.9	19.5	0	0.0	7	17.1	5	12.2	2	4.9	1
COEFF	236	163	38	13	8.0	27.6	0	0.0	8	4.9	21	12.9	5	3.1	2
DTUPDT	280	200	49	21	10.5	26.0	0	0.0	9	4.5	23	11.5	12	6.0	3
DXDT	537	332	150	74	22.3	28.0	0	0.0	12	3.6	44	13.3	23	6.9	6
ET	110	85	24	14	16.5	24.7	0	0.0	5	5.9	3	3.5	5	5.9	2
FACER	468	332	101	42	12.7	23.5	0	0.0	10	3.0	29	8.7	13	3.9	5
FLOWIN	764	452	214	143	31.6	22.8	0	0.0	18	4.0	45	10.0	19	4.2	9
getfld	56	3	45	36	1200.0	33.3	0	0.0	0	0.0	0	0.0	0	0.0	0
GETRV	164	9	97	67	744.4	33.3	0	0.0	0	0.0	1	11.1	0	0.0	0
GIT	50	26	19	9	34.6	19.2	0	0.0	0	0.0	2	7.7	2	7.7	1
inidr2	71	9	27	13	144.4	33.3	0	0.0	0	0.0	0	0.0	0	0.0	0
INPRIN	435	98	162	125	127.6	14.3	0	0.0	0	0.0	3	3.1	0	0.0	2
INTG	65	56	12	7	12.5	21.4	0	0.0	0	0.0	2	3.6	7	12.5	1
LECHMOD	60	29	17	7	24.1	20.7	0	0.0	2	6.9	5	17.2	0	0.0	0
METHOD	233	105	78	34	32.4	21.9	0	0.0	3	2.9	6	5.7	6	5.7	1
MXCLL	399	336	47	28	8.3	21.1	0	0.0	38	11.3	30	8.9	22	6.5	7
nefdit	138	44	51	28	63.6	27.3	0	0.0	0	0.0	5	11.4	0	0.0	1
opnfil	208	59	131	118	200.0	27.1	0	0.0	0	0.0	8	13.6	0	0.0	1
PRP	45	35	11	6	17.1	22.9	0	0.0	0	0.0	1	2.9	4	11.4	1
PTHLEN	315	77	182	151	196.1	22.1	0	0.0	0	0.0	7	9.1	0	0.0	1
RATIO	103	89	11	7	7.9	18.0	0	0.0	11	12.4	12	13.5	3	3.4	1
rdsmpl	197	42	92	67	159.5	19.0	0	0.0	2	4.8	6	14.3	0	0.0	1
SETDIS	269	204	33	8	3.9	23.5	0	0.0	12	5.9	22	10.8	8	3.9	3
SETUP	425	256	92	47	18.4	23.4	0	0.0	3	1.2	20	7.8	7	2.7	5
SIFT	24	21	4	4	19.0	23.8	0	0.0	3	14.3	2	9.5	2	9.5	0
skpfld	15	6	7	6	100.0	33.3	0	0.0	0	0.0	1	16.7	0	0.0	0
skpspc	18	8	12	9	112.5	25.0	0	0.0	1	12.5	1	12.5	0	0.0	0
SOURCE	472	316	118	49	15.5	23.4	0	0.0	9	2.8	33	10.4	8	2.5	3
split	11	6	2	0	0.0	33.3	0	0.0	0	0.0	1	16.7	0	0.0	0

Name	loc	sloc	cmnt	ncomt	ncomt /sloc	vg2 /sloc	cgoto	cgoto /sloc	ugoto	ugoto /sloc	bIF	bif /sloc	lIF	lif /sloc	Bhat
SRCIN	178	92	55	24	26.1	21.7	0	0.0	3	3.3	5	5.4	0	0.0	1
STOPPER	945	647	107	59	9.1	37.7	0	0.0	5	0.8	97	15.0	49	7.6	11
TIMER	155	101	31	11	10.9	26.7	0	0.0	6	5.9	5	5.0	8	7.9	2
timval	59	21	34	25	119.0	33.3	0	0.0	3	14.3	2	9.5	0	0.0	0
TPPRT	186	50	93	66	132.0	28.0	0	0.0	3	6.0	5	10.0	4	8.0	1
TRACER	104	51	31	11	21.6	17.6	0	0.0	2	3.9	4	7.8	2	3.9	1
TRNSPT	294	196	51	23	11.7	20.9	0	0.0	1	0.5	14	7.1	4	2.0	3
TSPFAC	118	85	25	19	22.4	15.3	0	0.0	1	1.2	7	8.2	1	1.2	3
upcase	35	7	26	19	271.4	42.9	0	0.0	0	0.0	1	14.3	0	0.0	0
updcur	105	22	46	32	145.5	27.3	0	0.0	0	0.0	2	9.1	0	0.0	0
upddr2	76	10	29	15	150.0	30.0	0	0.0	0	0.0	0	0.0	0	0.0	0
WORK	346	192	112	71	37.0	25.0	0	0.0	10	5.2	13	6.8	8	4.2	2
wrtdr2	92	16	35	18	112.5	25.0	0	0.0	0	0.0	1	6.3	0	0.0	0
XCHNG	97	44	22	6	13.6	25.0	0	0.0	1	2.3	3	6.8	0	0.0	1

Legend of Metrics in Report

loc -- lines of code
sloc -- number of executable statements
cmnt -- total number of commnts
ncomt -- number of non-blank COMMENT statements
100*ncomt/sloc -- percent, nonblank comments to number of executable statements
100*vg2/sloc -- percent, extended complexity of number of executable statements
cgoto -- number of COMPUTED GO TO statements
100*cgoto/sloc -- percent, computed GOTO's to number of executable statements
ugoto -- number of UNCONDITIONAL GO TO statements
100*ugoto/sloc -- percent, unconditional GOTO's to number of executable statements
bIF -- number of BLOCK IF statements
100*bif/sloc -- percent, Block IF statements to number of executable statements
lIF -- number of LOGICAL IF statements
100*lif/sloc -- percent, logical IF statements to number of executable statements
Bhat -- Halstead's predicted number of errors in writing code