SCIENTIFIC NOTEBOOK

170-4E

SCIENTIFIC NOTEBOOK

by

Carol Scherer

Printed: March 23, 2005

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

SCIENTIFIC NOTEBOOK

by

Carol Scherer

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

C. Scherer

11/3/98

Contacted by Michael Muller (Div. 15) and Rob Rice (consultant) about running Forstudy on latest version of TPA code.

11/5/98

Working in Room 204 which is shared by Rob and Michael. Using Michael's PC. This version of TPA is supposed to run on both PC and SUN. There are two new files: zportpc.f and zportunix.f. Zportpc.f is included in the compile if the executable is targeted for the PC and zportunix.f is included if the target is the SUN. The file sch.f is not included in this version. Peakfinder.f was renamed to peakfind.f.

My password(s) had changed or expired or something. They didn't work. Ray Kotara came by and reset them for me.

Copied latest version of source code from Michael Muller's directory /bscr0/carol and subdirectories to cscherer/working directory on bigbend. These files will be the source files. Copied all files to forstudy subdirectories. These files will be the ones that are modified for Forstudy.

Sitakanta came to talk. He wants me to train Michael on Forstudy so that there will be at least one more person who knows how to run it. Will have to work out a schedule with Michael since we both have odd work schedules.

Ran Forstudy on all six standalone source code files in codes subdirectory. Also ran all_exec.prj to see where the problems lie this time around.

Called Michael and arranged to meet with him early tomorrow morning. Will run the TPA source code files then.

11/6/98

Met with Michael Muller and explained how to use Forstudy. We ran Forstudy on all the source code files used to generate tpa.e. The file with the most messages is zportpc.f. This was written by Mike Epley using the Lahey FORTRAN compiler for the PC. Apparently, Lahey uses some FORTRAN 90 syntax that Forstudy does not recognize. Either these are extensions to the standard by Lahey or else they are "new" compared to what Forstudy accepts as valid FORTRAN 90. After discussion, Michael M. and I agreed that to keep the code maintainable in future, that it would be better to revert most of tnis FORTRAN 90 code back to FORTRAN 77 syntax conventions.

11/7/98

Nefmks.f is at the same stage as the last release. That is, there are Forstudy messages related to assigning real*8 variables to real*4 variables (implicit truncation). For the same reason that these remained in the last version, they will also remain in this version. Michael is also resolving some final messages for snllhs.f. I told him what we had done for similar messages in the past.

11/11/98 – 11/13/98

Snllhs.f is complete. Michael still needs to finish zportpc.f.

We made two new Forstudy project (.prj) files: tpaunx.prj and tpapc.prj. Tpaunx.prj will use Forstudy to analyze all the tpa source code including zportunx.f while tpapc.prj will use Forstudy to analyze all tpa source code including zportpc.f. Michael ran both these projects. The only messages remaining are missing interface warnings and warnings about the practice of passing constants to subroutines. Since Fortran passes all parameters by reference, there is the possibility that the subroutine may attempt to modify the constant, which is against programming rules. In the long term, these messages should be resolved, but it is probably not vital that they be resolved for this version.

11/16/98 –11/20/98

Michael Muller has tested both versions of tpa (pc and unix platforms) to compare the outputs of the pre-Forstudy code with the outputs of the post-Forstudy code. He reports that the results are the same except for timestamps and some platform-specific labels..

****

7/2/98

Report put together (included following this paragraph) for Sitakanta

Results of Forstudy Runs on TPA Code 11/5/98 – 11/20/98

Each source code file making up the TPA executable as well as the source code files for standalone executables was run through Forstudy, a code analyzer and pre-processor. Two Forstudy runs are made for each source code file using a project file (.PRJ) that directs Forstudy what to do: the first one generates an interface file (.NTF) and the second one analyzes the code for potential problems. Report files (.RPT) are generated.

The TPA executable and each standalone executable were recompiled. Warning and error messages generated by the FORTRAN compiler are included in this report for each module if any were generated.

TPA Modules:

TPA Files That Required No Changes:

Array.f	generated no Forstudy messages that require code modifications
Ashplumo.f	generated no Forstudy messages that require code modifications
Ashrmovo.f	generated no Forstudy messages that require code modifications
Condxyzt.f	generated no Forstudy messages that require code modifications
Dcags.f	generated no Forstudy messages that require code modifications
Dcagw.f	generated no Forstudy messages that require code modifications
Ebsfail.f	generated no Forstudy messages that require code modifications
Ebsrel.f	generated no Forstudy messages that require code modifications
Faulto.f	generated no Forstudy messages that require code modifications
Fileunit.f	generated no Forstudy messages
Findelev.f	generated no Forstudy messages that require code modifications
Invent.f	generated no Forstudy messages that require code modifications
Mv.f	generated no Forstudy messages that require code modifications
Nfenv.f	generated no Forstudy messages that require code modifications

Numrecip.f	generated no Forstudy messages
Peakfind.f	generated no Forstudy messages that require code modifications
Ran.f	generated no Forstudy messages that require code modifications
Reader.f	generated no Forstudy messages that require code modifications
Sampler.f	generated no Forstudy messages that require code modifications
Subarea.f	generated no Forstudy messages
Szft.f	generated no Forstudy messages that require code modifications
Uzflow.f	generated no Forstudy messages that require code modifications
Volcano.f	generated no Forstudy messages that require code modifications
Zportunx.f	generated no Forstudy messages that require code modifications

The Forstudy messages that require no code modifications are warnings related to module definitions in the interface (.NTF) file used to run Forstudy. The interface file is generated by Forstudy; however, Forstudy is not able to create the interface definition for some subroutines and functions for inline functions or for routines that are contained in separate source code files. The messages can be resolved by adding definitions for these files to the interface file(s) or by including all necessary source code files in one .prj file. This is what happens when all_exec.prj is run, which should resolve all the interface problems. The warnings have nothing to do with the code itself. If Forstudy generated only these kind of messages or generated no messages at all for a file, then that file was considered to be OK with no further code modifications.

TPA Files That Require Changes:

Exec.f	1 warning messages regarding loss of precision in assigning REAL*4 value to INTEGER*4 variable (using implicit truncation)
Seismo.f	1 warning message (include file never used)
Uzft.f	1 warning message (include file never used)
Zportpc.f	4 error messages (Forstudy does not recognize Lahey syntax; 6 warning messages (REAL*4 value to INTEGER*4 variable)

Forstudy was also run against all the TPA source code together. This run identifies potential problems with the interfaces between routines. One run was made with zportpc.f and one with zportunx.f. Other than missing interface warnings, the warning messages were generated for the practice of passing constants as parameters to subroutines. Although allowed by FORTRAN compilers, this practice is not good programming methodology because FORTRAN passes all parameters by reference. This raises the possibility that a subroutine could attempt to modify a constant value. These messages are not being resolved in this version.

Standalone Code:

Standalone Files That Require No Changes:

Ashplume.f	generated no Forstudy messages
Failt.f	2 warnings (missing interface and different type and size used in arg #12 of RKQC)
Ebsfilt.f	generated no Forstudy messages
Releaset.f	generated no Forstudy messages that require code modifications

Standalone Files That Require Changes:

Nefmks.f	10 error (intrinsic MAX() args differ REAL*4 vs. 8) and 67 warnings (1 missing interface and 66 loss of precision in assignment: REAL8 to REAL*4)
Snllhs.f	??? warnings (commons included but not used; mixed strings & arithmetic objects; loss of precision; common changed relative to prior use)
Snllhs.f	7 warnings generated by compile (variables never used; no argument for EXIT())

INITIAL ENTRIES

Scientific Notebook: #170-4e

Issued to: S. Mohanty

Issue Date: April 3, 1996

Account Number: 20-01402-762

Title: TPA 4.2

Participants: Carol Scherer

03/04/2002 – 03/08/2002 –

Will be working half-time in Div. 20 to modify code and test changes on next version of TPA. Worked with Div. 20 staff to set up work area in IMS lab. Copied latest version of TPA to cscherer directory on Spock. Began analyzing SPCR PA-SCR-344.

03/11/2002 – 03/15/2002 –

Set up testing subdirectory "tpatest". Compiled tpa.e after Ron J. updated .cshrc file to tie in Fortran compiler. Found parameters that need to be used in testing SCR #344 in tpa.inp: GenerateRestartFlag, NumberofRealizations, StartAtRealization, and StopAtRealization. There are also two flags set for subareas: StartAtSubarea and StopAtSubarea. Default number of subareas is 10.

Tried a base run of executable but having path problems. Can't access the /data subdirectory. Tried to change path and dpath in path.i, but after recompiling, still couldn't access data subdirectory.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Will work on scientific notebook and test plan documentation on my laptop, which already has word processor installed. Computer in IMS lab doesn't. Contacted Mike Muller about possibility of sharing office space (Rm 244) outside of lab.

03/18/2002 – 03/22/2002 –

Brought laptop to lab for documentation work. Worked out path problem. I'm using /net/spock/home/cscherer/tpatest as my working directory. Files pertaining to SCR #344 are in subdirectory .../scr344.

Set up skeleton forms for Test Plans and Test Results for the problem reports. Gave preliminary plan for SCR# 344 to Ron J. who returned it with suggested changes along with improvements for the test plan format.

Made several runs of tpa with different number of subareas and realizations. Originally hoped to be able to compare the restart output file check.pnt from different runs, but had forgotten that check.pnt is a binary file, so it cannot be examined directly. All check.pnt files are the same size regardless of number of subareas and realizations, so comparing sizes will not work either. Finally, decided to write small Fortran program to open check.pnt and read in the stored arrays. The plan is to examine one or two arrays where the values are stored by realization to see if values are filled in only for those realizations that are requested by the user. This will test part A of the SCR. Part B will be tested by making two runs of tpa.e with a tpa.inp that differs only in whether the GenerateRestart flag is set or not. Testing SCR 344 will take longer than originally anticipated, but should be done early next week.

Using the restart code in exec.f, created testcheck.pg. Test program will open check.pnt and read in the stored values. The specific values stored in totalmaxdosetime will be printed to an output file (check.out) by realization number and used for comparison with the output from other check.pnt files. Working on completing, compiling, and testing checktest.f.

03/25/2002 – 03/29/2002 –

Created three tpa.inp files to use in testing the restart/checkpoint code. These are named tpa011009.inp, tpa061009.inp, and tpa010403.inp. The first two digits in the name are the StartAtRealization, the second two digits are the StopAtRealization, and the last two digits refer to the last completed realization before the tester should abort the run. In all three tpa.inp files, the NumberOfRealizations is 10, StartAtSubarea is 7, and StopAtSubarea is 10. The GenerateRestartFlag is set to 1.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

The test plan for SCR# 344 was updated. Four system level tests were specified. Test 1 called for running tpa.e three times, using each of the three tpa.inp files for a run. Test 2 called for a tpa.e run using any of the three input files, but with GenerateRestartFlag set to 0. Test 3 involved inserting a STOP in exec.f before the 1st write to check.pnt (tpa_prestop.e). Test 4 involved inserting a STOP in exec.f after the last write to check.pnt (tpa_poststop.e).

Test 1: Tpa.e was run using each of the three tpa.inp files and were aborted after the designated realization was completed. A check.pnt file was created for each run and renamed cp011009.pnt, cp061009.pnt, and cp010403.pnt in accordance with the tpa.inp file used. Checktest.f was run on each check.pnt file and the results saved in cp.011009.out, cp061009.out, and cp010403.out. The cp.out files were visually inspected to ensure that values were written out only for the specified realizations. Test passed.

Test 2: After changing the flag to 0 in any of the three tpa.inp files, tpa was rerun. No check.pnt file was generated. Test passed.

Test 3: Mod was made to exec.f, tpa.e was recompiled and rerun using any of the 1st three input files. Program was STOPped before the first write to check.pnt. No check.pnt file was generated. Test passed.

Test 4: Mod was made to exec.f, tpa.e was recompiled and rerun using any of the 1st three input files. Program was STOPped after last write to check.pnt. This check.pnt file was the same as the one generated in test 1. Test passed

Copy of test plan and results was given to Ron J. Ron asked for test 4 to be eliminated.

All output files for tests 1 – 3 (test 4 output was deleted) were saved in /net/spock/home/cscherer/tpatest/scr344. A CD was created with .../tpatest subdirectory

04/01/2002 – 04/05/2002 –

Ron asked that a new test be added for check.pnt. This test should count the number of writes made to check.pnt to ensure that the code is only writing the correct number of times. Make copy of exec.f called exec_count.f and add the variable writecount. Add lines to code after "write (icheckpointinp):

```
write (icheckpoint)
close(icheckpoint)
writecount = writecount + 1
```

print *, 'writecount = ', writecount

Found minor problem, an unnecessary write of the check.pnt file. Generate new SCR #393 and fix along with #344.

Create new directory .../tpatest/scr393. Copy modified exec.f from /dev (v. 4.2a) and recompile. Create test plan for scr #393 based on the last test added to the test plan for #344.

Ran tpa.e w/ tpa061009.inp: test passed.

Ran tpa.e w/ tpa050909.inp: test passed

Made CD for scr #393 and gave to Ron.

Move on to scr # 376 dealing w/ sawidth; will need a new input parameter. SCR #376:

Affected Software Module(s), Description of Problem(s):

FT6 Reflect uncertainty in streamtube dimensions

The streamtube dimensions of width and length are specified in the *strmtube.dat* file as a constant table used for all realizations. It is desired to reflect the uncertainty of these data by subjecting them to the influence of sampled parameters. This process will require the generation of a new sampled parameter representing a streamtube width multiplier. All routines that read the *strmtube.dat* file to obtain streamtube dimensions will be modified to include the use of the new parameters. This width multiplier will be applied to all streamtubes for their entire lengths. The range of this multiplier will be consistent with the data values in the *strmtube.dat* file so that the total streamtube width remains within realistic limits.

Add to tpa.inp a sampled parameter called StreamTubeWidthMultiplier. Look at how new parameters are handled and how valuesp and ispquery work in reader. Code mods already worked out but not yet implemented. Made the following changes to code:

```
szft: lines 398,401
< cc
< cc css - 4/5/2002; SCR# 376:  add streamtube width multiplier
< cc
<       double precision widthmult
szft: lines 426,431
< cc
< cc css - 4/5/2002; SCR# 376:  add streamtube width multiplier;
```

```
< cc         declare logical & index
< cc
<       integer stwmindex
<       logical thistimeonly
szft: lines 452,459
< cc
< cc css - 4/5/2002; SPR# 376 - add streamtube width multiplier
< cc       set logical thistimeonly to true; save thistimeonly & stwmindex
<       data thistimeonly /.true./
<
<       save thistimeonly
<       save stwmindex
<
szft: lines 480,497
< cc
< cc css - 4/5/2002; SCR# 376:  add streamtube width multiplier
< cc
<       if (thistimeonly) then
<          spname = 'StreamTubeWidthMultiplier[]'
<          stwmindex = ispquery(spname)
<          thistimeonly = .false.
<        end if
<        widthmult = valuesp(stwmindex)
<        print *, 'szft: widthmult = ', widthmult
<        if (widthmult .le. 0) then
<           print *, ' '
<           print *, ' ***>>> Error in szft <<<*** '
<           print *, ' Invalid value for widthmult: ', widthmult,
<      &            '     (must be greater than 0.0)'
<           print *, ' '
<           STOP
<        endif
szft: lines 773
< cc
szft: lines 786,788
< c
< c  This assumes that the first tube length is strmtube.dat's 0.0.  This
enables the width of
< c  the width of the mixing zone to remain as given in strmtube.dat.
szft: lines 790
< cc
szft: lines 800,805
< cc
< cc css - 4/5/2002; SCR# 376:  add streamtube width multiplier
< cc
< cc        nefvel(inc)= flowrate/por(i)/sawidth(i)
<           nefvel(inc)= flowrate/por(i)/(sawidth(i)* widthmult)
< cc
```

Also, make the following change(s) to tpa.inp:

```
tpa.inp: line 3675
```

```
> ** css - 4/5/02:  SCR# 376 - add multiplier for streamtube width
> uniform
> StreamTubeWidthMultiplier
> -2.0, 2.0
> **
```

Saved copy of modified code as szft376.f Code compiles and runs. Peak Mean Dose is 1.04314E-02 rem/yr at 84700.0 yr.

04/08/2002 – 04/12/2002 –

Made corrections to text in test plans for SCR # 344 and SCR # 393 and gave them to Ron.

For SCR # 376: default abbreviation for new sampled parameter StreamTubeWidthMultiplier is STWM1. Check w/ Ron to see if that's OK.

Added print statements to szft.f for testing purposes. Saved this version as szft_test376.f Recompiled and ran w/ same Peak Mean Dose result as above. Need to look at ground water travel time (gwtt). Look at gwttuzsz.res.

May need to make changes to dcagw.f also.

Test plan for SCR # 344 approved. Made CD of all files modified or output for testing 376 and turned in to Ron. Additional change needed to exec for SCR # 393 to prevent writing out the file
for the stop realization. Made mod and re-ran test plan. Test plan passed. Test plan for 393 approved. Made CD and turned it in to Ron.

SCR # 376: Compile tpa.e with original szft.f and run with the baseline tpa.inp. Save results for comparison.

Ron confirmed that streamtube width is also used in dcagw.f (called awidth instead of sawidth). Will need to make mods to dcagw.f and revise the test plan. The following changes were made to dcagw.f:

dcagw.f: lines 162,167
< cc
< cc css - 4/11/2002; SCR #376 - add streamtube width multiplier

```
< cc
<      double precision widthmult
<      integer stwmindex
< cc
dcagw.f: lines
230c224
< cc
dcagw.f: lines 394,409
< cc
< cc css - 4/11/2002; SCR #376 - add streamtube width multiplier
< cc
<        call clearchar( 60, name )
<        name = 'StreamTubeWidthMultiplier[]'
<        stwmindex = ispquery(name)
<        widthmult = valuesp(stwmindex)
<        if (widthmult .le. 0) then
<          print *, ''
<          print *, ' ***>>> Error in dcagw <<<*** '
<          print *, ' Invalid value for widthmult:  ', widthmult,
<     &            '    (must be greater than 0.0)'
<          print *, ''
<          STOP
<        end if
< cc
dcagw.f: lines 426,430
< cc
< cc css - 4/11/2002; SCR #376 - add streamtube width multiplier
< cc
< cc           if (aifdist .le. dist) width = awidth
<              if (aifdist .le. dist) width = awidth * widthmult
```

Modified version of dcagw.f is dcagw376.f. Added print statements for testing and saved as

dcagw_test376.f.

Took out the "firsttimeonly" code in dcagw.f; doesn't seem to be necessary.

04/15/2002 – 04/19/2002 –

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Continued working on test plan for SCR # 376. Running w/ following parameters in tpa.inp:

NumberOfRealizations		5
StartAtRealization	1	
StopAtRealization	3	
StartAtSubarea		3
StopAtSubarea		8

Widthmix in szft.f also needs to be multiplied by widthmult.

Recompiled and reran tpa w/ new mods. Captured output file to show to Ron. He said results look reasonable. Wrote up code changes for SCR # 376. Worked on test plan. Added section for
general setup instructions. Functional test 1 will use the print statements in szft_test376.f and dcagw_test376.f to insure that the code mods are doing what they're supposed to do. For system level tests: 1) change StreamTubeWidthMultiplier from uniform to constant and set value to 1.0 (should be the same as base case); 2) set value to 0.5; 3) set value to 2.0.

Gave copy of captured display output to Ron for him to check. Ron wants abbreviation for
StreamTubeWidthMultiplier to be changed to STWidMul instead of default. Make change in tpanames.dbs and copy file down to data subdirectory.

04/22/2002 – 04/26/2002 –

Ran test plan. Peak Mean Dose for each test was:

basecase: Peak Mean Dose is 1.34670E-04 at 100000.0 yr.

STWidMul = 1.0: Peak Mean Dose is 1.34670E-04 at 100000.0 yr.

STWidMul = 0.5: Peak Mean Dose is 9.86291E-05 at 100000.0 yr.

STWidMul = 2.0: Peak Mean Dose is 1.50554E-04 at 100000.0 yr.

When STWidMul is set to uniform and allowed to vary the result was:

Peak Mean Dose is 1.04313E-02 at 84,700.0 yr.

Compared gwttuzsz.res files from the different test runs. Also showed results to J. Winterle. He wanted me to check nefii.inp and NEFII.VEL also. There were no differences in NEFII.VEL, but
nefiiinp differed in the migration path. One result of this check was that it was decided that we didn't need to make the widthmix modification; it was removed. Otherwise, results looked reasonable.

Since Ron did the initial analysis of the code and determined what mods needed to be made, it's OK for me to test it. Created directory scr376 under tpatest to store the results. Made subdirectories slt1, slt2, slt3, slt4, slt5 and flt1 under scr376 to save the results from each system and functional level test. Completed running the test plan (tp_scr376.wpd). All tests passed. Gave
completed plan to Ron. Completed text mods to scr376.wpd and tp_scr376 and turned them in to Ron.

04/29/2002 – 05/03/2002 –

Made the CD for SCR # 376 and turned it in to Ron. Next task is the test
plan for SCR #382. Roland Benke has made changes to the code (text version). Need to convert the files to unix format and add an error check before testing. Asked Roland for some reasonable values for the new sampled parameters. There will be twelve new parameters. Three existing ones
will need to be deleted. Mods need to be made to tpa.inp, dcags.f and dcagw.f. SCR # 382:

```
IA1   Divide mass loading and occupancy factors into inside and outside
      components

      Currently, only single values for the mass load and occupancy factor are
      used to determine the dose conversion factors for inhalation in the
      igneous case.  To accommodate the new definition of the lifestyle for
      the reasonably maximally exposed individual, five categories of
      disturbance parameters will be used, including one category for offsite
      time.  Each category will have a mass load and an occupancy time.  The
      doses from these five categories for inhalation will be summed and used
      in the current dose conversion factor expressions.  Even if the GENTPA
      code is used to generate the ground surface dose conversion factors for
      the other exposure pathways, the dose from the inhalation pathway will
      continue to be calculated by the TPA code and added to the results of
      the GENTPA code.
```

Made the following changes to tpa.inp:

```
commented out:

< **loguniform
< **AirborneMassLoadAboveFreshAshBlanket[g/m3]
< **1.2e-3, 1.6e-2


< **constant
< **OccupancyFactorForVolcanismDoseCalculation[-]
< **0.605

< **loguniform
< **AirborneMassLoadAboveSoil[g/m3]
< **5e-5, 3e-4

added:

tpa: lines 4783,4787d4782
< **          ***>>>  MASS LOADING AND OCCUPANCY FACTORS  <<<***
< ** SCR# 382:  delete AirborneMassLoadAboveFreshAshBlanket[g/m3],
< **            AirborneMassLoadAboveSoil[g/m3], and
< ** 5/3/02     OccupancyFactorForVolcanismDoseCalculation[-]
< **            add twelve new parameters below
4789,4836d4783
< constant
< AirborneMassLoadAboveFreshAshBlanketHeavyDisturbance[g/m3]
< 0.01
< **
< constant
< AirborneMassLoadAboveFreshAshBlanketLightDisturbance[g/m3]
< 0.001
< **
< constant
< AirborneMassLoadOutsideHeavyDisturbance[g/m3]
< 0.0001
< **
< constant
< AirborneMassLoadOutsideLightDisturbance[g/m3]
< 0.00001
< **
< constant
< AirborneMassLoadInsideHeavyDisturbance[g/m3]
< 0.000001
< **
< constant
< AirborneMassLoadInsideLightDisturbance[g/m3]
< 0.0000001
< **
< constant
```

```
< AirborneMassLoadOffsite[g/m3]
< 0.0
< **
< constant
< OccupancyFractionOutsideHeavyDisturbance[]
< 0.01
< **
< constant
< OccupancyFractionOutsideLightDisturbance[]
< 0.2
< **
< constant
< OccupancyFractionInsideHeavyDisturbance[]
< 0.24
< **
< constant
< OccupancyFractionInsideLightDisturbance[]
< 0.45
< **
< constant
< OccupancyFractionOffsite[]
< 0.1
< **
```

After Ron added modified code for 376 to the /dev directory, copied over to my area, tpatest/scr382.

Started test plan for SCR # 382. For testing purposes, needed to set flag in tpa.inp. Set VolcanismDisruptiveScenarioFlag to 1.

Compiled baseline version of tpa.e. Made subdirectory scr382/baseline. Using modified source code files, compiled a test version of tpa.e.

Files to compare: gs_cb_ad.dat, gs_cb_ci.dat, gs_pb_ad.dat, gs_pb_ci.dat, gw_cb_ad.dat, gw_cb_ci.dat, gw_pb_ad.dat, and gw_pb_ci.dat.

Ran into problem when running tpa.e. Forgot to change the path settings to /net/spock/home/
cscherer/tpatest/scr382. In .cshrc file, add:

setenv TPA_TEST /net/spock/home/cscherer/tpatest/scr382
setenv TPA_DATA /net/spock/home/cscherer/tpatest/scr382

Will need to remember to reset this each time I change SCRs.

Ran baseline version of code and saved data files to baseline subdirectory.

Modified dcagw.f and dcags.f and saved as dcagw_test382.f and dcags_test382.f. The following code changes were made:

```
      lve3 = .FALSE.
c
c     mass load values should be between 0.0 and 0.1, inclusive
c
      if ( (( massloadashheavy .lt. 0.0 ) .or.
     &      ( massloadashheavy .gt. 0.1 )      )    .OR.
     &     (( massloadashlight .lt. 0.0 ) .or.
     &      ( massloadashlight .gt. 0.1 )      )    .OR.
     &     (( massloadoutheavy .lt. 0.0 ) .or.
     &      ( massloadoutheavy .gt. 0.1 )      )    .OR.
     &     (( massloadoutlight .lt. 0.0 ) .or.
     &      ( massloadoutlight .gt. 0.1 )      )    .OR.
     &     (( massloadinheavy  .lt. 0.0 ) .or.
     &      ( massloadinheavy  .gt. 0.1 )      )    .OR.
     &     (( massloadinlight  .lt. 0.0 ) .or.
     &      ( massloadinlight  .gt. 0.1 )      )    .OR.
     &     (( massloadoffsite  .lt. 0.0 ) .or.
     &      ( massloadoffsite  .gt. 0.1 )      ) )  THEN
         lve1 = .TRUE.
      end if
c        1         2         3         4         5         6         7
c23456789012345678901234567890123456789012345678901234567890123456789012
c
c     occupancy values should be between 0.0 and 1.0, inclusive
c
      if ( (( occupancyoutheavy .lt. 0.0 ) .or.
     &      ( occupancyoutheavy .gt. 1.0 )      )    .OR.
     &     (( occupancyoutlight .lt. 0.0 ) .or.
     &      ( occupancyoutlight .gt. 1.0 )      )    .OR.
     &     (( occupancyinheavy  .lt. 0.0 ) .or.
     &      ( occupancyinheavy  .gt. 1.0 )      )    .OR.
     &     (( occupancyinlight  .lt. 0.0 ) .or.
     &      ( occupancyinlight  .gt. 1.0 )      )    .OR.
     &     (( occupancyoffsite  .lt. 0.0 ) .or.
     &      ( occupancyoffsite  .gt. 1.0 )      ) )  THEN
         lve2 = .TRUE.
      end if
c
c     occupancy values should add up to 1.0
c
      occupancytotal = occupancyoutheavy + occupancyoutlight
```

```
     &                + occupancyinheavy  + occupancyinlight
     &                + occupancyoffsite
      if ( occupancytotal .ne. 1.0) THEN
         lve3 = .TRUE.
      end if

c
c     if an error occurs, print out an error message and the mass load
c     and occupancy values; then ABORT processing
c
      if ((lve1) .or. (lve2) .or. (lve3)) then
         print *, ' '
         print *, ' ***>>> Error in DCAGS <<<*** '
         if (lve1) print *,
     & ' massload value out of range - must be between 0.0 and 0.1 '
         if (lve2) print *,
     & ' occupancy value out of range - must be between 0.0 and 1.0 '
         if (lve3) print *, ' occupancy values do not add up to 1.0 '
         print *, ' '
         print *,
     & 'AirborneMassLoadAboveFreshAshBlanketHeavyDisturbance[g/m3] = ',
     &            massloadashheavy
         print *,
     & 'AirborneMassLoadAboveFreshAshBlanketLightDisturbance[g/m3] = ',
     &            massloadashlight
         print *,
     & 'AirborneMassLoadOutsideHeavyDisturbance[g/m3]              = ',
     &            massloadoutheavy
         print *,
     & 'AirborneMassLoadOutsideLightDisturbance[g/m3]              = ',
     &            massloadoutlight
         print *,
     & 'AirborneMassLoadInsideHeavyDisturbance[g/m3]               = ',
     &            massloadinheavy
         print *,
     & 'AirborneMassLoadInsideLightDisturbance[g/m3]               = ',
     &            massloadinlight
         print *,
     & 'AirborneMassLoadOffsite[g/m3]                              = ',
     &            massloadoffsite
         print *,
     & 'OccupancyFractionOutsideHeavyDisturbance[]                 = ',
     &            occupancyoutheavy
         print *,
     & 'OccupancyFractionOutsideLightDisturbance[]                 = ',
     &            occupancyoutlight
         print *,
     & 'OccupancyFractionInsideHeavyDisturbance[]                  = ',
     &            occupancyinheavy
         print *,
     & 'OccupancyFractionInsideLightDisturbance[]                  = ',
     &            occupancyinlight
```

```
         print *,
     & 'OccupancyFractionOffsite[]                                  = ',
     &            occupancyoffsite
         print *, ' '
         print *,
     & 'OCCUPANCYTOTAL                                              = ',
     &            occupancytotal
         print *, ' '
         STOP
      end if
c        1         2         3         4         5         6         7
c23456789012345678901234567890123456789012345678901234567890123456789012
c
c    end of subroutine checkvalues
c
      return
      end
```

2. Changes to DCAGW:

```
2105      doss  = valuesp( i_doss )
c
c css - 5/2/02 - add range check to rmlf for SCR# 382
c       0.0 .le. AirborneMassLoadOutsideLightDisturbance[g/m3] .le. 0.1
c
      rmlf  = valuesp( i_rmlf  )
      if ((rmlf .lt. 0.0) .OR. (rmlf .gt. 0.1)) then
           print *, ' '
           print *, ' ***>>> Error in Gentpa <<<*** '
           print *,
     & ' value for AirborneMassLoadOutsideLightDisturbance[g/m3] ',
     & ' out of range: ', rmlf
           print *, '  (must be greater than or equal to 0.0 AND ',
     & ' less than or equal to 0.1) '
           print *, ' '
           STOP
      end if
2122      iyipip= ivaluesp( i_yipip )
```

Recompiled tpa.e w/ these changes. Save as tpa_test382.e. Run and save files to compare with baseline case. Work on test plan: tp_scr382.wpd. There were no differences in the following files:

gs_cb_ad.dat gs_cb_ci.dat gs_pb_ad.dat gs_pb_ci.dat
gw_cb_ci.dat gw_pb_ci.dat

However, there were changes in the ***inh*** column in gw_cb_ad.dat and gw_pb_ad.dat. Gave a listing of differences to Ron to see if he thinks they are reasonable.

Created the following system level tests for the plan:

1 - Airborne out of range (high).
2 - Occupancy out of range (low).
3 - Occupancy fractions do not add up to 1.
4 - good run; compare test case w/ base case
5 - change parameter values

05/06/2002 – 05/10/2002 –

Ron wanted abbreviations changed. Modified tpanames.dbs and copied to data subdirectory.

More tests for test plan. Roland set up values for the new parameters. Requires two runs for 5 pairs of values. The details are in tp_scr382.wpd. Tests are:

SLT1: Test AirborneMassLoad value out of bounds (high) in dcagw.f.
SLT2: Test AirborneMassLoad value out of bounds (high) in dcags.f. SLT3: Test OccupancyFraction out of bounds (low).
SLT4: Test OccupancyFraction total not equal to 1.0.
SLT5: Test AirborneMassLoad value at lower limit.
SLT6: Test OccupancyFraction value at upper limit.
SLT7: Test effects of modified code compared to unmodified code.
SLT8: Test new parameters (heavy and outside).
SLT9: Test new parameters (light and outside).
SLT10: Test new parameters (heavy and inside).
SLT11: Test new parameters (light and inside).
SLT12: Test new parameters (off site).

Ran modified code w/ Roland's test. The values did not come out exactly double where he
expected. Ran a test w/ all Occupancy Fractions set to 0.0. This would show if other code is influencing the results. This factor can then be subtracted out from the test runs to see if this result
is double. Results put in subdirectory zeroOF. Also look at airpkdos result files.

05/13/2002 – 05/17/2002 –

Moved to A123 to work. Set up computers. Ran test plan for SCR # 382.

Check results using calculator. Comparing files airpkdos, gw_cb_ad.dat, and gw_pb_ad.dat. Results still not ***exactly*** a twice or half as much, but could just be a precision problem. Checked with Ron. Results have been determined to be close enough.

05/20/2002 – 05/24/2002 –

The floppy where I had stored tp_scr382.wpd is corrupt. I cannot retrieve the file. I found an earlier version I had saved from 5/8 and used my hard copy to update that version. Finished modifying the test plan and turned it in to Ron along with the updated SCR and
a CD of the scr382 subdirectory.

Started SCR # 373:

```
FT3   Reflect uncertainty in CHnv thickness

      Currently, the thickness of the nonwelded vitric CHnv unsaturated zone
      layer below the repository is a fixed value specified in the tpa.inp
      file.  The proposed change will make the CHnv thickness a sampled
      parameter that reflects the mineralogic and lithologic thickness
      uncertainty.  With the addition of an overall unsaturated zone thickness
      parameter, the UZFT module will be modified to adjust the CHnz layer
      thickness to compensate for the variable CHnv thickness to maintain the
      desired overall unsaturated zone thickness for each subarea.
```

Mods required to uzft.f, szft.f and tpa.inp. Need to add 10 new parameters for total thickness of CH layers. Need to delete 10 existing parameters for thickness of CHnz which will now be
calculated. Change ChnvThickness parameters from constant to uniform.

Compiled and ran baseline version of TPA. Made baseline subdirectory to store results. Need to look at nefiiuz.inp.

Made the following changes in the code and input files:

```
SZFT.F

In the base case version of szft.f, in the subroutine getszunits, you will
see, beginning at line 1733:

        do j=1,numsalayers
```

```
         if (isa.lt.10) then
            spname=salayernam(j)(1:4)//'Thickness_'//
     &                    ctemp2(2:2)//'SubArea[m]'
            aleglen(j) = valuesp(ispquery( spname ))
         else if (isa.lt.100) then
            spname=salayernam(j)(1:4)//'Thickness_'//
     &                    ctemp2//'SubArea[m]'
            aleglen(j) = valuesp(ispquery( spname ))
         end if
      end do
```

Insert the following code after line 1732 and comment out the original lines 1733 - 1743:

```
c
c css - 5/22/2002:  SCR# 373 - read in total CH thickness and CHnv thickness;
c       calculate CHnz thickness; j=2 for CHnv and j=3 for CHtotal
c
      do j=1,numsalayers
         if (isa.lt.10) then
            if (j .eq. 3) then
               spname=salayernam(j)(1:2)//'_Total_Thickness_'//
     &                ctemp2(2:2)//'SubArea[m]'
            else
               spname=salayernam(j)(1:4)//'Thickness_'//
     &                ctemp2(2:2)//'SubArea[m]'
            end if
         else if (isa.lt.100) then
            if (j .eq. 3) then
               spname=salayernam(j)(1:2)//'_Total_Thickness_'//
     &                ctemp2//'SubArea[m]'
            else
               spname=salayernam(j)(1:4)//'Thickness_'//
     &                ctemp2//'SubArea[m]'
            end if
         end if
         if (j .eq. 3) then
            chtotthick = valuesp(ispquery( spname ))
            aleglen(3) = chtotthick - aleglen(2)
         else
            aleglen(j) = valuesp(ispquery( spname ))
         end if
c
c error check values
c
         if ((j .eq. 3) .AND. (chtotthick .lt. aleglen(2))) then
            print *, ' '
            print *, ' ***>>> Error in szft <<<*** '
            print *, 'CH Total Thickness value ', chtotthick,
     &                ' < CHnv value: ', aleglen(2)
            print *, ' '
            STOP
```

```
         end if
         if (aleglen(j) .lt. 0.0) then
            print *, ' '
            print *, ' ***>>> Error in szft <<<*** '
            print *, ' invalid negative value for ', spname,
     &                ':  ', aleglen(j)
            print *, ' '
            STOP
         end if
      end do
c
c css - 5/22/2002: comment out pre-SCR# 373 version
c
ccc   do j=1,numsalayers
ccc      if (isa.lt.10) then
ccc         spname=salayernam(j)(1:4)//'Thickness_'//
ccc  &                    ctemp2(2:2)//'SubArea[m]'
ccc         aleglen(j) = valuesp(ispquery( spname ))
ccc      else if (isa.lt.100) then
ccc         spname=salayernam(j)(1:4)//'Thickness_'//
ccc  &                    ctemp2//'SubArea[m]'
ccc         aleglen(j) = valuesp(ispquery( spname ))
ccc      end if
ccc   end do
```

Add the following to the declaration section of getszunits:

```
cc
cc css - 5/23/2002; SCR# 373:  add temporary CH total thickness variable
cc
      real*8 chtotthick
```

UZFT.F

In the base case version of szft.f, in the subroutine prenefmks, you will see, beginning at line 1088:

```
       do j=1,nlyers
         if (isa.lt.10) then
         spname=salayernam(j)(1:4)//'Thickness_'//
     &            ctemp2(2:2)//'SubArea[m]'
                   if(debug) print *, 'spname=',spname
           leglen(np,j) = valuesp(ispquery( spname ))
              if (debug) print *, 'leglen(np,j)=',leglen(np,j)
         else if (isa.lt.100) then
           spname=salayernam(j)(1:4)//'Thickness_'//
     &            ctemp2//'SubArea[m]'
                   if(debug) print *, 'spname=',spname
           leglen(np,j) = valuesp(ispquery( spname ))
              if (debug) print *, 'leglen(np,j)=',leglen(np,j)
         end if

cc rwj 8-23-00
```

```
cc      Remove old importance analysis method.
cc        leglen(np,j) = aiafilter(spname, leglen(np,j))
cc
        end do
```

Insert the following code after line 1087 and comment out the original lines 1088 - 1108:

```
c
c css - 5/22/2002:  SCR#373: read in total thickness for CH layers instead of
c        CHnz layer; since CHnv can now vary, calculate CHnz by subtracting
CHnv
c        from CHtotal (j = 2 for CHnv and j = 3 for CHtotal
c
        do j=1,nlyers

           if (isa.lt.10) then
              if (j .eq. 3) then
                 spname=salayernam(j)(1:2)//'_Total_Thickness_'//
     &                 ctemp2(2:2)//'SubArea[m]'
              else
                 spname=salayernam(j)(1:4)//'Thickness_'//
     &                  ctemp2(2:2)//'SubArea[m]'
              end if
           else if (isa.lt.100) then
              if (j .eq. 3) then
                 spname=salayernam(j)(1:2)//'_Total_Thickness_'//
     &                 ctemp2//'SubArea[m]'
              else
                 spname=salayernam(j)(1:4)//'Thickness_'//
     &                 ctemp2//'SubArea[m]'
              end if
           end if
        if (debug) print *, 'spname=',spname
c
c calculate CHnz based on values for CHtotal - CHnv
c
           if (j .eq. 3) then
              chtotthick = valuesp(ispquery( spname ))
              leglen(np,3) = chtotthick - leglen(np,2)
        if(debug) print *, ' - CHnv = CHnz = '
           else
              leglen(np,j) = valuesp(ispquery( spname ))
           end if
        if (debug) print *, 'leglen(np,j)=',leglen(np,j)
c
c error check values
c
           if ((j .eq. 3) .AND. (chtotthick .lt. leglen(np,2)))  then
              print *, ' '
              print *, ' ***>>> Error in uzft <<<*** '
              print *, 'CH Total Thickness value ', chtotthick,
```

```
     &                ' < CHnv value: ', leglen(np,2)
            print *, ' '
            STOP
          end if
          if (leglen(np,j) .lt. 0.0) then
            print *, ' '
            print *, ' ***>>> Error in uzft <<<*** '
            print *, ' invalid negative value for ', spname,
     &                ':  ', leglen(np,j)
            print *, ' '
            STOP
          end if
c
c css - 5/22/2002
c       comment out pre-SCR# 373 version
ccc      if (isa.lt.10) then
ccc      spname=salayernam(j)(1:4)//'Thickness_'//
ccc  &            ctemp2(2:2)//'SubArea[m]'
ccc                if(debug) print *, 'spname=',spname
ccc        leglen(np,j) = valuesp(ispquery( spname ))
ccc           if (debug) print *, 'leglen(np,j)=',leglen(np,j)
ccc      else if (isa.lt.100) then
ccc        spname=salayernam(j)(1:4)//'Thickness_'//
ccc  &            ctemp2//'SubArea[m]'
ccc                if(debug) print *, 'spname=',spname
ccc        leglen(np,j) = valuesp(ispquery( spname ))
ccc           if (debug) print *, 'leglen(np,j)=',leglen(np,j)
ccc      end if

cc rwj 8-23-00
cc     Remove old importance analysis method.
cc       leglen(np,j) = aiafilter(spname, leglen(np,j))
cc
       end do
```

Add the following to the declaration section of prenefmks:

```
c css - 5/22/2002:  SCR# 373 - declare temporary logical variable
c                              declare temporary CH total thickness
c                                 variable
c
      logical ltemp
      real*8  chtotthick
```

TPA.INP

In base case version of tpa.inp, you will see the following at the end of the UZFT section:

```
**
**          ***>>> UZFT  <<<***          CHANGE TO:
**
.
.
.
      constant                           uniform
      CHnvThickness_1SubArea[m]          CHnvThickness_1SubArea[m]
      19.0                               0.0, 19.0
      **                                 **
      constant                           constant
      CHnzThickness_1SubArea[m]          CH_Total_Thickness_1SubArea[m]
      72.0                               91.0
      .                                  .
      .                                  .
      .                                  .
      constant                           uniform
      CHnvThickness_2SubArea[m]          CHnvThickness_2SubArea[m]
      0.01                               0.0, 1.0
      **                                 **
      constant                           constant
      CHnzThickness_2SubArea[m]          CH_Total_Thickness_2SubArea[m]
      108.0                              108.0
      .                                  .
      .                                  .
      .                                  .
      constant                           uniform
      CHnvThickness_3SubArea[m]          CHnvThickness_3SubArea[m]
      24.0                               0.0, 24.0
      **                                 **
      constant                           constant
      CHnzThickness_3SubArea[m]          CH_Total_Thickness_3SubArea[m]
      55.0                               79
      .                                  .
      .                                  .
      .                                  .
      constant                           uniform
      CHnvThickness_4SubArea[m]          CHnvThickness_4SubArea[m]
      17.0                               0.0, 17.0
      **                                 **
      constant                           constant
      CHnzThickness_4SubArea[m]          CH_Total_Thickness_4SubArea[m]
      88.0                               105.0
      .                                  .
      .                                  .
      .                                  .
      **
      constant                           uniform
      CHnvThickness_5SubArea[m]          CHnvThickness_5SubArea[m]
      31.0                               0.0, 31.0
      **                                 **
      constant                           constant
```

```
CHnzThickness_5SubArea[m]
49.0
.
.
.
constant
CHnvThickness_6SubArea[m]
37.0
**
constant
CHnzThickness_6SubArea[m]
58.0
.
.
.
constant
CHnvThickness_7SubArea[m]
44.0
**
constant
CHnzThickness_7SubArea[m]
63.0
.
.
.
constant
CHnvThickness_8SubArea[m]
0.01
**
constant
CHnzThickness_8SubArea[m]
120.0
.
.
.
constant
CHnvThickness_9SubArea[m]
1.0
**
constant
HnzThickness_9SubArea[m]
128.0
.
.
.
constant
CHnvThickness_10SubArea[m]
0.01
**
constant
CHnzThickness_10SubArea[m]
```

```
CH_Total_Thickness_5SubArea[m]
80.0
.
.
.
uniform
CHnvThickness_6SubArea[m]
0.0, 37.0
**
constant
CH_Total_Thickness_6SubArea[m]
95.0
.
.
.
uniform
CHnvThickness_7SubArea[m]
0.0, 44.0
**
constant
CH_Total_Thickness_7SubArea[m]
107.0
.
.
.
uniform
CHnvThickness_8SubArea[m]
0.0, 1.0
**
constant
CH_Total_Thickness_8SubArea[m]
120.0
.
.
.
uniform
CHnvThickness_9SubArea[m]
0.0, 1.0
**
constant
CH_Total_Thickness_9SubArea[m]
128.0
.
.
.
uniform
CHnvThickness_10SubArea[m]
0.0, 1.0
**
constant
CH_Total_Thickness_10SubArea[m
]
```

```
137.0                                   137.0
**
**            ***>>>  SZFT  <<<***
**
```

```
CHnv parameters will change from constant to uniform with a range from
0.0 to the current constant value (or from 0.0 to 1.0 for those subareas
currently set to the constant 0.0). The sampled parameters
CHnzThickness_xSubArea[m] will be replaced by new parameters,
CH_Total_Thickness_xSubArea[m].  The values assigned to them will change to
the total thickness of the Calico Hills layers.  The thickness for CHnz will
be calculated at runtime and will depend upon the sampled value selected for
CHnv.
```

Modified code saved in uzft_mod.f, szft_mod.f, and tpa_test373.inp. Add print statements to modified FORTRAN code and save as uzft_test373.f and szft_test373.f. Compiled modified
version of tpa.e.

Ran a series of tests with modified code using a modified version of tpa.inp where the new
parameters were set to 700.0 and changed from uniform to constant. This is intended to exaggerate
the differences in the test results.

Ron approved the default abbreviations for the new parameters.

Modified SCR # 373 and started tp_scr373.wpd. Modified SCR sent to Ron.

Ron requested that I run a test of the new code on each of the 10 subareas individually and
compare test results with baseline results. Stored results in subdirectories sa1 through sa10. Saved
nefiiuz.inp and output file with captured screen results.

Functional test results passed. The code is performing exactly as expected. Ron would like to see some instances where more than 2 legs are selected in nefiiuz.inp. Need to run a series of tests using various values for one or more new parameters. Ended up concentrating on CHnv where a small value is sufficient to force addition of leg.

Tests included in test plan:

1. FLT1: Name: Modified Code Functionality Test
2. SLT1: Name: Test Negative Thickness Value in uzft.f.
3. Name: Test Total CH Thickness Value Less Than CHnv Value in uzft.f.
4. Name: Test Negative Thickness Value in szft.f.
5. Name: Test Total CH Thickness Value Less Than CHnv Value in szft.f.
6. Name: Modified Code Does Not Impact Other Processing Test
7. Name: Test SubArea 1
8. Name: Test SubArea 2
9. Name: Test SubArea 3
10. Name: Test SubArea 4
11. Name: Test SubArea 5
12. Name: Test SubArea 6
13. Name: Test SubArea 7
14. Name: Test SubArea 8
15. Name: Test SubArea 9
16. Name: Test SubArea 10
17. Name: Test Variation 1 on SubArea 2 - Adjust value of CHnv and CHnz so that both CHnz and TSw legs are included.
18. Name: Test Variation 2 on SubArea 2 - Adjust value of CHnv and CHnz so that both CHnv and TSw legs are included.
19. Name: Test Variation 3 on SubArea 2 - Adjust value of CHnv and CHnz so that CHnv, CHnz and TSw legs are included.

05/27/2002 – 05/31/2002 –

Worked on tp_scr373.wpd. Experiment with values until 4 legs are included.

For the new uniform sampled parameters where the old value was 0.0. Set the new values to 0.0, 1.0.

Finished mods to the test plan and gave to Ron. Returned with markups. Made final mods to SCR and test plan. Turned in to Ron. This test plan will have to be run (and modified) by someone else since I did the code analysis and mods.

06/03/2002 – 06/07/2002 –

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Made CD for SCR # 373 and turned in to Ron. Started work on SCR # 375. Copied latest version (4.2 d) of code from /dev into tpatest/scr375 directory.

SCR # 375:

```
FT5   Variable dispersivity for UZFT

      Currently, the dispersivity in the unsaturated zone is limited to a
      minimum of 0.1 m [0.33 ft] regardless of the layer thickness.  To
      provide a more realistic parameter value for the unsaturated zone
      transport module, this limit will be lowered to 0.01 m [0.033 ft] for
      layer thicknesses less than 40 m [131.2 ft].
```

Resolving this SCR will require mods to uzft.f, szft.f and tpa.inp. Compile and run base line version of tpa.e. Save results.

Made following changes to files:

```
UZFT.F

In the base case version of uzft.f, in the subroutine prenefmks, you will see,
beginning at line 1472:

C         Calculate dispersion.

            IF (media .EQ. 1) THEN
               disper(INC) = DMAX1(0.1d0, LEGLEN(NP,LAYUNS) * disperm)
            ELSE
               disper(INC) = DMAX1(0.1d0, LEGLEN(NP,LAYUNS) * disperf)
            END IF

Comment out the original lines 1474 - 1478 and insert/modify the following
lines:

C         Calculate dispersion.
c
c css - 6/3/02 - SCR# 375: dispersivity for layers less than 40 m should be
c                          changed to 0.01 instead of 0.1; dispersivity
c                          remains at .1 for layers > 40 m
cc            IF (media .EQ. 1) THEN
cc               disper(INC) = DMAX1(0.1d0, LEGLEN(NP,LAYUNS) * disperm)
cc            ELSE
cc               disper(INC) = DMAX1(0.1d0, LEGLEN(NP,LAYUNS) * disperf)
cc            END IF
            IF (LEGLEN(NP, LAYUNS) .GE. 40.0) THEN
               tempdisp = 0.1d0
            ELSE
```

```
            tempdisp = 0.01d0
         END IF
         IF (media .EQ. 1) THEN
            disper(INC) =
     &         DMAX1(tempdisp, LEGLEN(NP,LAYUNS) * disperm)
         ELSE
            disper(INC) =
     &         DMAX1(tempdisp, LEGLEN(NP,LAYUNS) * disperf)
         END IF
```

In the base case version of uzft.f, in the subroutine prenefmks, you will see, beginning at line 1524:

```
cc - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
cc    Point of early return.
cc - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

      REPDISPER = DMAX1(0.1d0, REPLEG * disperf)
```

Comment out the original line 1528 and insert/modify the following lines:

```
cc - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
cc    Point of early return.
cc - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

c
c css - 6/3/02 - SCR# 375: dispersivity for layers less than 40 m should be
c                          changed to 0.01 instead of 0.1; repository leg
c                          is set to one meter
cc      REPDISPER = DMAX1(0.1d0, REPLEG * disperf)
      REPDISPER = DMAX1(0.01d0, REPLEG * disperf)
```

Stored mods in uzft_mod375.f. Added print statements for testing purposes and saved as uzft_test375.f

Began test plan. Plan to look at various .res files for differences between test case and base case.

Came up with the following tests:

FLT1: Name: Modified Code Functionality Test
SLT1: Name: Test Effects of Modified Code
SLT2: Name: Test Single Run With Leg > 40 Meters and Leg < 40 Meters
SLT4: Name: Test Varying Values For "disperm" and "disperf".

Finished test plan, printed it out and left for Ron to see.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

06/10/2002 – 06/14/2002 –

Turned in signed COI form.

Made final mods for tp_scr375.wpd and turned in to Ron. Made CD for 375 and turned in.
Someone else will have to execute and modify test plan; I did the code.

Start on SCR # 374. This involves the interface to NEFTRAN. Checked out copy of NEFTRAN user manual from library.

SCR # 374:

```
FT4   Multiple fracture flow and matrix flow epochs

      The UZFT module currently restricts the flow media of a given layer to
      matrix or fracture for the entire simulation time.  It is desired to
      allow the flow to share or switch media types one or more times during a
      simulation.  This condition can be approximated by using two legs for
      one hydrostratigraphic unit in the NEFTRAN input data set.  This feature
      will be added to the UZFT module to increase the flexibility of the
      code.
```

Read through NEFTRAN manual for better understanding of what uzft.f code is doing.

SCR somewhat ambiguous. Several options possible. Consulted w/ Ron about changes needed. He called Tim McCartin for further clarification. For this SCR, need to add new variables to store fracture fraction and matrix fraction of source term. Calculate matrix fraction by dividing cond by infil for each layer at each time step. Select lowest matrix fraction of all layers at each time step. The maximum limit of 1.0 for matrix fraction is enforced. Subtract lowest matrix fraction from 1.0 to obtain fracture fraction for all layers. Multiply value of ciperyrinsaintoloweruz by fracture fraction and store in new array ciperyrinsaintouzfracture. Multiply value of ciperyrinsaintoloweruz by lowest matrix fraction and store in new array ciperyrinsaintouzmatrix. Send ciperyrinsaintouzmatrix to NEFTRAN. After NEFTRAN runs, results from ciperyrinsaintouzmatrix have been placed into ciperyrinsafromuz. Add ciperyrinsaintouzfracture back into ciperyrinsafromuz. .

Need to look at nefiiuz.inp and NEFII.VEL for this SCR.

Helped Brandi Winfrey come up to speed on SCR #373. She will be doing the official testing.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

06/17/2002 – 06/21/2002 –

Modified main uzft.f routine and prenefmks subroutine in uzft.f. The following code changes were made:

```
After the line:

      dimension ciperyrinsafromuz( mxntime, nnucl )

      Add:

c
c css - 6/17/02 - SCR #374: declare array returned from
c       prenefmks that contains the fracture portion of
c       the source flow
c
      dimension ciperyrinsaintouzfracture( mxntime, nnucl )
c css

Modify:

      call prenefmks(isa, mxntime, ntim_c, ntim, tim, nnucl, names,
     &               ciperyrinsaintoloweruz, qm3peryrinsa,
     &               tavgwtt, neflegs)

      To:

c
c css - 6/17/02 - SCR #374:  add ciperyrinsaintouzfracture argument
c       as an output argument from prenefmks
c
      call prenefmks(isa, mxntime, ntim_c, ntim, tim, nnucl, names,
     &               ciperyrinsaintoloweruz, qm3peryrinsa,
     &               tavgwtt, neflegs, ciperyrinsaintouzfracture)
c css

After the line:

      call aftnefmks(mxntime, ntim, tim, nnucl, names,
     &               ciperyrinsafromuz)

      Add:

c
c css - 6/17/02 - SCR#374:
c       add fracture portion, which is stored in
C       ciperyrinsaintouzfracture, back into ciperyrinsafromuz,
c       which contains ciperyrinsaintouzmatrix after it has
c       been modified by NEFTRAN
```

```
c
      do i = 1, ntim
         do j = 1, nnucl
            ciperyrinsafromuz( i,j ) = ciperyrinsafromuz( i,j ) +
     &          ciperyrinsaintouzfracture( i,j )
         end do
      end do

c css
```

Add to the header block of subroutine prenefmks:

```
c ciperyrinsaintouzfracture(maxntime,nnucl) =  output, double precision,
c               fracture portion of ciperyrinsaintoloweruz; css - SCR #374
                  .
                  .
                  .
cc        6-17-02 C. Scherer   (for SCR #374)
cc                Add ciperyrinsaintouzfracture to argument list.
```

After the line:

```
      dimension ciperyrinsaintoloweruz( mxntime, nnucl )
```

Add:

```
c
c css - 6/17/02 - SCR #374: declare two new arrays to store fracture
c       and matrix portions of ciperyrinsaintoloweruz; dimension new
c       array to store infiltration values per tpa time steps; add
c       new variables for fracture fraction, matrix fraction and
c       lowest matrix fraction (over all layers for time step)
c
      dimension ciperyrinsaintouzfracture( mxntime, nnucl )
      dimension ciperyrinsaintouzmatrix( mxntime, nnucl )
      dimension tpatime_infil(mxntime)
      double precision fracturefraction
      double precision matrixfraction
      double precision lowestmatfraction
c css
```

After the block:

```
      do ilayer = 1, NLYERU
        lmedia( ilayer ) = 1

        do itime = 1, ntim
           if ( infil(itime) .gt. cond(ilayer, 1 )) then
               lmedia( ilayer ) = 2
           end if
        end do
```

```
      end do

      Add:

c
c css - 6/17/02 - SCR #374:  separate ciperyrinsaintoloweruz
c       into an array for the matrix portion and an array for
c       the fracture portion of the source flow
c
c ntim passed into prenefmks from uzft and is called
c ntimg in prenefmks; ntim is calculated in prenefmks
c infil only has ntim values while ciperyr arrays have
c ntimg values so create a version of infil over all
c the original tpa time steps
c
       call maplist( ntim, tim(1), infil(1), ntimg, timg(1),
     &               tpatime_infil(1) )

       lowestmatfraction = 1.0d0

       do itime = 1, ntimg

c find lowest matrix fraction for layer in time step

           do ilayer = 1, NLYERU
c
c check that conductivity is positive
c
              if (cond(ilayer,1) .LT. 0) then
                 print *, ' '
                 print *, ' ***>>> Error in uzft: prenefmks <<<*** '
                 print *, ' negative value input for permeability',
     &                     perm(ilayer,1)
                 print *, ' in time step ', itime, ' and layer ', ilayer
                 print *, ' '
                 STOP
              end if
              matrixfraction = cond(ilayer,1) / tpatime_infil(itime)
              lowestmatfraction =
     &           DMIN1(lowestmatfraction, matrixfraction)
           end do
           fracturefraction = 1.0d0 - lowestmatfraction

c separate ciperyrinsaintoloweruz into fracture part and matrix part

           do inuc = 1, nnucl
              ciperyrinsaintouzfracture(itime, inuc) = fracturefraction
     &             * ciperyrinsaintoloweruz(itime, inuc)
              ciperyrinsaintouzmatrix(itime, inuc) = lowestmatfraction
     &             * ciperyrinsaintoloweruz(itime, inuc)
           end do
```

```
       end do
c
c css

After:

  499 CONTINUE

      CLOSE (UNIT=i3)

      Add:

c
c css - 6/17/02 - SCR #374:  call writesource using matrix only
c       array instead of ciperyrinsaintoloweruz which also contains
c       fracture portion
c       WARNING:  DO NOT MODIFY WRITESOURCE - IT IS ALSO CALLED
c                 FROM prenefmksa in szft.f
c

      Modify:

cc      call writesource(mxntime, ntimg, timg, nnucl, name,
cc     &                 names, ciperyrinsaintoloweruz)

      To:

      call writesource(mxntime, ntimg, timg, nnucl, name,
     &                 names, ciperyrinsaintouzmatrix)
c css
```

CAVEATS: Don't change writesource because it is also called from szft.f. Don't change the name of the parameter ciperyrinsaintoloweruz. This variable is also used in szft.f.

Discovered that ntim in prenefmks is not the same as the ntim that is passed into prenefmks. ntim becomes ntimg. This caused confusion during mods. Different arrays are dimensioned using different numbers of time steps. Use maplist to create the right size array of infil data.

Answered more questions from Brandi about SCR # 373.

Decided on the following tests for the test plan:

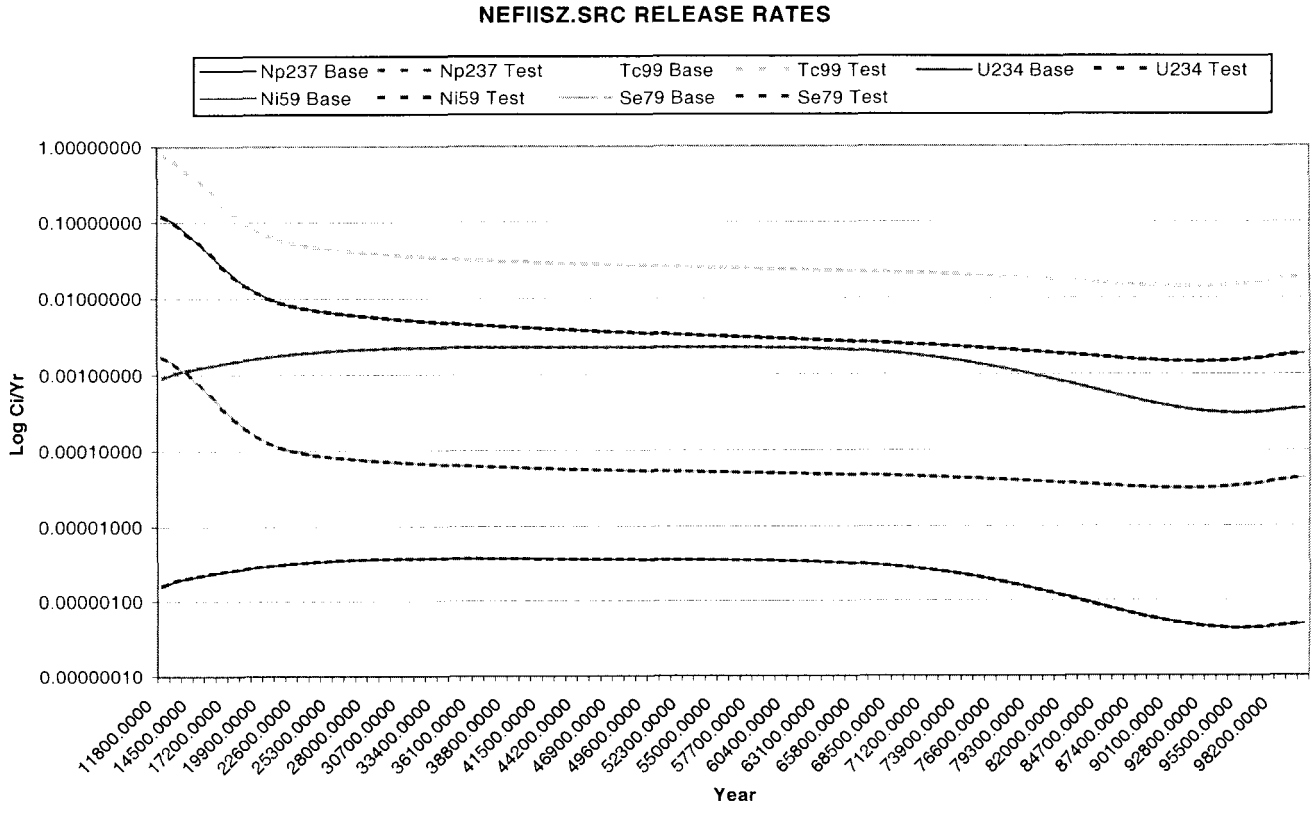
FLT1: Name: Modified Code Functionality Test
SLT1: Name: Test Effects of Modified Code
SLT2.: Name: Test Permeability Error Code

SLT3: Name: Test Matrix Fraction = 1.0
SLT4: Name: Test Interaction of DripShieldThickness[m], DefectiveFractionOfWPs/cell and Matrix Fraction

Added print statements to uzft.f for the functional level test.

06/24/2002 – 06/28/2002 –

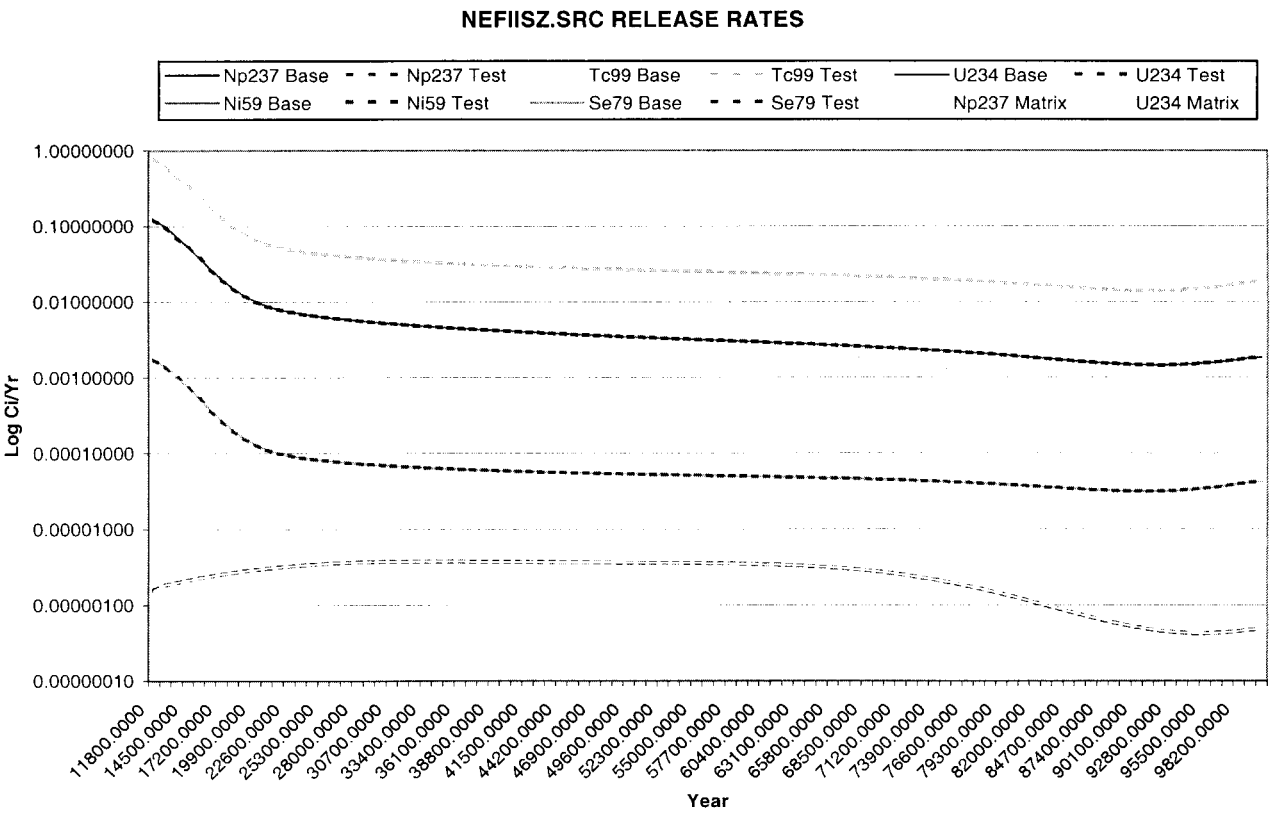
Start test runs. Make subdirectories flt1, slt1, slt2, slt3. Under slt1 make directories



testres, compres, baseres to store results for different cases.

Results from test runs somewhat confusing. Trend appears to be that release rates go up at the same or at an earlier time. This seems reasonable. But you must look at each nuclide individually to see the trend. Different nuclides behave differently, due to half-life? layer

conductivity?, retardation rates? or other unspecified characteristics. Means, averages, sums and



annual doses seem to mask the trend.

Picked Tc99, Ni59, Np237, Se79, and U234 to concentrate on. Ran 100000 years instead of
10000 years.

Graphed results using Excel. There are no differences in nefiisz.src between base and test cases, but there are significant results in nefiiuz.src files.

Graphed more results, which show that the code seems to be working OK.

07/01/2002 – 07/05/2002 –

Begin working three-quarter time through December.

Question about C14. In uzft.f, values for C14 are zeroed out. Ron said he had heard from Sitakanta that nonzero values still showed up for C14 after uzft processing. Ron asked me to look for this, but I could not find any examples of this. I asked Sitakanta about it, but he said that wasn't
the problem. He thinks that we should not zero out values of C14 in uzft even though it is a gas. Talked to Ron again; he will look into this.

Finished test plan for SCR # 374 and turned in to Ron.

07/08/2002 – 07/12/2002 -

Start looking at SCR # 377.

Remade CDs for SCR #373 and #375. I wrote them at 16x but they need to be at 12x speed.

Made final mods to tp_scr374.wpd per Ron's markups. Made CD and gave to Ron.

Remade CDs for SCR # 375, # 373, & # 382. I made them originally using the default speed of 16x. Ron couldn't read them. Need to use 12x speed. Also made and printed out directory listings
for each CD so Ron can keep them in the SCR files.

SCR # 377:

```
FT7   Update streamtube flux after climate change

      Currently, the streamtube flux is implemented as a constant value read
      from the strmtube.dat file.  The TPA code should permit the flux to
      change in response to climatic conditions.  The change to climatic
      conditions relative to present day conditions is currently available in
      the UZFLOW file called climato2.dat.  Normally, the saturated zone
      transport code considers a fixed porosity and fixed velocity for the
      transport legs. To consider time dependent flow, however, a velocity
      file will be generated to reflect the time dependent flux change at the
      assigned porosity.  The degree of this effect will be controlled by an
      input factor which will control the scaling of the climatic deviations
      relative to present day conditions as applied to the flux rates.  The
      present day flux rates will continue to be specified in the strmtube.dat
      file.  This change will be made with time permitting.
```

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

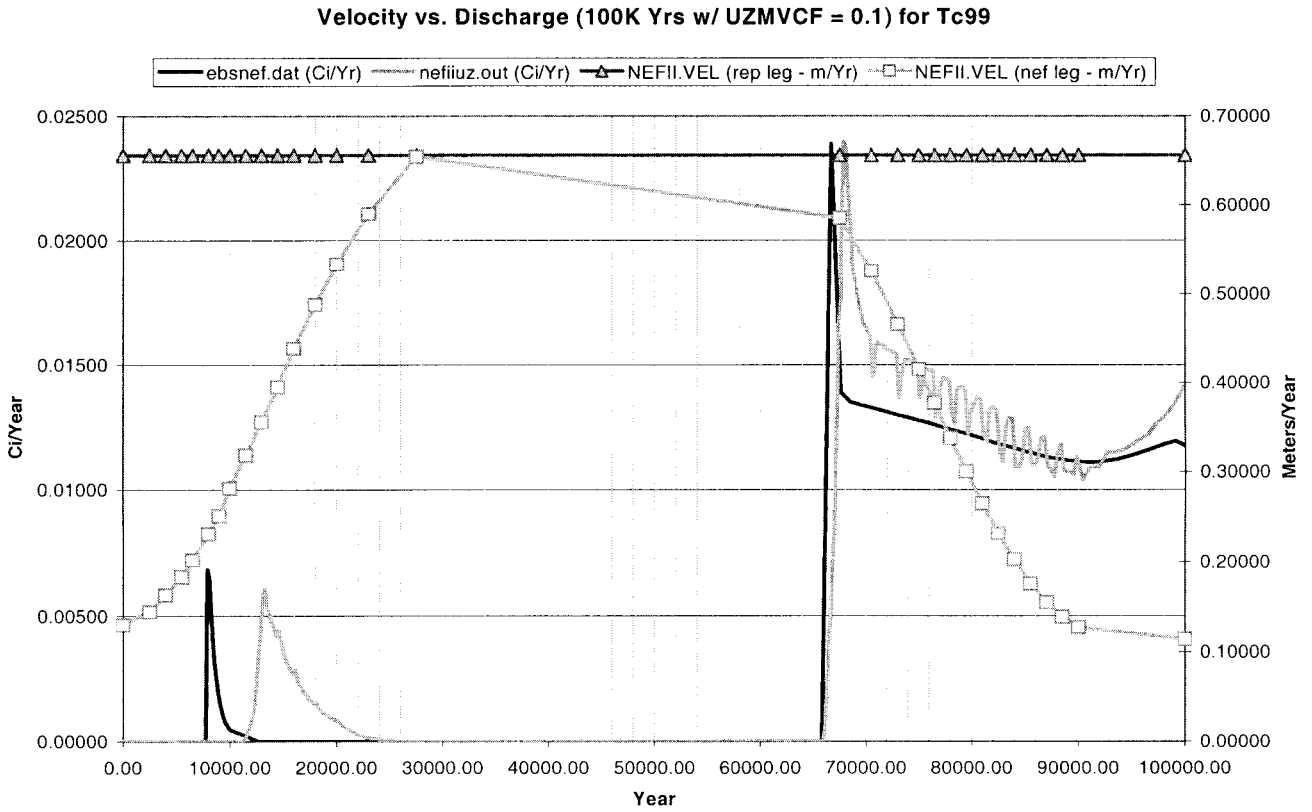
This SCR will require changes to szft.f similar to velocity processing in uzft.f. Ron & Sitakanta have noted an apparent glitch either in uzft.f or in NEFTRAN where discharge rates are too abrupt and have a disconnect between the velocity step boundaries. Ron wants me to look at this before adding similar code to szft. I can run the baseline code and save the intermediate and output files. These can be modified and used to run ***nefmks*** directly to test the glitch. Mods to NEFII.VEL. Look at nefii.out.

07/15/2002 - 07/19/2002 -

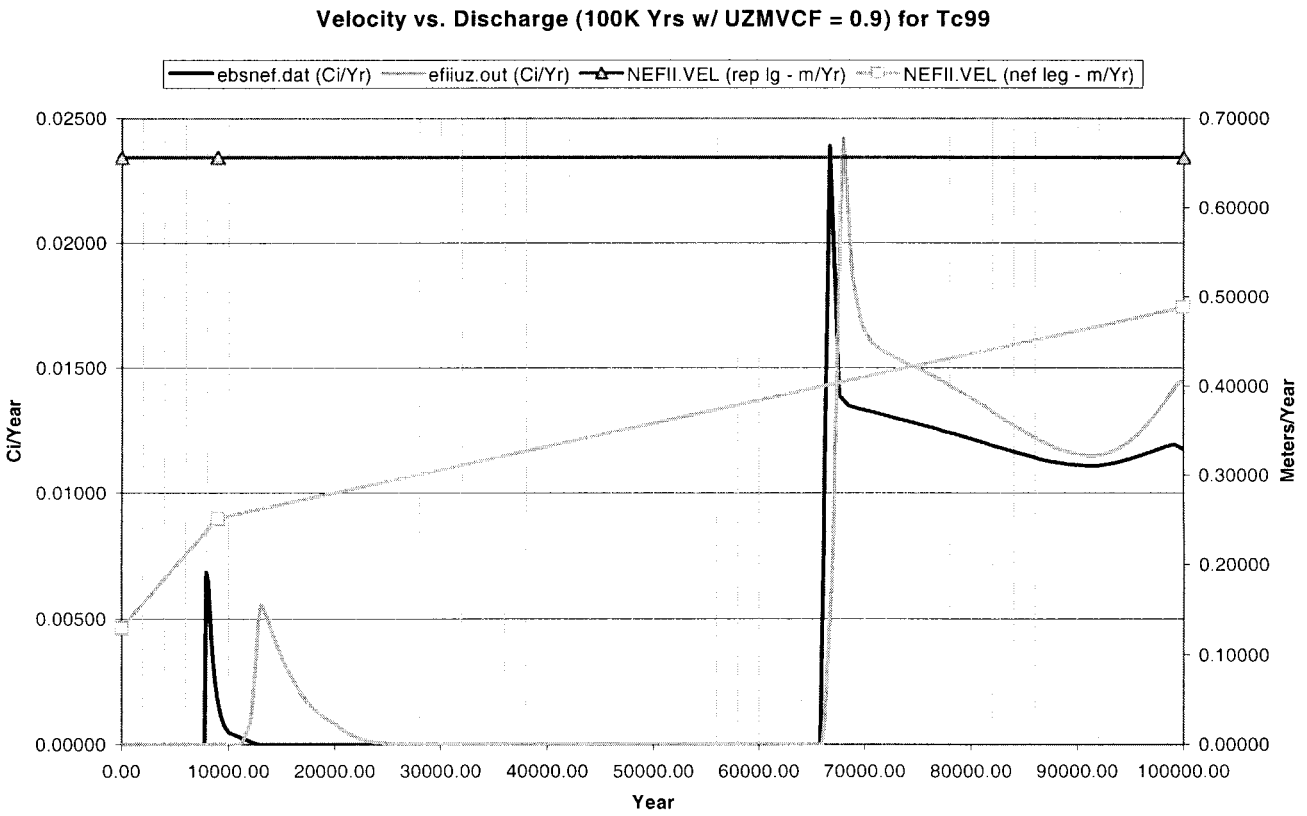
Meeting w/ Ron J., Sitakanta, Randy Fedors, George Adams, Brandi, Bob Rogers, and Michael Smith to discuss what we need to finish on TPA testing and by when. Ron wants all coding finished by Sept. 3. Delivery to NRC is at end of Nov. The period of time Sept.- Oct. is for more testing.

The sampled parameter UnsaturatedZoneMinimumVelocityChangeFactor[Fraction] in tpa.inp influences the number of time steps saved for velocity (NEFII.VEL). The smaller the fraction
(called velfac or UZMVCF in my files), the more time steps saved. Runs made with baseline code (v. 4.2f) for velfacs of 0, 1, 3, 5, 6, 7, 9 and 10. Saved nefii intermediate and output files in subdirectories velfac0, velfac1, velfac3, velfac5, velfac6, velfac7, velfac9, and velfac10.

Plot velocities from NEFII.VEL and discharge rates from nefiiuz.out using Excel. Look for the discontinuties. To begin with, use velfac1, velfac9, and base100K (same as UZMVCF = 0.4).
The results from both files will be plotted on one graph using scatter plot w/ 2 Y axes.



Graphs stored in tpatest/scr377/neftran_glitch/vel_ci01.xls, vel_ci04.xls, and vel_ci09.xls.



Results indicate a probable NEFTRAN bug. Ron is sending the data to Tim McCartin.

Question regarding saturation fraction used in writing out NEFII.VEL. It is the last column of the write statement for repvel. It is set to 1.0 for UZ. It can't be set any higher for SZ. Should it be 1.0 for both?

SCR # 377 is on hold. There is no agreement on what this SCR mod should do yet. Move on to SCR # 379. Ron has already made the mods. I need to write a test plan and execute it. SCR # 379:

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

D1 Add plume capture model

Currently, the DCAGW module handles the plume capture due to pumping at 10 km [6.2 mi] differently than at 20 km [12.4 mi]. An option will be provided in the *tpa.inp* file to select a single algorithm to be used for both locations consistent with the rule specification of a pumping rate of 3,000 acre-feet/year.

Talked with George Adams about SCR #375. He will be doing the official test for this SCR.

07/22/2002 -

Working full time starting this week through end of current fiscal year (9/27).

Test plan for SCR # 379 on hold until version 4.2i is available. Ron still needs to complete merging code for version 4.2h, which will include SCR # 395. In the meantime, updated electronic notebook.

Used command ssh -l cscherer scratchy1 to log onto scratchy1.

Copied modified files for SCR # 379 from /net/export/home/janetzke/tpa/test/scr379 on scratchy1 to /net/spock/home/cscherer/tpatest/scr374/janetzke. TPA version 4.2i is not in /dev yet.

Ran diff on all the files to see what exactly has changed for this SCR against version 4.2h, which is the current version in the /dev directory..

07/23/2002 -

There were too many differences between Ron's modified files and version 4.2h that were not directly related to SCR # 379. According to the tpa.inp in Ron's SCR379 directory, he was modifying version 4.2b code. I didn't have 4.2b in any of my directories, but I did have version 4.2c in my scr382 directory, so I compared Ron's files against 4.2c. This is much cleaner. The differences are contained in six files: ashplumo2.dif, dcags2.dif, dcagw2.dif, szft2.dif, exec2.dif and tpa2.dif. None of the .i files in Ron's directory have any mods.

(NOTE: I have TPA version 4.2 in scr344, scr393, and scr376. Version 4.2 c in scr382 and scr373. Directories scr374 and scr375 used TPA version 4.2d. Scr377 contains version 4.2f. I have versions 4.2g and 4.2h in their own subdirectories under tpatest/v42g and tpatest/v42h.)

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Copied TPA v4.2b from Ron's directory on spock to scr379. Ran diff on the five .f files and tpa.inp again (ashplumo3.dif, dcags3.dif, dcagw3.dif, exec3.dif, szft3.dif, and tpa3.dif). Working on setting up the test plan. Saved baseline version of tpa.e (tpa_base.e) and compiled modified code version of tpa.e (tpa_mod379.e).

Looks like Ron modified these files in April/May after I had made changes for SCR #344, SCR #393, and SCR #376. Those SCRs had not been included in /dev yet so those changes also show up in the .dif files.

Code failed to compile first time because it contained two include statements for *ieee/part1.i* and *ieee/part2.i*, which the compiler couldn't find. Ron said to delete the two lines from exec.f.

Ran modified code tpa.e, but execution was aborted (see tpa_base1.out) w/ error message:

```
***>>> Error in DCAGW <<<***
check pump rates in dilution.dat and the
values specified in "tpa.inp"
- the sampled value for pump rate is outside
  of the range for the data in dilution.dat

sampled value for pump rate (m3/d) =     24323.088745211

for screen length:
in dilution.dat - pump rate lower limit (m^3/d) =     3.4000000000000
in dilution.dat - pump rate upper limit (m^3/d) =    1000.00000000000

for capture widths and thicknesses:
in dilution.dat - pump rate lower limit (m^3/d) =     3.4000000000000
in dilution.dat - pump rate upper limit (m^3/d) =    1000.00000000000
```

07/24/2002 -

Met with Ron in morning. Error message probably due to changes made in dilution.dat which don't appear in version 4.2b file. Copied dilution.dat from version 4.2h. Re-ran tpa.e - results in test2 subdirectory.

Hard-copy of SCR # 379 not accurate on changes made.

In tpa_mod379.inp, the following old sampled parameters are defunct:

DilutionModel

DistanceToReceptorGroup[km][should_be_10_or_20]
MixingZoneThickness20km[m]
PlumeCaptureModel
PlumeThickness5km[m]
PluvialWellPumpingRateAtReceptorGroup20km[gal/day]
UserDefinedDilutionVolume[gal/day]
UserDefinedPlumeCaptureFraction
WellPumpingRateAtReceptorGroup10km[gal/day]
WellPumpingRateAtReceptorGroup20km[gal/day]

New/replacement sampled parameters have been added. These are:

DistanceToReceptorGroup[km]
MixingZoneThickness[m]
Note: mixing zone thickness = plume thickness ?????????
PlumeCaptureModel(1=UserDefined,2=Calculated)
PlumeThickness[m]
PluvialDilutionModel(1=UserDefined,2=PumpingRate)
PluvialWellPumpingRateAtReceptorGroup[gal/day]
PresentDayDilutionModel(0,1,2,3,4)
Note: (0=Calc,1=User,2=Pumped,3=UZq,4=Plume)
ReceptorGroup(1=Farming,2=Residential)
UserDefinedPlumeCaptureFraction
UserDefinedPluvialDilutionVolume[gal/day]
UserDefinedPresentDayDilutionVolume[gal/day]
WellPumpingRateAtReceptorGroup[gal/day]

The only two existing sampled parameters to remain after the mods are:

AquiferThickness5km[m]
PluvialSwitchTime[m]

Mods to the code are as follows:

ASHPLUMO.F:

Use of sampled parameter DistanceToReceptorGroup[km][should_be_10_or_20] was changed to DistanceToReceptorGroup[km].

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

DCAGS.F

"New parameters for climate and location of receptor group permit the use of explicit specification of farming or residential DCGs. Pluvial DCF determination was also changed for consistency with the dcagw.f method."

Use of sampled parameter DistanceToReceptorGroup[km][should_be_10_or_20] was changed to DistanceToReceptorGroup[km].

Added declarations:

```
integer receptorgroup
integer pluv
        WHAT ABOUT pluvswitch?
```

Removed declarations:

```
parameter      (maxtubes = 50)
dimension      szflowrate(maxtubes)
character*80   aline
```

Use of sampled parameter DistanceCutoffForDoseConversionDualityInDCAGS[km] removed. However, parameter still remains in tpa.inp. DOES IT STILL NEED TO BE THERE?

Added use of sampled parameters ReceptorGroup(1=Farming,2=Residential) and PluvialSwitchTime[yr].

Commented out code that opened and read in strmtube.dat because "Distance is no longer limited to near or far as specified in strmtube.dat, therefore strmtube.dat is not needed." Commented out code checking anearfielddist, farfielddist, and cutoffpt. DO WE STILL NEED VARIABLES ANEARFIELDDIST, FARFIELDDIST, CUTOFFPT?

Added code to calculate dKGvalue at pluvial switch time, to be used to determine time of return to current conditions.

Moved reading of DCF files outside of time step loop. Inside time loop, processing branches due to receptorgroup value instead of using dMAP and dKGboundary variables.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

DCAGW.F

Added declarations:

```
dimension     dwdcfc(43)
dimension     dwdcfp(43)
        Note: WHAT ARE THESE? WHAT IS SIGNIFICANCE OF 43?
integer       receptorgroup
integer       pbdilflg

common / dcagw30 / irg, receptorgroup
        Note: add receptor group classification variable.
```

Modified declarations:

```
common / dcagw2/ dcf(43)  to common / dcagw2 / dcf(43), dcfp(43)
```

Use of sampled parameter DistanceToReceptorGroup[km][should_be_10_or_20] was changed to DistanceToReceptorGroup[km]. WellPumpingRateAtReceptorGroup10km[gal/day] and WellPumpingRateAtReceptorGroup20km[gal/day] was changed to WellPumpingRateAtReceptor Group[ga/day]. PlumeThickness5km[m] was changed to PlumeThickness[m].

"Repository footprint mixing zone thickness used in determination of the SZ flow volume for dilution model 4." MixingZoneThickness20km[m] was changed to MixingZoneThickness[m].

"Replace distance to cutoff parameter with receptor group id for group selection." DistanceToDoseConversionDualityInDCAGS[km] replaced with ReceptorGroup(1=Farming,2=Residential".

Use PlumeCaptureModel(1=UserDefined,2=Calculated) instead of PlumeCaptureModel.

"Add new flag for pluvial biosphere dilution" PluvialDilutionModel(1=UserDefined,2=PumpingRate) and pbdilflg. Replace DilutionModel with PresentDayDilutionModel(0=Calc,1=User,2=Pumped,3=UZq,4=Plume). Error check dilflg.

Replace UserDefinedDilutionVolume[gal/day] with UserDefinedPresentDayDilutionVolume[gal/ day]. Add

UserDefinedPluvialDilutionVolume[gal/day]. Replace PluvialWellPumpingRateAtReceptor Group20km[gal/day] with PluvialWellPumpingRateAt ReceptorGroup[gal/day].

Large block of 278 lines (approx. 788:1145) commented out and rewritten. New block of 236 lines added.

Added new code regarding epadosetotal, epadosesum, and epadose (approx. 1309:1318 in dcagw_mod379.f).

"Delay entering time step loop until after the DCF files have been read."

Replace branching according to cutoffpt with branching based on receptorgroup and pluv.

Looks like additional reads of variables dcfde, dcfinh, dcfinga, dcfingp, dcfdw, dcfmlk for use in calculations. ???????

Redo time step block of 295 lines of code (approx 1479:1941); replace with 320 lines. Remove vestigial if block.

EXEC.F

Use of sampled parameter DistanceToReceptorGroup[km][should_be_10_or20] was changed to DistanceToReceptorGroup[km].

SZFT.F

Use of sampled parameter DistanceToReceptorGroup[km][should_be_10_or20] was changed to DistanceToReceptorGroup[km].

NOTE: Still need to move reading of DCF files out of time step loop for szft.f..

Uses new parameters to decide branching path.

*****

To begin with, test using 10,000 years, 1 realization, and 1 subarea. Test new/modified parameters to see if changes occur and in the right direction. Check tpameans.out and totdose.res for test run against base case.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Recompiled base case tpa.e and ran it. Saved results in base1 subdirectory.

When I compared the tpameans.out files, I discovered that the disruptive scenario flags were all set to 1 in tpa_mod379.inp, but only the seismic one was set to 1 in tpa_base.inp. I don't think that any of them need to be set, but I will confirm that with Ron. Anyway, need to re-run these basic tests with the flags set to the same values in each case. In totdose.res, the dilution volume is less for the test case than the base case and remains constant throughout for both cases. Tede is 0.0 all through the base case, but does show a value at 4291 years that then decreases for the test case.

Modified tpa_base.inp and tpa_mod379.inp so that all 3 disruptive scenario flags set to 0. Re-ran both base and test cases (see subdirectories base_nodisrupt and test_nodisrupt). This time, there were no tede values in the test case other than 0. Asked Ron about disruptive scenario flags. He said the base case should have Seismic turned on and the others turned off. He also said that setting the Volcanism flag will effect results, and I will need to run tests with this flag set and with it turned off.

Ron sent me an email with some notes from his scientific notebook when he made the changes. There is mention of the sampled parameter DistanceToTuffAlluviumInterface[km] from the SZFT section. Need to check that the correct leg length is sent to NEFTRAN in the case where the receptor group is located closer than the DistanceToTuffAlluviumInterface[km]. The parameter is only referenced in szft.f. distTAI is compared to farlength and w.r.t. disttocg (distance to critical group).

Ron's email:

```
Carol,

Here are some notes from my scientific notebook.


Apr. 14, 2002 DCAGW.F

If the dilution flag is modified to support 4 dilution models:

dilflg =0 => maximum of models 2,3,4
dilflg = 1 => user defined dilution volume
dilflg = 2 => use pumping rate
dilflg = 3 => use average UZ flow rate
dilflg = 4 => use SZ mixing volume, mixing zone thickness and plume
thickness can be interpreted as providing the same information and can
be combined into one (plume thickness).

and 2 receptor groups:
```

```
receptor group = 1 => farming
receptor group = 2 => residential

then the Version 4.1 case for distance to receptor group of 10km can be
simulated by using dilflg=2, and setting the receptor group to group 2
and setting the pluvial pumping rate equal to the regular pumping rate,
and the 20km case can be simulated by using dilflg = 0 and setting the
receptor group to group 1 and fraction of mass captured to 1.


April 19, 2002 DCAGW

This is the dcagw portion of the mean value tpa.inp file that produced
identical results to the base case mean value file.

constant
DistanceToTuffAlluviumInterface[km]
14.950000000000
**
** ***>>> DCAGW <<<***
**
iconstant
ReceptorGroup(1=Farming,2=Residential)
1
**
constant
DistanceToReceptorGroup[km]
20.0
**
iconstant
PlumeCaptureModel(1=UserDefined,2=Calculated)
1
**
constant
UserDefinedPlumeCaptureFraction
1.0d0
**
constant
PlumeThickness[m]
55.000000000000
**
constant
AquiferThickness5km[m]
500.00000000000
**
**uniform
**MixingZoneThickness20km[m]
**50.0, 200.0
**
**PresentDayDilutionModel(0=Calc,1=User,2=Pumped,3=UZq,4=Plume)
iconstant
PresentDayDilutionModel(0,1,2,3,4)
0
**
constant
UserDefinedPresentDayDilutionVolume[gal/day]
```

```
10901787.249406d0
**
**uniform
**WellPumpingRateAtReceptorGroup10km[gal/day]
**1.5e4, 2.64e5
**
constant
WellPumpingRateAtReceptorGroup[gal/day]
8750000.0000000
**
constant
PluvialSwitchTime[yr]
13000.0
**
iconstant
PluvialDilutionModel(1=UserDefined,2=PumpingRate)
2
**
constant
UserDefinedPluvialDilutionVolume[gal/day]
10901787.249406d0
**
constant
PluvialWellPumpingRateAtReceptorGroup[gal/day]
6215000.0000000
```

April 23, 2002 DCAGW

The following was sent via e-mail to DCAGW folk:

The DCAGW code was reviewed prior to designing changes to implement the addition of the plume capture model for the 20km receptor group. The module proved to be very difficult to read, containing somewhat convoluted logic, and greatly in need of restructuring for performance reasons. Here is a list of issues found with the code:

1) Distance to receptor group is limited to 10km or 20km, and this fact is hard coded into the parameter name DistanceToReceptorGroup[km][should_be_10_or_20].

2) Parameters

WellPumpingRateAtReceptorGroup10km[gal/day]
WellPumpingRateAtReceptorGroup20km[gal/day]
PlumeThickness5km[m]
MixingZoneThickness20km[m]

bloat the tpa.inp file with names that could be generic rather than specific (because TPA can only run one receptor group at a time), using generic names would reduce the parameter count.

3) Using a new ?iconstant? parameter to specify the receptor group type as farming or residential provides a consistent flag that can be used for multiple areas that currently use various parameters to determine the same information. For example:

DistanceToReceptorGroup[km][should_be_10_or_20] - is cryptically used to specify receptor group type.

DistanceCutoffForDoseConversionDualityInDCAGS[km] - is used in combination with the parameter above to specify the receptor group yet again.

4) The PlumeCaptureModel parameter has no indication of acceptable values. I suggest
PlumeCaptureModel(=UserDefined,2=Calculated).

5) With restructured code it would be trivial to add parameters to specify one of four dilution models for present day biosphere and one of two pluvial dilution models. These would be available for both farming and residential cases. I suggest a set of 4 parameters:

PresentDayDilutionModel(0=Calc,1=User,2=Pumped,3=UZq,4=Plume)
UserDefinedPresentDayDilutionVolume[gal/day]

PluvialDilutionModel(1=UserDefined,2=PumpingRate)
UserDefinedPluvialDilutionVolume[gal/day]

Present day model 0 means the maximum of models 2, 3 and 4 (this is the TPA4.1j method).
1 means user defined volumes for present day and pluvial.
2 means dilution volume always equals pumped volume while respecting the pluvial switch for different pumping rates.
3 means dilution volume always equals the average UZ flow.
4 means dilution volume always equals the SZ flow down to the depth of the plume thickness.

6) The PluvialWellPunpingRateAtReceptorGroup20km[gal/day] parameter has the receptor group location hard coded in the name and requires a code modification to change it. With distances other than 18km for the receptor group the user should have the freedom to design any scenario for other distances.

7) The plume capture algorithm is restricted unnecessarily to the 5km case.

8) The current code has a major performance problem by reading the DCF files inside the time loop. That means the DCF files are opened, read and closed once for each time step!! These will be moved outside the time step loop.

9) Although time does not permit right now, the sections that read strntube.dat, dilution.dat, and the DCF files should be put in separate subroutines in order to clean up DCAGW.

Please let me know, either YES, NO or DON?T CARE, as to whether these changes should be made and if they cause more problems than they solve.

ron j

Ron says that PlumeThickness[m], in dcagw.f, refers to plume thickness at the receptor site and that MixingZoneThickness[m] is a footprint of the repository. So, in that case, they are not referring to the same thing.

Started a table of test cases to cover all the new parameters and modified code. Use values from tpameans.out to set constants. Check interactions among sampled parameters. Have Ron check table before first draft of test plan. Need to find other results suitable for comparison & test purposes.

07/26/2002 -

Received email from Ron Janetzke:

```
Carol,

It appears that the fracture values are being sent to prenefmks. I
assume that it should actually be the matrix values. Please make
changes to your files and send a new one to me.

thanks,

ron j
```

Fracture values are returned from prenefmks. Need to talk to Ron to see what the problem is.

Continued working on table of test cases for SCR #379. Relationships among sampled parameters:

DCAGW:

ReceptorGroup (1 & 2)
    PluvialSwitchTime (13000 & ?)
DistanceToReceptorGroup (10 km & 20 km?)
PlumeCaptureModel (1 & 2)
    1: UserDefinedPlumeCaptureFraction (1.0d0 & ?0.5d0?)
    2: PlumeThickness (constant from tpameans.out) ?always
       AquiferThickness (constant from tpameans.out) ?always
       WellPumpingRateAtReceptorGroup (constant from tpameans.out) ?always
PresentDayDilutionModel (0, 1, 2, 3, & 4)
    0: MixingZoneThickness (constant from tpameans.out) ?always

PluvialWellPumpingRateAtReceptorGroup (constant from tpameans.out) ?always
WellPumpingRateAtReceptorGroup (constant from tpameans.out) ?always
1: UserDefinedPluvialDilutionVolume (10901787.249406d0) ?always
UserDefinedPresentDayDilutionVolume (10901787.249406d0) ?always
2: WellPumpingRateAtReceptorGroup (constant from tpameans.out) ?always
PluvialWellPumpingRateAtReceptorGroup (constant from tpameans.out) ?always
3: none
4: MixingZoneThickness (constant from tpameans.out) ?always
PluvialDilutionModel (1 & 2) - read in but not used?????

DCAGS:

ReceptorGroup
DistanceToReceptorGroup
PluvialSwitchTime

SZFT:

DistanceToTuffAlluviumInterface - don't know if effect of this parameter has been taken into account for SCR #379.

At least one test should compare test results against base results. Need more than one? Need to exercise modified code in both dcags.f and dcagw.f? Code changes in exec.f, ashplumo.f, and szft.f are minimal - a different name for one parameter - do they need to be tested at all?

Need to determine which output files to look at when parameters are being tested (gw_*.dat sometimes, totdose.res, gs_*.dat, others?).

Started running tests on effects of DTTAI to see if already taken care of or whether needs to be added to code.

7/29/2002 -

Possible Tests for SCR # 379:

Functional:

1. Print statements to see that parameters are read in correctly and that calculations match hand calculations.

System:
1. Compare base case to test case at 10 km (as per Ron's email).
2. Compare base case to test case at 20 km (as per Ron's email).
3. Compare PlumeCaptureModel w/ different PlumeCaptureFraction & using calculations.
    a. PlumeCaptureModel = 1, PlumeCaptureFraction = 1.0
    b. PlumeCaptureModel = 1, PlumeCaptureFraction = 0.5
    c. PlumeCaptureModel = 2
4. PluvialSwitchTime - does it switch back to present day when it should?
    a. dcags
        1) 13,000 yrs.
        2) 30,000 yrs.
    b. dcagw
        1) 13,000 yrs.
        2) 30,000 yrs.
5. PluvialDilutionModel . . . (depends on what Ron decides)
6. ReceptorGroup/DistanceToReceptorGroup
    a. ReceptorGroup = 1, DistanceToReceptorGroup = 10 km.
    b. ReceptorGroup = 1, DistanceToReceptorGroup = 20 km.
    c. ReceptorGroup = 2, DistanceToReceptorGroup = 10 km
    d. ReceptorGroup = 2, DistanceToReceptorGroup = 20 km
7. PresentDayDilutionModel
    a. 0 - Calculated
        1) WellPumpingRateAtReceptorGroup, MixingZoneThickness, & PluvialWellPumpingRateAtReceptorGroup =
    b. 1 - User defined
        1) UserDefinedDilutionVolume = UserDefinedPluvialDilutionVolme = ? vary
    c. 2 - Pumped
        1) WellPumpingRateAtReceptorGroup = PluvialWellPumpingRateAtReceptorGroup = ? vary
    d. 3 - UZq
        1) use aveqm3peryrallsafromuz passed into dcagw
    e. 4 - Plume
        1) MizingZoneThickness =

? vary

Continued with tests on effects of DTTAI to see if already taken care of or whether needs to be added to code. Used five test cases: 1) DistanceToTuffAlluvium = 7, DistanceToReceptorGroup = 10; 2) DTTA = 7, DTRG = 20; 3) DTTA = 15, DTRG = 10; 4) DTTA = 15, DTRG = 20; and 5) DTTA = 25, DTRG = 20. Checked leg lengths in nefii.inp for each case. They all seem to be correct. Since this has already been taken care of in SZFT in an earlier mod, don't need to re-test for SCR # 379.
Output files stored in cscherer/tpatest/scr379/distTAI/t7r10, t7r20, t15r10, t15r20, and t25r20 subdirectories.

Output files of interest for testing SCR #379 are totdose.res, totdos_c.res, gw_cb_ad.dat, gw_cb_ci.dat, gw_pb_ad.dat, and gw_pb_ci.dat. If selected legs and their length are of interest, look at nefii.inp. Save tpameans.out from base cases to obtain constants for test cases. Other information of interest will come from print statements. Need to look at gs_cb_ad.dat, gs_cb_ci.dat, gs_pb_ad.dat, and gw_pb_ci.dat to determine if any of these are affected, too.

Ron is looking into whether we need to remove PluvialDilutionModel or add code to use it. The dcagw.f code reads the parameter in, but never makes use of it.

Ron looked at uzft.f and the fracture/matrix code again and decided there is no problem.

Tim McCartin is looking into the NEFTRAN glitch problem. We'll wait to hear back from him.

Worked on test plan (tp_scr379.wpd). Set up for Function Level Test # 1 (FLT #1) and System Level Test # 1 (SLT #1). Copied ashplumo_mod379.f, dcags_mod379.f, dcagw_mod379.f, exec_mod379.f, and szft_mod379.f to _test379.f versions to which I added print statements for use in testing.

Ran tpa_mod379.e w/ the tpa.inp file from Ron's scr379 directory. Saved tpameans.out for values to use in testing. The DCAGW portion was:

```
**
**          ***>>>   DCAGW   <<<***
**
iconstant
ReceptorGroup(1=Farming,2=Residential)
1
**
```

```
constant
DistanceToReceptorGroup[km]
20.0
**
iconstant
PlumeCaptureModel(1=UserDefined,2=Calculated)
1
**
constant
UserDefinedPlumeCaptureFraction
1.0d0
**
constant
PlumeThickness[m]
     55.000000000000
**
constant
AquiferThickness5km[m]
     500.00000000000
**
constant
MixingZoneThickness[m]
     125.00000000000
**
**PresentDayDilutionModel(0=Calc,1=User,2=Pumped,3=UZq,4=Plume)
iconstant
PresentDayDilutionModel(0,1,2,3,4)
0
**
constant
UserDefinedPresentDayDilutionVolume[gal/day]
10901787.249406d0
**
**uniform
**WellPumpingRateAtReceptorGroup10km[gal/day]
**1.5e4, 2.64e5
**
constant
WellPumpingRateAtReceptorGroup[gal/day]
     8750000.0000000
**
constant
PluvialSwitchTime[yr]
13000.0
**
iconstant
PluvialDilutionModel(1=UserDefined,2=PumpingRate)
2
**
constant
UserDefinedPluvialDilutionVolume[gal/day]
10901787.249406d0
**
constant
PluvialWellPumpingRateAtReceptorGroup[gal/day]
     6215000.0000000
```

For testing PluvialTimeSwitch, need to check dKG and dKGValue to make sure processing switches back from Pluvial to Present Day when the value goes back to the pre-pluvial value (to switch off pluvial). Pluvial is turned on when time exceeds the value input for PluvialTimeSwitch.

Made tpa_test379.e version of the executable code using the five *_test379.f files.

Looking at the TPA 5.0 Development Status chart Ron gave us at the last meeting, I see that SCR #380 is supposed to be included w/ #379. I retrieved the SCR to read. The only question it raised is whether or not strmtube.dat needs to be or has been changed. Also, should I use 18 km for testing (not 20)? Need to check w/ Ron. There is another meeting tomorrow afternoon to discuss #377 (szft velocity) & #379 (PluvialDilutionModel).

Used values from tpameans.out to set up constant values in the DCAGW section of tpa_test379.inp. Used one realization and subarea one for 10,000 yrs.

7/30/2002 -

Ron found an error in my logic in uzft.f for SCR #374. I initialized lowestmatfraction outside the time step loop. Should be initialized inside the time step loop before the layer loop. Already in the system, so Ron will generate SCR # 396 (?) to take care of this one. Should be an easy fix. Will need to force a very low matrix fraction and show that it does not carry over into the next time step.

Functional test for SCR #379. Ran tpa_test379.e. With PluvialSwitchTime set to 13000 yrs. and MaximumTime set to 10000 yrs., can't see the climate switch processing. Reset PST to 1300 yrs. That run did show the switch TO pluvial time, but never switched back. Renamed directory test379_1300. Tried 13000 yrs and 100000 yrs. which did allow enough time for the climate to switch back to present day.

Too much output for one functional level test. FLT1 will look at dilution volumes, fraction mass captured and pluvial climate switch. FLT2 will look at the calculations for remperyrnuclgw. Both will print out the sampled parameters and values read in. Tpa_flt1.inp will be set up to use user-defined values and receptor group 1 while tpa_flt2.inp will use calculated values and receptor group 2. Ran flt1 and flt2. However, nothing printed out from ashplumo.f or dcags.f. Need to turn on disruptive scenario flag for volcanism and rerun both cases. Also want to print out values only for some cases, not all cases.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Reviewed SCR # 377. Met at 1:30 w/ Ron Janetzke, Randy Fedors, David Farrell, and James Winterle to discuss SCR #377, SCR #379 and SCR #380. Results of discussion follow:

1. Agreed on solution for adding time-dependent velocity to szft.f. Uses value(s) from climato2.dat at given time steps (cl2) and flux value(s) from strmtube.dat.. Velocity = (flux * (1 + cl2 (SZFMAGM - 1))) / mobile porosity. This should be similar to the way Precipitation (modern day) is currently calculated. SZFMAGM is a new sampled parameter SZFluxMultiplierAtGlacialMaximum to be added (w/ option to correlate w/ the existing sampled parameter MeanAveragePrecipitationMultiplierAt GlacialMaximum (MAPMAGM) to tpa.inp). Will also need new parameter SaturatedZoneMinimumVelocityChangeFactor (SZMVCF). Start this value at 0.1. Need to be able to access both strmtube.dat and climato2 from within szft.f. Test by holding all parameters constant except new one. Will be able to look at velocities in nefiisz.vel.

2. Dilution.dat needs to be expanded to account for higher pumping rates. Need to be able to meet EPA rule of 3,000 acre ft./yr. James Winterle will get some new rows for the tables by the end of August. Currently, for the new sampled parameters and code changes (#379, #380), using a table where the last row changed pumping rate from 1000 to 100000. Allows the code to function, although some of the numbers may be off the wall.

3. Strmtube.dat is usable as is. May be able to get rid of the near field section altogether. I will check the code to see if it is still accessed.

4. Do not need the sampled parameter PluvialDilutionModel tentatively added for SCR #379. Delete from tpa.inp and the code (dcagw.f).

5. Agreed to update SCR #377 w/ summary of solution from meeting.

Ron will be on vacation for the rest of this week.

Removed the PluvialDilutionModel parameter from tpa.inp and the code from dcagw.f. Remade the flt1 executable

7/31/2002 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Analyzed results of FLT #1. All the new sampled parameters were read in with the correct values. Calculations for converting the pump rate and the pluvial pump rate were done correctly. So were the calculations to convert dilution volume and pluvial dilution volume. With PlumeCaptureModel set to 1, fractionmasscaptured was set to the user defined value of 1.0. Need to run this one again in FLT #2 to see if fractionmasscaptured is calculated correctly when PlumeCaptureModel is 2. Also ran several dKG and dKGvalue calculations with a hand calculator. Those results are fine, too. The PluvialSwitchTime toggle is working correctly. But, I don't understand why the time step loop in dcags.f starts at time step 144 instead of 1. Files are in tpatest/scr379/flt1.

Added and/or deleted print statements to dcagw.f and dcags.f for FLT #2. Not quite sure what to do with new arrays dwdcfc and dwdcfp. I believe that there use will be fully tested during system level tests and I shouldn't need to do anything special with them for functional level tests. But I will look more closely at how they're used before finishing this test. Too much info being printed out so added the conditional to only print out results every 20 time steps or when pluv changes.

Remade the flt2 executable and reran. Analyzed results. Getting the wrong fractionmasscaptured when plumflg = 2. It really wasn't wrong. Result was limited to a max of 1.0 no matter what. Will modify UserDefinedPlumeCaptureFraction to 0.5 in tpa_flt2.inp instead of 1.0 so difference will stand out. Oops. There was an error. Fractionmasscaptured is set to the userplumefraction even though plumflg is 2. Oops again. There was an error in the print statement, not the actual code. Fractionmasscaptured is being calculated correctly. However, this only tests one version of calculating fractionmasscaptured. The if-else depends on screen length, capture width and thickness. Those are not being exercised yet. Will have to determine what I need to manipulate to get into these blocks.

All the remperyrpernuclgw values are printing out 0. Not sure why. Looking at dcagw.f again.

Results are being stored in tpatest/scr379/flt2. Peak Mean Dose is slightly less for flt2 than flt1. Flt2 is residential instead of farming using calculated values instead of user defined values.

8/02/2002 -

Continued analyzing output from flt1 & flt2. Worked on test plan. Used tpa_flt1.out and tpa_flt2.out for examples of what should be seen to pass these two tests.

Not all remperyrpernuclgw values are 0 in tpa_flt2.out. The ones that were printed out every 20 time steps w/ only nuclide = 1, however, were. Just coincidence. Is it a problem that dcags processing begins with time step 144 in the time step loop? Why does it?

Tests to be included in test plan:

All tests use these value in tpa.inp:

StartAtRealization = 1
StopAtRealization = 1
StartAtSubarea = 1
StopAtSubarea = 1
MaximumTime = 1.0e5

Functional level:

1. Does code read in new parameters and values correctly? Does the code for PluvialSwitchTime
work correctly? Are calculated values calculated correctly? Check computations for dKG and dKGvalue. In addition to above, use these values for the new sampled parameters:

ReceptorGroup = 1 (residential)
DistanceToReceptorGroup[km] = 20.0
PlumeCaptureModel(1=UserDefined,2=Calculated) = 1
UserDefinedPlumeCaptureFraction = 1.0d0
PlumeThickness[m] = 55.0
AquiferThickness5km[m] = 500.0
MixingZoneThickness = 125.0
PresentDayDilutionModel(0,1,2,3,4) = 1 (userdefined)
UserDefinedPresentDayDilutionVolume = 10901787.249406d0
WellPumpingRateAtReceptorGroup[gal/day] = 8750000.0
PluvialSwitchTime = 13000
UserDefinedPluvialDilutionVolume[gal/day] = 10901787.249406d0
PluvialWellPumpingRateAtReceptorGroup[gal/day] = 6215000.0

2. Check PluvialSwitchTime w.r.t. remperyrpernuclgw. Does Peak Mean Dose differ for this receptor group? Modify these parameters from FLT # 1:

Receptor Group = 2 (farming)
PlumeCaptureModel(1=UserDefined,2=Calculated) = 2
UserDefinedPlumeCaptureFraction = 0.5d0
PresentDayDilutionModel(0,1,2,3,4) = 0 (calculated)

These two tests test all affected processing branches except for the different blocks in the if-else that calculates fractionmasscaptured. Need to look at how to control screen length, capture width, and capture thickness to get into the blocks. Also, what about new arrays dwdcfc & dwdcfp? Do I need to look explicitly at how these are filled and used?

System level:

1. Compare base case at 10m with modified code - see Ron's email. Results should be the same. Use:

   ReceptorGroup = 2 (Residential)
   DistanceToReceptorGroup[km] = 10
   PresentDayDilutionModel(0,1,2,3,4) = 0 (Calculated)
   PlumeCaptureModel(1=UserDefined,2=Calculated) = 1
   UserDefinedPlumeCaptureFraction = 1.0d0
   WellPumpingRateAtReceptorGroup = 8750000.0
   PluvialWellPumpingRateAtReceptorGroup[gal/day] = 8750000.0

   DistanceToTuffAlluviumInterface[km] = 14.95

   SET OLD PARAMETERS:

2. Compare base case at 20 km with modified code - see Ron's email. Results should be the same. Use:

   ReceptorGroup = 1 (Farming)
   DistanceToReceptorGroup[km] = 20
   PresentDayDilutionModel(0,1,2,3,4) = 0 (Calculated)
   PlumeCaptureModel(1=UserDefined,2=Calculated) = 2
   UserDefinedPlumeCaptureFraction = 1.0d0
   WellPumpingRateAtReceptorGroup = 8750000.0
   PluvialWellPumpingRateAtReceptorGroup[gal/day] = 6215000.0

   DistanceToTuffAlluviumInterface[km] = 14.95

SET OLD PARAMETERS:

3. Show can do Farming at 10 km. Use:

ReceptorGroup = 1
DistanceToReceptorGroup[km] = 10
PlumeCaptureModel(1=UserDefined,2=Calculated) = 1
PlumeCaptureFraction = 1.0d0
PresentDayDilutionModel = 1

4. Show can do Residential at 20 km. Use:

ReceptorGroup = 2
DistanceToReceptorGroup[km] = 20
PlumeCaptureModel(1=UserDefined,2=Calculated) = 1
PlumeCaptureFraction = 1.0d0
PresentDayDilutionModel = 1

5. Compare different results when each of 5 PresentDayDilutionModel(0,1,2,3,4) values are selected For all 5 runs, use:

ReceptorGroup = 1
DistanceToReceptorGroup[km] = 20

When PresentDayDilutionModel = 0, use:

PlumeCaptureModel(1=UserDefined,2=Calculated) = 2

6. Compare two different PluvialSwitchTime[yr] values (13,000 & 45,000). Use:

ReceptorGroup = 1
DistanceToReceptorGroup[km] = 18
PresentDayDilutionModel(0,1,2,3,4) = 0

7. Compare two different values for UserDefinedPluvialDilutionVolume[gal/day]. Use:

ReceptorGroup = 1
DistanceToReceptorGroup[km] = 18
PresentDayDilutionModel(0,1,2,3,4) = 0

8. Compare two different values for UserDefinedPresentDayDilutionVolume[gal/day]. Use:

   ReceptorGroup = 1
   DistanceToReceptorGroup[km] = 18
   PresentDayDilutionModel(0,1,2,3,4) = 0

9. Let PluvialWellPumpingRateAtReceptorGroup[gal/day] vary over 10 realizations. Use:

   ReceptorGroup = 1
   DistanceToReceptorGroup[km] = 18
   PresentDayDilutionModel(0,1,2,3,4) = 0

10. Let WellPumpingRateAtReceptorGroup[gal/day] vary over 10 realizations. Use:

    ReceptorGroup = 1
    DistanceToReceptorGroup[km] = 18
    PresentDayDilutionModel(0,1,2,3,4) = 0

11. Let AquiferThickness5km vary over 10 realizations. Use:

    ReceptorGroup = 1
    DistanceToReceptorGroup[km] = 18
    PresentDayDilutionModel(0,1,2,3,4) = 0

12. Let PlumeThickness vary over 10 realizations. Use:

    ReceptorGroup = 1
    DistanceToReceptorGroup[km] = 18
    PresentDayDilutionModel(0,1,2,3,4) = 0

13. Let MixingZoneThickness vary over 10 realizations. Use:

    ReceptorGroup = 1
    DistanceToReceptorGroup[km] = 18
    PresentDayDilutionModel(0,1,2,3,4) = 0

8/05/2002 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Set up and ran first four System Level tests. Updated test plan. Working on finishing pass/fail criteria on SLT #3 & #4. Decided not to make test cases to vary values that were already being used in the same way before the mods. Delete cases 6 - 13 above.

8/12/2002 -

Finished SLT #3 & #4. Instead of SLT #5 having all dilution models, separate into separate tests. SLT #2 already does PresentDayDilutionModel(0,1,2,3) = 0. SLT #5 has PDDM = 1, #6 has PDDM = 2, #7 has PDDM = 3, and #8 has PDDM = 4. Ran all tests and ran diff on outputs. Added analysis results to test plan. Diff files are saved in tpatest/scr379. Completed draft of TP_SCR379.WPD for Ron to review. Besides changes to this, still need to complete SCR377.

WordPerfect says my disk space is full. Deleted what I could.

Ron will be out on travel Wed. & Thurs.

8/13/2002 -

While waiting for Ron's comments on tp_scr379, worked on scr #377 as per meeting on 7/30/02. Moved scr377 dir to neftran_glitch. Made new scr377 dir and copied over version 4.2i. (Ron said it was i. The tpa.inp says it is h.) Reviewed scientific notebook entries from last month when I first looked at scr #377.

Copied szft.f to szft_orig.f & szft_mod377.f. Copied tpa.inp to tpa_base.inp. Made code and copied tpa.e to tpa_base.e. Looked at accessing strmtube.dat and climato2.dat from szft.f.

Climato2.dat is read in uzflow.f. The data columns are:

CliTim - time instant (yr)
AAPglacfrac - corresponding fraction of full glacial MAP (Mean Annual Precipitation)
AATglatfrac - corresponding fraction of full glacial MAT (Mean Annual Temperature)

A comment in uzflow.f states that time steps must be constant. Will this matter?

Reminded Ron about the need for an SCR to correct the misplacement of the initialization of lowestmatfraction in uzft.f. It should be placed inside the time step loop but still remain outside the layer loop. Created SCR #396 and turned in to Ron. SCR #396:

```
Move initialization of lowestmatfraction.

Currently, lowestmatfraction is initialized to zero outside the
time step loop in the prenefmks subroutine in uzft.f. But it needs
to be re-initialized every time the time step loop is entered.  As
it is now, a large number for the matrix fraction in an early time
step may be carried over and used in a later time step.
```

The fix is easy. Move the statement "lowestmatfraction = 1.0d0" from before the statement "do itime = 1, ntimg" to after it. Should be able to use the earlier test plan with some minor mods to test it. SCR # 374.

Re: SCR #377. Szft.f is already calculating nefvel values. Ron says this should be replaced with the new calculation. Need to turn on the flag written to nefii.inp that says use velocity flag. Need to copy NEFII.VEL to nefiisz.vel. Need to re-dimension nefvel as in uzft.f. Run a test with zeroing out velocity values written out by uzft.f to nefii.inp. See if it makes any difference in outputs. This should tell us whether or not what is written out to nefii.inp is significant or not when the use velocity file flag is set and can be removed from szft.f safely. Want to use writevelocities already in uzft.f.

Need to add reading climato2.dat inside the time step loop. What about different time steps? Use maplist? Fluxes (flow rates?) are 3 values near the top of strmtube.dat. Which one or ones should be used & when?

Highlighted appropriate parts of code in uzft.f, szft.f, and uzflow.f to use to help figure out exactly what to add to szft.f. Still need clarification on the part about the option to correlate with MeanAveragePrecipitationMultiplierAtGlacialMaximum.

8/14/2002 -

Started out with computer problems. Spock wasn't logging in correctly - seems to be related to a timeout within the login process perhaps due to several concurrent tpa.e runs that were all going and/or temperature problems because air conditioning was turned off yesterday. I'd get as far as the system displaying the 'no mail' message and then it hung. Solution: enter ctrl-C and if my .cshrc hadn't executed enter 'source .cshrc'. Added "alias ' ls -lsd * | drw' " to my .cshrc so I wouldn't be relying on Ron's .cshrc for that one - it's too useful to do without. Could also use "alias lD 'ls -Al | grep ^d' " (from Hollen).

Ron hadn't finished reviewing the test plan for SCR #379 before he left on travel; will pick that back up on Friday. He wants some more tests, wants more explanations of why results are they way they are, and wants to discuss some points.

Worked on SCR #377.

8/15/2002 -

Analyzing code changes for SCR #377.

8/16/2002 -

Still having problems with WordPerfect and my electronic scientific notebook. Error messages about OLE prefix packet. These started on Wednesday, but are still occurring. Still have same logon problem with Spock, too.

Continued with SCR #377.

Uzflow.f reads in precipitation values from climato2.dat and stores them in array AAPmu. AAPseq = AAPmu + correlated perturbation * sdevAAP. Perturbation is stored in AAPpert. Correletated perturbation = AAPpert + AAPpert*rhoPT. AAP array = dlog10(AAPseq). AAPseq is mapped into QinSA and also AAP to correlate time steps from nCliTS & CliTim to ntim & tim. AAPseq and AAPmu are stored in commons in uz_climr.i. Which one should be used - m ay not have to reread climato2.dat.

Ran tpa_base.e to generate base case files. Error in DSFAIL - cannot copy dsfailt.def file. Change $TPA_TEST and $TPA_DATA to .../scr377 in .cshrc file. Also, version 4.2i does not have the correct tpanames.dbs. Saved nefii.inp and NEFII.VEL files in subdirectory base. Other output files were not created.

Modified nefii.inp so that 0's appear in velocity column. Copied codes/nefmks.e to run directory and re-ran tpa.e. Made nefii.out which was moved to vel0 subdirectory. But can't compare w/ base case because base case didn't make nefii.out.

Ron was not back from travel yet today.

8/19/2002 -

No problem logging onto Spock this morning. However, still get the error message from Word Perfect about the missing OLE prefix package. I'm working with a new copy of the

scientific notebook, but this does not appear to be a self-mending problem. I split the file into ESN_part1.wpd and ESN_part2.wpd, but I still get the message for both parts. I think it may have something to do with the links to Excel spreadsheets.

SCR # 377:

AAPseq & AAPmu are stored for each subarea in uz_climr.i Also use uz_parms.i. (Other related include files are uz_climz.i, uz_flowi.i, uzflowr.i and uzflowz.i.) Need a version of the glacial fractions read in from climato2.dat that are stored according to tpa time steps.

Who calls qclimato? No one at this time.

Ron believes that we can get by using values from AAPseq and dividing by the value in AAPseq(1). Have to check with James Winterle, who is out today. The glacial fraction values are read in as a set from climato2.dat and start out unitless. The precipitation value is converted to mm/yr by

sets(2,m1) = AAP0 * (done + (AAPmaxmul - done)) * sets(2,m1)

AAP0 is set to a constant of 162.8. done = 1.0d0. Don't know the significance of the number 162.8. AAPmaxmul is MAPMAGM, which is read in from tpa.inp and is a uniform value between 1.5 and 2.5. Once converted, the values are stored in AAPmu. AAPmu is converted in AAPseq by

AAPseq(i) = AAPmu(i) + Ppert * sdevAAP

Standard deviation is read in from tpa.inp and is currently set to 0. Perturbation is a calculated value based on values read from climato1.dat. To have the values as the original glacial fractions, without any conversions, we would have to save the sets read in from climato2.dat into another array and then convert it to tpa time steps (AAPglacfrac ?). This would have to be added to uzflow.f. Another option is to include uz_climr.i in szft and do some reverse conversion to get back to the unitless condition. My last choice would be to re-open climato2.dat in szft and read it all in again.

Flux is the flowrate of the selected streamtube in szft.f. Use mobile porosity for the sz mixing zone (pormix ?). Need valid range for new parameter SZFluxMultiplierAtGlacialMaximum - will use 0.0 to 1.0 until get a better set of numbers from Winterle. SZMinimumVelocityChangeFactor will be a constant of .1 for now. Where should it fit in the equation? Optional correlation between SZFMAGM and MAPMAGM (MeanAverage

PrecipitationMultiplierAtGlacialMaximum) will be added at the end of the tpa.inp with other correlations. Need a value (Winterle). This can be done last minute as it does not require code changes.

Still working with equation

velocity = (flux * (1 + cl2 * (SZFMAGM -1))) / mobile porosity or

maybe velmix = (flowrate * (1 + AAPglacfrac(j) * (SZFMAGM - 1))) / pormix and

maybe nefvel(i) = (flowrate * (1 + AAPglacfrac(j) * (SZFMAGM - 1)))/por(i)

Need to figure out how SZMVCF fits in. Will need to create new arrays for diffusion, dispersion & gwtt - maybe others, too.

Still plan to use writevelocities to create NEFII.VEL from szft.f. Instead of a repleg, will need a mixing zone leg (mixleg ?) Need to write out as many values as currently are written to neffisz.inp at a minimum. Two equations of interest from szft:

nefvel(inc) = flowrate/por(i)/(sawidth * widthmult) and

velmix = flowrate/pormix/widthmix

Velmix is written out where repvel is in uzft.f Nefvel is written out the same to nefii.inp for both uzft.f and szft.f. Neflegs is calculated the same way, too. Find where UZMVCF (UnsaturatedZoneMinimum
VelocityChangeFactor) is used in uzft.f and mimic in szft.f. Determine how legs are currently selected in szft.f.

Question: In climato2.dat, if the temperature and precipitation glacial fractions are always the same, why not have just one column of fractions, instead of two?

8/20/2002 -

Ron and I talked with James Winterle. SZFluxMultiplierAtGlacialMaximum[] should be uniform ranging from 1.0 to 3.0. He says we should just use the fractions from climato2.dat as they are. So I will go into uzflow.f and save those values a new array AAPglacfrac stored in the common /climater/ in uz_climr.i. That means changes, so far, to tpa.inp, szft.f, uzflow.f and uz_flowr.i. The remaining decisions are about where to put the time step loop in prenefmksa in

szft.f and how many new arrays I have to make and fill. Uzft.f has changed the dimensions on nefvel and other arrays (neflen, moistc, nefrd, and nefrdi) and I need to determine if I have to follow suit in szft.f. One dimension of the arrays used to be the number of layers but was replaced by the PARAMETER MAXNEF which is MAXLYR*2+1.The array NEFRDI is declared REAL*8 but is never dimensioned or accessed.

Found some code in uzft.f that looks like it could be condensed. There are several do loops back to back that are all looping from 1 through all the layers. There is a statement 499 CONTINUE that can be deleted. Szft has a loop "do 1, npart" but npart is hard coded to 1, so it only does the loop once. The major write section for nefii.inp is INSIDE this loop but probably shouldn't be. I told Ron I'd write up the possible changes to the routines. Might be able to do szft.f along with scr #379. Both szft.f and uzft.f have a type on the write statements that create nefii.inp.

In uzft.f/prenefmks, there are two versions of time steps using ntim (500-yr time steps) and ntimg (tpa time steps). The arrays gwtt, avgwtt, vell, tvel, and vtim are dimensioned using 500-yr time steps. The arrays nefvel & neflen are dimensioned using MAXNEF, which is based on number of layers. Do I need to set up the 500-yr time steps in szft.f? Do I need an array for the mixing zone leg like repvel, which is dimensioned using max500yr or will a single value be OK? Do I need versions of AAPglacfrac for tpa time steps and 500-yr time steps both? Ron says to pattern szft.f on uzft.f so that the file gwttuzsz.res is using the same basis for sz values as for uz.

EN_css202.wpd was divided into three sections: ESN_part1.wpd, ESN_part2.wpd, and ESN_part3.ESN. ENa.wpd was divided into ESN_part1.wpd and ESN_part2a.wpd. ESN_part2a.wpd was later divided into ESN_part2b.wpd and ESN_part3a.wpd. In part three, I managed to get rid of the OLE prefix message. The other two parts still have some error, apparently related to the graphs and perhaps tied to a link between this document and the Excel files that was corrupted last week when we had the computer problems. I'll have to look at the files again another day.

8/21/2002 -

More work on SCR #377.

Made modifications to uzflow (uzflow_mod377.f) in get_climean subroutine. Added array tempsets to save the values read in from climato2.dat. Declared new array AAPglacfrac to hold the values read from climato2.dat. Used tempsets to interpolate values for AAPglacfrac

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Made modifications to szft (szft_mod377.f) in prenefmksa. Added new variables or redimensioned existing one to implement time-dependent velocity feature. Gwtt, nefvel, and neflen need to change. Will need to return tavgwtt to szft instead of gwtt, which is now an intermediate array to be used the same way as in uzft.f. Should be able to use vell, tvel, & vel as used in uzft.f to calculate time-dependent velocity and then set nefvel from these new arrays. Question about streamtube width multiplier: New equation does not use sawidth. Should we? How? Flowrate comes from strmtube.dat as m^3/yr/m. Need to check with J. Winterle when he is back in the office Friday.

Potential problem?: mnCliTS = 1001; max500yr = 2001; maxntime = 401; ntim is set in reader.f; actual climate steps in climato2.dat is 101 (counting 0 time) 1000-yr time steps which is converted to 500-yr time steps in get_climean, so size is doubled. Use ntim to dimension arrays in szft.f:prenefmksa instead of max500yr.

Made copy of uzft (uzft_mod377.f) in case mods needed there. Why does uzft.f use MAXNEF instead of maxlyr? Do I need to do something similar in szft.f? Also corrected some typos in case we get to do mods here.

Made modifications to tpa (tpa_mod377.inp). Added (after StreamTubeWidthMultiplier[]):

```
constant
SaturatedZoneMinimumChangeVelocityFactor[Fraction]
0.1

uniform
SZFluxMultiplierAtGlacialMaximum[]
1.0, 3.0
```

Values for parameters came from J. Winterle and/or R. Fedors & D. Farrell.

Made modifications to uz_climr.i in COMMON climater. Declared REAL*8 AAPglacfrac(mnCliTS) and added AAPglacfrac to end of common block.

Ron J. says he has comments on the test plan for SCR #379. We plan to talk tomorrow afternoon.

8/22/2002 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Worked on mods to szft.f. Made new subroutine called calc_velocities which, currently, calculates the velocities and ground water travel times over the current layer and returns these values to prenefmksa for use in other calculations and to be written out. Not planning to change diffusion or dispersion calculations at this time.

In the header block of subroutine prenefmksa in szft.f, it says that gwtt is an average, like uzft, but it looks like a total to me.

Met w/ Ron Janetzke re: SCR #379. He wants some new checks on existing tests plus two new tests. There are also markups on the printed copy to take care of. The most concern was the direction of some of the trends in the results.

The SSH Shell window where I was editing szft_mod377.f hung (recurring problem) and I lost some of the mods I had made to the file. Will need to figure out what changes got lost and redo them.

8/23/2002 -

Re-checked trends in the output for SCR #379. Results seem to be correct. Did most of the corrections redlined by Ron, except for those requiring a new or modified test. Began working on adding new tests to the test plan. New/modified tests requested:

FLT #1: check dilvol = uservol and pluvdilvol = userpluvdil

FLT #2: add verbiage that setting the UserDefinedPlumeCaptureFraction has no effect when the PlumeCapture Model is set to 2. Also, print out values for pumprate and pluvpump to allow hand calculations. Check max of (pump,uzq,plume).

SLT #1: Check max of (pump,uzq,plume) if don't already.

SLT #2: Check all files if already don't.

SLT #3: Check gs = more vs. 20 km & dilvol = userdilvol, if already don't.

SLT #5: Add a base case run (old version) for comparison w/ dilflg = UserDefined and UserDilVol = to 1.2098e+07, which is what the calculated dilution volume comes out to, instead of 1.5073e+07. See if two runs are the same. See how the two test runs differ w/ different dilution volumes.

SLT #6: Check w/ old version w/ same pumping rate.

SLT #7: Compare w/ old version with same well pump & plume cap.

SLT #8: Compare w/ old version w/ same plume thickness.

SLT #9: NEW. DistToRG = 18. Check NEFTRAN leglen and dilution vol. Results should be higher does than SLT #2.

SLT #10: NEW. Plume cap = 1; user plume = 1; dil mod = 4. Compare w/ plume cap = 1; user plume = 0.5l dil mod = 4.

SLT #11: NEW. Dilflg = 2 (pump); well pump = 3.4 X 10^7; pluv well pump = 1.2 X 10^6. Check dilvol or use print statements to check dilvol changes at pluvial switch time.

Still having problems with the SSH Shell windows accessing spock. They will lock up on me so I can't continue the session. The only way out seems to be to close the window which will disconnect the session, but anything I was working on in the window will be lost.

8/26/2002 -

Gwttuzsz.res is written out from exec.f. Need to look at all variables used to write it and make sure the information from szft.f gets passed back. Maybe it is already taken care of automatically. I'll have to check the exec.f code more closely.

I spoke to James Winterle about the streamtube width multiplier question. Our new equation does not divide by width, so there is no value with which to multiply the streamtube width multiplier. He will check on it and get back to me. He thinks we may need to include the division by width * widthmultiplier to get the real velocity. Yes, we do need to divide the existing new equation by (streamtube width * streamtube width multiplier).

Put the changes to szft:prenefmksa and calc_velocity as I remember them before the computer problem. Have printed current version so I can go through it and check that all variables are declared and initialized where needed and to follow the logic through. One problem left with comparing current velocity calculation with gwttmin and 10 percent of total average gwtt. Tavgwwt is not calculated until return from calc_velocity and so is not available within calc_velocity.

8/27/2002 -

Continued with changes to szft.f for SCR #377. Moved reading of StreamTubeWidthMultiplier[] to calc_vel_gwtt. Need to resolve questions about total gwtt vs. total average gwtt and whether or not szft needs to do the extra averaging for the compliance period. Also, should szft have the gwttmin check? Ron says yes. Will need to restructure how I'm doing the mod.

Proofread tp_scr379.wpd; ready for next pass.

8/28/2002 -

Took new velocity calculation subroutine and broke it into two parts: calc_vel_gwtt and setvelfile. Calc_vel_gwtt calculates time-dependent velocity (vell) and gwtt per time step and layer. Average gwtt per layer is passed back to prenefmksa where total average gwtt is calculated. Tavgwtt and vell are passed into setvelfile and used to fill the tvel array which, in turn, is used to generate the NEFII.VEL file for input into NEFTRANS.

Modified exec.f to pass ntim_c (used to determine itime_c) to szft.f. Added ntim_c to parameter lists for szft and prenefmksa. Itime_c is set in prenefmksa and passed to calc_vel_gwtt.

Modified prenefmksa according to the pattern in uzft:prenefmks: changed ntim to ntimg and tim to timg. Calculated new ntim based on 500-yr time steps instead of geometric time steps. New ntim and ntimg are passed to calc_vel_gwtt and setvelfile. Data from AAPglacfrac already in 500-yr time steps.

Have gone through printout of current szft_mod377.f and compared with uzft.f to see if it looks like I've covered all the bases.

Szft returns gwtt to exec - should it be changed to tavgwtt? Remember, still need to look at how gwttuzsz.res is written out and whether all the correct information is being passed back to exec. Should also check again that all variables are declared correctly and passed correctly.

8/29/2002 -

Modified szft_mod377.f according to last night's markups. Re-checked variable declarations and passed parameters. Returned total average gwtt from prenefmksa instead of what gwtt contains now.

It looks like no mods will be required in exec.f for generating the output file gwttuzsz.res.

Szft.f contains comment 'jmm LEFT OFF HERE!'. Asked Marty if he remembers why he put this comment here once before - need to ask again.

Compiled tpa.e w/ newly modified routines and files: exec_mod377.f, uzflow_mod377.f, szft_mod377.f, uzft_mod377.f, and uz_climr_mod377.i.

Tried to run TPA. However, run bombs because of an error in an isotope name? This isn't near the code I changed, so not sure what's happening. I printed out diff files on all the files I changed, but don't see anything obvious. Put in print statements, recompiled and reran TPA several times. Looked at results in tpa_test1.out to try to figure out what's wrong. The chainlist array is getting trashed somehow. It works OK in uzft but not in szft. Getelements is called which calls indexperiso where TPA bombs when getelements is called from szft.

8/30/2002 -

Tried the base case run again, just in case. But it works OK. So somehow my changes seem to be trashing something. Eureka! When ntim_c was added to parameters for prenefmksa, the line was slightly too long, and that trashed the names parameter.

THERE SEEMS TO BE DUPLICATE CODE IN SZFT and UZFT REGARDING INVENTORY INFORMATION.

Then got bus error. Probably a parameter or array problem.

SCR #396: made uzft_mod396.f and moved the lowestmatfraction initialization statement inside the time step loop but still outside the layer loop.

8/31/2002 - 9/2/2002 -

Ran more test runs. Fixed bus i/o error. Fixed a couple of indexing bugs. Found bug associated with calculating avgwtt. The values seem to get lost somewhere.

Keep seeing warnings about incompatible lengths for common ccorol in releaset.f while making executable. Told Ron. He's already seen it.

9/3/2002 -

Avgwtt problem due to misspelled parameter name: avgwwt instead of avgwtt.

Running w/ only 2 subareas for a while - output file way too big otherwise.

Found another index bug in setvelfile - itimestep instead of itime.

Moved call writevelocities from subroutine setvelfile up to prenefmksa after call writesource; it was being called more than once which was incorrect.

Not getting output files because there is no significant release after prenefmksa. Ran w/ MaximumTime set to 100,000 yrs, but it aborted in NEFTRAN. NEFII.VEL is not correct. However, there was a release so it got further than at 10,000 yrs.

The values for vell when layer is 2 print out within calc_vel_gwtt but not after the return to prenefmksa.

9/4/2002 -

Seem to have fixed problem with the missing values from vell - just not absolutely sure how.

Tpa_test2.e now runs and produces a viable NEFII.VEL at 10,000 yrs. No differences between base and test2 nefii.inp except for IF to OF and the NEFII.VEL flag. Doesn't call NEFTRAN though. Test3 runs for 100,000 yrs. Still using subarea 4, but NEFTRAN wasn't called again. Try w/ subareas 1 - 2 (test4). Test4 crashes NEFTRAN!

Problem: in setvelfile, inc should end up the same number as inc (& neflegs) in prenefmksa, but it isn't. May be related to my do loops in setvelfile. Need to check w/ uzft and see which loop(s) have layers as the outer loop and time steps as the inner and vice versa. Also, may not need to return numsteps and numlegs from setvelfile. Corrected do loops - need 3, not 1, layer loops.

Re-ran w/ new code and just subarea 4 for 10,000 yrs. NEFII.VEL looks correct except for some 0 velocities. Maybe failed gwttmin test - no else, so would default to 0. Saved in test2. No release so NEFTRAN not called.

9/5/2002 -

Fixed problem w/ tvel values = 0. Ran test2 (subarea 4 for 10,000 yrs.) and stored results. NEFII.VEL looks OK.

Re-ran w/ new code and subareas 1-2 for 10,000 yrs. (test4). Looks good.

Re-ran w/ subareas 1-2 for 100,000 yrs. (test5). It ran and called NEFTRAN w/o bombing. Results look good.

Re-ran w/ SZMinimumVelocityChangeFactor set to 0.9 instead of 0.1. Works! YAY!!!!

Made clean copies of modified files for Ron - put in subdirectory mods. Look at if test in uzft and szft where avgwtt is compared to gwttmin and tavgwtt. It looks like there may be a problem if the test fails for all layers - no value would be stored for velocity and it would contain either garbage or default init value. Ron says take a day to test this. Then Ron may have another coding task for me. Ran diff on the original source code files and modified files. Put diff output files (*.dif) in subdirectory diffs. Second question: why is the test for all layers and not just current one?

Re-compiled tpa w/ modified source files and re-ran using tpa_test5.inp. Checked out OK.

9/6/2002 -

Inserted code into uzft.f (uzft_gwttmin.f) and szft.f (szft_gwttmin.f) to reset gwttmin for some time steps to a high enough value that the gwtt test would fail for all layers. This resulted in velocities in tvel that were defaulted to 0.0. Those velocities were written out to NEFII.VEL and caused NEFTRAN to abort. Results stored in subdirectory testnef.

Because of the possibility of zero velocities when all gwtt tests fail for all layers, added code to szft & uzft to keep track of gwtt test results. Logical failedgwtttest is initialized to true and reset to false if any layer passes. If failedgwtttest remains true then the itm counter will be decremented and the time step will be tossed. Also added check for zero index when resetting the last time step in tvel to etime.

Updated SCR 377 with the meeting results from July and code changes. Printed it out, signed it and turned it in to Ron. Ron copied the six modified files from scr377/mods. The files are exec.f, szft.f, uzflow.f, uzft.f, uz_climr.i, and tpa.inp.

For now, working on finishing the mods to the test plan for SCR 379. Need to run them and make CD for Ron before that one is finished. There may be more coding next week. There will definitely be more testing to do.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

9/9/2002 -

Went back to working on the test plan for SCR #379. Made modifications of redlines from my last pass through the document. Ron wanted a dilution volume test added as part of SLT #5 where a run of the base case (DilutionModel = 0, still one subarea at 100,000 yrs) was made. Dilution volume shows up in the output file totdose.res. A user defined input for dilution volume is converted from gal/day to m^3/yr in dcagw.f using the formula dilution volume (m^3/yr) = dilution volume (gal/day) * 365.25 / 264.172. Taking the value of dilutvol (m^3/yr) from totdose.res (1.0379d+07) and converting (dilutvol * 264.172 / 365.25) back to gal/day (7,506,751.9d0), this value was then used to set UserDefinedDilutionVolume[gal/day] in tpa_slt5b.inp. Tpa_mod379.e was then run using tpa_slt5b.inp. The expected result was that dilutvol in totdose_slt5b.res would match the dilutvol in totdose_baseslt5.res. They did. However, the Total Mean Dose and TEDE do not. Will check w/ Ron to make sure that the differences in dose are also expected.

Ron said to change the user defined pluvial purmping rate, too. But that didn't change the outcome.

Added tests SLT # 9 - #11. Started running the new tests and collecting data.

9/10/2002 -

Finished running tests SLT #9 - #11. SLT #11 did not pass; no changes in dilutvol show up for the pluvial time switch. Turns out it shouldn't because dilutionvolume which is passed back to exec from dcagw is a single value; it should be a time-dependent array. Generated SCR #399 to request change. Deleted test 11 from the test plan. Still need to rerun FLT #2 and SLT #5. Then I should be done with the test plan.

Reran SLT # 5. Same results. For Ron's added test, I can generate the same dilutvol in the test case as in the base case, but the tedes and Peak Mean Doses are different. After talking to Ron, he decided that the results are correct. All we're interested in is the dilutvols.

Reran FLT #2 and regenerated correct results for the test plan.

9/11/2002 -

Made final (I hope) changes to the test plan for SCR #379. Printed copy and gave it to Ron. Still need to make a CD for it.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Ron wants me to find everywhere routines reference the fact the number of nuclides is 43, except for invent.f and reader.f. I will need to find everywhere that functions/subroutines in inventory are referenced externally. Using grep to locate hard-coded uses of 43 as well as calls to the functions and subroutines from invent.f. Also looking at releaset.f and ebsfilt.f from the codes subdirectory. There may be references in nefmks.f as well. There are no problems with .i files and there are no calls or uses of subroutines & functions outside of the standalone source code files. Started working through the list of references to see which ones can be discarded or if I can discover other related uses.

Problem noticed w/ TPA versions p & q: IEEE exception flags being raised (inexact and underflow). Can't find a problem with the code for SCR #379 (version q) and SCR #377 (version p). Worked with Ron re-running earlier versions; it comes up somewhere version m or earlier. Ron says that Perry installed a new compiler last week and we may have a problem with one of the linked libraries. Ron plans to table a fix until later; there's not enough time to track it down now.

9/12/2002 -

Finished looking for hardcoded references to the number of nuclides as 43 and calls & invocations to subroutines & functions contained in invent.f. Used grep to generate listings and concatenated the listings together. Then I went in and checked the code, deleting lines that didn't apply to nuclides. One question: pertaining to the comment in maxnnucl.i regarding AQUEOUS PATHWAYs: are there two different maximum number of nuclides depending on which subroutines you're in?

Turned in a copy of the report to Ron. Report is as follows:

USE OF 43 FOR NUMBER OF NUCLIDES IN TPA & STANDALONE CODE

hardcoded references to 43 in source code files (except invent.f & reader.f):

in subdirectory codes/

EBSFILT

84 parameter (maxnnucl = 43)
used to dimension several arrays

NEFMKS

```
443     cc rwj 10-10-97 Increase FN to handle 43 chains, when generating DISCHARGE file
444     CHARACTER FN(43)*9, TITLE*80, TTLCEN*80, DUM*1, DLST(NNINP)*20

456     cc rwj 10-10-97 Increase FN to handle 43 chains, when generating DISCHARGE file
468     &           'Chain41', 'Chain42', 'Chain43'
```

in TPA source:

ASHRMOVO

```
parameter(maxnnucsvol = 43 )

dkd(43) = valuesp( ikdcarbon )

sol(43) = valuesp( isolcarbon )

do i=1,43

do idoloop=1,43

do j=1,43

Must do a call to decayremove43mol to get the radionuclide inventory
call decay43mol( dt, tempdnt1(1) )

do n=1,43

do m=1,43

call decayremove43mol( dt, dlt, tempdnt1, tempdn )

do j=1,43

do j=1,43
```

do inuc=1,43

do i=1,43

DCAGS

```
138     dimension dcfde(43),dcfinh(43),dcfingp(43),dcfinga(43)
140     double precision dcfmlk(43)
141     dimension resuspendablefraction(maxntime),dcf(maxntime,43)
```

do k=1,43

do k=1,43

do k=1,43

do k=1,43

do k=1,43

```
c       REAL DCFEXT(43), DCFINH(43), DCFANI(43), DCFPLA(43), DCFDW(43),
```

```
REAL DCFEXT(43), DCFINH(43), DCFANI(43), DCFPLA(43),
&       DCFMILK(43)
```

do i=1,43

I = 43

DO 722 I = 1, 43

DO 732 I = 1, 43

DCAGW

dimension dcfde(43),dcfinh(43),dcfinga(43),dcfingp(43),dcfdw(43)

```
dimension dcfmlk(43)

dimension dwdcf(mxntime,43)

cc  dimension rnconc(ntim, 43)
dimension rnconc(maxntime, 43)

common / dcagw2 / dcf(43)

character*6 nucname(43)

common / dcagw29 / dosefactor(43,8)

dimension epadose(maxntime,43,8)

if( (nnucl .le. 0) .or. (nnucl .gt. 43) ) then

print *, ' (nnucl .le. 0)  .or. (nnucl .gt. 43)'

do iorgandf = 1,43

do k=1,43

do k=1,43

do k=1,43

do k=1,43

do ifind = 1,43

do ifind = 1,43

REAL DCFEXT(43), DCFINH(43), DCFANI(43), DCFPLA(43), DCFDW(43),
 &      DCFMILK(43)

do i=1,43

ELSE IF (RADNAME(IN) .EQ. "CM243" .OR. RADNAME(IN) .EQ.
```

```
 &          "PU243") THEN

ELSE IF (RADNAME(IN) .EQ. "AM243" .OR. RADNAME(IN) .EQ.

I = 43

DO 722 I = 1, 43

DO 732 I = 1, 43


EBSREL

dimension ammtu(43)

call decay43mol( rzero, ammtu)

call decay43molglass( rzero, ammtu)


EXEC

dimension ciperm2gsatCP(maxntime, 43 )

dimension remperyrgsnr(maxntime, 43 )

dimension remperyrgsna(maxntime, 43 )

dimension ipt(43)

dimension temp(43)

dimension gsnmaxdose(maxrealizations,43)

dimension gsnmaxdose_c(maxrealizations,43)

dimension dwdcf(maxntime,43)

dimension amolepermtuejected(43)
```

```
external decay43mol

columnheader2(20:43) = 'conc(pCi/L)        time'

ilength = (43+3) * 16

do i = 1,43+3

&  (columnheader1(1:((43+3)*16)))

&  (columnheader1(1:((43+3)*16)))

ilength = (43+3) * 16

do i = 1,43+3

&  (columnheader1(1:((43+3)*16)))

&  (columnheader1(1:((43+3)*16)))

do ik = 1, 43

do ik = 1, 43

call zero( maxntime*43,      remperyrgsna(1,1) )

call zero( maxntime*43, remperyrgsnr(1,1) )

call zero( maxntime*43, remperyrgsnr(1,1)  )

 &          (namesisotopesave(j),j=41,43)

 &           (ciperm2gsatCP(j,k),k=41,43)

do i = 1, 43

&          (namesisotopesave(j),j=41,43)

&           (ciperm2gsatCP(j,k),k=41,43)
```

```
c          call dcags( maxntime, 43,
call dcags( ir, maxntime, 43,

 &       ntim, tim(1), 43, namesisotopesave(1),

 &          (namesisotopesave(j),j=41,43)

 &           (remperyrgsnr(j,k),k=41,43)

do inuc = 1, 43

do ijk = 1, 43

call sortqr( 43, temp(1), ipt(1) )

cutoff = temp( ipt( MAX( 1, 43-5 ) ) )

if ( temp( ipt(43) ) .eq. 0.0d0) then

do inuc = 43,39,-1

print *, '    The remaining ',43-inonzerocount,' '

do inuc = 43,1,-1

call writedata( irgsnr, maxntime, ntim, 43, tim(1),

call addto( maxntime*43,

do ijk=1, 43

do inuclide = 1, 43

do inuclide = 1, 43

call decay43mol(voltimeofevent, amolepermtuejected)

do k = 1, 43
```

```
do k = 1, 43

write( formatstr,'(a,i3,a)') '(5x,i5,6x,', (43+3),

 &      (gsnmaxdose(ireal,j), j=1,43)

 &      (gsnmaxdose_c(ireal,j), j=1,43)

 &         maxntime*43, remperyrgsna(1,1) )

call writedata( irgsna, maxntime, ntim, 43, tim(1),

write( iunit, 343 ) m2

common / wrtccdf2 / iown(43)

common / wrtccdf3 / imv(43)
```

PEAKFIND

```
common / peak9  / iownpeakdosenv(43)

common / peak10 / iownpeakdosent(43)

common / peak11 / ipeakdosenv(43)

common / peak12 / ipeakdosent(43)

if( (nnucl .le. 0) .or. (nnucl .gt. 43) ) then

print *, ' (nnucl .le. 0)  .or. (nnucl .gt. 43)'
```

SZFT

```
parameter (maxpth=128,maxlyr=15,maxprt=13,maxiso=43,maxchn=43)

character*6 name(43)
```

```
character*6 namelj(43)


UZFT

c          the 43(=inucl) nuclides

c          the 43(=inucl) nuclides

PARAMETER (MAXPTH=128,MAXLYR=15,MAXPRT=13,MAXISO=43,MAXCHN=43,

character*6 name(43)

character*6 namelj(43)

c          the 43(=inucl) nuclides

dimension citemp(43)

character*6 nuclidename(43)


hardcoded references to 43 in include files (except invent*.i):

maxchain.i:     parameter (maxnumchains=43)

maxnnucl.i:cc NOT 43 being tracked in INVENT

maxnnucl.i:     parameter (maxnnucl      =  43 )


invent.f functions:

c                function cipermtuat10periso is no longer used,
    double precision function dmyexp(x)
    integer function indexperiso( name )
    Integer function indexperisonostop( name )
    character*2 function nameelem( index )
```

```
character*6 function nameiso( index )
function epalimperiso( index )
function activityperiso( index )
function wmoleperiso( index )
function halflifeperiso( index )
function alambdaperiso( index )
function cipermtuat10periso( index )
function amolepermtuatemplaceperiso( index )
function cipermtuatemplaceperiso( index )
function qpermtu( time )
```

invent.f subroutines:

```
subroutine setage( timeatemplacement )
subroutine decay43mol( time, amolepermtu )
subroutine decayremove43mol( dt, dlremove, dnt1, dnt2 )
subroutine chains(maxnt, nt, time, dlt, dlr, dnt1, dn )
subroutine chainsolver(len, maxnt, nt, tim, dnt1, dlt, dlr, dn)
subroutine newinventdb()
subroutine decay43molglass( time, amolepermtuglass )
```

1) grep dmyexp *.f

```
condxyzt.f:     external dmyexp

condxyzt.f:    &  ( dmyexp( - z**2 / ( 4.0d0 * alpha * (tend-t) ) )  -

condxyzt.f:    &    dmyexp( - (z-2.0 * aH)**2/ ( 4.0d0 * alpha * (tend-t) ) )  )


invent.f:        dn(j,i) = dnt1(i) * dmyexp(-dlt(i) * (time(j) -time(1)))
invent.f:        dn(it,i) = dn(it,i) + dnn / den * dmyexp(-dlt(j) * time(it))
invent.f:     double precision function dmyexp(x)
invent.f:c dmyexp      = output, double precision, output value
invent.f:       dmyexp = 0.d0
invent.f:       print *, ' dmyexp: WARNING: large exponential has been'
invent.f:       dmyexp = 1.d0 + 99
```

```
invent.f:       dmyexp = dexp(x)


 2)  grep indexperiso *.f

dcags.f:     external indexperiso

dcags.f:        ik = indexperiso( namesdr(k) )

dcags.f:       ik = indexperiso( namesdr(k) )


dcagw.f:        ik = indexperiso( names(k) )

dcagw.f:        ik = indexperiso( names(k) )


ebsrel.f:     external indexperiso

ebsrel.f:        indexset(in) = indexperiso( names(in) )

ebsrel.f:       itempindex = indexperiso(names(i))

ebsrel.f:       itempindex = indexperiso(names(i))

ebsrel.f:       idex = indexperiso( nameisoebspac )

ebsrel.f:       idextpa =  indexperiso( nametemp )


exec.f:           iindex = indexperiso( names(itotaldose2) )

exec.f:           iindex = indexperiso( names(itotaldose2) )

exec.f:         index = indexperiso( names(k) )

exec.f:         index = indexperiso( namesisotopesave(k) )

exec.f:         index = indexperiso( names(k) )
```

```
exec.f:        index = indexperiso( namesisotopesave(k) )

exec.f:     external indexperiso

exec.f:       index = indexperiso( names(k) )


invent.f:     integer function indexperiso( name )
invent.f:c  indexperiso          = output, integer, index used to identify isotope
invent.f:        indexperiso = i
invent.f:     print *, ' ***>>> Error in indexperiso <<<***'
invent.f:     integer function indexperisonostop( name )
invent.f:c  indexperiso          = output, integer, index used to identify isotope
invent.f:c              indexperisonostop = 0
invent.f:        indexperisonostop = i
invent.f:     indexperisonostop = 0


reader.f:     external indexperiso
reader.f:     external indexperisonostop
reader.f:       ind43( 1) =  indexperiso('U238  ')
reader.f:       ind43( 2) =  indexperiso('Cm246 ')
reader.f:       ind43( 3) =  indexperiso('Pu242 ')
reader.f:       ind43( 4) =  indexperiso('Am242m')
reader.f:       ind43( 5) =  indexperiso('Pu238 ')
reader.f:       ind43( 6) =  indexperiso('U234  ')
reader.f:       ind43( 7) =  indexperiso('Th230 ')
reader.f:       ind43( 8) =  indexperiso('Ra226 ')
reader.f:       ind43( 9) =  indexperiso('Pb210 ')
reader.f:       ind43(10) =  indexperiso('Cm243 ')
reader.f:       ind43(11) =  indexperiso('Am243 ')
reader.f:       ind43(12) =  indexperiso('Pu239 ')
reader.f:       ind43(13) =  indexperiso('U235  ')
reader.f:       ind43(14) =  indexperiso('Pa231 ')
reader.f:       ind43(15) =  indexperiso('Ac227 ')
reader.f:       ind43(16) =  indexperiso('Cm245 ')
reader.f:       ind43(17) =  indexperiso('Pu241 ')
reader.f:       ind43(18) =  indexperiso('Am241 ')
reader.f:       ind43(19) =  indexperiso('Np237 ')
```

```
reader.f:      ind43(20) = indexperiso('U233  ')
reader.f:      ind43(21) = indexperiso('Th229 ')
reader.f:      ind43(22) = indexperiso('Cm244 ')
reader.f:      ind43(23) = indexperiso('Pu240 ')
reader.f:      ind43(24) = indexperiso('U236  ')
reader.f:      ind43(25) = indexperiso('U232  ')
reader.f:      ind43(26) = indexperiso('Sm151 ')
reader.f:      ind43(27) = indexperiso('Cs137 ')
reader.f:      ind43(28) = indexperiso('Cs135 ')
reader.f:      ind43(29) = indexperiso('I129  ')
reader.f:      ind43(30) = indexperiso('Sn126 ')
reader.f:      ind43(31) = indexperiso('Sn121m')
reader.f:      ind43(32) = indexperiso('Ag108m')
reader.f:      ind43(33) = indexperiso('Pd107 ')
reader.f:      ind43(34) = indexperiso('Tc99  ')
reader.f:      ind43(35) = indexperiso('Mo93  ')
reader.f:      ind43(36) = indexperiso('Nb94  ')
reader.f:      ind43(37) = indexperiso('Zr93  ')
reader.f:      ind43(38) = indexperiso('Sr90  ')
reader.f:      ind43(39) = indexperiso('Se79  ')
reader.f:      ind43(40) = indexperiso('Ni63  ')
reader.f:      ind43(41) = indexperiso('Ni59  ')
reader.f:      ind43(42) = indexperiso('Cl36  ')
reader.f:      ind43(43) = indexperiso('C14   ')
reader.f:        ijunk = indexperisonostop(names(k))
reader.f:          inames(kkcount) = indexperiso( names(kkcount) )
reader.f:        indexnames(i) = indexperiso(names(i))
reader.f:     external indexperiso
reader.f:     external indexperisonostop


szft.f:     external indexperiso

szft.f:     nucind = indexperiso(chainlist(nelm))


uzft.f:     external indexperiso

uzft.f:     nucind = indexperiso(chainlist(nelm))
```

```
uzft.f:     index = indexperiso(chainlist(1))

uzft.f:       index = indexperiso(chainlist(i))
```

3) grep indexperisonostop *.f

```
invent.f:     integer function indexperisonostop( name )
invent.f:c              indexperisonostop = 0
invent.f:        indexperisonostop = i
invent.f:     indexperisonostop = 0


reader.f:     external indexperisonostop
reader.f:        ijunk = indexperisonostop(names(k))
reader.f:     external indexperisonostop
```

4) grep nameelem *.f

```
invent.f:     character*2 function nameelem( index )
invent.f:c nameelem   = output, character*2, name of element
invent.f:        print *, ' ***>>> Error in nameelem <<<***'
invent.f:     nameelem = nameselemsave(index)


szft.f:     character*2 nameelem

szft.f:     external nameelem

szft.f:     elemnam = ljs2(nameelem(nucind))


uzft.f:     character*2 nameelem

uzft.f:     external nameelem

uzft.f:     elemnam = ljs2(nameelem(nucind))
```

```
uzft.f:     character*2 nameelem

uzft.f:     external nameelem

uzft.f:     symbol = nameelem(index)

uzft.f:       symbol = nameelem(index)
```

5) grep nameiso *.f

```
dcags.f:     character*6 nameiso

dcags.f:     external    nameiso

dcags.f:        WRITE(igsaddat,61) ljs(nameiso(i)),

dcags.f:         WRITE(igscidat,62) ljs(nameiso(i)), DCFEXT(I), DCFINH(I)

dcagw.f:     character*6 nameiso

dcagw.f:     external    nameiso

dcagw.f:        WRITE(igwaddat,61) ljs(nameiso(i)),

dcagw.f:         WRITE(igwcidat,62) ljs(nameiso(i)), DCFDW(I)

ebsrel.f:    character*6 nameisoebspac

ebsrel.f:    character*6 nameiso

ebsrel.f:    external nameiso

ebsrel.f:      read( iebsnefdat, fmt='(a6)' ) nameisoebspac

ebsrel.f:      idex = indexperiso( nameisoebspac )
```

```
ebsrel.f:      nametemp = nameiso( idex )

ebsrel.f:    &         nameisoebspac,tmaxfrac(is),rrmaxfrac(is)

ebsrel.f:      ljustify = ucljs(nameisoebspac)

ebsrel.f:     character*6 nameisoebspac

ebsrel.f:      read ( iebssfdat,    fmt='(a6)' ) nameisoebspac

ebsrel.f:      read ( iebsglassdat, fmt='(a6)' ) nameisoebspac

ebsrel.f:      write( iebsnefdat,   fmt='(a6)' ) nameisoebspac


invent.f:     character*6 function nameiso( index )
invent.f:c nameiso      = output, character*6, name of isotope
invent.f:        print *, ' ***>>> Error in nameiso <<<***'
invent.f:     nameiso = namesisotopesave(index)


reader.f:     character*6 nameiso
reader.f:     external nameiso
reader.f:     print *, ' which is the nuclide = ',nameiso(indexnames(mold))
reader.f:       print *, ' nuclide = ',nameiso(indexnames(ijk))
reader.f:       formatstr(1:6) = nameiso(indexallowedchains(ijk1,1))
reader.f:    &       nameiso(indexallowedchains(ijk1,ijk4))
reader.f:     print *, ' which is the nuclide = ',nameiso(indexnames(mold))
reader.f:         print *, ' nuclide = ',nameiso(indexnames(ijk))
reader.f:     print *, ' For ',nameiso(indexnames(mold))
reader.f:    &    nameiso(indexallowedchainmembers(indexnames(mold),ijk))
```

6) grep epalimperiso *.f

```
exec.f:     double precision epalimperiso

exec.f:     external epalimperiso
```

```
exec.f:        epalim = epalimperiso( index ) * amturep
exec.f:        epalim = epalimperiso( index ) * amturep
exec.f:        epalim = epalimperiso( index ) * amturep
exec.f:        epalim = epalimperiso( index ) * amturep

exec.f:     double precision epalimperiso

exec.f:     external epalimperiso

exec.f:       epalim = epalimperiso( index ) * amturep


invent.f:     function epalimperiso( index )
invent.f:c epalimperiso          = output, double precision, EPA release limit
invent.f:        print *, ' ***>>> Error in epalimperiso <<<***'
invent.f:     epalimperiso = epalim(index)


szft.f:     external epalimperiso

szft.f:     external epalimperiso

szft.f:     epawgt(nelm) = epalimperiso( nucind )


uzft.f:     external epalimperiso

uzft.f:     external epalimperiso

uzft.f:     epawgt(nelm) = epalimperiso(nucind)
```

7) grep activityperiso *.f

```
ebsrel.f:     double precision activityperiso

ebsrel.f:     external activityperiso

ebsrel.f:c    &          * activityperiso( itempindex)
```

```
ebsrel.f:    &          * activityperiso( itempindex)


exec.f:        gsnucrelease = amtuejected*activityperiso(index)*

exec.f:        gsnucrelease = amtuejected*activityperiso(index)*


invent.f:     function activityperiso( index )
invent.f:c activityperiso= output, double precision,
invent.f:        print *, ' ***>>> Error in activityperiso <<<***'
invent.f:     activityperiso = activity(index)
```

8) grep wmoleperiso *.f

```
ebsrel.f:     double precision wmoleperiso

ebsrel.f:     external wmoleperiso

ebsrel.f:       fmolwt = wmoleperiso( itempindex)


invent.f:     function wmoleperiso( index )
invent.f:c wmoleperiso          = output, double precision, molecular weight
invent.f:        print *, ' ***>>> Error in wmoleperiso <<<***'
invent.f:     wmoleperiso = wmole(index)


szft.f:     external wmoleperiso

szft.f:     amass(nelm) = wmoleperiso(nucind)


uzft.f:     external wmoleperiso

uzft.f:     amass(nelm) = wmoleperiso(nucind)
```

9) grep halflifeperiso *.f

```
ashrmovo.f:      external halflifeperiso

ashrmovo.f:        dlr(i)=0.6931472d0/halflifeperiso(i)


ebsrel.f:     double precision halflifeperiso

ebsrel.f:     external halflifeperiso

ebsrel.f:       halflife   = halflifeperiso(itempindex)

ebsrel.f:       halflife = halflifeperiso(itempindex)


invent.f:     function halflifeperiso( index )
invent.f:c halflifeperiso= output, double precision, halflife [yr]
invent.f:        print *, ' ***>>> Error in halflifeperiso <<<***'
invent.f:     halflifeperiso = halflife(index)


szft.f:     external halflifeperiso

szft.f:     hlife(nelm) = halflifeperiso(nucind)


uzft.f:     external halflifeperiso

uzft.f:     hlife(nelm) = halflifeperiso(nucind)
```

10) grep alambdaperiso *.f

```
invent.f:     function alambdaperiso( index )
invent.f:c alambdaperiso = output, double precision, transformation constant
invent.f:        print *, ' ***>>> Error in alambdaperiso <<<***'
invent.f:     alambdaperiso = alambda(index)
```

11) grep cipermtuat10periso *.f

```
invent.f:c                    function cipermtuat10periso is no longer used,
invent.f:     function cipermtuat10periso( index )
invent.f:c cipermtuat10periso = output, double precision,  reference
invent.f:        print *, ' ***>>> Error in cipermtuat10periso <<<***'
invent.f:     cipermtuat10periso = cipermtuat10yr(index)
```

12) grep amolepermtuatemplaceperiso *.f

```
invent.f:     function amolepermtuatemplaceperiso( index )
invent.f:c amolepermtuatemplaceperiso = output, double precision,  reference
invent.f:        print *, ' ***>>> Error in amolepermtuatemplaceperiso <<<***'
invent.f:     amolepermtuatemplaceperiso = amolepermtuatemplace(index)
```

13) grep cipermtuatemplaceperiso *.f

```
invent.f:     function cipermtuatemplaceperiso( index )
invent.f:c cipermtuatemplaceperiso = output, double precision,  reference
invent.f:        print *, ' ***>>> Error in cipermtuatemplaceperiso <<<***'
invent.f:     cipermtuatemplaceperiso = cipermtuatemplace(index)
```

14) grep qpermtu *.f

```
condxyzt.f:c also requires external function "qpermtu" which returns [W/MTU]

condxyzt.f:     external qpermtu

condxyzt.f:c also requires external function "qpermtu" which returns [W/MTU]

condxyzt.f:     external qpermtu

condxyzt.f:        qpp = qpermtu( t ) * (1.d0 - hloss_fact) * aml / 4047.0d0

condxyzt.f:        qpp = qpermtu( t ) *  aml / 4047.0d0
```

```
invent.f:     function qpermtu( time )
invent.f:c qpermtu      = output, double precision, heat output of waste
invent.f:       print *, ' ***>>> Error in qpermtu <<<*** '
invent.f:     qpermtu  = (pwrblend * qp + bwrblend * qb)


nfenv.f:     external qpermtu

nfenv.f:            qwp = qpermtu( tim(it) ) * wppayload * (1.d0-hloss_fact)

nfenv.f:            qwp = qpermtu( tim(it) ) * wppayload

nfenv.f:c        qheat[W/m2]  (qpermtu [W/MTU])(AML [MTU/acre])(acre/m2)

nfenv.f:       external qpermtu

nfenv.f:    &              / (qpermtu(tim(it1))*(1.d0-hloss_fact) )

nfenv.f:    &              / (qpermtu(tim(it1)) )


15)  grep  call setage  *.f

exec.f:       call setage( age )


16)  grep  call decay43mol  *.f

ashrmovo.f:       call decay43mol( dt, tempdnt1(1) )


ebsrel.f:       call decay43mol( rzero, ammtu)

ebsrel.f:       call decay43molglass( rzero, ammtu)


exec.f:       call decay43mol(voltimeofevent, amolepermtuejected)
```

17) grep call decayremove43mol *.f

```
ashrmovo.f:      call decayremove43mol( dt, dlt, tempdnt1, tempdn )


invent.f:    call decayremove43mol( time, dlremove, amolepermtuatemplace,
invent.f:    call decayremove43mol( time, dlremove, amolepermtuatemplaceglass,
```

18) grep call chains *.f *.f

```
invent.f:    call chains(maxnt, nt, time,
invent.f:    call chains(maxnt, nt, time,
invent.f:    call chains(maxnt, nt, time, dltotal, alambda, dnt1, dn )
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:      call chains(maxnt, nt, time,
invent.f:      call chains(maxnt, nt, time,
invent.f:    call chains(maxnt, nt, time,
invent.f:    call chains(maxnt, nt, time,
invent.f:    call chains(maxnt, nt, time, dltotal, alambda, dnt1, dn )
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:      call chains(maxnt, nt, time,
invent.f:      call chains(maxnt, nt, time,
```

19) grep call chainsolver *.f

```
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
```

```
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
invent.f:    call chainsolver(iknt,maxntime, nt,time,tempdnt1,
```

20) grep call newinventdb *.f

```
ashrmovo.f:      call newinventdb()

invent.f:    if (ikey .ne. 929318)  call newinventdb()
invent.f:       call newinventdb()
invent.f:       call newinventdb()
invent.f:       call newinventdb()
invent.f:        call newinventdb()
invent.f:       call newinventdb()
invent.f:       call newinventdb()
invent.f:        call newinventdb()
invent.f:       call newinventdb()
invent.f:        call newinventdb()
invent.f:        call newinventdb()
invent.f:        call newinventdb()
invent.f:        call newinventdb()
invent.f:        call newinventdb()
invent.f:        call newinventdb()
invent.f:       call newinventdb()
invent.f:       call newinventdb()
```

21) grep call decay43molglass *.f

```
ebsrel.f:      call decay43molglass( rzero, ammtu)
```

After I gave Ron the report, he asked me to redo the test plan for SCR #379. He said he'd had so much trouble merging in the code, that he wanted it retested. SCR #379 was merged in for tpa version 4.2q. So I copied the test report and am working on the Test Plan Version B for SCR #379 in tpatest/scr379B. I copied the current version (q) over to this subdirectory and made

another subdirectory called tpatest/scr379B/tpa42p - it contains version 4.2 p, of course. Version p will be the baseline and I'll test the differences in q against it.

For this test plan, things need to be set up somewhat differently. Since I'm testing the complete TPA code w/ the changes merged in against an earlier version of TPA, I'll need to have two complete sets of the code. So, I'll also need to change the values for $TPA_TEST and $TPA_DATA on the fly. I made a command file show_env that echoes the current values for the environment variables and another, ch_env, which uses setenv to change them. Show_env will execute, but you need to use the command source for ch_env. Only needed to switch environment variables for SLT #1, SLT #2, and SLT #5.

The copyfiles and diffslt* command files are still there from SCR #379. They will still work. Results from any runs in the tpa42p subdirectory will be copied up a level and put in subdirectories on the same level as the slt results.

Ran the two Functional Level Tests (FLT #1 and FLT #2) as well as the first two System Level Tests (SLT #1 and SLT #2). They all passed. Will work on the remaining 8 tomorrow. There is also a meeting scheduled with Ron Janetzke and the tpa programmers/testers at 1:30 in the conference room near Ron's office.

9/13/2002 -

Continued with the Test Plan Version B for SCR #379 (379B). Started with SLT #3. Finished all tests around noon. All tests passed. Then went on to make changes to the Test Plan for Version B.

The meeting was postponed until 2:00 p.m.

9/17/2002 -

Made the rest of the redline changes to the Version B Test Plan for SCR #379. After looking at the results from SLT # 9 and discussing them with Ron, decided to add another test (11). With the most uptodate code, the differences between 18 km and 20 km were too close, with greatest TEDE values fluctuating between the two distances. So added 11a and 11b at 15 and 12 km, respectively. NEFTRANS bombs for 11a. There are 3 legs in nefiisz.inp but only two legs show up per time step in NEFII.VEL (would be nefiisz.vel if code got that far.) This looks like bug in my code for SCR #377, not Ron's code for SCR #379, so I need to find it quickly. The 3rd leg in nefii.inp is short (57.5 meters).

Need to write out the values in vell and tvel, etc. as well as gwtt. Also need to proofread Version B Test Plan for 379 one more time.

9/18/2002 -

The problem with the discrepancy in number of legs between nefii.inp and NEFII.VEL was because, the way I added the time-dependent velocities for szft, tvel values are limited by several value checks but nefvel values aren't. In order to add these checks to nefvel, the code for calculating nefvel had to be moved down until after vell and gwtt are determined as well as avgwtt and tavgwtt. So, I moved the calculations for nefvel and neflen into setvelfile. This required passing in three more input parameters and returning two more output parameters (nefvel and neflen). The call to calc_vel_gwtt is moved up to the top of the do layer loop in prenefmksa. This solves the problem of there being a possibility that nefvel (for nefii.inp) and tvel (for NEFII.VEL) would have different numbers of legs.

Just noticed today that an IEEE flag for division by zero is being raised. Not sure if it's my code changes or not. Check tomorrow.

9/19/2002 -

Looked at my code changes in szft.f. If porosity or streamtube width is 0, there is a possibility of a divide by 0. Took care of it in my code changes, but the flag is still being raised. Porosity and width are input as sampled parameters in tpa.inp. Neither of these are being validated. (Streamtube width multiplier could have been a culprit, too, but it IS validated when it is read in.

Finished testing my "bug fix" for number of legs. Looks ready to go. However, there is also a problem with some of the flags being written out to nefii.inp. By comparing it with uzft, have found three flags that need to be changed, plus one flag change requires inputting sixteen new values. Ron wants me to generate a new SCR - #408 - that will include the number of legs problem as well as the flag changes and add validation checks for streamtube width (greater than 0.0) and porosity (range of 0.0 to 1.0, excluding end points). Also need to check what happens if number of time steps is 1. This could also cause a divide by zero if allowed. Number of time steps may be checked elsewhere; need to resolve.

Was able to run tpa.e w/ DistanceToReceptorGroup[km] set to 15.0 and get correct results. Ran slt9 and slt11 input files w/ patched code to get totdose.res files to compare to complete slt11 in Version B Test Plan for SCR #379. See following spreadsheet (/net/spock/home/cscherer/tpatest

/scr379B/slt11/totdose_all.xls) - seems to support statement that tede from totdose.res generally increases as DistanceToReceptorGroup[km] decreases. At 20 km and 18 km, there are three legs in the velocity arrays, while there are only two legs (with new checks) at 15km and 12 km.

			TOTDOSE	.RES			
		tede	tede	tede	tede		
	time	slt2-20 km	slt9-18km	slt11a-15km	slt11b-12km		dilutvol
1	8.3262e+02	0.0000e+00	0.0000e+00	4.0858e-07	4.8232e-07		1.2098e+07
2	8.5442e+02	0.0000e+00	0.0000e+00	5.2757e-07	6.0808e-07		1.2098e+07
3	8.7674e+02	0.0000e+00	0.0000e+00	1.0521e-06	1.0601e-06		1.2098e+07
4	8.9957e+02	0.0000e+00	7.7423e-10	1.5889e-06	1.5228e-06		1.2098e+07
5	9.2294e+02	0.0000e+00	6.1225e-09	2.1383e-06	1.9963e-06		1.2098e+07
6	9.4686e+02	0.0000e+00	1.2570e-08	2.7005e-06	2.4808e-06		1.2098e+07
7	9.7134e+02	3.2974e-09	1.9168e-08	3.2759e-06	2.9767e-06		1.2098e+07
8	9.9639e+02	1.0012e-08	3.2987e-08	3.7749e-06	3.4243e-06		1.2098e+07
9	1.0220e+03	2.2800e-08	6.5274e-08	4.1971e-06	3.8234e-06		1.2098e+07
10	1.0483e+03	3.5888e-08	9.8317e-08	4.6292e-06	4.2319e-06		1.2098e+07
11	1.0751e+03	4.9281e-08	1.3213e-07	5.0714e-06	4.6500e-06		1.2098e+07
12	1.1026e+03	6.2989e-08	1.9265e-07	5.5240e-06	5.0778e-06		1.2098e+07
13	1.1307e+03	8.8513e-08	2.5476e-07	5.6423e-06	5.2358e-06		1.2098e+07
14	1.1595e+03	1.2366e-07	3.1832e-07	5.4929e-06	5.1778e-06		1.2098e+07
15	1.1890e+03	1.5962e-07	3.6738e-07	5.3400e-06	5.1185e-06		1.2098e+07
16	1.2191e+03	1.9643e-07	4.1653e-07	5.1835e-06	5.0579e-06		1.2098e+07

17	1.2500e+03	2.3319e-07	4.6271e-07	5.0281e-06	4.9888e-06		1.2098e+07
18	1.2815e+03	2.6464e-07	4.6529e-07	4.9007e-06	4.8714e-06		1.2098e+07
19	1.3138e+03	2.9683e-07	4.6792e-07	4.7703e-06	4.7512e-06		1.2098e+07
20	1.3469e+03	3.2977e-07	4.5627e-07	4.6368e-06	4.6282e-06		1.2098e+07
21	1.3808e+03	3.5986e-07	4.2338e-07	4.5038e-06	4.5044e-06		1.2098e+07
22	1.4154e+03	3.6082e-07	3.8986e-07	4.3967e-06	4.3956e-06		1.2098e+07
23	1.4508e+03	3.6181e-07	3.4876e-07	4.2870e-06	4.2843e-06		1.2098e+07
24	1.4871e+03	3.6282e-07	3.0581e-07	4.1748e-06	4.1704e-06		1.2098e+07
25	1.5242e+03	3.5375e-07	2.6623e-07	4.0673e-06	4.0614e-06		1.2098e+07
26	1.5622e+03	3.3088e-07	2.3010e-07	3.9672e-06	3.9602e-06		1.2098e+07
27	1.6011e+03	3.0747e-07	1.9608e-07	3.8646e-06	3.8566e-06		1.2098e+07
28	1.6409e+03	2.8340e-07	1.7037e-07	3.7601e-06	3.7510e-06		1.2098e+07
29	1.6816e+03	2.5608e-07	1.4513e-07	3.6633e-06	3.6544e-06		1.2098e+07
30	1.7233e+03	2.2812e-07	1.2819e-07	3.5642e-06	3.5556e-06		1.2098e+07
31	1.7660e+03	1.9951e-07	1.1124e-07	3.4628e-06	3.4545e-06		1.2098e+07
32	1.8096e+03	1.7721e-07	9.9960e-08	3.3664e-06	3.3588e-06		1.2098e+07
33	1.8543e+03	1.5522e-07	8.8833e-08	3.2686e-06	3.2618e-06		1.2098e+07
34	1.9000e+03	1.3271e-07	8.1186e-08	3.1685e-06	3.1625e-06		1.2098e+07
35	1.9468e+03	1.1829e-07	7.3570e-08	3.0729e-06	3.0681e-06		1.2098e+07
36	1.9947e+03	1.0384e-07	6.7733e-08	2.9753e-06	2.9717e-06		1.2098e+07
37	2.0437e+03	9.0453e-08	6.2425e-08	2.8772e-06	2.8748e-06		1.2098e+07

38	2.0939e+03	8.1792e-08	5.7661e-08	2.7831e-06	2.7819e-06		1.2098e+07
39	2.1452e+03	7.2929e-08	5.3565e-08	2.6868e-06	2.6867e-06		1.2098e+07
40	2.1977e+03	6.6313e-08	4.9669e-08	2.5940e-06	2.5950e-06		1.2098e+07
41	2.2515e+03	6.0930e-08	4.6340e-08	2.5023e-06	2.5042e-06		1.2098e+07
42	2.3065e+03	5.5828e-08	4.3212e-08	2.4102e-06	2.4131e-06		1.2098e+07
43	2.3628e+03	5.2199e-08	4.0349e-08	2.3229e-06	2.3268e-06		1.2098e+07
44	2.4204e+03	4.8484e-08	3.7823e-08	2.2336e-06	2.2384e-06		1.2098e+07
45	2.4794e+03	4.5567e-08	3.5523e-08	2.1518e-06	2.1574e-06		1.2098e+07
46	2.5398e+03	4.2678e-08	3.3372e-08	2.0692e-06	2.0754e-06		1.2098e+07
47	2.6015e+03	4.0096e-08	3.1509e-08	1.9927e-06	1.9993e-06		1.2098e+07
48	2.6648e+03	3.7607e-08	2.9778e-08	1.9176e-06	1.9246e-06		1.2098e+07
49	2.7295e+03	3.5309e-08	2.8193e-08	1.8471e-06	1.8544e-06		1.2098e+07
50	2.7957e+03	3.3104e-08	2.6735e-08	1.7787e-06	1.7862e-06		1.2098e+07
51	2.8634e+03	3.1090e-08	2.5445e-08	1.7142e-06	1.7221e-06		1.2098e+07
52	2.9328e+03	2.9165e-08	2.4249e-08	1.6515e-06	1.6596e-06		1.2098e+07
53	3.0038e+03	2.7468e-08	2.3152e-08	1.5930e-06	1.6015e-06		1.2098e+07
54	3.0764e+03	2.5827e-08	2.2129e-08	1.5352e-06	1.5440e-06		1.2098e+07
55	3.1507e+03	2.4446e-08	2.1183e-08	1.4833e-06	1.4923e-06		1.2098e+07
56	3.2268e+03	2.3105e-08	2.0291e-08	1.4319e-06	1.4411e-06		1.2098e+07
57	3.3046e+03	2.1951e-08	1.9464e-08	1.3854e-06	1.3947e-06		1.2098e+07
58	3.3843e+03	2.0873e-08	1.8685e-08	1.3410e-06	1.3504e-06		1.2098e+07

59	3.4659e+03	1.9873e-08	1.7953e-08	1.2988e-06	1.3082e-06		1.2098e+07
60	3.5493e+03	1.8996e-08	1.7265e-08	1.2605e-06	1.2700e-06		1.2098e+07
61	3.6347e+03	1.8159e-08	1.6614e-08	1.2236e-06	1.2331e-06		1.2098e+07
62	3.7221e+03	1.7389e-08	1.5995e-08	1.1892e-06	1.1987e-06		1.2098e+07
63	3.8116e+03	1.6685e-08	1.5407e-08	1.1579e-06	1.1674e-06		1.2098e+07
64	3.9031e+03	1.6581e-08	1.4857e-08	1.1867e-06	1.1807e-06		1.2098e+07
65	3.9968e+03	1.7114e-08	1.5891e-08	1.2906e-06	1.2706e-06		1.2098e+07
66	4.0927e+03	1.6389e-08	1.5273e-08	1.2272e-06	1.2525e-06		1.2098e+07
67	4.1908e+03	1.5830e-08	1.4687e-08	1.0568e-06	1.0849e-06		1.2098e+07
68	4.2912e+03	1.5143e-08	1.4126e-08	9.9390e-07	1.1044e-06		1.2098e+07
69	4.3940e+03	1.4461e-08	1.3535e-08	1.0828e-06	1.1125e-06		1.2098e+07
70	4.4992e+03	1.3821e-08	1.2755e-08	1.0027e-06	1.0090e-06		1.2098e+07
71	4.6068e+03	1.3297e-08	1.1873e-08	9.7715e-07	9.7437e-07		1.2098e+07
72	4.7170e+03	1.2588e-08	1.1169e-08	9.5961e-07	9.5527e-07		1.2098e+07
73	4.8297e+03	1.1686e-08	1.0677e-08	9.4350e-07	9.3905e-07		1.2098e+07
74	4.9451e+03	1.0938e-08	1.0293e-08	9.2826e-07	9.2380e-07		1.2098e+07
75	5.0632e+03	1.0402e-08	9.9535e-09	9.1365e-07	9.0920e-07		1.2098e+07
76	5.1840e+03	1.0010e-08	9.6291e-09	8.9989e-07	8.9545e-07		1.2098e+07
77	5.3077e+03	8.5074e-01	8.5796e-01	8.7919e-01	9.2463e-01		1.2098e+07
78	5.4343e+03	1.9548e-02	2.4511e-02	3.8400e-02	7.1687e-02		1.2098e+07
79	5.5638e+03	1.4851e-02	1.8631e-02	2.9208e-02	5.4558e-02		1.2098e+07

80	5.6964e+03	1.1505e-02	1.4460e-02	2.2730e-02	4.2551e-02		1.2098e+07
81	5.8321e+03	8.9940e-03	1.1344e-02	1.7919e-02	3.3679e-02		1.2098e+07
82	5.9709e+03	7.0682e-03	8.9552e-03	1.4236e-02	2.6892e-02		1.2098e+07
83	6.1130e+03	5.5705e-03	7.0940e-03	1.1357e-02	2.1576e-02		1.2098e+07
84	6.2584e+03	4.3959e-03	5.6293e-03	9.0805e-03	1.7353e-02		1.2098e+07
85	6.4073e+03	3.4707e-03	4.4700e-03	7.2662e-03	1.3969e-02		1.2098e+07
86	6.5596e+03	2.7402e-03	3.5497e-03	5.8150e-03	1.1251e-02		1.2098e+07
87	6.7154e+03	2.1629e-03	2.8181e-03	4.8548e-03	9.3000e-03		1.2098e+07
88	6.8750e+03	1.7069e-03	2.2374e-03	4.1562e-03	7.7089e-03		1.2098e+07
89	7.0382e+03	1.3476e-03	1.7759e-03	3.4534e-03	6.3175e-03		1.2098e+07
90	7.2053e+03	1.0635e-03	1.4079e-03	2.8554e-03	5.1632e-03		1.2098e+07
91	7.3763e+03	8.3839e-04	1.1147e-03	2.3814e-03	4.2354e-03		1.2098e+07
92	7.5513e+03	6.6019e-04	8.8157e-04	2.0038e-03	3.4886e-03		1.2098e+07
93	7.7304e+03	5.1935e-04	6.9620e-04	1.6974e-03	2.8845e-03		1.2098e+07
94	7.9137e+03	4.0808e-04	5.4891e-04	1.4594e-03	2.4046e-03		1.2098e+07
95	8.1013e+03	3.2023e-04	4.3200e-04	1.2695e-03	2.0195e-03		1.2098e+07
96	8.2933e+03	2.5092e-04	3.3930e-04	1.1179e-03	1.7116e-03		1.2098e+07
97	8.4897e+03	1.9628e-04	2.6589e-04	1.0028e-03	1.4697e-03		1.2098e+07
98	8.6908e+03	1.5327e-04	2.0787e-04	9.0867e-04	1.2757e-03		1.2098e+07
99	8.8966e+03	1.1948e-04	1.6209e-04	8.4330e-04	1.1302e-03		1.2098e+07
100	9.1072e+03	9.3315e-05	1.2646e-04	8.4430e-04	1.0709e-03		1.2098e+07

101	9.3227e+03	7.2597e-05	9.8159e-05	7.4653e-04	9.2126e-04		1.2098e+07
102	9.5433e+03	5.6169e-05	7.5779e-05	7.1825e-04	8.5172e-04		1.2098e+07
103	9.7690e+03	4.3420e-05	5.8485e-05	6.9852e-04	8.0113e-04		1.2098e+07
104	1.0000e+04	3.3642e-05	4.5108e-05	6.8643e-04	7.6473e-04		1.2098e+07
105	1.0900e+04	1.3599e-05	1.7613e-05	7.5313e-04	7.8181e-04		1.2098e+07
106	1.1800e+04	7.1466e-06	8.5250e-06	8.7716e-04	8.7500e-04		1.2098e+07
107	1.2700e+04	4.1148e-06	4.5339e-06	9.6909e-04	9.6454e-04		1.2098e+07
108	1.3600e+04	4.6923e-06	4.7701e-06	1.8517e-03	1.8388e-03		1.2098e+07
109	1.4500e+04	4.0872e-06	4.0759e-06	2.1427e-03	2.0382e-03		1.2098e+07
110	1.5400e+04	3.0843e-06	3.0600e-06	2.0479e-03	2.0464e-03		1.2098e+07
111	1.6300e+04	2.6935e-06	2.6708e-06	2.1494e-03	2.1476e-03		1.2098e+07
112	1.7200e+04	2.4311e-06	2.4127e-06	2.1108e-03	2.1070e-03		1.2098e+07
113	1.8100e+04	2.4819e-06	2.4680e-06	2.2707e-03	2.0782e-03		1.2098e+07
114	1.9000e+04	2.1208e-06	2.1184e-06	1.9686e-03	1.9698e-03		1.2098e+07
115	1.9900e+04	2.0301e-06	2.0452e-06	1.9721e-03	1.9722e-03		1.2098e+07
116	2.0800e+04	1.9593e-06	2.1477e-06	1.9971e-03	1.9966e-03		1.2098e+07
117	2.1700e+04	1.9016e-06	2.5988e-06	2.0322e-03	2.0316e-03		1.2098e+07
118	2.2600e+04	1.8524e-06	3.2641e-06	2.0710e-03	2.0702e-03		1.2098e+07
119	2.3500e+04	1.7640e-06	1.5002e-05	2.0246e-03	2.0804e-03		1.2098e+07
120	2.4400e+04	1.7702e-06	8.9427e-05	2.1431e-03	2.1425e-03		1.2098e+07
121	2.5300e+04	1.7349e-06	2.4246e-04	2.1764e-03	2.1756e-03		1.2098e+07

122	2.6200e+04	1.7037e-06	3.7576e-04	2.2061e-03	2.2051e-03		1.2098e+07
123	2.7100e+04	1.6744e-06	4.5316e-04	2.2328e-03	2.2317e-03		1.2098e+07
124	2.8000e+04	1.6472e-06	5.2356e-04	2.1199e-03	2.1102e-03		1.2098e+07
125	2.8900e+04	1.6229e-06	6.0051e-04	1.6241e-03	1.5973e-03		1.2098e+07
126	2.9800e+04	1.6039e-06	6.6764e-04	1.1645e-03	1.1533e-03		1.2098e+07
127	3.0700e+04	1.6169e-06	7.1656e-04	9.7781e-04	9.7324e-04		1.2098e+07
128	3.1600e+04	1.7386e-06	7.4538e-04	8.9968e-04	8.9733e-04		1.2098e+07
129	3.2500e+04	2.0669e-06	7.5748e-04	8.5884e-04	8.5728e-04		1.2098e+07
130	3.3400e+04	3.7034e-06	7.3617e-04	8.3167e-04	8.3045e-04		1.2098e+07
131	3.4300e+04	1.5762e-05	6.5662e-04	8.1030e-04	8.0927e-04		1.2098e+07
132	3.5200e+04	6.4293e-05	5.3971e-04	7.9215e-04	7.9125e-04		1.2098e+07
133	3.6100e+04	1.6825e-04	4.3601e-04	7.7629e-04	7.7547e-04		1.2098e+07
134	3.7000e+04	2.9639e-04	3.6831e-04	7.6227e-04	7.6154e-04		1.2098e+07
135	3.7900e+04	4.0256e-04	3.2886e-04	7.4986e-04	7.4921e-04		1.2098e+07
136	3.8800e+04	4.8352e-04	3.0595e-04	7.3889e-04	7.3829e-04		1.2098e+07
137	3.9700e+04	5.5746e-04	2.9441e-04	7.2918e-04	7.2866e-04		1.2098e+07
138	4.0600e+04	6.2685e-04	2.9261e-04	7.2071e-04	7.2024e-04		1.2098e+07
139	4.1500e+04	6.8380e-04	2.9484e-04	7.1332e-04	7.1290e-04		1.2098e+07
140	4.2400e+04	7.2302e-04	2.9434e-04	7.0686e-04	7.0646e-04		1.2098e+07
141	4.3300e+04	7.4309e-04	2.9045e-04	7.0116e-04	7.0081e-04		1.2098e+07
142	4.4200e+04	7.3605e-04	2.8547e-04	6.9622e-04	6.9590e-04		1.2098e+07

143	4.5100e+04	6.8734e-04	2.8069e-04	6.9197e-04	6.9169e-04		1.2098e+07
144	4.6000e+04	5.9733e-04	2.7636e-04	6.8839e-04	6.8815e-04		1.2098e+07
145	4.6900e+04	4.9447e-04	2.7245e-04	6.8537e-04	6.8515e-04		1.2098e+07
146	4.7800e+04	4.0968e-04	2.6892e-04	6.8286e-04	6.8267e-04		1.2098e+07
147	4.8700e+04	3.5321e-04	2.6575e-04	6.8083e-04	6.8065e-04		1.2098e+07
148	4.9600e+04	3.1976e-04	2.6290e-04	6.7926e-04	6.7910e-04		1.2098e+07
149	5.0500e+04	3.0163e-04	2.6035e-04	6.7811e-04	6.7797e-04		1.2098e+07
150	5.1400e+04	2.9429e-04	2.5804e-04	6.7730e-04	6.7718e-04		1.2098e+07
151	5.2300e+04	2.9331e-04	2.5598e-04	6.7677e-04	6.7665e-04		1.2098e+07
152	5.3200e+04	2.9327e-04	2.5413e-04	6.7640e-04	6.7631e-04		1.2098e+07
153	5.4100e+04	2.9103e-04	2.5248e-04	6.7619e-04	6.7609e-04		1.2098e+07
154	5.5000e+04	2.8686e-04	2.5101e-04	6.7602e-04	6.7591e-04		1.2098e+07
155	5.5900e+04	2.8213e-04	2.4972e-04	6.7567e-04	6.7557e-04		1.2098e+07
156	5.6800e+04	2.7761e-04	2.4858e-04	6.7565e-04	6.7554e-04		1.2098e+07
157	5.7700e+04	2.7351e-04	2.4761e-04	6.7655e-04	6.7643e-04		1.2098e+07
158	5.8600e+04	2.6978e-04	2.4677e-04	6.7853e-04	6.7841e-04		1.2098e+07
159	5.9500e+04	2.6643e-04	2.4603e-04	6.8157e-04	6.8140e-04		1.2098e+07
160	6.0400e+04	2.6342e-04	2.4538e-04	6.8563e-04	6.8542e-04		1.2098e+07
161	6.1300e+04	2.6072e-04	2.4477e-04	6.9084e-04	6.9058e-04		1.2098e+07
162	6.2200e+04	2.5829e-04	2.4423e-04	6.9724e-04	6.9689e-04		1.2098e+07
163	6.3100e+04	2.5611e-04	2.4376e-04	7.0480e-04	7.0434e-04		1.2098e+07

164	6.4000e+04	2.5415e-04	2.4335e-04	7.1337e-04	7.1278e-04		1.2098e+07
165	6.4900e+04	2.5240e-04	2.4298e-04	7.2309e-04	7.2235e-04		1.2098e+07
166	6.5800e+04	2.5084e-04	2.4266e-04	7.3428e-04	7.3338e-04		1.2098e+07
167	6.6700e+04	3.7477e-04	3.9850e-04	4.4033e-03	4.6272e-03		1.2098e+07
168	6.7600e+04	4.1545e-04	3.7999e-04	7.9021e-03	7.9499e-03		1.2098e+07
169	6.8500e+04	2.5080e-04	2.4430e-04	9.0100e-03	9.0328e-03		1.2098e+07
170	6.9400e+04	2.4757e-04	2.4316e-04	9.7960e-03	9.8155e-03		1.2098e+07
171	7.0300e+04	2.4688e-04	2.4312e-04	1.0528e-02	1.0545e-02		1.2098e+07
172	7.1200e+04	2.4631e-04	2.4308e-04	1.1207e-02	1.1222e-02		1.2098e+07
173	7.2100e+04	2.4581e-04	2.4300e-04	1.1837e-02	1.1848e-02		1.2098e+07
174	7.3000e+04	2.4536e-04	2.4289e-04	1.2414e-02	1.2424e-02		1.2098e+07
175	7.3900e+04	2.2054e-04	2.1862e-04	1.2977e-02	1.3359e-02		1.2098e+07
176	7.4800e+04	2.2050e-04	2.1865e-04	1.3422e-02	1.3418e-02		1.2098e+07
177	7.5700e+04	2.2028e-04	2.1856e-04	1.3279e-02	1.3295e-02		1.2098e+07
178	7.6600e+04	2.2009e-04	2.1857e-04	1.2712e-02	1.2734e-02		1.2098e+07
179	7.7500e+04	1.0817e-04	1.0776e-04	6.5333e-03	6.5800e-03		1.2098e+07
180	7.8400e+04	1.0821e-04	1.0829e-04	6.1737e-03	6.2151e-03		1.2098e+07
181	7.9300e+04	1.0818e-04	1.0875e-04	5.8301e-03	5.8726e-03		1.2098e+07
182	8.0200e+04	9.5405e-05	9.6083e-05	4.4855e-03	5.0664e-03		1.2098e+07
183	8.1100e+04	9.5641e-05	9.6325e-05	5.1590e-03	5.1733e-03		1.2098e+07
184	8.2000e+04	9.5637e-05	9.6240e-05	4.8184e-03	4.8323e-03		1.2098e+07

185	8.2900e+04	8.5921e-05	9.0803e-05	4.0867e-03	4.3717e-03		1.2098e+07
186	8.3800e+04	8.6061e-05	8.6488e-05	4.3326e-03	4.2716e-03		1.2098e+07
187	8.4700e+04	8.6062e-05	8.6920e-05	4.0677e-03	4.0075e-03		1.2098e+07
188	8.5600e+04	7.6263e-05	7.8830e-05	3.3481e-03	3.3293e-03		1.2098e+07
189	8.6500e+04	7.6536e-05	8.3556e-05	3.5943e-03	3.5611e-03		1.2098e+07
190	8.7400e+04	7.6666e-05	9.1872e-05	3.4057e-03	3.3728e-03		1.2098e+07
191	8.8300e+04	7.5475e-05	1.0348e-04	3.2318e-03	3.2130e-03		1.2098e+07
192	8.9200e+04	6.8958e-05	1.0326e-04	2.9627e-03	2.6191e-03		1.2098e+07
193	9.0100e+04	6.9391e-05	1.1238e-04	2.9371e-03	2.9458e-03		1.2098e+07
194	9.1000e+04	6.9571e-05	1.1963e-04	2.8456e-03	2.8591e-03		1.2098e+07
195	9.1900e+04	6.9683e-05	1.2525e-04	2.7760e-03	2.7882e-03		1.2098e+07
196	9.2800e+04	6.9756e-05	1.2978e-04	2.7319e-03	2.7426e-03		1.2098e+07
197	9.3700e+04	6.9799e-05	1.3369e-04	2.7125e-03	2.7218e-03		1.2098e+07
198	9.4600e+04	6.9833e-05	1.3731e-04	2.7128e-03	2.7212e-03		1.2098e+07
199	9.5500e+04	6.9861e-05	1.4080e-04	2.7338e-03	2.7410e-03		1.2098e+07
200	9.6400e+04	6.9895e-05	1.4423e-04	2.7785e-03	2.7845e-03		1.2098e+07
201	9.7300e+04	6.9929e-05	1.4763e-04	2.8460e-03	2.8508e-03		1.2098e+07
202	9.8200e+04	6.9972e-05	1.5115e-04	2.9344e-03	2.9381e-03		1.2098e+07
203	9.9100e+04	7.0013e-05	1.5518e-04	3.0435e-03	3.0463e-03		1.2098e+07
204	1.0000e+05	7.0059e-05	1.6026e-04	3.1156e-03	3.1260e-03		1.2098e+07

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Michael Muller will be testing SCR #379 and, I guess, SCR #408, which I assume I will be coding.

List of things to tie up:

TP #379B
SCR #408:

Affected Software Module(s), Description of Problem(s): szft.f

Correct number of legs discrepancy; modify flags for nefii.inp; validate values for input parameters; check for number of time steps = 1

1) After merging code modifications in TPA v. 4.2q, a bug in the modifications for SCR #377 was revealed. With DistanceToReceptorGroup[km] set to 15.0, the code crashed in NEFTRAN. There were two legs in NEFII.VEL and three legs in nefii.inp after szft returned from prenefmksa. 2) Three more flags need to be changed in writing to nefii.inp and some values added (modeled on uzft.f). The flags 'RESTRICTIONS ON THE TIME STEPS', 'VELOCITY FIELDS FROM UNIT 11', and 'USE DEFAULTS (0) OR SUPPLY ALL 16 ' need to be changed from 0 to 1. The 16 values should be added to szft.f just as the appear in uzft.f. 3) An IEEE flag is being raised for Division by Zero; this may result in the new/modified code in szft.f for SCR #377. Porosity and streamtube width are input sampled parameters and are used in the denominators of calculations for vell and tvel. The values are never range-checked and may be coming in as zero. This is also true for uzft.f.

List of things that need to be looked at/changed in TPA:
- make a writeinputfile for both szft and uzft to use instead of both having nearly identical in-line code
- add validation of input parameters
- get rid of NP in uzft?
- look for possible divide by 0 in the code
- dilution.dat
- strmtube.dat and nearfield/farfield in szft.f & ?others

Made new directory tpatest/scr408. Wrote up SCR for 408 and gave to Ron. Used gettpa to get latest version of TPA (4.2v) from dev. Started mods for SCR #408.

Finished mods to the Version B Test Plan for SCR #379. Turned in hard copy to Ron.

Had to back track and delete version v (there's a problem with it) and replace it with version u from spock.

Ron wants me to do the code mods for SCR #408 (fix bug from #377, add range checks, and modify flags for nefii.inp) and then for SCR #399 (time-dependent dilution volumes).

Began mods for SCR #408. Was able to copy szft_mod379.f directly to szft_mod408.f. Apparently, no other changes have occurred in the szft.f file. Now, I just need to add the range checks and the flag changes. Uzft.f doesn't use width to calculate tvel or nefvel, just have to check porosity. Put range check in twice for uzft & once for szft. In szft.f, sawidth CAN be zero for the 1st leg, but values read in from strmtube.dat are all greater than zero. I think I can check for farwidth(i,j) .ne. zero as long as i is not 1. Szft.f is done except for the sawidth check. Uzft.f will be done after I add the range checks for porosity.

9/20/2002 -

Finished code changes for SCR #408. The divide by zero was caused by my not passing widthmult down to setvelfile after I moved the nefvel and neflen calculations down there. Uzft.f only uses mobile porosity; there is no width. Put in range check for porosity. Put in checks for porosity and streamtube width in szft.f. Ran tpa_mod408.e & passed tests where tpa ran using:

1. original tpa.inp
2. 3 realizations & 1 subarea
3. 1 realization w/ DistanceToReceptorGroup[km] = 15 km
4. 3 realizations & 10 subareas
5. a width in strmtube.dat set to 0.0
6. a UZFT Matrix Porosity in tpa.inp set to 0.0
7. a UZFT Fracture Porosity in tpa.inp set to 0.0
8. 10 realizations & 1 subarea

Started SCR #399 using version u and szft_mod408.f & uzft_mod408.f.

Affected Software Module(s), Description of Problem(s): exec.f, dcagw.f

The variable dilutionvolume needs to be a time-dependent array.

Currently, dilutionvolume is determined in dcagw and returned to exec as a double precision variable. Exec uses this value to print out the column values for dilutvol in

the files totdose.res and totdose_c.res. However, this does not reflect the changes in dilution volume due to the pluvial time switch.

Description of Change(s) or Problem Resolution (*If changes not implemented, please justify*): Dcagw already contains a time step loop where it checks for the pluvial time switch and sets dilutionvolume accordingly. The correct value for dilutionvolume is used to calculate remperyrpernuclgw. But dilutionvolume is either set to pluvial dilution volume or the non-pluvial dilution volume depending on whether the pluvial time switch is triggered on or off. The times at which the dilution volume switches back and forth are lost when the latest value for dilutionvolume is passed back to exec. This can be corrected by changing dilutionvolume to an array dimensioned to ntim and storing the current value for dilutionvolume by current time step. All uses of dilutionvolume in dcagw and exec are already contained within a time step loop.

Modified code in exec.f and dcagw.f. Dilution volume is now an array instead of a variable. In dcagw.f, either pluvial dilution volume or regular dilution volume is stored for each time step depending on whether the conditions are pluvial or not. The new array dilutvol is set to dilutionvolume or pluvialdilutionvolume. Dilutvol(timestep) is then used in calculations instead of the old dilutionvolume or pluvialdilutionvolume.

Ran tests w/ the baseline tpa.inp, with 3 realizations & 10 suabareas, and with 10 realizations & 10 subareas.

Loose ends to tie up: update SCR #399, update SCR #408, make CD for SCR #379, make CD for SCR #379B, turn in Scientific Notebook, give Ron a list of possible areas in the source code that need attention.

Mods for SCR # 408 are in tpatest/scr408/mods. Files affected are szft.f and uzft.f. Mods for SCR #399 are tpatest/scr399/mods. Files affected are exec.f and dcagw.f. I also put the modified files from #408 into the mods directory for #399. I built the mods for 399 on top of the ones for 408.

09/23/2002 -

Ran one more test for #399 where DistanceToTuffAlluviumInterface[km] is set to the same value as DistanceToReceptorGroup[km]. Test ran OK - no problem there since the problem with number of legs was resolved.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Modified tpa.inp for Ron for his mods to add colloids. Added KDs, RDs and RD_Inverts to tje SZFT, UZFT, and EBSREL sections of tpa.inp for Ja (parent Americium Am), Jc (parent Curium Cm), Jp (parent Plutonium Pu), and Jt (parent Thorium Th). Used version v of tpa.inp.

Turned in SCR #399 w/ description of software changes to Ron. Worked on finishing up SCR #408, but didn't quite finish.

09/30/2002 -

Finished the mods for SCR #408 and turned in a hard copy to Ron. Made CDs for SCR #379 and SCR #379B and turned them in to Ron.

Met with Brandy to discuss her testing the code mods I made for SCR #377. Ron says the latest version of the code is now 5.0 beta. New charge number is 20.06002.01.113.

Began work on testing SCR #346. Once that is finished, will move on to SCR #385.

10/01/2002 -

Turned in hard copy and electronic copy on floppy of Scientific Notebook for period March - September, 2002 to Ron.

Created subdirectory /net/spock/home/cscherer/tpatest/scr346 and used ~rjanetz/gettpa to copy current version of the code to it. Made new executable tpa.e. Began running George Adams' test plan for SCR #346 on TPA 5.0beta.

Makefile makes tpa.e and most of the standalone executables. However, it doesn't make itym.e. There is a makefile for it in codes/itym. Had some trouble making itym.e. There was an error in the makefile in codes/itym. The line:

```
"$(ODIR)%.o  : $(SDIR)%.f $(INC1)
        f77 -c $(FFLAGS) -o $@.e $(SDIR)$(@F:.o=.f)
```

should be:

```
$(ODIR)%.o   : $(SDIR)%.f $(INC1)
        f77 -c $(FFLAGS) -o $@ $(SDIR)$(@F:.o=.f)
```

Code compiles OK after making the mod.

Finished test PL-1 if it's OK to have ******* for output and raise IEEE flag for underflow. Need to check w/ Ron. Left off running PL-2 test. There are 4 versions w/ 50, 100, 500, and 1000 realizations, respectively. These are taking forever to run. Started making mods to the test plan to reflect working out of my directory. Need to modify specified directories. Several references to files and/or scientific notebooks that aren't specified fully. The ancillary .dat files are the same for TPA 4.1j and TPA 5.0 beta.

Copied tpa41j from janetzke/tpa41j. Need to make a base case run w/ 400 realizations for PL-2. May need to clarify w/ George Adams what parameters need to be in itym.dat for the base run. Looks like I need to get Randy Fedors to generate some plots on the data (once I get it all). Will have to check w/ George for the details.

10/03/2002 -

Continued with the test plan for SCR #346. Running three different TPA runs - taking a LONG time to finish. Started looking at the test plan for SCR #390 to work on at the same time. Runs for pltest/pl-2 case C, pl-3, and version 4.1j.

Meeting at 9:00 a.m. with Ron J., George Adams, Bob Rogers, and Michael Muller to discuss the next few months of testing, etc. Dates of interest:

1/1/03 - deliver validation test plan
2/1/03 - deliver 5.0 code
5/31/03 - complete validation of 5.0

Talked with Randy Fedors about missing information on SCR-390. He sent an email with additional information.

10/04/2002 -

Ran pl-2 case D and checked results from yesterday's runs. One run disappeared overnight.

Results from the runs in pl-2 so far do not match the test plan. Will have to start runs using version 4.2b instead of version 5.0beta and see if they are different. Have the code ready to go to start runs and let them go over the weekend.

10/07/2002 -

The seven runs I had going when I left Friday finished OK. Once all tests have been run, can show results to Randy Fedors and get his opinion on the results as well as get his help with the plots that are called out. The results in the table for pl-2 are the same running version 4.1j, but they differ quite a bit using 4.2b or 5.0 beta. The results for 4.2b and 5.0 beta are the same. Had trouble interpreting the results based on the test plan. Turns out George and Randy had set up some Excel spreadsheets to look at the results. Got the files from George. Ran sl-1 and started sl-2.

Got updated test plan from George Adams for SCR #390.

10/11/2002 -

Ran sl-3. There are three runs w/ 50 realizations (A, B, C) and they took quite a while. Left while one of them was still running. SL-3B bombed.

10/15/2002 -

All runs completed. Re-started Sl-3B after checking tpa.inp file to see if it bombs again. One difference: ArealAverageMeanAnnualInfiltrationAtStart[mm/yr] is a constant of 8.5 instead of a uniform distribution (A & C) of 4.0, 13.0.

Talked to George Adams about analyzing the output of the system level tests. Had to copy over later version of extract-subarea from his utiltest/sl-3 subdirectory. Re-ran it. My summary.dat is now consistent with his. Asked George to provide a description of what to be looking for in the .xls files to decide if tests pass.

Problem w/ my Scientific Notebook that I turned in 10/1. There are some changes in text from the last time. Probably due to the editing I had to do after the computer problems 8/14/2002. Maria Padilla is checking w/ Bruce Mabrito about what they want to do. I don't have the original electronic copy of the notebook; it was corrupted. Had to patch together files and edit to recreate the notebook (obviously, not perfectly).

Unix note: command 'top' shows the top processes running. The command 'ps -u cscherer' shows me the processes I have running.

Looked at results from SL-1 in sl-1.xls. Can't figure out where George got the data for the MAI Input column. Will have to ask him.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Reran sl-3b using George's idea of commenting out the correlation stuff at the end of tpa.inp for ArealAverageMeanAnnualInfiltrationAtStart[mm/yr]. Still bombed.

Ron J. also got the same error at realization 95 of a run. He says it's related to the new sampled parameter pH_AllUZ_SZ_Layers[StandardUnits]. It was set to a normal distribution from 6.3 - 9.6. Ron checked with ? Turner who said to change the 9.6 to 8.7. Started run before I left.

10/16/2002 -

The run I left overnight bombed on realization 3, instead of 33. Looks like lowering the max pH value isn't the way to go. Started run (testB) with pH_AllUZ_SZ_Layers[StandardUnits] ranging from 6.3 to 12.3 (pH scale is 0 - 14, acid to base). This run bombed on realization 3, too. Additional runs:

set max = 8.7 - bombed on realization 3 (test A)
set max = 12.3 - bombed on realization 3 (test B)
set max = 9.9 - bombed on realization 33 (test C)
set max = 9.0 - bombed on realization 33 (test D)

Error message at end of nefii.out:

```
DIMENSIONS EXCEEDED FOR THE TRANSFER FRACTIONS ARRAY, BF
*** PROGRAM STOP FROM FACER
```

This is actually an array out of bounds message: The code from FACER in NEFMKS.F:

```
      IF(LADD .GT. NDBF) THEN
        WRITE(6,9071)
9071      FORMAT(//,' DIMENSIONS EXCEEDED FOR THE TRANSFER FRACTIONS',
  1     ' ARRAY, BF')
        GO TO 5000
      ENDIF
          .
          .
          .
5000 WRITE(6,9050)
9050 FORMAT(' *** PROGRAM STOP FROM FACER')
   STOP
```

LADD is an index into the array BF; NDBF is set to 150000.

Talked to Ron about what to look at next. Look at results from testA (failed at realization 3 after changing value for pH_All...) and the results from sl3-b (failed at realization 33) and see if there are any clues. Files in scr346 directory. Look at nefii_sl3b.out and nefii_sl3b.inp and then compare against nefii_testAsl3b.out and nefii_testAsl3b.inp.

10/17/2002 -

Test E: run tpa.e w/ tpa_testAsl3b.inp but with the MatrixKDs for the five elements uncommented out after pH_AllUZ_SZLayers[StandardUnits], called tpa_testEsl3b.inp (Five elements: Am, Np, U, Pu and Th). Didn't mess with the FractureRDs at all. EUREKA! This one runs!

The NEFII.VEL file, after crash, contains only one time step for 10000 yrs. Try for more time steps? Change UnsaturatedZone MinimumVelocityChangeFactor[Fraction]? See Test F. The single time step is for end time. Would it still bomb if that step remained at 0 time? NO. When test E ran to completion the last nefiiuz.vel contains only one time step at end time. Dead end.

Test F: run tpa.e w/ tpa_testAsl3b.inp with UZMinimumVelocityChangeFactor set to 0.1. Still failed in realization 3.

Test G: run tpa.e w/ tpa_testEsl3b.inp with Matrix KDs for Americium (Am) commented out again. Ran to completion.

Test H: run tpa.e w/ tpa_testEsl3b.inp with Matrix KDs for Neptunium (Np) commented out again. Ran to completion.

Test I: run tpa.e w/ tpa_testEsl3b.inp with Matrix KDs for Uranium (U) commented out again. Ran to completion.

Test J: run tpa.e w/ tpa_testEsl3b.inp with Matrix KDs for Plutonium (Pu) commented out again. Bombed in realization 21. One problem w/ tpa.inp for this run. The Matrix KD for Thorium for TSw was also commented out. Run Test Jb.

Test K: run tpa.e w/ tpa_testEsl3b.inp with Matrix KDs for Thorium (Th) commented out again. Ran to completion.

Unix note: command "ps -u cscherer" shows processing time (not real time elapsed) for a process. Tpa.e today is taking approximately 42 (42.22 for the last one w/ complete run) minutes.

10/18/2002 -

Test L: run tpa.e w/ tpa_testEsl3b.inp with Matrix KDs for Plutonium and Americium commented out. These two will be calculated; the others will be set in tpa.inp. Ran to completion.

Test M: run tpa.e w/ tpa_testAsl3b.inp with Matrix KDs for Plutonium only uncommented out. In other words, Plutonium will be set in tpa.inp; the other four will be calculated in the code. Bombed in realization 17.

Test Jb: run tpa.e w/ tpa_testJsl3b.inp with correction made to uncomment out MatrixKD_TSw_Th. Ran to completion.

Test N: run tpa.e w/ tpa_testAsl3b.inp. Increase NTBF from 150000 to 200000 in SIZE.INC; recompile nefmks. Postpone until later.

Test O: run tpa.e w/ tpa_testEsl3b.inp with Matrix KDs for both Americium and Thorium commented out. Run aborted because I started another run in the same directory. Postpone until later.

Test P: run tpa.e w/ tpa_testAsl3b.inp with Matrix KDs for Americium only uncommented out.
In other words, Americium will be set in tpa.inp; the other four will be calculated in the code. Bombed in realization 17.

Test Q: run tpa.e w/ tpa_testAsl3b.inp with Matrix KDs for Neptunium only uncommented out. In other words, Neptunium will be set in tpa.inp; the other four will be calculated in the code. Bombed in realization 17.

Test R: run tpa.e w/ tpa_testAsl3b.inp with Matrix KDs for Uranium only uncommented out.
In other words, Uranium will be set in tpa.inp; the other four will be calculated in the code. Bombed in realization 17.

Test S: run tpa.e w/ tpa_testAsl3b.inp with Matrix KDs for Thorium only uncommented out.

In other words, Thorium will be set in tpa.inp; the other four will be calculated in the code. Bombed in realization 17. Wow. Coincidence?

Test T: run tpa.e w/ tpa_testAsl3b.inp with Matrix KDs for Americium, Uranium, and Neptunium commented out. I'm wondering if all runs with 2 KDs calculated or all runs with 3 KDs calculated will end the same (like the runs w/ 1 KD and 4 KDs). There are 720 versions of 3 KDs and 24 versions of 2 KDs. Ran to completion.

Observations and conclusions of test runs starting w/ tpa_testAsl3.inp:

14. No one of the 5 elements of interest (Am, Np, U, Pu, Th) alone causes the abort conditions.
15. Abort conditions are not caused by number of time steps selected.
16. Abort conditions can be affected (but not necessarily so, see SCR-346, test sl-1) by setting ArealAverageMeanAnnualInfiltrationAtStart[mm/yr] to constant: different processing path.
17. No abort conditions occur if all Matrix KDs are set in tpa.inp.
18. Values for pH_AllUZ_SZLayers[StandardUnits] do affect abort conditions.
19. Abort conditions may be induced if some Matrix KDs are commented out and some not for a given element. (see J and Jb)
20. Calculating values for any 4 of the 5 elements causes error condition. THIS IS WEIRD! All the nefii files (*.inp, *.vel, *.out) are the same except for the time stamp. Same for the tpa_test*.out files.
21. Those runs with just 2 elements being calculated ran to completion.
22. Those runs that end in the same realization have the same result file values. (See testsl3b and testBsl3b as well M, P, Q, R, & S)

I talked with Michael Muller about SCR 371 (where all these KD changes occurred). I was misinterpreting some things. Regardless whether the Matrix KDs are commented out or not in tpa.inp, the calculation of the values for the 5 elements is controlled by the file coefkdeq.dat, which currently has all 5 being calculated. So the results are really not WEIRD, they behaved as expected. Values in tpa.inp are only used IFF they are not calculated according to the coefkdeq.dat data file.

Michael things problem may be due to something screwy with sampled parameters and lhs. If a uniform parameter is changed to a constant, then the seeds for all subsequent parameters are changed. Constants don't use lhs. Suggested test: add a dummy uniform variable ahead of my constant ArealAverageMeanAnnualInfiltrationAtStart[mm/yr] so that changing it from uniform

to constant doesn't affect the remaining file. If I do this for my sl-3b case and it runs, then the problem moves out of the arena for SCR 371 and into tpa.inp problems.

So, Test U: add the dummy variable to tpa_testsl3b.inp and run. That was the input file that caused an abort in realization 3.

I do have a way now to get around sl-3b bombing. I can uncomment the 5 elements and let it run. I will also have to re-run sl-3a and sl-3b. If U works, that might be a better workaround: just add the dummy variable to tpa_sl3b.inp. That will be the end of the tests I need to do for SCR-346. Still have to input data into spreadsheets, check a couple of questions with George, and get Randy Fedors to look at the results and tell me if they are OK or not. Finish up planned testing for SCR-346 next Monday if I can. Ron wants me to continue pursuing the problem with sampled parameters.

Runs of tpa.e were taking about 42:32.33 (processor time only; not elapsed time) today (42:39, 42:28, 42:30) for 50 realizations.

10/21/2002 -

Test U ran to completion.

Test V: use tpa_testsl3b.inp and change pH_AllUZ_SZLayers[StandardUnits] back to the original 6.3, 9.6. The only difference between this tpa.inp and the one for sl-3b is the dummy variable. Ran to completion. Use this one for the sl3-b test results for SCR-346.

Test W: use tpa_sl3b.inp and change random seed + 1 (original = 188910452.0). No dummy variable. Ran to completion. Saved nefii* files and lhs.out.

Test X: use tpa_sl3b.inp and change random seed - 1 (original = 188910452.0). No dummy variable. Ran to completion. Saved nefii* files and lhs.out.

Test Y: use tpa_sl3b.inp and change random seed + 5 (original = 188910452.0). No dummy variable. Tpa_testY.inp is ready. Postpone run for now.

Test Z: use tpa_sl3b.inp and change random seed - 5 (original = 188910452.0). No dummy variable. Tpa_testZ is ready. Postpone run for now.

The values Michael Muller put in for KDs are just rough guesses. He is in the process of getting better values from the experts. Will resume testing on the problem of NEFTRAN

bombing after the numbers have been refined. Should be able to tell then if the problem is an extreme value being fed to NEFTRAN based on calculating KDs or there is some other kind of weirdness going on.

10/22/2002 -

Back to finishing off SCR-346 and on to SCR-390. Moved tests and results for pH values and other test related to NEFTRAN bombing to subdirectory pHtests.

Copied results from testV run down to sltest/sl-3 to replace the sl-3b files. By using these, all tests have now been run for SCR-346. Just need to update the .xls files and verify results. All .xls files are uptodate, although I need to check on where the "Uncertainty Parameter" comes from for sl-3.xls. Made plans with George Adams to go over the spreadsheets tomorrow. Starting reviewing the changes I had already made to the test plan and verifying that I ran all the tests appropriately and that I have the files archived that need to be. Made it through PL-2, everything except the plots that I need help from Randy Fedors to finish.

Test 1024 realizations: started a run in scr346/co-runC using tpa_testX.inp (subtract 1 from seed) with NumberOfRealizations and StopAtRealization changed to 1024. Many IEEE inexact and underflow flags raised.

10/23/2002 -

Test 1024 bombed in realization 61, failed in NEFTRAN, same error message. Looking at lhse.out and lhs.out. pH_AllUZ_SZLayers[StandardUnits] is sampled parameter #176.

Test 1024B: start w/ tpa_testW.inp (add 1 to seed) and change from 50 to 1024 realizations. Bombed in realization 103. Same error in NEFTRAN.

Test 1024C: start w/ tpa_testVsl3b.inp (dummy variable added) and change from 50 to 1024 realizations. Bombed in realization 148.

SCR-346: Talked with George Adams about the results of the system level tests. We match on sl-1. George only put Randy Fedors earlier results in the test plan. George's results were the same as mine. Sl-2 indicated a problem. I re-ran it to double check that I had run it according to the plan; my results were identical to my first run but still way off on George's. We discovered that George had run his for 100,000 years, not 10,000. So I re-ran it again (SL-2c) and now my results match George's. We may still have a problem with the results on sl-3. For one thing, I had to rerun extract-subarea.e using a different version of maydtbl.dat, one that had the

current climate as the first case, not somewhere in the middle. This changed summary.dat to the same results as George had. Need to change the test plan to mention this. My results are still different. For my results on sl-3B, need to use uzflow_testVsl3b.rlt instead of uzflow-B.rlt because the B run bombed. However, rerunning SL3-B and SL3-C for 100000 yrs. ??use revised maydtbl.dat ?? if needed?????

Finished putting my results into the spreadsheets (sl-1.xls, sl-2.xls, and sl-3.xls) except for data from the current runs. Randy Fedors will be generating the plots called out by the test plan and will check the other results for reasonableness.

10/24/2002 -

The run for sl-3b (see PA-SCR-346_SL3-Bb.out) at 100,000 years bombed even though the dummy variable was there. ARGH!!! PA-SCR-346_SL3-Cb.out ran to completion, however. Updated sl-3.xls. Can't do much w/ sl-3b w/o a successful run at 100,000 years. So, I guess these are as good as it will get for now. I'm waiting for Randy Fedors to get back to me on the results (he'll be on travel next week).

Ron asked Sitakanta to revise the numbers for coefkdeq.dat. Sitakanta left on travel. Don't know if someone else is working on fixing the KD calculation numbers or not.

Test dummy2: added 2nd dummy variable after pH_AllUZ_SZLayers[StandardUnits]. Used tpa_sl3-Bb.inp (100,000 years, 50 realizations). Failed in realization 26. This time array JPN, the Transfer Pointer Array, in FACER in nefmks.f went out of bounds.

Test NDBF300K: changed NDBF from 150,000 to 300,000 in SIZES.INC in codes. Recompiled nefmks.e. Running tpa_sl3b.inp (50,000 years, 50 realizations). Failed in realization 33, just like the previous incarnation, BF.

10/25/2002 -

Made changes to test plan for SCR #346. Can't finalize it until Randy Fedors has a chance to look over the results, which won't be until week after next. It looks like the results are pretty good even though I can't run sl-3b for 100,000 years.

Moved on to SCR #390. Will try to have all the results for both SCR #346 and SCR #390 ready for Randy after he gets back from travel week after next.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Test dummy8: added 6 more dummy variables after dummy2. Changed back to 50,000 yrs. and 1024 realizations. Started about 2:00 p.m. Bombed in realization 116, array BF array out of bounds in NEFTRAN.

10/28/2002 -

Re-worded scr_408.wpd and gave it to Ron. Made a floppy with all scr_???.wpd files I had changed and gave it to Ron.

Made table for Ron for getting input from Sitakanta on the hydrogeologic characteristics (KDs and RDs) sampled parameters in tpa.inp. The table compares SPs from tpa.inp v4.1j and tpa.inp v5.0beta. Many of the values do not have "real" or meaningful numbers. At least some of them are probably related to the bombing problem we are seeing.

10/29/2002 -

Continued with the table. It's around 34 pages long now. It has the old SP name and the new along w/ PDF type and values. In addition, there is a reference column that ids the source of the data. For now it is stored in cscherer as table.wpd.

Started copying files for SCR #390. It used 4.2u. I'll be testing with version 5.0 beta.

Met with Ron and Michael Muller to discuss KDs and RDs.

10/30/2002 -

Started 1st run of itym.dat for SCR #390 (pl-1A). Runs 500 realizations. While waiting for run to finish, started looking at SCR #395. SCR #394 is not yet implemented, so I can't test it yet. SCR #384 went to Bob Rogers who is more familiar w/ that area and has worked with Mike Smith.

Generated SCR #412, based on some problems Michael Muller uncovered when he was testing SCR #408. 408 is complete. Completed code modes to szft.f and gave copies of the code and SCR to Ron. It is ready for testing.

SCR #412: Values for the array pori (immobile porosity) and the variable pormix are not range checked.

If pormix is allowed to be zero, a divide by zero will occur. If pori is set to zero, NEFTRAN will abort.

In szft.f: after ImmobilePorosity is read in, add code to verify that the value read is greater than 0.0 and less than 1.0. After pormix is read in, add code to verify that the vale read is greater than 0.0.

Changes to SZFT:

After line 998 of szft.f (after pormix value is read in), add the following lines:

```
ccc
c css - 10/30/2002:  SCR #412
c   add value check for pormix
c   a zero value could result in a divide by zero
         if (pormix .le. 0.0) then
            print *, ' ***>>> Error in szft:prenefmksa <<<*** '
            print *, ' Pormix value out of range - must be',
     &               ' greater than 0.0 '
            print *, ' ', spname, ' = ', pormix
            print *, '  '
            STOP
         endif
ccc - SCR #412:  end change
```

After line 1037 of szft.f (after immobile porosity is read in,)add the following lines:

```
ccc
c css - 10/30/2002:  SCR #412
c   add value range check for immobile porosity pori()
c   a zero value for immobile porosity will cause NEFTRAN to abort
         if ((pori(i) .le. 0.0) .or. (pori(i) .ge. 1.0)) then
            print *, ' ***>>> Error in szft:prenefmksa <<<*** '
            print *, ' Porosity value out of range - must be',
     &               ' greater than 0.0 and less than 1.0'
            print *, ' ', spname, ' = ', pori(i)
            print *, '  '
            STOP
         endif
ccc - SCR #412:  end change
```

SCR #394: I have a hard copy of a test plan that George gave Ron, but I can't find a directory for this scr in George's directory on spock. Does he have it on the PC? Couldn't find George when I looked for him.

SCR #390: Started 2nd run of itym.dat for SCR #390 (pl-1B). Runs 500 realizations

Left both runs going when I left for the day.

10/31/2002 -

SCR #390: Only the 2nd run completed successfully overnight. Restarted the 1st run. Need results from both runs to compare to complete pl-1. Getting the itym.dat, maidtbl.dat, maydtbl.dat, smaydtbl.dat, and tpa.inp files set up for sl-1, too. Re-made subareas.dat and maidtbl-extB.dat. Talked to George about missing details in the test plan directions. Started 3rd and 4th runs to generate files for the sl-1 tests. ran extract_difference.e on the result files from the two runs. The files produced for my runs matched George's exactly except for time/date stamps. Need to ask Randy to plot them when he returns next week.

George says that subareas.dat which is made by copying the subarea information from tpa.inp, removing the commas, and some minor formatting has to be corrected. For subareas 9 & 10, the first and last coordinates (? out of 5 sets) aren't the same and they need to be. Just for this test? or always?

SCR #395: George Adams has restored the files for this scr in his directory. Copied the test plan over to my scr395/docs directory. Reviewed test plan. Copied version 5.0 beta over to scr395.

Two runs still going when I left for the day: PIDs 526 and 584. The 1st run took a little over 4½ hours to complete (itym.e w/ 500 realizations) and over 300 minutes processor time.

11/1/2002 -

Last night's runs ran to completion. Copied files to the correct directories and started 1st tpa run w/ the generated data files (sl-1A). Ran sl-1B. All runs for SCR #390 are complete and the result files are archived. Need to give copy of test plan to Randy Fedors and ask him to look at results. Completion of test plan waiting for expert opinion.

SCR #395: The plan references some *.xls files generated by George, but I can't find them in his PA-SCR-395 files. He says they should be the same as the ones for PA-SCR-348, which is the scr that required dsfailt to be created. He will send me the spreadsheet files by mail.

Received spreadsheets, except for sl-3.xls, which is referenced in the test plan. Started analysis on sl-1 results.

Gave copy of the test plan for SCR #390 to Randy Fedors. He thinks he will be able to look at the output files for #390 and #346 sometime next week.

There will be a code walkthrough on Monday. George Adams will present SCR #346 and SCR #390 (the two I'm almost finished with pending Randy's help). Printed out file from George to review.

Made file of possible improvements/changes for tpa and gave copy to Ron.

TPA: Potential Code and Input File Changes

0.23 Expand dilution.dat for higher pumping rates. Need to be able to meet EPA rule of 3,000 acre ft./yr. (J Winterle ?)

0.24 Nearfield and farfield concept no longer used w/ addition of sampled parameter DistanceToReceptorGroup[km]. Modify strmtube.dat and code that uses it?

0.25 Still need to add option to correlate SZFluxMultiplierAtGlacialMaximum[] w/ MeanAveragePrecipitationMultiplerAtGlacialMaximum, now called MeanAnnualPrecipitationMultiplierAtGlacialMaximum.

0.26 Uzft.f and szft.f both have inline code to generate nefii.inp. Suggest making one subroutine and have both modules call it.

0.27 Both uzft.f and szft.f have similar code w.r.t. initializing inventory variables. Could possibly be made into subroutine(s) called by both modules.

0.28 There is a problem in the code with the use of "global variables" (like ntim & tim, nefvel) which are passed as parameters and renamed in subroutines (ntimg & timg, tvel). Then a local variable is declared using the old name (ntim & tim) that mean something different. Since it is so hard to keep track of where you are in the code (modules that are way too long), this usage can make it much harder to understand and correctly modify code. I've added comments where I've noticed this which should, hopefully, make it easier the next time through the code, but it would make code analysis easier in the future if we could clean these up.

0.29 Look for input parameters whose values needs to be checked and add code for error trapping. Look for code where it is possible to divide by zero., too.

0.30 Break up code modules that are too long into subroutines (exec, uzft, szft, dcagw, dcags, . . .); pull out the set up code and place into subroutines (e.g., read and set sampled parameters, initialize arrays and variables from input files, write output files); your idea of putting declarations into include files.

0.31 Get rid of implicit statements (use implicit none); declare and comment all variables used.

0.32 Get rid of NP in uzft.f since it really isn't used; the arrays will be smaller as well as less complex.

0.33 I personally wouldn't mind seeing tpa.inp simplified by providing file names which contain the actual parameters for specific code sections. The file is getting bulkier all the time and it is difficult to wade through it if you need to look for parameters whose names you don't know exactly. Could still generate tpameans.out as one long file. Could provide a utility that would build tpa.inp as one file from a series of smaller files; existing code wouldn't have to be modified then.

0.34 I wouldn't be surprised if there are similar parameters for the same thing in tpa.inp. Perhaps some could be used for multiple purposes instead of adding new parameters. Anything that reduces the number of parameters would be good. Of course, I don't think any one person is totally familiar with all the parameters. It would be nice to have an index or cross-referenced table (gee, if we made it a database, we could set it up to automatically generate appendix A between major revisions) that could be used to search existing parameters to see if they fit the requirements for new functionality.

0.35 As Bob? George? suggested, get rid of the data files that aren't being used.

0.36 Clean up comments - especially comments that have no code following them.

0.37 Make sure header blocks and comments are updated to reflect what is actually going on in the code. Ex.: uzft.f:prenefmks says that we only have fracture or matrix flow, but not both. This is no longer correct, but I'm not sure what we should say instead.

0.38 I think we probably have some arrays that are dimensioned much larger than they need to be (like storing a value per time step per layer when we really just want an overall value for the layer).

0.39 There may be code inside loops that only needs to be executed once. We could improve performance a bit if we can locate this code and move it outside the loop.

0.40 I've seen places where the code is structured by setting up a nested loop followed by another nested loop and so on. These loops usually process one array. This may be the result of multiple modifications to the code. I know I try to keep my changes simple and self-contained, so I don't break existing code, but over the years, it makes the code harder to follow. It may be possible to combine some of them to process multiple arrays within the same nested loop.

0.41 It probably isn't productive to expend the effort needed to do some of these things, but we could generate a list of things to look for in code that code developers touch. If we don't take care of it immediately, we could update a list of specific lines of code that need future attention.

0.42 Do we have a reference anywhere that lists the differences between the unix, pc, and linux versions of the code or identifies things that are unique to or required by each platform?

0.43 Do we have a central list of suggestions for code improvements that we could update as needed or see if someone else has already made the suggestion?

11/4/2002 -

Finished runs of all the process level tests for SCR #395.

Code walkthrough at 2:00 p.m. George Adams. SCR #346 and SCR #390. I was tasked by Ron to generate a summary of the review.

11/6/2002 -

Completed draft of peer review. Gave copy to Ron and waiting for editing suggestions to finish it.

Received email from Ron forwarded from Stacey at NRC. She said:

```
 "The importance analysis flags that are causing a problem with
the code execution are:
Component_STFF_SAV_Study
ComponentDripShieldStudy
```

```
When these flags are turned on, the code doesn't run and the
results files are blank.
Thanks,
Stacey"
```

Ron added:

```
"George,

Could you check on this drip shield flag and see what the error
is?

Carol,

Could you check on the STFF_SAV flag and see what the error is?

thanks,

ron j"
```

TPA does bomb with an error message from ispquery when the ImportanceAnalysisFlag is set to 1 along w/ the identified Component Study flags. George and I got together and determined that the problem is with ia.dat. When the Importance Analysis flag is set, iareader is called. Iareader reads ia.dat and calls ispquery to look for the sampled parameter names ia.dat in tpanames.dbs. If the names have changed or been deleted from tpa.inp (there are a lot of those in 5.0beta!), ispquery will report an error. Talked with Marty and Michael about the Studies and which variables should matter.

Made a new version of ia.dat with the correct name changes. Also added some other new parameters that we thought mattered and commented out parameter names that are no longer used. Set all study flags in tpa.inp to 1 and ran TPA. (See cscherer/tpatest/comp_study/tpa*.out as well as ia*.dat.). TPA still bombed with an error about the Well Pumping Rate At Receptor Group. The old value used in ia.dat is 1.0. That is out of range. Changed the 1.0 to 2000.0 and TPA ran to completion. Sent copy of new ia.dat, with comments about what I did and what we need to look at next) to Ron.

Found a couple of errors in error messages in iareader.f and sampler.f.

IAREADER.F:

```
print *, ' ***>>> Error in ia.f <<<*** '
print *, ' not able to copy ia.f'
```

This should say 'Error in iareader.f' or 'ERROR in IAREADER', 'not able to copy ia.dat'.

SAMPLER.F:

```
print *, ' Can not ADD pdf after calling newrealization  '
print *, ' can not find name of parameter  '
print *, ' can not find name of parameter  '
print *, ' Can not ADD userdist after calling newrealization  '
```

These should say 'cannot' and 'Cannot'.

Sent email to Ron about these for inclusion in SCR #392, an SCR to be used for general cleanup.

Sent draft version of code walkthrough to Ron. He suggested using another form.

11/7/2002 -

Re-entered code walkthrough summary info in new form. Sent to Ron. Still need to resolve a couple of questions to finish it. George, I think, isn't going to be over here today so it will have to wait until tomorrow.

Ran a test in scr346/co-runA with no maydtbl.dat available. Error message says 'Inconsistent entries in DTBL < maydtbl.dat >'. You also get a cp error from unix 'No such file or directory.'

SCR #395: Started system level tests. SL-1: TPA run failed. array BF, array out of bounds in NEFTRAN. Failed on realization 1, out of 3. Ron said he had some luck changing NumberOfRealizations to 1023 and setting StopAtRealization to the number he really wanted. UNBELIEVABLE - it ran! Unfortunately, it didn't generate the dsfail.rlt and dsfail.ech files needed, probably because it didn't run through all 1023 realizations?. Tried a run w/ 1023 realizations, but it bombed in realization 10. Ran 3 realizations using a different seed. It ran successully to completion, but the output files still aren't there. Do I need to set OutputMode to 2? Ran test. This time I got dsfail.ech and dsfail.rlt. Re-run NumberOfRealizations = 1023 and StopAtRealization = 3 and OutputMode set to 2. This time it did create the expected files. Need to add to changes for tpa.inp in sl-1 test.

Sl-1: sl-1.xls: started entering data into the spreadsheet.

Made directory tpatest/tpa50beta. Made executable. Ran base case. Ran 1 realization successfully to completion. Ran w/ NumberOfRealizations = 2047, StopAtRealization = 1024 (pid 5174). Whoops. Seems the maximum number of realizations is 1024.

11/8/2002 -

Received email from Ron stating that we should stop using the default f77 compiler on spock and immediately change to /solapps/SUNWspro.4.2/bin/f77. It finds errors that the default compiler allowed. NRC has the solapps compiler. Five files need revisions: mechfail.f (George Adams), nfenv.f (George), seismo2.f (George), szft.f (me), and uzft.f (me). I didn't re-compile itym.e. The old Makefile we've been using needs to change 'f77' to 'solapps/SUNWspro.4.2/bin/f77'. The first line of the Makefile should be 'FC = /solapps/SUNWspro.4.2/bin/f77'.

The SCR for George's changes will be #415; mine will be #416.

All the errors are the same: bad dimension list for an array that cannot be an adjustable array. If the dimension and the array are both passed as parameters, the declaration is OK. But, if the dimension is passed as a parameter and then used to dimension a local array, it is an error.

Rons wants us to check all work we've been doing w/ TPA 5.0beta w/ the new compiler code. For me, this means I need to redo SCRs #346, 390 and 395.

Talked to George about the code walkthrough summary from Monday. Finished my mods and gave hard copy and electronic copy (see cscherer/walkthrough110402.wpd) of summary to Ron. There will be another code walkthrough next Monday (Nov. 11), also presented by George Adams.

Started looking at what I need to do to fix the new compiler-related problems in szft.f and uzft.f. Made mods to szft.f, using 'include maxntime.i' and include 'include uzszft.i' so I could access the variables maxntime and maxlayers to dimension local variables. Problems: maxntime and mxntime need to stay in lockstep; numsalayers from uzszft.i results in warning numsalayers not used. Plan to comment heavily re: maxntime and mxntime. Plan to check that numsalayers and numlayers are the same and that maxlayers and maxlyr are the same. Potential problem: szft.i has a maxlayers set to 20 and uzszft.i has a maxlayers that is set to 15 and saved. Both of these include files are used in different places in szft.f.

11/11/2002 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Tested whether this version of FORTRAN would allow the parameter mxntime to be changed. Good news: it won't! Put in statement 'maxntime = maxntime + 1' and got error message: 'Error: assignment to a non-variable.'. So, long as maxntime isn't assigned the value of maxntime and modified before using it as a parameter in a subroutine call, we should be OK.

Tested what would happen if included both uzszft.i and szft.i in the same subroutine. Good news again: error message: 'Error: maxlayers redefined'. Would still recommend getting rid of this descrepancy so no possible problem arises in the future. I think that the use of szft.i to set maxlayers was forgotten sometime and the local constant maxlyr was declared in szft.f. 'maxlyr is passed down to subroutines and used to dimension arrays passed as parameters. The old default compiler allowed the use of the parameter maxlyr to dimension local arrays the same size; the solapps compiler does not. Now we need to include either szft.i or uzszft.i or delcare yet another local constant and set it to the same value. But, with the proliferation of local constants, the possibility arises that at some time in the future, some values will be changed while others are not. The include file szft.i is included in szft.f and dcagw.f. Szft.i only contains maxlayers which is used to dimension arrays when it is included. I think we need to get rid of it or set the value to 15 in szft.i and remove maxlayers from uzszft.i and include both files where needed. Right now, if you include uzszft.i, you get an error message about numsalayers not being used.

My recommendation is to remove maxlayers from uzszft.i. Set the correct value for maxlayers in szft.i. Wherever uzszft.i is included, also include szft.i. Will check with Ron. (Note: I don't really know which version of numsalayers is meant in uzszft.i. Sometime it means number of layers in a subarea; sometimes it means number of stream tubes.)

CLEANUP: change sa* variables to st* variables when they refer to stream tubes and not subareas?

Trying to compile, got message "All 16 license(s) for Sun Workshop Compiler F77 are currently in use, queuing the request". It happened several times during lunchtime and the waiting period varied from a few seconds to minutes.

In uzft.f, also had to modify arrays using passed dimension mxnnucl to use maxnnucl after including maxnnucl.i

Code Walkthrough 2:00 - 4:00 p.m. over DSFAIL, SCR #s348, 395, 397. George presented.

11-12-2002 -

Szft.i was included in dcagw.f, uzft.f and szft.f. So, all these files had arrays declared with maxlayers = 20, although maxlayers was expected to b 15.

Created tpatest/scr416 directory for SCR #416. Using TPA 5.0 beta code w/ new compiler specified.

Changed szft.i so maxlayers = 15. Changed uzszft.i so maxlayers isn't in it. Changed szft.f and uzft.f to include szft.i so maxlayers is available for local arrays. Also added 'include szft.i' above any 'include uzszft.i'. Changed uzft.f by deleting parameter = 15 for maxlayers and including szft.i. Changed dimensioning of arrays from MAXLYR to maxlayers. Ran tpa_mod416.e against tpa_50beta.e. Same output (screen display) files except for time/date.

Made command file (tpatest/scr416/diff_files) to look at differences in files between the two runs:

```
echo '  1) diff FILENAME.DAT  ../tpa50beta/FILENAME.DAT'
diff FILENAME.DAT  ../tpa50beta/FILENAME.DAT
echo '  2) diff NEFII.VEL     ../tpa50beta/NEFII.VEL'
diff NEFII.VEL     ../tpa50beta/NEFII.VEL
echo '  3) diff airpkdos.res  ../tpa50beta/airpkdos.res'
diff airpkdos.res  ../tpa50beta/airpkdos.res
echo '  4) diff arpkds_c.res ../tpa50beta/arpkds_c.res'
diff arpkds_c.res ../tpa50beta/arpkds_c.res
echo '  5) diff ashout.res    ../tpa50beta/ashout.res'
diff ashout.res    ../tpa50beta/ashout.res
echo '  6) diff burnup.dat    ../tpa50beta/burnup.dat'
diff burnup.dat    ../tpa50beta/burnup.dat
echo '  7) diff chlrdmf.dat   ../tpa50beta/chlrdmf.dat'
diff chlrdmf.dat   ../tpa50beta/chlrdmf.dat
echo '  8) diff corrode.out   ../tpa50beta/corrode.out'
diff corrode.out   ../tpa50beta/corrode.out
echo '  9) diff cp.tpa        ../tpa50beta/cp.tpa'
diff cp.tpa        ../tpa50beta/cp.tpa
echo ' 10) diff cumrel.res    ../tpa50beta/cumrel.res'
diff cumrel.res    ../tpa50beta/cumrel.res
echo ' 11) diff cumrel_c.res  ../tpa50beta/cumrel_c.res'
diff cumrel_c.res  ../tpa50beta/cumrel_c.res
echo ' 12) diff cumrelse.out  ../tpa50beta/cumrelse.out'
diff cumrelse.out  ../tpa50beta/cumrelse.out
echo ' 13) diff deltaec.inp   ../tpa50beta/deltaec.inp'
diff deltaec.inp   ../tpa50beta/deltaec.inp
echo ' 14) diff diagnose.out  ../tpa50beta/diagnose.out'
diff diagnose.out  ../tpa50beta/diagnose.out
echo ' 15) diff drifts.dat    ../tpa50beta/drifts.dat'
diff drifts.dat    ../tpa50beta/drifts.dat
echo ' 16) diff drythick.dat  ../tpa50beta/drythick.dat'
diff drythick.dat  ../tpa50beta/drythick.dat
```

```
echo ' 17) diff dsfailt.dat   ../tpa50beta/dsfailt.dat'
diff dsfailt.dat   ../tpa50beta/dsfailt.dat
echo ' 18) diff dsfailt.def   ../tpa50beta/dsfailt.def'
diff dsfailt.def   ../tpa50beta/dsfailt.def
echo ' 19) diff dsfailt.inp   ../tpa50beta/dsfailt.inp'
diff dsfailt.inp   ../tpa50beta/dsfailt.inp
echo ' 20) diff dsfailt.out   ../tpa50beta/dsfailt.out'
diff dsfailt.out   ../tpa50beta/dsfailt.out
echo ' 21) diff ebscld.out    ../tpa50beta/ebscld.out'
diff ebscld.out    ../tpa50beta/ebscld.out
echo ' 22) diff ebsfail.def   ../tpa50beta/ebsfail.def'
diff ebsfail.def   ../tpa50beta/ebsfail.def
echo ' 23) diff ebsfail.inp   ../tpa50beta/ebsfail.inp'
diff ebsfail.inp   ../tpa50beta/ebsfail.inp
echo ' 24) diff ebsfilt.def   ../tpa50beta/ebsfilt.def'
diff ebsfilt.def   ../tpa50beta/ebsfilt.def
echo ' 25) diff ebsfilt.inp   ../tpa50beta/ebsfilt.inp'
diff ebsfilt.inp   ../tpa50beta/ebsfilt.inp
echo ' 26) diff ebsfilt.out   ../tpa50beta/ebsfilt.out'
diff ebsfilt.out   ../tpa50beta/ebsfilt.out
echo ' 27) diff ebsflo.dat    ../tpa50beta/ebsflo.dat'
diff ebsflo.dat    ../tpa50beta/ebsflo.dat
echo ' 28) diff ebsnef.dat    ../tpa50beta/ebsnef.dat'
diff ebsnef.dat    ../tpa50beta/ebsnef.dat
echo ' 29) diff ebsnef.out    ../tpa50beta/ebsnef.out'
diff ebsnef.out    ../tpa50beta/ebsnef.out
echo ' 30) diff ebsnef2.dat  ../tpa50beta/ebsnef2.dat'
diff ebsnef2.dat  ../tpa50beta/ebsnef2.dat
echo ' 31) diff ebspac.nuc    ../tpa50beta/ebspac.nuc'
diff ebspac.nuc    ../tpa50beta/ebspac.nuc
echo ' 32) diff ebsrel.def    ../tpa50beta/ebsrel.def'
diff ebsrel.def    ../tpa50beta/ebsrel.def
echo ' 33) diff ebsrel.inp    ../tpa50beta/ebsrel.inp'
diff ebsrel.inp    ../tpa50beta/ebsrel.inp
echo ' 34) diff ebssf.dat     ../tpa50beta/ebssf.dat'
diff ebssf.dat     ../tpa50beta/ebssf.dat
echo ' 35) diff ebstrh.dat    ../tpa50beta/ebstrh.dat'
diff ebstrh.dat    ../tpa50beta/ebstrh.dat
echo ' 36) diff ebstrhc.inp   ../tpa50beta/ebstrhc.inp'
diff ebstrhc.inp   ../tpa50beta/ebstrhc.inp
echo ' 37) diff echofail.dat  ../tpa50beta/echofail.dat'
diff echofail.dat  ../tpa50beta/echofail.dat
echo ' 38) diff echofilt.dat  ../tpa50beta/echofilt.dat'
diff echofilt.dat  ../tpa50beta/echofilt.dat
echo ' 39) diff epa_ave.out   ../tpa50beta/epa_ave.out'
diff epa_ave.out   ../tpa50beta/epa_ave.out
echo ' 40) diff epapktim.out  ../tpa50beta/epapktim.out'
diff epapktim.out  ../tpa50beta/epapktim.out
echo ' 41) diff failt.out     ../tpa50beta/failt.out'
diff failt.out     ../tpa50beta/failt.out
echo ' 42) diff fluoride.dat  ../tpa50beta/fluoride.dat'
diff fluoride.dat  ../tpa50beta/fluoride.dat
```

```
echo ' 43) diff frac_rel.out  ../tpa50beta/frac_rel.out'
diff frac_rel.out  ../tpa50beta/frac_rel.out
echo ' 44) diff gbioac1.dat   ../tpa50beta/gbioac1.dat'
diff gbioac1.dat   ../tpa50beta/gbioac1.dat
echo ' 45) diff gdefault.def  ../tpa50beta/gdefault.def'
diff gdefault.def  ../tpa50beta/gdefault.def
echo ' 46) diff gdefault.inp  ../tpa50beta/gdefault.inp'
diff gdefault.inp  ../tpa50beta/gdefault.inp
echo ' 47) diff gdosinc2.dat  ../tpa50beta/gdosinc2.dat'
diff gdosinc2.dat  ../tpa50beta/gdosinc2.dat
echo ' 48) diff gentoo.out    ../tpa50beta/gentoo.out'
diff gentoo.out    ../tpa50beta/gentoo.out
echo ' 49) diff genv.in       ../tpa50beta/genv.in'
diff genv.in       ../tpa50beta/genv.in
echo ' 50) diff genv.out      ../tpa50beta/genv.out'
diff genv.out      ../tpa50beta/genv.out
echo ' 51) diff gftrans.def   ../tpa50beta/gftrans.def'
diff gftrans.def   ../tpa50beta/gftrans.def
echo ' 52) diff gftrans.inp   ../tpa50beta/gftrans.inp'
diff gftrans.inp   ../tpa50beta/gftrans.inp
echo ' 53) diff ggamen.dat    ../tpa50beta/ggamen.dat'
diff ggamen.dat    ../tpa50beta/ggamen.dat
echo ' 54) diff ggenii.def    ../tpa50beta/ggenii.def'
diff ggenii.def    ../tpa50beta/ggenii.def
echo ' 55) diff ggenii.inp    ../tpa50beta/ggenii.inp'
diff ggenii.inp    ../tpa50beta/ggenii.inp
echo ' 56) diff ggenii.out    ../tpa50beta/ggenii.out'
diff ggenii.out    ../tpa50beta/ggenii.out
echo ' 57) diff ggrdf.dat     ../tpa50beta/ggrdf.dat'
diff ggrdf.dat     ../tpa50beta/ggrdf.dat
echo ' 58) diff gmedia.out    ../tpa50beta/gmedia.out'
diff gmedia.out    ../tpa50beta/gmedia.out
echo ' 59) diff gnewdf.dat    ../tpa50beta/gnewdf.dat'
diff gnewdf.dat    ../tpa50beta/gnewdf.dat
echo ' 60) diff grmdlib.dat   ../tpa50beta/grmdlib.dat'
diff grmdlib.dat   ../tpa50beta/grmdlib.dat
echo ' 61) diff gsccdf.res    ../tpa50beta/gsccdf.res'
diff gsccdf.res    ../tpa50beta/gsccdf.res
echo ' 62) diff gsccdf_c.res  ../tpa50beta/gsccdf_c.res'
diff gsccdf_c.res  ../tpa50beta/gsccdf_c.res
echo ' 63) diff gw_cb_ad.dat  ../tpa50beta/gw_cb_ad.dat'
diff gw_cb_ad.dat  ../tpa50beta/gw_cb_ad.dat
echo ' 64) diff gw_cb_ci.dat  ../tpa50beta/gw_cb_ci.dat'
diff gw_cb_ci.dat  ../tpa50beta/gw_cb_ci.dat
echo ' 65) diff gw_pb_ad.dat  ../tpa50beta/gw_pb_ad.dat'
diff gw_pb_ad.dat  ../tpa50beta/gw_pb_ad.dat
echo ' 66) diff gw_pb_ci.dat  ../tpa50beta/gw_pb_ci.dat'
diff gw_pb_ci.dat  ../tpa50beta/gw_pb_ci.dat
echo ' 67) diff gwccdf.res    ../tpa50beta/gwccdf.res'
diff gwccdf.res    ../tpa50beta/gwccdf.res
echo ' 68) diff gwccdf_c.res  ../tpa50beta/gwccdf_c.res'
diff gwccdf_c.res  ../tpa50beta/gwccdf_c.res
```

```
echo ' 69) diff gwork.buf     ../tpa50beta/gwork.buf'
diff gwork.buf     ../tpa50beta/gwork.buf
echo ' 70) diff gwpkdos.res   ../tpa50beta/gwpkdos.res'
diff gwpkdos.res   ../tpa50beta/gwpkdos.res
echo ' 71) diff gwpkds_c.res  ../tpa50beta/gwpkds_c.res'
diff gwpkds_c.res  ../tpa50beta/gwpkds_c.res
echo ' 72) diff gwttuzsz.res  ../tpa50beta/gwttuzsz.res'
diff gwttuzsz.res  ../tpa50beta/gwttuzsz.res
echo ' 73) diff infilper.res  ../tpa50beta/infilper.res'
diff infilper.res  ../tpa50beta/infilper.res
echo ' 74) diff inv1000.out   ../tpa50beta/inv1000.out'
diff inv1000.out   ../tpa50beta/inv1000.out
echo ' 75) diff lhs.csv       ../tpa50beta/lhs.csv'
diff lhs.csv       ../tpa50beta/lhs.csv
echo ' 76) diff lhs.inp       ../tpa50beta/lhs.inp'
diff lhs.inp       ../tpa50beta/lhs.inp
echo ' 77) diff lhs.out       ../tpa50beta/lhs.out'
diff lhs.out       ../tpa50beta/lhs.out
echo ' 78) diff lhse.out      ../tpa50beta/lhse.out'
diff lhse.out      ../tpa50beta/lhse.out
echo ' 79) diff maxrel.dat    ../tpa50beta/maxrel.dat'
diff maxrel.dat    ../tpa50beta/maxrel.dat
echo ' 80) diff maydtbl.dat   ../tpa50beta/maydtbl.dat'
diff maydtbl.dat   ../tpa50beta/maydtbl.dat
echo ' 81) diff mechfail.dat  ../tpa50beta/mechfail.dat'
diff mechfail.dat  ../tpa50beta/mechfail.dat
echo ' 82) diff mechfail.def  ../tpa50beta/mechfail.def'
diff mechfail.def  ../tpa50beta/mechfail.def
echo ' 83) diff mechfail.inp ../tpa50beta/mechfail.inp'
diff mechfail.inp ../tpa50beta/mechfail.inp
echo ' 84) diff mechfail.out  ../tpa50beta/mechfail.out'
diff mechfail.out  ../tpa50beta/mechfail.out
echo ' 85) diff multifaf.dat  ../tpa50beta/multifaf.dat'
diff multifaf.dat  ../tpa50beta/multifaf.dat
echo ' 86) diff multifbe.dat  ../tpa50beta/multifbe.dat'
diff multifbe.dat  ../tpa50beta/multifbe.dat
echo ' 87) diff mv.tpa        ../tpa50beta/mv.tpa'
diff mv.tpa        ../tpa50beta/mv.tpa
echo ' 88) diff nearfld.res   ../tpa50beta/nearfld.res'
diff nearfld.res   ../tpa50beta/nearfld.res
echo ' 89) diff nefii.dis     ../tpa50beta/nefii.dis'
diff nefii.dis     ../tpa50beta/nefii.dis
echo ' 90) diff nefii.inp     ../tpa50beta/nefii.inp'
diff nefii.inp     ../tpa50beta/nefii.inp
echo ' 91) diff nefii.out    ../tpa50beta/nefii.out'
diff nefii.out    ../tpa50beta/nefii.out
echo ' 92) diff nefii.rel     ../tpa50beta/nefii.rel'
diff nefii.rel     ../tpa50beta/nefii.rel
echo ' 93) diff nefiisz.dis   ../tpa50beta/nefiisz.dis'
diff nefiisz.dis   ../tpa50beta/nefiisz.dis
echo ' 94) diff nefiisz.inp   ../tpa50beta/nefiisz.inp'
diff nefiisz.inp   ../tpa50beta/nefiisz.inp
```

```
echo ' 95) diff nefiisz.out  ../tpa50beta/nefiisz.out'
diff nefiisz.out  ../tpa50beta/nefiisz.out
echo ' 96) diff nefiisz.src   ../tpa50beta/nefiisz.src'
diff nefiisz.src   ../tpa50beta/nefiisz.src
echo ' 97) diff nefiisz.vel   ../tpa50beta/nefiisz.vel'
diff nefiisz.vel   ../tpa50beta/nefiisz.vel
echo ' 98) diff nefiiuz.dis   ../tpa50beta/nefiiuz.dis'
diff nefiiuz.dis   ../tpa50beta/nefiiuz.dis
echo ' 99) diff nefiiuz.inp   ../tpa50beta/nefiiuz.inp'
diff nefiiuz.inp   ../tpa50beta/nefiiuz.inp
echo '100) diff nefiiuz.out   ../tpa50beta/nefiiuz.out'
diff nefiiuz.out   ../tpa50beta/nefiiuz.out
echo '101) diff nefiiuz.src   ../tpa50beta/nefiiuz.src'
diff nefiiuz.src   ../tpa50beta/nefiiuz.src
echo '102) diff nefiiuz.vel   ../tpa50beta/nefiiuz.vel'
diff nefiiuz.vel   ../tpa50beta/nefiiuz.vel
echo '103) diff nefmks.log    ../tpa50beta/nefmks.log'
diff nefmks.log    ../tpa50beta/nefmks.log
echo '104) diff npkdoset.res  ../tpa50beta/npkdoset.res'
diff npkdoset.res  ../tpa50beta/npkdoset.res
echo '105) diff npkdst_c.res  ../tpa50beta/npkdst_c.res'
diff npkdst_c.res  ../tpa50beta/npkdst_c.res
echo '106) diff nuclides.dat  ../tpa50beta/nuclides.dat'
diff nuclides.dat  ../tpa50beta/nuclides.dat
echo '107) diff organdf.dat   ../tpa50beta/organdf.dat'
diff organdf.dat   ../tpa50beta/organdf.dat
echo '108) diff pkmndose.out  ../tpa50beta/pkmndose.out'
diff pkmndose.out  ../tpa50beta/pkmndose.out
echo '109) diff pkreltim.res  ../tpa50beta/pkreltim.res'
diff pkreltim.res  ../tpa50beta/pkreltim.res
echo '110) diff pkrltm_c.res  ../tpa50beta/pkrltm_c.res'
diff pkrltm_c.res  ../tpa50beta/pkrltm_c.res
echo '111) diff rel_flow.out  ../tpa50beta/rel_flow.out'
diff rel_flow.out  ../tpa50beta/rel_flow.out
echo '112) diff relccdf.res   ../tpa50beta/relccdf.res'
diff relccdf.res   ../tpa50beta/relccdf.res
echo '113) diff relcum.out    ../tpa50beta/relcum.out'
diff relcum.out    ../tpa50beta/relcum.out
echo '114) diff releaset.out  ../tpa50beta/releaset.out'
diff releaset.out  ../tpa50beta/releaset.out
echo '115) diff relfrac.out   ../tpa50beta/relfrac.out'
diff relfrac.out   ../tpa50beta/relfrac.out
echo '116) diff relgwgs.res   ../tpa50beta/relgwgs.res'
diff relgwgs.res   ../tpa50beta/relgwgs.res
echo '117) diff repdes.dat    ../tpa50beta/repdes.dat'
diff repdes.dat    ../tpa50beta/repdes.dat
echo '118) diff rgwna.tpa     ../tpa50beta/rgwna.tpa'
diff rgwna.tpa     ../tpa50beta/rgwna.tpa
echo '119) diff rgwnapani.tpa ../tpa50beta/rgwnapani.tpa'
diff rgwnapani.tpa ../tpa50beta/rgwnapani.tpa
echo '120) diff rgwnapdw.tpa  ../tpa50beta/rgwnapdw.tpa'
diff rgwnapdw.tpa  ../tpa50beta/rgwnapdw.tpa
```

```
echo '121) diff rgwnapext.tpa ../tpa50beta/rgwnapext.tpa'
diff rgwnapext.tpa ../tpa50beta/rgwnapext.tpa
echo '122) diff rgwnapinh.tpa ../tpa50beta/rgwnapinh.tpa'
diff rgwnapinh.tpa ../tpa50beta/rgwnapinh.tpa
echo '123) diff rgwnapmlk.tpa ../tpa50beta/rgwnapmlk.tpa'
diff rgwnapmlk.tpa ../tpa50beta/rgwnapmlk.tpa
echo '124) diff rgwnappla.tpa ../tpa50beta/rgwnappla.tpa'
diff rgwnappla.tpa ../tpa50beta/rgwnappla.tpa
echo '125) diff rgwnr.tpa     ../tpa50beta/rgwnr.tpa'
diff rgwnr.tpa     ../tpa50beta/rgwnr.tpa
echo '126) diff rgwsa.tpa     ../tpa50beta/rgwsa.tpa'
diff rgwsa.tpa     ../tpa50beta/rgwsa.tpa
echo '127) diff rgwsap.tpa    ../tpa50beta/rgwsap.tpa'
diff rgwsap.tpa    ../tpa50beta/rgwsap.tpa
echo '128) diff rgwsr.tpa     ../tpa50beta/rgwsr.tpa'
diff rgwsr.tpa     ../tpa50beta/rgwsr.tpa
echo '129) diff rlccdf_c.res  ../tpa50beta/rlccdf_c.res'
diff rlccdf_c.res  ../tpa50beta/rlccdf_c.res
echo '130) diff rlgwgs_c.res  ../tpa50beta/rlgwgs_c.res'
diff rlgwgs_c.res  ../tpa50beta/rlgwgs_c.res
echo '131) diff seisbs1.dis   ../tpa50beta/seisbs1.dis'
diff seisbs1.dis   ../tpa50beta/seisbs1.dis
echo '132) diff seisbs2.dis   ../tpa50beta/seisbs2.dis'
diff seisbs2.dis   ../tpa50beta/seisbs2.dis
echo '133) diff smaydtbl.dat  ../tpa50beta/smaydtbl.dat'
diff smaydtbl.dat  ../tpa50beta/smaydtbl.dat
echo '134) diff sotnef.dat    ../tpa50beta/sotnef.dat'
diff sotnef.dat    ../tpa50beta/sotnef.dat
echo '135) diff sp.tpa        ../tpa50beta/sp.tpa'
diff sp.tpa        ../tpa50beta/sp.tpa
echo '136) diff totdos_c.res  ../tpa50beta/totdos_c.res'
diff totdos_c.res  ../tpa50beta/totdos_c.res
echo '137) diff totdose.res   ../tpa50beta/totdose.res'
diff totdose.res   ../tpa50beta/totdose.res
echo '138) diff tpa.inp       ../tpa50beta/tpa.inp'
diff tpa.inp       ../tpa50beta/tpa.inp
echo '139) diff tpa_.out      ../tpa50beta/tpa_.out'
diff tpa_.out      ../tpa50beta/tpa_.out
echo '140) diff tpameans.out  ../tpa50beta/tpameans.out'
diff tpameans.out  ../tpa50beta/tpameans.out
echo '141) diff trelease.out  ../tpa50beta/trelease.out'
diff trelease.out  ../tpa50beta/trelease.out
echo '142) diff weldfail.out  ../tpa50beta/weldfail.out'
diff weldfail.out  ../tpa50beta/weldfail.out
echo '143) diff wpflow.dat    ../tpa50beta/wpflow.dat'
diff wpflow.dat    ../tpa50beta/wpflow.dat
echo '144) diff wpflow.def    ../tpa50beta/wpflow.def'
diff wpflow.def    ../tpa50beta/wpflow.def
echo '145) diff wpsfail.res   ../tpa50beta/wpsfail.res'
diff wpsfail.res   ../tpa50beta/wpsfail.res
```

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

This is the output from running tpatest/scr416/diff_files > scr416files50.diff:

```
  1) diff FILENAME.DAT  ../tpa50beta/FILENAME.DAT
  2) diff NEFII.VEL     ../tpa50beta/NEFII.VEL
  3) diff airpkdos.res  ../tpa50beta/airpkdos.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
  4) diff arpkds_c.res ../tpa50beta/arpkds_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
  5) diff ashout.res    ../tpa50beta/ashout.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
  6) diff burnup.dat    ../tpa50beta/burnup.dat
  7) diff chlrdmf.dat   ../tpa50beta/chlrdmf.dat
  8) diff corrode.out   ../tpa50beta/corrode.out
  9) diff cp.tpa        ../tpa50beta/cp.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 10) diff cumrel.res    ../tpa50beta/cumrel.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 11) diff cumrel_c.res  ../tpa50beta/cumrel_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 12) diff cumrelse.out  ../tpa50beta/cumrelse.out
 13) diff deltaec.inp   ../tpa50beta/deltaec.inp
 14) diff diagnose.out  ../tpa50beta/diagnose.out
 15) diff drifts.dat    ../tpa50beta/drifts.dat
 16) diff drythick.dat  ../tpa50beta/drythick.dat
 17) diff dsfailt.dat   ../tpa50beta/dsfailt.dat
 18) diff dsfailt.def   ../tpa50beta/dsfailt.def
 19) diff dsfailt.inp   ../tpa50beta/dsfailt.inp
 20) diff dsfailt.out   ../tpa50beta/dsfailt.out
 21) diff ebscld.out    ../tpa50beta/ebscld.out
 22) diff ebsfail.def   ../tpa50beta/ebsfail.def
 23) diff ebsfail.inp   ../tpa50beta/ebsfail.inp
 24) diff ebsfilt.def   ../tpa50beta/ebsfilt.def
```

```
 25) diff ebsfilt.inp   ../tpa50beta/ebsfilt.inp
 26) diff ebsfilt.out   ../tpa50beta/ebsfilt.out
 27) diff ebsflo.dat    ../tpa50beta/ebsflo.dat
 28) diff ebsnef.dat    ../tpa50beta/ebsnef.dat
 29) diff ebsnef.out    ../tpa50beta/ebsnef.out
 30) diff ebsnef2.dat  ../tpa50beta/ebsnef2.dat
 31) diff ebspac.nuc    ../tpa50beta/ebspac.nuc
 32) diff ebsrel.def    ../tpa50beta/ebsrel.def
 33) diff ebsrel.inp    ../tpa50beta/ebsrel.inp
 34) diff ebssf.dat     ../tpa50beta/ebssf.dat
 35) diff ebstrh.dat    ../tpa50beta/ebstrh.dat
 36) diff ebstrhc.inp   ../tpa50beta/ebstrhc.inp
 37) diff echofail.dat  ../tpa50beta/echofail.dat
 38) diff echofilt.dat  ../tpa50beta/echofilt.dat
 39) diff epa_ave.out   ../tpa50beta/epa_ave.out
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 40) diff epapktim.out  ../tpa50beta/epapktim.out
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 41) diff failt.out     ../tpa50beta/failt.out
4c4
<  Tue Nov 12 11:10:06 2002
---
>  Tue Nov 12 11:17:47 2002
 42) diff fluoride.dat  ../tpa50beta/fluoride.dat
 43) diff frac_rel.out  ../tpa50beta/frac_rel.out
 44) diff gbioac1.dat   ../tpa50beta/gbioac1.dat
 45) diff gdefault.def  ../tpa50beta/gdefault.def
 46) diff gdefault.inp  ../tpa50beta/gdefault.inp
 47) diff gdosinc2.dat  ../tpa50beta/gdosinc2.dat
 48) diff gentoo.out    ../tpa50beta/gentoo.out
 49) diff genv.in       ../tpa50beta/genv.in
 50) diff genv.out      ../tpa50beta/genv.out
 51) diff gftrans.def   ../tpa50beta/gftrans.def
 52) diff gftrans.inp   ../tpa50beta/gftrans.inp
 53) diff ggamen.dat    ../tpa50beta/ggamen.dat
 54) diff ggenii.def    ../tpa50beta/ggenii.def
 55) diff ggenii.inp    ../tpa50beta/ggenii.inp
 56) diff ggenii.out    ../tpa50beta/ggenii.out
27c27
<  Executed on: 12-Nov-02 at 11:10:20
   Page     0
---
>  Executed on: 12-Nov-02 at 11:18:05
   Page     0
35c35
```

```
<  Executed on: 12-Nov-02 at 11:10:20
   Page A.  1
---
>  Executed on: 12-Nov-02 at 11:18:05
   Page A.  1
 57) diff ggrdf.dat     ../tpa50beta/ggrdf.dat
 58) diff gmedia.out    ../tpa50beta/gmedia.out
7c7
<  Executed on: 12-Nov-02 at 11:10:21
   Page     1
---
>  Executed on: 12-Nov-02 at 11:18:05
   Page     1
 59) diff gnewdf.dat    ../tpa50beta/gnewdf.dat
 60) diff grmdlib.dat   ../tpa50beta/grmdlib.dat
 61) diff gsccdf.res    ../tpa50beta/gsccdf.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 62) diff gsccdf_c.res  ../tpa50beta/gsccdf_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 63) diff gw_cb_ad.dat  ../tpa50beta/gw_cb_ad.dat
 64) diff gw_cb_ci.dat  ../tpa50beta/gw_cb_ci.dat
 65) diff gw_pb_ad.dat  ../tpa50beta/gw_pb_ad.dat
 66) diff gw_pb_ci.dat  ../tpa50beta/gw_pb_ci.dat
 67) diff gwccdf.res    ../tpa50beta/gwccdf.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 68) diff gwccdf_c.res  ../tpa50beta/gwccdf_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 69) diff gwork.buf     ../tpa50beta/gwork.buf
 70) diff gwpkdos.res   ../tpa50beta/gwpkdos.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 71) diff gwpkds_c.res  ../tpa50beta/gwpkds_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 72) diff gwttuzsz.res  ../tpa50beta/gwttuzsz.res
3c3
```

```
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 73) diff infilper.res  ../tpa50beta/infilper.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 74) diff inv1000.out   ../tpa50beta/inv1000.out
 75) diff lhs.csv       ../tpa50beta/lhs.csv
 76) diff lhs.inp       ../tpa50beta/lhs.inp
 77) diff lhs.out       ../tpa50beta/lhs.out
 78) diff lhse.out      ../tpa50beta/lhse.out
 79) diff maxrel.dat    ../tpa50beta/maxrel.dat
 80) diff maydtbl.dat   ../tpa50beta/maydtbl.dat
 81) diff mechfail.dat  ../tpa50beta/mechfail.dat
2c2
< DATE/Time: Tue Nov 12 11:08:03 2002
---
> DATE/Time: Tue Nov 12 11:15:17 2002
 82) diff mechfail.def  ../tpa50beta/mechfail.def
 83) diff mechfail.inp ../tpa50beta/mechfail.inp
2c2
< DATE/Time: Tue Nov 12 11:08:03 2002
---
> DATE/Time: Tue Nov 12 11:15:17 2002
 84) diff mechfail.out  ../tpa50beta/mechfail.out
 85) diff multifaf.dat  ../tpa50beta/multifaf.dat
 86) diff multifbe.dat  ../tpa50beta/multifbe.dat
 87) diff mv.tpa        ../tpa50beta/mv.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 88) diff nearfld.res   ../tpa50beta/nearfld.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
 89) diff nefii.dis     ../tpa50beta/nefii.dis
2c2
< EXECUTION DATE   Nov 12   AND TIME  11:10:1
---
> EXECUTION DATE   Nov 12   AND TIME  11:17:5
 90) diff nefii.inp     ../tpa50beta/nefii.inp
 91) diff nefii.out    ../tpa50beta/nefii.out
8c8
<      *              EXECUTION DATE  Nov 12   AND TIME  11:10:1
          *
---
>      *              EXECUTION DATE  Nov 12   AND TIME  11:17:5
          *
```

```
 92) diff nefii.rel     ../tpa50beta/nefii.rel
 93) diff nefiisz.dis   ../tpa50beta/nefiisz.dis
2c2
< EXECUTION DATE   Nov 12   AND TIME  11:10:1
---
> EXECUTION DATE   Nov 12   AND TIME  11:17:5
 94) diff nefiisz.inp   ../tpa50beta/nefiisz.inp
 95) diff nefiisz.out  ../tpa50beta/nefiisz.out
8c8
<      *              EXECUTION DATE  Nov 12   AND TIME  11:10:1
          *
---
>      *              EXECUTION DATE  Nov 12   AND TIME  11:17:5
          *
 96) diff nefiisz.src   ../tpa50beta/nefiisz.src
 97) diff nefiisz.vel   ../tpa50beta/nefiisz.vel
 98) diff nefiiuz.dis   ../tpa50beta/nefiiuz.dis
2c2
< EXECUTION DATE   Nov 12   AND TIME  11:10:0
---
> EXECUTION DATE   Nov 12   AND TIME  11:17:5
 99) diff nefiiuz.inp   ../tpa50beta/nefiiuz.inp
100) diff nefiiuz.out   ../tpa50beta/nefiiuz.out
8c8
<      *              EXECUTION DATE  Nov 12   AND TIME  11:10:0
          *
---
>      *              EXECUTION DATE  Nov 12   AND TIME  11:17:5
          *
101) diff nefiiuz.src   ../tpa50beta/nefiiuz.src
102) diff nefiiuz.vel   ../tpa50beta/nefiiuz.vel
103) diff nefmks.log    ../tpa50beta/nefmks.log
104) diff npkdoset.res  ../tpa50beta/npkdoset.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
105) diff npkdst_c.res  ../tpa50beta/npkdst_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
106) diff nuclides.dat  ../tpa50beta/nuclides.dat
107) diff organdf.dat   ../tpa50beta/organdf.dat
108) diff pkmndose.out  ../tpa50beta/pkmndose.out
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002

---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002

109) diff pkreltim.res  ../tpa50beta/pkreltim.res
```

```
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
110) diff pkrltm_c.res  ../tpa50beta/pkrltm_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
111) diff rel_flow.out  ../tpa50beta/rel_flow.out
112) diff relccdf.res   ../tpa50beta/relccdf.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
113) diff relcum.out    ../tpa50beta/relcum.out
114) diff releaset.out  ../tpa50beta/releaset.out
3c3
<  Tue Nov 12 11:10:07 2002
---
>  Tue Nov 12 11:17:47 2002
115) diff relfrac.out   ../tpa50beta/relfrac.out
116) diff relgwgs.res   ../tpa50beta/relgwgs.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
117) diff repdes.dat    ../tpa50beta/repdes.dat
118) diff rgwna.tpa     ../tpa50beta/rgwna.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
119) diff rgwnapani.tpa ../tpa50beta/rgwnapani.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
120) diff rgwnapdw.tpa  ../tpa50beta/rgwnapdw.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
121) diff rgwnapext.tpa ../tpa50beta/rgwnapext.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
122) diff rgwnapinh.tpa ../tpa50beta/rgwnapinh.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
```

```
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
123) diff rgwnapmlk.tpa ../tpa50beta/rgwnapmlk.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
124) diff rgwnappla.tpa ../tpa50beta/rgwnappla.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
125) diff rgwnr.tpa     ../tpa50beta/rgwnr.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
126) diff rgwsa.tpa     ../tpa50beta/rgwsa.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
127) diff rgwsap.tpa    ../tpa50beta/rgwsap.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
128) diff rgwsr.tpa     ../tpa50beta/rgwsr.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
129) diff rlccdf_c.res  ../tpa50beta/rlccdf_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
130) diff rlgwgs_c.res  ../tpa50beta/rlgwgs_c.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
131) diff seisbs1.dis   ../tpa50beta/seisbs1.dis
132) diff seisbs2.dis   ../tpa50beta/seisbs2.dis
133) diff smaydtbl.dat  ../tpa50beta/smaydtbl.dat
134) diff sotnef.dat    ../tpa50beta/sotnef.dat
135) diff sp.tpa        ../tpa50beta/sp.tpa
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
136) diff totdos_c.res  ../tpa50beta/totdos_c.res
3c3
```

```
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
137) diff totdose.res   ../tpa50beta/totdose.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
138) diff tpa.inp       ../tpa50beta/tpa.inp
139) diff tpa_.out      ../tpa50beta/tpa_.out
140) diff tpameans.out  ../tpa50beta/tpameans.out
7c7
< ** ** submitted for the run on Tue Nov 12 11:07:43 2002.
---
> ** ** submitted for the run on Tue Nov 12 11:14:53 2002.
141) diff trelease.out  ../tpa50beta/trelease.out
142) diff weldfail.out  ../tpa50beta/weldfail.out
143) diff wpflow.dat    ../tpa50beta/wpflow.dat
144) diff wpflow.def    ../tpa50beta/wpflow.def
145) diff wpsfail.res   ../tpa50beta/wpsfail.res
3c3
<  TPA 5.0beta, Job started: Tue Nov 12 11:07:43 2002
---
>  TPA 5.0beta, Job started: Tue Nov 12 11:14:53 2002
```

There are no differences in the files from the two runs other than date/time. So, it looks like my changes for SCR (#416) cause no ill effects.

Added five items to my list of potential code changes and sent original doc to Ron. Added:

43.01 Replace hard code usage of number 22 for SelectAppendFile w/ use of parameter that is set in only one place (esp. in files updated for SCR #395).

43.02 Look at usage of variables containing 'sa' in uzft.f and szft.f. In uzft.f, sa stands for subarea; in szft.f, it sometimes stands for subarea and sometimes streamtube. Eventually, it becomes the number of subareas/streamtubes selected for inclusion in the NEFII.VEL files. However, the sa and associated comments is confusing when you are looking at streamtubes. Numsalayers is being saved in the include file uzszft.i. I don't know whether a problem could arise is numsalayers is saved in uzft.f and then it is reset in szft.f.

43.03 See if we can do away with the parameter MAXNEF in uzft.f and reduce the size of some of the arrays.

43.04 We probably should confirm that everything required is being written out to the .res files and that they are still in the NRC-approved format per Sitakanta's comments at 11-11-02 Code Walkthrough.

43.05 Ditto on updating header blocks w/ implementation date and Chris Grossman's name and number.

I know I need to look at my new subroutines and make sure I put in an implementation date.

11-13-2002 -

Started test plan for SCR #416. Discovered I hadn't really changed the parameter (maxlayers = 20) to a value of 15. When I did, the code generated different numbers and broke NEFTRAN because a) length in Migration Path in nefii.inp was to large for format, b) por(i) was set to 0.0 for 'SAV' layer; and c) rdi was set to 0.0 for 'SAV' layer. NEFTRAN bombs for immobile porosities that are zero and RDs that are less than 1.0. Need to update SCR #412 and add more range checking to trap these errors. Check uzft.f and see if the potential for bombing exists there, too. In the meantime, since it seems that 20 is a magic number, try TPA run w/ maxlayers set to 20 and see if the numbers revert back to TPA 5.0beta numbers.

Added new item to tpa_changes.wpd:

Seems to be something magic about the number 20 for maxlayers (see dcagw.f, uzft.f, and szft.f). Why does 20 work and 15 fail miserably. Should nearlegs in dcagw.f be dimensioned by maxlayers or maxtubes or something else. Since nearlegs is storing streamtube width would a better name be helpful? Should there be a maxsalayers = 20 and a maxuzlayers = 50. Do we need to add names for sz layers (SAV and STFF)? If using 20 works, then SCR #416 is done, but need to redo #412 and add the mystery of 20 to the list of things to look at or change in tpa.

Maxlayers = 20 worked. Have noted the problem when the value changes from 20 to 15; hopefully will be able to track it down later.

Updated scr_416.wpd and tp_scr_416.wpd and gave electronic and hard copies to Ron.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Made mods for SCR #412. Updated scr_412.wpd and gvae electronic and hard copies to Ron.

11-14-2002 -

Started run in SCR #416 for 10 realizations. Successful (see tpa_10real.out). Ran for 10 realizations at 100,000 years. Also, successful (see tpa_10r100k.out).

Copied tpa.e from scr/416 to use in finishing test plan for SCR #395. I had left off in sl-1 ready to do the spreadsheet. Started from there. Sl-1 passed. Ran sl-2; it passed. Ran sl-3. It passed, too, if you consider that the file dcf.cum is no longer generated. Now you get a dcfgs.cum and dcfgw.cum instead. Re-word test plan accordingly. This test plan (tp_scr395.wpd) has passed overall.

Received a zip file from Ron for SCR #394. I am supposed to integrate changes made by George Adams (version 4.1.1) into TPA 5.0beta, and test it after I finish w/ #395.

11-15-2002 -

Updated the test plan for SCR #395. Gave hard copy to Ron and signed SCR. Made directory listing of scr395 directory and all subdirectories. Burned CD. Gave listing and CD to Ron.

Started comparing the files modified for SCR #394 (v4.1.1) to try and determine what changes need to be made to 5.0 to incorporate the modifications. Looking at exec.f, ebsrel.f, ebsfilt,f and releaset.f. Had to convert the version 4.1.1 files from dos format to unix. Saved and printed out the diff files. Will incorporate George's changes (v4.1.1.) into v5.0beta (5.0betaB, if it's available in time).

UNIX NOTE: use "dos2unix <filename> <newfilename>" to remove the ctrl-M at the end of the lines. The command "ls -R" gives a recursive listing of the current directory and its subdirectories.

11-18-2002 -

Code walkthrough (George seismo) 2:00 - 4:00 p.m. plus a few minutes. There seemed to be a lot of questions about how this new version of seismo, the drip shield and drift should interrelate w/ other modules and what data needs to be accumulated. Next walkthrough is next Monday afternoon.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

SCR #394: Merged in George's changes from 4.1.1 to 5.0beta in ebsfilt.f, and ebsrel.f. Started on exec.f. Still releaset.f to go after that.

11-19-2002 - 11 - 22-2002 -

SCR #394 put on hold for now. New task for Sitakanta; needs some results ASAP. He's leaving Friday for a month.

Flow through the invert and release is determined by ebsfilt (called by ebsrel) which considers decay but not the chaining process like NEFTRAN does. The question is: are we losing much in accuracy by doing it this way? Would it be better to send the invert through NEFTRAN, too?

SETUP: Use TPA version 5.0betaB. Ran TPA in cscherer/tpatest/invert/tpa50betaB. Ran nefmks.e in cscherer/tpatest/invert/nefmks. Saved data from runs in subdirectories in both places. Also used cscherer/tpatest/invert/compare to store some files and to use excel to compare results from one run while next run going on in tpa50betaB and/or nefmks subdirectories.

To use nefmks for invert, we'll need the appropriate input files: nefii.inp and sotnef.dat. For the two calls to nefmks, nefii.inp is created by uzft.f and szft.f.. There is no code in place that could do the same for invert, so we need to take an existing nefii.inp and modify it to work for invert. The data of interest is release rates stored in the array ciperyrinsaintoloweruz in the TPA code. Ebsrel uses this array to write ebsnef.dat. Sotnef.dat is created by ebsfilt.f from ebsnef.dat. The output from ebsfilt.f is ebsnef2.dat. (Note: the file ebsnef2.dat is padded w/ blanks; this increases its size significantly.) Once we send invert through nefmks, we can compare the ebsnef2.dat file with the nefii.dis file created from the special nefmks run. Information in the constructed nefii.inp should be consistent with tpa.inp and data used in ebsfilt.f.

PERTINENT DATA IN TPA.INP: Sitakanta wanted to look at decreased flow rate so that flow would be matrix-only flow (AAMAIAS) and to use flowthrough model instead of bathtub for the initially defective failures (WCM_I).

>>> DCAGS <<<
ArealAverageMeanAnnualinfiltrationAtStart[mm/yr]

>>> EBSREL <<<
WaterContactMode_Initial (Other WaterContactMode parameters not used)
RD_Invert_. . . (for each of 20 elements) - copy these values to nefii.inp for invert

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

>>> EBSFILT <<<
InvertRockPorosity - started set to .3; varying this affects ksat in ebsfilt.inp & thus velocity
InvertThickness (don't use InvertHeight in Seismo section)
InvertDiffusionCoefficient[m^2/yr] - not used in tests
InvertMatrixPermeability[m^2] - not used in tests

PERTINENT DATA IN NEFII.INP:

1) Number of Legs - must be at least two - won't accept just one; this is artificial, so, for the initial tests divide the .75 width of the invert into two equal legs w/ all characteristics the same.

2) Velocity - affects the number of time steps generated by NEFTRAN. Not currently calculated in ebsfilt.f. Used uzft.f velocity calculation (pore velocity = infil/por*sat) as model for test. TPA writes out infilper.res which reports the average reflux rate per saved time step. Sum these rates and divide by number of time steps to get average relux rate over all time steps. Used this value for infil. Ebsfilt.f reads in flux, por, ksat, x, dx from ebsfilt.inp. Use por and ksat values from ebsfilt.inp in the velocity calculation. Edit nefii.inp and use calculated velocity in velocity fields for MIGRATION PATH PROPERTY ARRAY.

3) Porosity (Immobile, InvertRockPorosity) - can't be zero, affects the number of time steps generated by NEFTRAN. Set in both tpa.inp and nefii.inp. Should be the same in both. Written out to ebsfilt.inp and read in by ebsfilt.f. Used both 0.3 and 0.1 for different tests.

4) Dispersion: use .1 - affects the number of time steps generated by NEFTRAN.

5) RDs - use the 20 "RD_Invert_... " sampled parameters from tpa.inp.

6) Diffusion - for now, leave it turned off (set to 0).

7) Mobile Porosity - no affect? leave set to 1.0 all elements.

8) Number of Times Steps - should be set correctly in nefii.inp to match the TPA run.

PROCESS:

1) Create nefii.inp. Starting with a nefiisz.inp from before the addition of time-dependent velocities (NEFII.VEL or nefiisz.vel) so we can use constant velocity. An example of the constructed nefii.inp follows (with pertinent data highlighted):

```
 AREA & PATH #S  1,   1  INFIL. =  0.000E+00  MASS FRC.=  0.100E+01
     GROUP 1 - OPTIONS, STANDARD PRINT, 0=NO PRINT
    1     LIST IF ALL THE  PARAMETERS
    1     THE NETWORK LEG PROPERTIES
    1     THE NETWORK JUNCTION PROPERTIES
    1     THE MIGRATION PATH PROPERTIES ARRAY
    1     THE DECAY CHAIN PROPERTIES ARRAY
    1     THE ELEMENT PROPERTIES ARRAY
    1     FLUID VELOCITIES & SOURCE/DISCH FLOWS
    1     SUBCHAINS, PRODUCTION & VELOCITIES
    1     SPACE & TIME STEPS AND SOURCE TYPE
    1     TIME INDEPENDENT OUTPUT, IF +-N, RATES/CONCS EVERY NTH TIME STEP
    0     DISCHARGE SUMMARY
     EXTERNAL FILES, NONZERO => FILE WRITTEN/READ
    0     SRATE.DAT, SOURCE RATES (CI/Y) - WRITTEN TO UNIT 25
   -1     nefii.dis, DISCHARGE RATES (CI/Y) - WRITTEN TO UNIT 30
    1     EXTERNAL SOURCE FLAG (=1 =>READ UNIT 14)
    1     SAMP.DAT, SAMPLED DATA FOR REPEATED TRIALS - READ FROM UNIT 10
    0     VFIELD.DAT, UNIT 11, TIME-DEPENDENT VELOCITIES READ FROM
    0     SFLOW.DAT, UNIT 12, TIME-DEPENDENT SOURCE FLOW RATES READ FROM
                RUN CONTROLS
    0     USE DVM (0) OR ANALYTIC SOLN (NONZERO)
    1     SOLVE NETWORK (0) OR INPUT VELOCITIES (NONZERO)
    0     LEACH (0), SOLUBILITIES (1), OR BOTH (2) FOR SOURCE RATES
    0     FLOWTHRU (0), MIXCELL (1), OR CHOOSE (2) FOR SOURCE RATES
    0     CONSTANT (0) OR EXPONENTIAL (1) LEACH RATE MODEL
    0     FORCE (NONZERO) USE OF LEG-TO-LEG TRANSFER ALGORITHM
                DEBUG PRINTS, NONZERO => PRINT
    0     DVM MULTIPLIERS FOR TRANSPORT, DISCHARGE, AND SOURCE
    0     DIAGNOSTIC INFO FOR SUBROUTINE BAND
    0     DIAGNOSTIC INFO FOR FUNCTION GIT
    0     MATRIX/VECTOR SYSTEM SOLVED FOR FLOW
    0     THE DATA READ FROM UNIT 10, SAMP.DAT
    1     CONC. FLAG (=1 => WRITE TO UNIT 29 IN CI/M**3)
    1     INVENTORY AT RELEASE TIME
    0     RESTRICTIONS PLACED ON THE TIME STEPS
    0     LEG/JUNCTION & JUNCTION/LEG CONNECTIONS
    0     VELOCITY FIELDS FROM UNIT 11
    0     THE ATOM COUNT SUMMARY
    0     JUNCTION PRESSURES & LEG FLOWRATES
    0     LEG-TO-LEG TRANSFER FRACTIONS
    0     TIME SPENT AS EACH ISOTOPE IN A SUBCHAIN
    0     TIME DEPENDENT FLOWRATES THROUGH SOURCE REGIME
                GROUP 2  -  PROBLEM SIZE
    2     NUMBER OF NETWORK LEGS
    0     NUMBER OF NETWORK JUNCTIONS
    2     NUMBER OF MIGRATION PATH LEGS
   18     NUMBER OF DECAY CHAINS INPUT
   20     NUMBER OF ELEMENTS INPUT
   CHN1  CHN2  CHN3  CHN4  CHN5  - # PER CHAIN
    2    3    2    1    4    1    1    1    1    1
    1    1    1    1    2    2    1    1
```

```
   CHN1  CHN2  CHN3  CHN4  CHN5 - TRANSPORT-0, SOURCE-1, NEITHER-2
    0     0     0     0     0     0     0     0     0     0
    0     0     0     0     0     0     0     0
                      GROUP 3 - SOURCE/FLOW PARAMETERS
 0.100E+01      FRACTION OF INVENTORY ACCESSED
 0.000E+00      PORE VOLUME CONTAINING SOURCE (M**3) USED FOR MIXING CELL
 0.000E+00      PORE AREA OF SOURCE  (M**2)
 0.100E-04      LEACH RATE (1/Y) IF LEACH LTD
     1000.00      DENSITY OF SOURCE FLUID (KG/M**3)
 0.000E+00      PORE AREA AT DISCHARGE (M**2)
                      GROUP 4 - TIME PARAMETERS
 0.300E+05      TIME TO END OF SIMULATION (Y)
 0.000E+00      TIME OF ONSET OF MIGRATION (Y)
 0.000E+00      TIME OF ONSET OF LEACHING (Y)
 0.000E+00      TIME STEP FOR SOURCE (Y)
CHAIN1 CHAIN2 CHAIN3 CHAIN4 CHAIN5 TIME STEPS BY CHAIN (Y)
  0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0
  0.0    0.0    0.0    0.0    0.0    0.0    0.0    0.0
          GROUP 5 - INCREMENT DETERMINATION
    0     USE DEFAULTS (0) OR SUPPLY ALL 16 VALUES
          NETWORK LEG PROPERTIES ARRAY
LEG  INLET  OUTLET    LENGTH    AREA    HYDRAULIC   BRINE
 #    JCT    JCT       (M)     (M**2)   K  (M/YR)   CONC.
 1      1      2       0.375     0.0       0.0       0.0
 2      2      3       0.375     0.0       0.0       0.0
                      MIGRATION PATH PROPERTIES ARRAY
LEG   DISPERS.    SPA. STEP  DIFFUS  MOBILE  IMMOB   MASS  XFER   VELOCITY
 #      (M)         (M)     N/Y=0/1  POROS.  POROS.  COEF(1/Y)     (M/YR)
  1 1.0000E-01     0.0      0    1.000E-03   3.000E-01   0.000E+00   1.0621E-01
  2 1.0000E-01     0.0      0    1.000E-03   3.000E-01   0.000E+00   1.0621E-01
                                DECAY CHAIN ARRAY
NAME      ATM#  ELE-   LOC  PAR  PAR  FRAC  FRAC     INVEN    HALF LIFE  WEIGH
  A6     (AMU)  MENT   NDX   #1   #2  FROM1 FROM2     (CI)       (Y)      FAC
'CM246 '  246.     1     1    0    0  0.000  0.000   0.000E+00 0.473E+04
0.100
'U238  '  238.     2     2    1    0  1.000  0.000   0.000E+00 0.447E+10
0.100
'CM245 '  245.     1     1    0    0  0.000  0.000   0.000E+00 0.850E+04
0.100
'AM241 '  241.     3     2    1    0  1.000  0.000   0.000E+00 0.432E+03
0.100
'NP237 '  237.     4     3    2    0  1.000  0.000   0.000E+00 0.214E+07
0.100
'AM243 '  243.     3     1    0    0  0.000  0.000   0.000E+00 0.738E+04
0.100
'PU239 '  239.     5     2    1    0  1.000  0.000   0.000E+00 0.241E+05
0.100
'PU240 '  240.     5     1    0    0  0.000  0.000   0.000E+00 0.654E+04
0.100
'U234  '  234.     2     1    0    0  0.000  0.000   0.000E+00 0.244E+06
0.100
```

```
'TH230 '  230.    6    2    1    0  1.000  0.000   0.000E+00 0.770E+05
0.010
'RA226 '  226.    7    3    2    0  1.000  0.000   0.000E+00 0.160E+04
0.100
'PB210 '  210.    8    4    3    0  1.000  0.000   0.000E+00 0.223E+02
1.000
'CS135 '  135.    9    1    0    0  0.000  0.000   0.000E+00 0.230E+07
1.000
'I129  '  129.   10    1    0    0  0.000  0.000   0.000E+00 0.157E+08
0.100
'TC99  '   99.   11    1    0    0  0.000  0.000   0.000E+00 0.213E+06
10.000
'NI59  '   59.   12    1    0    0  0.000  0.000   0.000E+00 0.800E+05
1.000
'C14   '   14.   13    1    0    0  0.000  0.000   0.000E+00 0.573E+04
0.100
'SE79  '   79.   14    1    0    0  0.000  0.000   0.000E+00 0.110E+07
1.000
'NB94  '   94.   15    1    0    0  0.000  0.000   0.000E+00 0.203E+05
1.000
'CL36  '   36.   16    1    0    0  0.000  0.000   0.000E+00 0.301E+06
1.000
'JC246 '  246.   17    1    0    0  0.000  0.000   0.000E+00 0.473E+04
0.000
'JC245 '  245.   17    1    0    0  0.000  0.000   0.000E+00 0.850E+04
0.000
'JA241 '  241.   18    2    1    0  1.000  0.000   0.000E+00 0.432E+03
0.000
'JA243 '  243.   18    1    0    0  0.000  0.000   0.000E+00 0.738E+04
0.000
'JP239 '  239.   19    2    1    0  1.000  0.000   0.000E+00 0.241E+05
0.000
'JP240 '  240.   19    1    0    0  0.000  0.000   0.000E+00 0.654E+04
0.000
'JT230 '  230.   20    1    0    0  0.000  0.000   0.000E+00 0.770E+05
0.000
                    ELEMENT PROPERTIES ARRAY
ELEM.    SOLUBILITY    LEG   MOBIL RD        IMMOBILE RD
INDEX      (KG/KG)      #
    1    0.000E+00      1    0.100E+01       6.000E+03
                        2    0.100E+01       6.000E+03
    2    0.000E+00      1    0.100E+01       6.010E+02
                        2    0.100E+01       6.010E+02
    3    0.000E+00      1    0.100E+01       3.000E+03
                        2    0.100E+01       3.000E+03
    4    0.000E+00      1    0.100E+01       1.200E+03
                        2    0.100E+01       1.200E+03
    5    0.000E+00      1    0.100E+01       3.000E+03
                        2    0.100E+01       3.000E+03
    6    0.000E+00      1    0.100E+01       3.000E+03
                        2    0.100E+01       3.000E+03
    7    0.000E+00      1    0.100E+01       6.010E+02
```

		2	0.100E+01	6.010E+02
8	0.000E+00	1	0.100E+01	3.010E+02
		2	0.100E+01	3.010E+02
9	0.000E+00	1	0.100E+01	1.210E+02
		2	0.100E+01	1.210E+02
10	0.000E+00	1	0.100E+01	0.700E+01
		2	0.100E+01	0.700E+01
11	0.000E+00	1	0.100E+01	0.100E+01
		2	0.100E+01	0.100E+01
12	0.000E+00	1	0.100E+01	6.100E+01
		2	0.100E+01	6.100E+01
13	0.000E+00	1	0.100E+01	6.100E+01
		2	0.100E+01	6.100E+01
14	0.000E+00	1	0.100E+01	0.100E+01
		2	0.100E+01	0.100E+01
15	0.000E+00	1	0.100E+01	6.100E+02
		2	0.100E+01	6.100E+02
16	0.000E+00	1	0.100E+01	0.100E+01
		2	0.100E+01	0.100E+01
17	0.000E+00	1	0.100E+01	6.000E+03
		2	0.100E+01	6.000E+03
18	0.000E+00	1	0.100E+01	3.000E+03
		2	0.100E+01	3.000E+03
19	0.000E+00	1	0.100E+01	3.000E+03
		2	0.100E+01	3.000E+03
20	0.000E+00	1	0.100E+01	3.000E+03
		2	0.100E+01	3.000E+03

2) Set sampled parameters in tpa.inp.

ArealAverageMeanAnnualinfiltrationAtStart[mm/yr] = 4.0
WaterContactMode_Initial (0=BathTub,1=FlowThrough) = 1
RD_Invert_. . . (for each of 20 elements) - leave as is
InvertRockPorosity = 0.3 or 0.1
InvertThickness[m] = 0.75

SimulationTime[yr] = 1.0e4, 1.0e5, or 3.0e4 for different tests

NumberOfRealizations = 1
StartAtSubArea = 6 (picked at random to limit run to 1 subarea)
StopAtSubArea = 6 (picked at random)

3) Run tpa.e. Save ebsnef.dat and ebsnef2.dat.

4) Create sotnef.dat from ebsnef.dat. Replace header with the appropriate header for sotnef.dat and add line 0.0 0.0 after each element id in ebsnef.dat. Make sure that number of nuclides and number of steps is correct in header.

ebsnef.dat header:

```
release rate [ci/yr]

   27  300
```

sotnef.dat header:

```
TITLE - Source release rates in Ci/yr for NEFTRAN.
   1 Sub-area number
   27  301 number of nuclides, number of times
```

Note that in ebsnef.dat, there is one less time step listed for each nuclide. The initial time step is not there. Sotnef.dat requires the initial time step, as follows:

```
CM246
   0.  0.
```

5) Run nefmks.e. Save nefii.dis.

6) Edit nefii.dis (vi seems to accept the largest file) and save the lines for the elements of interest (Np237, Am243, Pu239, Tc99, and Pb210). (Note: Sometimes, if the nefii.dis file was very large, I would write out the elements I wanted to several smaller files. Once, with a file that was too large for any of my editors to handle, I captured the first part of the file using 'cat > filename' and aborting part way through. I was able to create a file small enough for vi to handle, but I only could capture Np237, Am243, and Pu239. Tc99 and Pb210 were in the latter part of the file. Need to look at awk, grep and other unix tools to see if I can't come up with a good way to look at specific parts of files. Of course, I could always write a program to open the file and copy out the parts I wanted, but it might not be worth the effort.)

7) Open ebsnef2.dat in excel and group the elements of interest in columns. Delete the rest. First column is time steps. Subsequent columns are the elements you want to graph. Save as ebsnef2.xls.

8) Using excel, compare (graph - scatter plot, smooth line) output from tpa.e (ebsnef2.dat) and from nefmks.e (nefii.dis). (Usually, I started by opening ebsnef2.xls and bringing in the data I needed from nefii.dis. Worked best to graph one element per graph with both ebsnef2.dat and

nefii.dis data. If all the elements were on the graph, it obscured some trends because of the differences in release rate. Tc99 needed to be graphed on a separate y axis and Am243 tended to overpower Np237 and Pu239.) Graphs stored on separate sheets.

ELEMENTS TO ANALYZE: Initially: Np237, Am243, Pu239, and Tc99. Added Pb210 for last analysis run 11/22/02.

PROBLEMS ENCOUNTERED:

Excel limitations:

65532 (?) rows max can be read into Excel: if your data file is longer than 65,532 lines, then you need to break the file into smaller files before you open the files in Excel.

32000 data points max per series: One series on an Excel graph can contain up to 32000 data pts, no more. So if one element has more than 32, rows, you need to graph it as 2 different series.

Nefmks taking too long, nefii.dis file too large to process, too many time steps generated: sometimes, after a long run (the more time steps NEFTRAN decides to use, the longer the run takes, and the bigger the nefii.dis & nefii.out files will be), the resulting output was not usable. Obviously, the bigger the data files, the longer it took to graph them.

Disk space: The long NEFTRAN runs threatened to use up all available disk space. Usually, on my account on spock, usage (UNIX note: use command 'df -k .') runs 97% - 98%. The longer runs use space for the 18 temporary Chain01 - Chain18 files as well as the space required for nefii.dis and nefii.out. Had to watch the runs and abort if it looked like all the available space would be eaten up. It would be useful if we had a good way to influence the number of time steps NEFTRAN decides to use.

ANALYSIS:

Some of the earlier runs yielded results showing relatively small differences between test output numbers and base case output numbers. But this was before we learned to set dispersion to a small number.

Made some test runs trying to run only 1 nuclide through nefmks. Results seemed to show that no matter which nuclide was selected, it graphed out the same and didn't always match the characteristic plot for that nuclide that we'd get for the base case and other test cases. Of

course, all the base case plots came from running all nuclides, so perhaps it doesn't make sense to compare them.

The latter test runs yielded results where the test numbers seemed to plot right on top of the base case numbers. This doesn't seem to make much sense. Plan to look over the runs and plots to see if I made any stupid mistakes while rushing through the tests trying to finish before Sitakanta left. The .xls files are saved in the directories noted above.

NEXT STEPS:

1) make invert larger to see if taking more time causes more disparate results.

2) change proportion of the legs going into nefmks; instead of making the two legs equal, make one leg about 10% of the other.

3) both larger invert and disproportionate legs

4) other WaterContactMode_. . . parameters; some are set to bathtub and some to flowthrough; set all to flowthrough?

5) compare nefii.dis files from both base case and test run? um . . . no

6) run tpa w/ InvertBypass turned on; compare results of ebsnef.dat and ebsnef2.dat - are they different?

Plan to make some more runs when the SCRs I'm working on are completed or perhaps while I'm waiting for a long test run to finish. Want all SCRS completed before Jan. 1, 2003.

11-25-2002 -

Worked on organizing the results of last weeks tests of the invert for Sitakanta. Looked at the process used to see if I inadvertently introduced any errors or missed something vital. Want to make sure I didn't make any foolish mistakes either; things were very rushed last week. Check the files I saved and the processing steps. Checked to make sure I had the RD_Inverts correctly copied to the ELEMENT PROPERTIES ARRAY in nefii.inp. One small error there: I put 6.10E+02 for RD for element #15 (Nb) instead of 6.01E+02.

Finished updates to the code walkthrough summary for 11/4/02 and gave it to Ron.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Checked w/ Randy Fedors about the output from SCR #346 & #390. He hasn't gotten to it yet, but it's in his pile and he may get to it this week or early next week. He will be going out of town again the following week.

There weill be no more code walkthroughs for awhile; we need to finish up the SCRs first.

Ron has updated TPA to preclude the excessively large Kds/Rds. Need to try the test runs that bombed before (SCR #346, #390). Use TPA version 5.0betaB. Also contains the mods George and I made so the old compiler will work.

11-26-2002 -

Since I started, changes were made to exec.f for versions TPA 5.0betaA and 5.0betaB. Merged those changes in, too. Also, got TPA version 5.0betaB for tpatest/scr394 for testing. Finished both exec.f and releaset.f.

Compiled test version and base version of tpa.e. When I compiled the modified releaset.f, I ran into a problem. Dick Codell had modified the code during the summer and removed the "tlt" calculation. But George's code writes it out. Now, I'm waiting for George to get back to me so I can ask him about it. There was also a compile problem with one local automatic array; fixed it. George says to remove the reference to tlt - hope that's the end of it. The Makefile that comes with TPA version 5.0betaB does not have the solapps (old v 4.2 compiler) specified. So re-compiled with old compiler, too. Have 3 versions of executables: _v50B, _mod394, and _mod394_solapps.

Started running/updating George's test plan for SCR #394. Tpa.e is bombing in sampler for checklhsout:lhs.out

11-27-2002 -

Recompiled TPA 5.0betaB entirely using the default compiler for Spock and using the /solapps/SUNWspro.4.2/bin/f77 compiler. Ran the base case tpa.inp for both executables and compared files generated by the runs. (See directories tpatest/scr394/50Bdefaultfiles and tpatest/scr394/50Bsolappsfiles as well as file tpatest/scr394/diff_files.) The values are different in the files; perhaps due to the use of r8 declarations. The make files I used follow:

TPA Makefile - default compiler

```
FFLAGS = -O4

OBJECTS = array.o     \
          ashplumo.o  \
          ashrmovo.o  \
          dcagw.o     \
          dcags.o     \
          dsfail.o    \
          ebsfail.o   \
          ebsrel.o    \
          faulto.o    \
          iareader.o  \
          invent.o    \
          mv.o        \
          nfenv.o     \
          szft.o      \
          reader.o    \
          sampler.o   \
          seismo2.o    \
          subarea.o   \
          uzflow.o    \
          uzft.o      \
          volcano.o   \
          ran.o       \
          numrecip.o  \
          condxyzt.o  \
          peakfind.o  \
          fileunit.o  \
          findelev.o  \
          codes/linintrp.f  \
          codes/srchpos.f  \
          zportunx.o


all : tpa
      cd codes; make
      cd codes/gentpa; make

tpa : $(OBJECTS)
      f77 exec.f -o tpa.e $(OBJECTS)

clean:
      rm *.o
      rm *.e
```

TPA Makefile - solapps compiler

```
FC = /solapps/SUNWspro.4.4/bin/f77
FFLAGS = -O4

OBJECTS = array.o     \
```

```
            ashplumo.o  \
            ashrmovo.o  \
            dcagw.o     \
            dcags.o     \
            dsfail.o    \
            ebsfail.o   \
            ebsrel.o    \
            faulto.o    \
            iareader.o  \
            invent.o    \
            mv.o        \
            nfenv.o     \
            szft.o      \
            reader.o    \
            sampler.o   \
            seismo2.o    \
            subarea.o   \
            uzflow.o    \
            uzft.o      \
            volcano.o   \
            ran.o       \
            numrecip.o  \
            condxyzt.o  \
            peakfind.o  \
            fileunit.o  \
            findelev.o  \
            codes/linintrp.f  \
            codes/srchpos.f  \
            zportunx.o


all : tpa
      cd codes; make
      cd codes/gentpa; make

tpa : $(OBJECTS)
      f77 exec.f -o tpa.e $(OBJECTS)

clean:
      rm *.o
      rm *.e
```

Standalones Makefile - default compiler

```
all : mechfail nefmks failt releaset ashplume snllhs ebsfilt dsfailt

mechfail : mechfail.f
      f77 -r8 -O4 mechfail.f ../zportunx.o -o mechfail.e

nefmks : nefmks.f
      f77 -r8 -O4 nefmks.f ../zportunx.o -o nefmks.e
```

```
failt : failt.f weldfail.o integrt.o linintrp.o srchpos.o corrosn.o
      f77 -O4 failt.f weldfail.o integrt.o linintrp.o srchpos.o corrosn.o
../zportunx.o -o failt.e

releaset : releaset.f
#     f77 -r8 -O4 releaset.f ../zportunx.o -o releaset.e
      f77    -xtypemap=real:64,double:64,integer:64    -O4    releaset.f
../zportunx.o -o releaset.e


ashplume : ashplume.f
      f77 -O4 ashplume.f ../zportunx.o -o ashplume.e

snllhs : snllhs.f \
         lhs1.i \
         lhs2.i \
         lhs3.i \
         lhs4.i \
         lhs5.i \
         lhs6.i \
         lhs7.i \
         lhs8.i
#     f77 -O4 snllhs.f ../zportunx.o -o snllhs.e
#       Use next line when comparing UNIX output with run on PC.
      f77 -r8 -O3 snllhs.f ../zportunx.o -o snllhs.e

ebsfilt : ebsfilt.f
      f77 -O4 ebsfilt.f -o ebsfilt.e

dsfailt : dsfailt.f integrt.o linintrp.o srchpos.o
      f77 -O4 dsfailt.f integrt.o linintrp.o srchpos.o -o dsfailt.e

integrt.o : integrt.f
      f77  -c -O4 integrt.f -o integrt.o

linintrp.o : linintrp.f
      f77  -c -O4 linintrp.f -o linintrp.o

srchpos.o : srchpos.f
      f77  -c -O4 srchpos.f -o srchpos.o

corrosn.o : corrosn.f
      f77  -c -O4 corrosn.f -o corrosn.o

weldfail.o : weldfail.f integrt.o linintrp.o srchpos.o corrosn.o
      f77 -c -O4 weldfail.f -o weldfail.o
```

Standalones Makefile - solapps compiler

```
FC = /solapps/SUNWspro.4.2/bin/f77
all : mechfail nefmks failt releaset ashplume snllhs ebsfilt dsfailt
```

```
mechfail : mechfail.f
      f77 -r8 -O4 mechfail.f ../zportunx.o -o mechfail.e

nefmks : nefmks.f
      f77 -r8 -O4 nefmks.f ../zportunx.o -o nefmks.e

failt : failt.f weldfail.o integrt.o linintrp.o srchpos.o corrosn.o
      f77 -O4 failt.f weldfail.o integrt.o linintrp.o srchpos.o corrosn.o
../zportunx.o -o failt.e

releaset : releaset.f
      f77 -r8 -O4 releaset.f ../zportunx.o -o releaset.e
#     f77    -xtypemap=real:64,double:64,integer:64    -O4    releaset.f
../zportunx.o -o releaset.e

ashplume : ashplume.f
      f77 -O4 ashplume.f ../zportunx.o -o ashplume.e

snllhs : snllhs.f \
         lhs1.i \
         lhs2.i \
         lhs3.i \
         lhs4.i \
         lhs5.i \
         lhs6.i \
         lhs7.i \
         lhs8.i
#     f77 -O4 snllhs.f ../zportunx.o -o snllhs.e
#       Use next line when comparing UNIX output with run on PC.
      f77 -r8 -O3 snllhs.f ../zportunx.o -o snllhs.e

ebsfilt : ebsfilt.f
      f77 -O4 ebsfilt.f -o ebsfilt.e

dsfailt : dsfailt.f integrt.o linintrp.o srchpos.o
      f77 -O4 dsfailt.f integrt.o linintrp.o srchpos.o -o dsfailt.e

integrt.o : integrt.f
      f77  -c -O4 integrt.f -o integrt.o

linintrp.o : linintrp.f
      f77  -c -O4 linintrp.f -o linintrp.o

srchpos.o : srchpos.f
      f77  -c -O4 srchpos.f -o srchpos.o

corrosn.o : corrosn.f
      f77  -c -O4 corrosn.f -o corrosn.o

weldfail.o : weldfail.f integrt.o linintrp.o srchpos.o corrosn.o
      f77 -c -O4 weldfail.f -o weldfail.o
```

I left it with Ron to decide if we need to change the solapps Makefiles. In the meantime, we all need to know that we can't compare the results from TPA runs using different Makefiles. However, I should now be able to test my merges for SCR #394.

12-2-2002 -

Ran tpa.e compiled with solapps compiler and including modified code for SCR #394. Get error when ebsrel calls subarea: gsanwpglass:

```
Error in gsanwpglass: isa .gt. nsastore
```

See tpa_test1.out. Error message results if isa > nsastore or if isa < or equal to 0. Isa is the current subarea and is passed into gsanwpglass from ebsrel. Ebsrel receives isa from exec as an input parameter. Nsastore is a constant in the include file subareag.i which is included in gsanwpglass. Is isa set wrong somehow in ebsrel? exec? Ebsrel also calls ebsfilt, but not until after the call to gsanwpglass (line 962). Doesn't seem likely that anything in ebsrel could affect isa before the call to gsanwpglass. Putting in print statements to check value.

Print statements caused "error" to move to:

```
error reading ebsnef.dat
```

See tpa_isa.out. I've seen this one before; it seemed to be tied in to mixed-compiler executables. Will try to recompile the *_isa.f code using the newer default compiler for everything and see if the same thing happens.

IMPORTANT NOTE: It can cause problems using executables made by different compilers in the same run. So, to ensure that all code is recompiled at the same time, delete the .e files first. Make sure the correct Makefile is used each time. Don't forget codes and codes/gentpa subdirectories.

12-3-2002 -

NRC is here through Thursday.

Same problem with the code persists from yesterday. Recompiled EVERYTHING using the default compiler for spock. Used the code with the print statements. Still bombed in ebsrel

trying to read ebsnef.dat. Put in two print statements in ebsrel just before the message is printed out. Recompiling EVERYTHING again.

When I run TPA, releaset does NOT make ebsnef.dat with the 3rd column affected_flag. When I run releaset standalone, it does. WHY!??? Because ebsrel.f has 2 new subroutines (added by Ron for SCR 368) addebsnef & releasecolloids that do not read and write i_affected_flag, the 3rd column added by George.

Went back to solapps v4.2 compiler since I'm having trouble with both compilers. Same errors occur. I believe problem is the loss of the i_affected_flag in Ron's new subroutines. Will need to write and read it in the intermediate files (ebsnef.out and ebscld.out) so it remains in ebsnef.dat and ebsnef2.dat.

Added more print statements. Also getting a bus error further down in the code. Not sure why. Have to look at maplist and the new arrays that are built. Something to do with maxntime vs. mxntime in the solubilityaffected arrays? Doesn't appear to be. Start from here tomorrow.

ALSO, STILL NEED TO ASK RON ABOUT THE use of maxnnucl and nnucl in George's code - DOES IT NEED TO CONSIDER COLLOIDS?

12-4-2002 -

The bus error was caused because of a discrepancy between array dimensions in arrays that were passed between exec and ebsrel. Once I changed the use of maxnnucl to maxaqnuc and nnucl to nnucl + numclnuc in George's code modifications, the bus error problem went away. Adding the i_affected_flag to ebsrel code seems to have taken care of the other problems.

Removed print statements and recompiled. TPA will now run to completion.

I've asked Ron to double check the changes I made for colloids once he gets the modified files from me.

Moved on to testing the modified code. First run of pl-1 OK except for the output file relcum.out. It didn't contain the right information. Turned out that George's modified code using unit 25 for reading and writing to relcum.out. However, when he did another problem report (#365), he used unit 25 for rel_flow.out. Unfortunately, out compilers don't catch a 2nd open of an already open file. Changed unit to 30 for relcum.out. Second run of pl-1 seems much better. I think relcum.out is correct now. Need to check with George about one of the conditions (8.5, #4), but I think we passed this time.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Went on to system level test. Plan to add one more test to show that mods don't change output files that it shouldn't. First run of sl-1 still going when I left.

12-5-2002 -

TPA ran to completion and made all the files, but George tells me that there is information missing from relcum.out. So, started tracking down the problem. It seems that, although Dick Codell removed the tlc calculation from the releaset code, he did not remove where it was printed out to relcum.out under the column header SF-LCH End[yr]. Also, the use of variables tftc, tleach, and addtime seems to have been commented out, but the variables remain. Tftc and tleach are also printed out to relcum.out as column headers SF-LCH Onset[yr] and SF-LCH Span[yr]. Addtime was used to calculate those values. Looks like they should be removed from the code, too. Need to check w/ Ron.

12-6-2002 -

Talked with Ron this morning about releaset.f. He agrees that tftc, tleach, and addtime should be removed from the code along with code printing them out to files, since the values would always be 0.0 now. While I'm at it, he wants me to get rid of the unused local variables and common alignment problems that generate so many warnings when we compile. I will also be adding the new system level test to show that I didn't break any of the other code.

In releaset.f, the warnings about local variables not used was handled by commenting out the declarations. The warnings about incompatible lengths for common blocks was caused by common variables being declared real*8 in most places and double precision in one place. Changed all declarations to real*8. Now compiles cleanly.

Also discovered that dissolution_time was set to tleach, which has had the value 0.0 since Dick Codell's last changes. Commented out the use of dissolution time in releaset.f, ebsrel.f and exec.f. It was being written out to relfrac.out and relcum.out. Talked to Ron and George. As far as can be determined, no other code makes use of dissolution time. Have to take it out of the test plan.

Re-running sl-1 test and pl-1 test. Once I've confirmed the results, I'll run the new test and compare the output from all files to make sure I didn't affect any other files indadvertently.

Added to list of future changes:

1) need to check open/close statements for errors (see releaset.f)

2) need to check read/write statements for errors (see ebsrel.f)

3) need header blocks for functions/subroutines (see releaset.f)

12-9-2002 -

In addition to removing tftc, tlt, tleach and addtime, Ron told me today to get rid of ileach as well. This involves releaset.f, ebsrel.f, ebsrel.tmp, and tpa.inp.

In re-running George's sl-1 test, ran into a difficulty w/ releaset.f. The TPA I left running Friday never finished. It paused after ebsrel was called (subarea 4, realization 2) with this message:

```
PAUSE: stepsize underflow in rkqs
To resume execution, type go
Any other input will terminate the program
```

If you type go, the message is repeated. If you type <CR>, "STOP" is displayed (I assume releaset.f stops) and TPA continues to completion. I've looked at releaset.f to try to see exactly what caused the problem, but the code is hard to follow.

Saved the output files in tpatest/scr394/sltest/sl-1. Couldn't save the scr394_sl1.out file, however. It paused for a response but wouldn't read anything input from the keyboard, so had to abort the run. However, did run tpa.e w/o and output file, entered <CR> when it paused and it did run to completion. Made the ileach changes and will recompile and run my new test on it. If there are no problems with the results from sl-2 (System Level 2 test for SCR #394), then I believe that we're OK. Pl-1 and sl-1 have passed twice so far. Running an sl-1b w/ different parameters: 10,000 yrs instead of 100,000; 201 time steps instead of 251, RatioOfLastToFirstTimeStepInCompliancePeriod = 100.0 instead of 1.0. I'll check those results, too, against the criteria for sl-1. (If it runs to completion w/o pauses, I'll save the scr394_sl1b.out file in subdirectory sl-1b.)

12-10-2002 -

Sl-1b ran to completion. Saved files in tpatest/scr394/sltest/sl-1b. Compared results to test plan criteria. Used diff_files_sl2 command file to compare the output files from two runs. Passed George's criteria.

Brandi had some questions. She is testing SCRs #415 (George's) and #416 (mine). These are the SCRs that include the changes made to be able to use the old solapps 4.2 compiler. I'm not sure exactly where her problems were coming from. They didn't seem to be tied into the changes made in the code. Could be from having mixed executables. TPA won't run correctly unless all the executables are created using the same compiler.

Ron was having some problems merging in the code changes from SCR #412, so we went through the merge together. I think it's working OK now.

Added new test to test plan to compare a base case run (TPA 5.0betaB) with the modified code. Some files have changed format, but most files and values should be the same.

12-11-2002 -

Looked at the differencess in files from basecase and sl-2. The new formats show up, but the values are different for releases - I wasn't expecting that. Why are they different?

Other problems: some of the files have the affected flag showing up as real either all the time or some of the time:

ebscld.out	affected_flag in real format
ebsnef.out	affected_flag sometimes real and sometimes integer

Caused by errors in the FMT descriptors in the write statements.

Mod releaset.f - real*8 vs double precision (scr394_sl2b.out) - this accounts for most of value differences. Compile releaset.f using the xtypemap line and the difference problem goes away.

Mod ebsrel.f - fix format of write i_affected_flag (scr394_sl2c.out) - this fixes the output problem but doesn't affect the values.

Using releaset_mod394_d.e (put DOUBLE PRECISION) back in - this is most of value differences.

Using releaset_mod394_e.e (put if then ileach back in)no value diff; delete executable; renamed source releaset_test_ileach.f. WHAT ELSE COULD CAUSE THE SLIGHT VALUE DIFF?

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

12-12-2002 -

Reviewed modified code modules to look for possible causes of changes in values.

Found extra line in releaset.f that should have been commented out: "if concn = amass1(ielem)) concn = sol(ielem)". Corrected it, but it made no difference in the output. Tried commenting out other new code, but nothing made any difference. Repeating the ileach test and making sure that changes made in ebsrel.f, releaset.f and ebsrel.def (see tpatest/scr394/scr394_sl2f.out). Then want to try a run with releaset executable made using xtypemap w/ solapps 4.2 compiler (see scr394_sl2g.out). Compare with a basecase run (see scr394_basecaseg.out).

Using the xtypemap line in the makefile for releaset.f makes a difference in values and consistency. Running the basecase against the TPA modified for SCR #394 w/ the ileach code still in results in the same output values. I must have screwed up the previous test putting ileach back in. This time, it made a difference. After ileach is commented out everywhere, including the following lines (line 2129 in releaset.f, TPA 5.0betaB), the output values change (compare cscherer/tpatest/scr394/scr394_sl2g with .../scr394_sl2i.out):

```
        else if (ileach.eq.1) then
          sareap = 4. * pi* r0z**2
c--       waste matrix leach rate from total wet mass UO2/wp
          sareat = sareap*(amass0*xfrac)/(4./3.*pi*r0z**3*fueden)
```

Sent email to Ron with attached files. Need to check w/ Dick Codell to see if we sould really comment out the lines or not. If not, what about the if condition: do we need to retain ileach? I need to know what state to leave it in while we're waiting for an answer.

The good news is that SCR #394 has passed all it's tests!

With the newer versions of TPA, Ron is providing the usual Makefile plus a Makefile4.2 which specifies the solapps compiler. UNIX NOTE: can enter make all -f Makefile4.2 to specify a make file other than Makefile.

Was told I'd be moving offices after Christmas. A new hydrologist will be moving into this office. May be using another computer - will have to make sure it has all the necessary software loaded. Looking into getting my email transferred to Div. 20. See if we can get password for messages on new office phone. Lisa will be moving, too, to the same office.

12-13-2002 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Talked with Ron - decided to remove the ileach if-then-else block. (This means removing ileach from ebsrel.inp) Test by setting ileach = 0 in ebsrel and see if the two results are the same. A question remains regarding oxwidth. If it is no longer calculated and is always 0 (BTW, add an explicit statement setting oxwidth to 0), it could be removed from the code. Several calculations would be simplified by removing oxwidth terms. Maybe later?

Re-ran base case: see tpatest/scr394/sltest/basecase. Then, re-ran basecase setting ileach to 0 in ebsrel.def and ebsrel.f (tpatest/scr394/sltest/basecase_ileach0). This case should match the modified version of the code. Last, ran the modified code (tpatest/scr394/sltest/sl-2). The test passed (sl-2). There are no difference between the base case w/ ileach set to 0 and the modified code case except for time/date in headers and the formatting differences.

Moved on to modifying the SCR and the test plan.

Sent Ron an updated tpa_change.wpd. Included the oxwidth question and getting rid of the warning about using date when you run TPA.

Noted that we don't necessarily use xtypemap for compiling nefmks.e, snllhs.e, mechfail.e and some of the other standalones. I guess as long as we don't go back and forth between compilers, it shouldn't make a difference, although some problems might slide by if we use it that would show up as value differences otherwise.

Modified files are exec.f, ebsrel.f, ebsfilt.f, releaset.f, ebsrel.def, and tpa.inp. The final form of these files are in tpatest/scr394/mods.

12-16-2002 -

Finished the mods for SCR #394 and turned it in to Ron. Finished the mods to the test plan for SCR #394 and turned it into Ron. Burned CD and printed listing of files; turned them in to Ron. On CD, write date, id for contents (TPA SCR #394), file formats (unix, Windows) and attach listing.

Return to re-testing some of the runs that bombed when I was running the tests for SCR #346 and SCR #390. After that (or maybe before), Ron wants me to finish the write up about check.pnt. He wants the write up before January.

12-17-2002 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Talked to Brandi on my way out yesterday. She wants me to look at her test plan for SCR #377 and let her know if I can think of anything else she needs to test. I made suggestions to change one test and add two others.

George is helping Randy check the output for SCR #346 and #390. He came by to check with me where the files were.

Started on check.pnt writeup for User's Guide. Reviewed source code to see how it is implemented. Reviewed 4.0 User's Guide looking for best place to put writeup. Ran tpa.e from version TPA 5.0betae to generate a check.pnt file so I could see how big they are. Dependent upon number of time steps. Decided I should test check.pnt again anyway; it hasn't been tested for several versions. Base tpa.inp w/ restart flag set, 3 realizations. Interrupted-restarted files were copied to tpatest/tpa50betae/checkpoint subdirectory to compare with a normal run (tpatest/tpa50betae) and a run with the restart flag set but no interruptions. One problem on the restart version: check.pnt is not deleted at the end. However, it was deleted when the restart flag was set, but there was no interruption and restart (saved a copy of the check.pnt before it was deleted in subdirectory restart_no_interrupt). Same end result for the two runs with the restart flag set. The base case at 3 realizations (no restart flag) ran OK and the results were the same.

Pulled out my files for SCR #346 and #390. Need to look at the old test runs that bombed. Now there is code in place to limit the range for Kds and Rds; want to see if TPA will still bomb in those cases. Started re-running SCR #346, sl3-b w/ TPA 5.0betae. Runs for 50 realizations. This is the one that originally bombed in realization 33 when I changed ArealAverageMeanAnnualInfiltrationAt
Start[mm/yr] to a constant value of 8.5 instead of uniform from 4.0 - 13.0. Also, couldn't get a clean run at 100,000 yrs. This first run is at 10,000 years only. See tpa_testRDlimits.out. Left it running overnight.

Spock will be down in the morning from about 8:00 a.m. on, for I don't know how long, due to some upgrades.

12-18-2002 -

Generated a rough draft of the check.pnt writeup and emailed it to Ron. Reviewed the test plans for SCR #346 and #390 which I will turn in as soon as the output has been verified by Randy Fedors.

Spock was down all morning and in the afternoon until I left.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

12-19-2002 -

Spock is back up, but I couldn't log in until Hollen put my account name back into the system. Had to change to a new password.

Ron edited the check.pnt writeup and left corrections for me. Put them in and resent it to Ron. He wants me to do some testing with the disruptive scenario flags set and using different models to see if we need to be saving other files as well, e.g., the gs files along with the gw files. We'll see. Working in tpatest/checkpoint using TPA 5.0betae.

First test: 3 realizations, all disruptive scenario flags set; one run w/o interruptions (tpatest/checkpoint/disflags_3r); one run w/ interruptions (tpatest/checkpoint/disflags_3r_interrupt). There were no differences.

Second test: start w/ 1st, change Imodel to 4: same results.

Third test: start w/ 1st, change MaximumTime from 10,000 yrs. to 100,000 yrs.; set all WaterContactModes to 1: bombed in NEFTRAN, FACER: BF array out of bounds even w/ limits on KDs and RDs. Talk to Ron tomorrow. Re-try 3rd test at 10,000 yrs. w/ no interruptions. If it runs, run it w/ interruptions tomorrow. It ran w/ just slightly different PMD. Test more tomorrow.

12-20-2002 -

Found no differences in the checkpoint tests. However, the run at 100,000 yrs for 3 realizations bombs in subarea 1 of realization 1. Talked to Ron about the problem and we've been trying to track it down.

Bombs in uzft - called NEFTRAN - BF array, out of bounds again.

Note: mobil and immobile rds in nefii.inp - still huge numbers that aren't going through the limit checks. But happens in both bomb runs and clean runs.

Michael uses UZ. . .ForceFactor. . . set to 0 to mean limit the RDs to 1.0 Why isn't it working. The elements with really big numbers are the 5 elements (Am, Np, Pu, Th, and U) that are calculated from Michael's/Sitakanta's new table coefkdeq.dat.

It will run with 99 realizations selected and StopAtRealization = 3. There don't seem to be any apparent differences between the input files to NEFTRAN among the runs that bomb and those that don't.

Ron is not seeing the error in TPA 5.0betaG, but George added a bunch of new uniform distribution sampled parameters and we know that has the effect of moving the mystery problem around. Will G run for 1024 realizations?

Ron wants me to put in some print statements and see where the RD values are getting messed up. Moving to tpatest/neftran_bomb. Will copy over checkpoint and delete checkpoint related stuff. Need to get NEFTRAN tests out of checkpoint, too.

In uzft.f, the array NEFRD is written out to nefii.inp. However, both matrix and fracture values are written to the Mobil RD column and the Immobile RD column is hard-coded and set to 0.0. In szft, the array rd (mobile) is written to the Mobil RD column and array rdi (immobile) is written to the Immobile RD. But talking to some of the scientists, fracture should be mobile and matrix should be immobile ??????????????????!!!!!!!!!!!!!!! I would ***really*** like to get that clarified!

Started going through the NEFTRAN manual again to see if anything stands out as possibly contributing to this problem and the problem of NEFTRAN picking such tiny time steps/intervals that the files are too big to be useful and take forever to run.

Ron is planning a telecon w/ Tim McCartin at 0900 next Monday, the 23rd.

12-23-2002 -

Ron and I talked to Tim McCartin (telecon) about data going into NEFTRAN. Got the following definition: mobile (where the water is moving) vs. immobile (water is not mpving/stagnant). Either mobile or immobile can be fracture sometimes and sometimes matrix. (In other words, matrix is not synonomous with immobile; nor is fracture synonomous with mobile.) Re: David Pickett's question: 100% of sat zone is fracture. Re: too small time step question: no good way to affect the number of time steps selected by NEFTRAN w/o losing accuracy. Suggests we turn off changing media from maxtrix to fracture and vice versa once infiltration reaches a certain level.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Started looking at uzft & szft to see what it would take to turn off switching between maxtrix & fracture (MEDIA = 1 or 2) once infiltration reaches a certain level like that at 10,000 or 25,000 (compliance period) or nK years.

Brandi - should results from test plan for SCR #377 linear when manipulating the flux multiplier, SZ_FluxMult? No, only affects 1 term of velocity equation not whole equation; glacfrac is changing per time step, also.

George Adams - SCR #346 & #390: reran sl-1 in SCR #390 - use tpameans.out for tpa.inp. He's getting files ready for Randy to verify results.

12-30-2002 -

At George's request, I re-ran sl-1 for SCR #390 again - I hadn't done both test cases last week using the means. He is in the process of putting the data from the test runs together for Randy Fedors to verify. Waiting on Randy's verdict on runs from SCR #346 and SCR #390.

Checked w/ Brandi. She has turned the test plan for SCR #377 in to Ron and is waiting on his approval on whether or not it's done.

Finished editing the writeup for checkpoint/restart. Gave a hard copy to Ron and sent him the electronic copy, too. Actually, after re-reading it, I'd like to make a few more changes. Will see if Ron wants to change anything else as well.

Started working on testing SCR #414 (another one of George's). Printed out SCR and test plan to review.

12-31-2002 -

Helped Ron merge in changes for SCR #394 (it will be TPA version 5.0betaI). There were some problems merging SCR #394 w/ SCR #411 and SCR #414. Divide by zero (or close to it) showing up in releaset.

George had a question about my output files for SCR #346 - sl-3, test case B & C. I think that problem was resolved. He should be able to get all the files to Randy Fedors now for his opinion. I can't wait for SCR #346 and #390 to be finished!

Made last (I hope) changes to checkpoint.wpd, the writeup for the user's guide about the check.pnt file and the restart flag. Attached the file to an email to Ron.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Continued w/ SCR #414. Used ~rjanetz/gettpa to bring current version (5.0betaH) over to tpatest/scr414. Made mods to SCR and Test Plan documents. Compiled TPA using old compiler. Began running tests. Made directory tpatest/tpa50betaE and copied over the files for that version to use for comparison purposes in testing 414. Also re-compiled E but saved existing executables that were probably made using the default compiler.

PL-1: I didn't get any screen output from running mechfail.e. Turns out I needed to set the debug flag in mechfail and recompile. Not only should that create screen output, it should also create mechfail.out which is needed to generate the .xls spreadsheet for the "hand calculations". Will have to modify instructions. Do we need a standalone subdirectory and if so, shouldn't we move some files to it in order to run anything? Anyway, this is where I need to start up again next Monday (next year?).

I understand that once 414, 346, 390 and 417 (George is doing that one) are finished, that all the SCRs will be "done". So, we can either get on with validation or generate some more SCRs.

1-6-2003 -

Some problems this morning: there was a screen dump on the screen when I turned on the monitor this morning. After rebooting, got some error messages pertaining to winscan.exe. Don't know what happened. Had trouble opening this file; got message: "\\Spock is not accessible. No more connections can be made to this remote computer at this time because there are already as many connections as the computer can accept." Don't know what's going on. Spock is 99% full. Power outages in the office in the afternoon. Perry added my new e-mail account for cscherer@cnwra.swri.edu. I hope it works OK.

Modified mechfail.f so that DEBUG_FLAG is set to .TRUE. and recompiled. Re-ran pl-1. This time I got the screen output mentioned in George's test plan. Copied data to pl1.xls and compared the results from the new run against George's and finished pl-1 in the test plan (passed). There are three system level tests remaining.

George left me a test plan report for SCR #390 with Randy's opinions. I'll mention this in my test plan and include his results with mine. I need to check with George to make sure that this finishes 390 (I believe it does) and check status for SCR #346. Check w/ George; 390 is complete. Now, just need to update the test plan, make a CD and turn it in to Ron.

1-7-2003 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Continued with test plan for SCR #414 (tp_scr414.wpd). Problems running mechfail.e. If I compile mechfail using the 4.2 compiler, it will abort reading in mechfail.dat (apparently writes some *** instead of integers in a couple of places). Why? If I compile mechfail with the default compiler, even if everything else is compiled with 4.2, then it runs OK.

Finished running tests sl -1 and sl-2. They passed, except for the compiler thing.

Checked w/ George re: SCR #346 - Randy is still looking at the results.

1-8-2003 -

Looked at results from sl-2 for the Test Plan for SCR #414. It calls for plotting results, but doesn't say how. Checked with George and found out what he used: Excel. I should be able to duplicate that with few problems. Plotted the data; it looks good (just like George's earlier one, in fact). See tpatest/scr414/sltest/sl-2/sl2.xls

Ran sl-3. It passed. Other than determining what the compiler problem is, the test plan for SCR #414 is complete. Need to finish typing in the mods (also for SCR #390) and get the listing and CD made for Ron.

There was more room on spock today, so I recompiled mechfail with the 4.2 compiler and tried it. It still doesn't work. Copied over a version of mechfail.e that George had compiled (he said he used the command 'make all -f Makefile4.2') and it worked in my area! It was a lot smaller than my compiles (??). Ron wants me to put in print statements, run some tests, and find out what's going wrong.

The problem seems to be related to formatting the output. Located the write statements for the subarea comment at the top and the message and flag at the bottom of mechfail.dat that are being trashed. If the write statements contain the explicit formatting commands, then the output is garbled in some way. Asterisks are printed (the number depending upon the number of spaces specified for the number in the FMT list) instead of the number, like when a number is too big for the specified field. The subarea number is read correctly from mechfail.inp. The other number at the bottom is hard coded to be either 0 or 1 with a different message printed with each flag. How did we ever get that 2?!! If the write statement is free format, the numbers come out correctly, but there is an added space at the beginning of the line.

Ran a test using the FMT list but increasing I2 to I12. It printed out a large (4294967296) number instead of subarea number (should be 1 - 0) or the failure type flag (should be 0 or 1).

George says that Randy didn't like some of the numbers for sl-3B (from the test plan for SCR #346). So, they've come up with some code mods that have more accurate output. Ron wants to include the code changes w/ SCR 346 if Randy decides the modified version is correct. Can't see how it will be finished this week.

Moved offices today from A244 to A229. Guardian and Birch were both moved. Lisa isn't back yet, so we had to go ahead without her; most of her stuff is still in her desk in A244. Perry got the email working yesterday afternoon.

1-9-2003 -

Continued looking at the compiler problem w/ mechfail. Re-compiled everything again this morning. Mechfail still has problems, although George's version still runs OK in my area.

Went to tpatest/tpa50betaH and compiled everything w/ the 4.2 compiler. Ran the base case. The run aborted reading mechfail.dat. Went to codes and recompiled using 'make all -f Makefile4.2' like George does. Re-ran the code. Also aborted. Made a third run using George's mechfail.e from his 'current_code' directory (beta I). This ran OK.

Created directory tpatest/tpa50betaI and copied the current version to it. Compiled using the 4.2 compiler. Ran the base case. It aborted reading mechfail.dat. What are the differences between compiling in my area and in George's?

Talked to George some more. He makes a new directory and uses gettpa to bring over the current version and enters 'make all -f Makefile4.2' at the tpa level, not the codes level. So I made a tpatest/test directory and brought over the tpa code again and compiled like George. I'm still getting a different sized executable than George and it still aborts reading in the last line of mechfail.dat ('incomprehensible list input'). Then I re-made just mechfail.e using the default compiler. It runs.

During the default compiler compile, there was a warning message about the obsolete use of -r8 instead of xtypemap, so I made a version (in tpatest/tpa50betaI) of the makefile using xtypemap for mechfail.e, like nefmks and releaset use. Re-made the code. Guess what? It worked! It ran to completion, and mechfail.dat is fine. Why does the 4.2 Makefile use xtypemap for nefmks while the default compiler Makefile does not? Went back to version H to see if the same thing would work there. The code also ran in H. Still don't know ***WHY*** it makes a difference - actually don't know why George doesn't have a problem on his system..

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Need to check results from compiled version 4.2 w/ and w/o xtypemap to make sure the numbers match. If so, should become part of next version. Both mechfail.dat and the screen display capture file are the same except for time/date stamp. Sent email to Ron. Went back to completing the test plan for SCR #414.

Ron says that David Pickett provided some new numbers for the KD/RD calculations and they seem to work. Once they're part of tpa50betaJ (?), I can test it and see if I can break it again. Ron has run 100 realizations w/o it bombing.

Called Michael Muller to let him know where his notebook and Guardian have moved. Also moved a Mathcad book that I assume belongs with Guardian.

The power is going down in the building this weekend. The IMS (?) - computer - group wants all computer equipment, including power supplies, shut down by 3:00 p.m. tomorrow. Will have to check if that means we can't use the PCs at all, even locally.

1-10-2003 -

Finished modifying the test plan for SCR #414. Went on to modify the test plan for SCR #390. Made CD's for both SCRs, printed listings, and updated the SCR.wpd file. Turned it all in to Ron.

George and Randy Fedors want to make some changes related to SCR #346. This will include code changes (probably) and modifying the test plan. I'll have to wait until Ron gets the updates and then re-do the test plan.

Started work on SCR #371. Start by updating SCR.wpd. May need to check with Michael Muller. Ron sent me some files to start with. Have to wait for Monday until Ron has a chance to update the numbers, etc. In the meantime, the current version is TPA50betaI.

Ron wants me to update the implementation section of scr_390.wpd, too. May need to check with George. Stu Stothoff authored the changes. I have emails from Randy, too.

Talked to Michael about SCR #371. He seems to recall some potential code mods needing to be done relating to the problem of extreme values for KDs/RDs. He will come over Monday afternoon to check his scientific notebook because he doesn't really remember what he was working on before the break.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

All the computer systems have to be turned off by 3:00 today. They're shutting down the systems to get ready for a power outage this weekend.

1-13-2003 -

Ron sent me the latest file (scr_390.wpd) for SCR #390 along with the diff files so I can update it. Copied them to the tpatest/scr390/docs subdirectory. Finished the changes to the file, printed it out, signed my part, and turned it in to Ron. This should complete my input for SCR #390.

Created a draft version for scr_371.wpd. Generated diff files for the modified code between version 4.2w and 4.2x. Note: Coefkdeq.dat does not exist in Ron's directory for version 4.2x. Got the one from version 4.2y instead. It differs from the version contained in TPA 5.0betaI. This shouldn't make any real difference, though.)

Talked to Michael Muller about SCR #371. He is going to look up his notes from his Scientific Notebook and let me know where he left off. He thinks he may have come across a problem but doesn't remember what or what its resolution was.

David Pickett's updated numbers for tpa.inp have not been included yet. Ron will let me know when so I can get the latest version to work with.

1-14-2003 -

Talked to Michael Muller about where he left SCR #371: ..... Gave him a copy of the SCR (scr371.wpd) that I had updated (implementation sectino) and asked him to check it over.

Test Plan for SCR #371 (tp_scr371.wpd):

George Adams - SCR #346 - email:

1-15-2003 -

Continued working on the test plan for SCR #371.

Signed paperwork for Ron from SCRs that I've worked on the past several months.

1-16-2003 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Finished signing paperwork for Ron.

Made some corrections (from Ron) on scr_371.wpd. Also added the difference files for exec.f, szft.f, and uzft.f. Should be complete barring any changes Michael wants me to make.

Continued working on the test plan for SCR #371. Functional level test calls for inserting print statements into the code to monitor what's happening. Need to mod szft.f and uzft.f. Not much is happening in exec.f - just passing in ir (current realization number) to szft.

Was going to do a comparison test between pre-371 code and post-371 code, but there are too many changes between TPA 4.2w and TPA 5.0betaL. I can (I think) "turn off" the new stuff relating to coefkdeq.dat simply by removing the file and un-commenting out the sampled parameters in tpa.inp for Am, Np, Pu, Th, and U. It should revert to the old method of getting values from tpa.inp then. Need to ensure that the values remained constant while being commented out. Will also test what happends when coefkdeq.dat is missing and the old values are not returned to tpa.inp. Oops! It's an error condition w/o the file. Will an empty (somewhat) file work set to 0 elements?

Would I get the same RDs, KDs in nefii.inp from 4.2w as 5.0betaL by using the old values in tpa.inp? Maybe I should be testing just the differences between 4.2w and 4.2x??? Need to check w/ Ron. Talked to Ron. We looked at the exec header and decided that I should be using 4.3a since there was a bug fix put in then. The fixes put in after will be taken care of by SCR #418 (George?). So, I'll need to bring over another version and recheck the diff files. Ron wants me to add a table he created for Michael about KDs to the SCR. He also wants me to double check that the last row of the table is implemented correctly.

1-17-2003 -

Brought in TPA version 4.3a to replace tpa50betaL. Made new diff files - they're ready to be merged into scr_371.wpd after the previous ones are deleted. See if there are more things I need to add to SCR #371 (scr_371.wpd). Did add Ron's table. Abandoned changes I was making in tpa50betaL to szft.f and uzft.f to add print statements. Need to use 4.3a instead.

Both 4.3a and 4.2w were pre-solappsv4.2 compiler, so can't test using it. Started making mods to uzft.f and szft.f - adding print statements for the functional level test.

Ron said there might be some coding to do next??? Check next week.

1-20-2003 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Continued working on test plan for SCR #371. Putting prunt statements into uzft.f and szft.f to capture the processing. This will be for FLT #1. Also reviewing the code to see what's being done and what cna be tested. Trying to fit the code to the table Ron gave me last week. Trying to match code and variables to names and symbols so I can check that equations are being calculated correctly.

1-21-2003 -

Continued w/ test plan for SCR #371. Finished putting print statements in uzft.f and szft.f for FLT #1. Have a compiled version, tpa_flt1.e, for the functional testing. Lots of data to wade through. Ran and captured the screen output: tpa_flt1.out. It's a very long printout with 1 realization, 10 subareas, and all the nuclides with all the information from szft and uzft. I will need to limit the info in some way for practical purposes in the test plan. First, I want to go through this printout. I could always separate szft and uzft into two different functional level tests. Looking at reading in coefkdeq.dat could be a separate test, too. Let's see what makes sense.

Next test, run base versus modified and look at some output files. These two tests should show me how many others I need to run.

Formulas in the code should match D. Turner's table:

Matrix							
Unsaturated Zone				**Saturated Zone**			
Quantity	**Symbol**	**Units**	**Source**	**Quantity**	**Symbol**	**Units**	**Source**
matrix porosity	Φ	-	sampled	matrix porosity	Φ	-	sampled
matrix grain density	ρ	kg/m^3	sampled	matrix grain density	ρ	kg/m^3	sampled
matrix pore radius	r	m	sampled	matrix pore radius	r	m	sampled
specific surface area	ssa	m^2/kg	$= \frac{3\Phi}{\rho r}$	specific surface area	ssa	m^2/kg	$= \frac{3\Phi}{\rho r}$
K_a	K_a	mL/m^2	table lookup	K_a	K_a	mL/m^2	table lookup
Kd	Kd	m^3/kg	$= K_a\ ssa\ (10^{-6}\frac{m^3}{mL})$; or sampled	Kd	Kd	m^3/kg	$= K_a\ ssa\ (10^{-6}\frac{m^3}{mL})$

				moisture content	n	kg/m^3	1
				Rd	Rd	m^3/kg	$= 1 + \frac{\rho(1-\Phi)}{n} Kd$; or sampled

Fracture							
Unsaturated Zone				**Saturated Zone**			
Quantity	**Symbol**	**Units**	**Source**	**Quantity**	**Symbol**	**Units**	**Source**
specific surface area	ssa	m^2/kg	1	specific surface area	ssa	m^2/kg	1
K_a	K_a	mL/m^2	table lookup	K_a	K_a	mL/m^2	table lookup
Kd	Kd	m^3/kg	$= K_a \text{ ssa } (10^{-6} \frac{m^3}{mL})$	Kd	Kd	m^3/kg	$= K_a \text{ ssa } (10^{-6} \frac{m^3}{mL})$
fracture aperture	A	m	sampled	fracture aperture	A	m	sampled
Rd	Rd	m^3/kg	$= 1 + \frac{2}{A} K_a$; or sampled	Rd	Rd	m^3/kg	$= 1 + \frac{2}{A} K_a$; or sampled

1-22-2003 -

Continued with test plan for SCR #371. Had error in my print statements for FLT #1. Re-did them. Re-ran w/ only 1 subarea. Only 52 pages this time. Checked results in tpa_flt1.out against coefkdeq.dat and Turner's table for KDs (see tpa_flt1.xls for some of the calculation checks):

- uzft.f reads in coefkdeq.dat correctly (5 radioelements, 10 rows & 9 parameters each)
- uzft.f calculates original number for log PCO2 correctly given base (10) and log

Started looking at the results from SLT #1 as well. I printed out nefiisz.inp, nefiiuz.inp, nefiisz.vel, and nefiiuz.vel from both slt1_42w and slt1-43a directories for comparison. Results: different legs (last leg selected) in nefiiuz.inp and nefiisz.inp. Differences in mobile and immobile RDs, but some differences even for elements not included in coefkdeq.dat (anywhere in decay chain).

1-27-2003 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Continued looking at the data from SLT #1. Discovered I had run tpa42w/tpa.e for 10 subareas whereas I had run tpa43a/tpa.e for only 1. So, I re-ran tpa42w/tpa.e after changing StopAtSubarea from 10 to 1. However, it made no difference in the nature of the results - still seeing changes in RDs for elements that I expected to remain unchanged.

Note: in coefkdeq.i, the parameter nradioelementsmax is set to 16, which does not allow for colloids. Don't know if colloids are ever intended to be included in the KD/RD calculations, but I'll check with Ron.

Continued checking results in tpa_flt1.out against coefkdeq.dat and Turner's table for KDs (see tpa_flt1.xls for some of the calculation checks).

Made some changes to make the screen captured data more usable. Re-ran tpa.e and printed out the tpa_flt1.out file again (now 56 pages long for 1 subarea for 1 realization).

Looking at the print statements when the KDs and RDs are read in, some results seemed confusing. Multiple reads of the same values for elements are made because of multiple isotopes. The values for Am and U are within the range of the former values in tpa.inp, but the values for Np, Pu, and Th are not. Don't know if that's a problem. Still need to confirm that the values for those 5 are coming from coefkdeq.dat. Since they are commented out in tpa.inp, I don't see where else they could be coming from, but need to verify it.

1-28-2003 -

Continued testing SCR #371. The flag (ireflag) is set correctly for the 5 elements that should be read from coefkdeq.dat. Confirmed that values for Am, Np, Pu, Th, and U match the KDs/RDs calculated using the data in coefkdeq.dat. Added some more print statement to szft_flt1.f. Tpa_flt1.out is now 60 pages long. Got some strange results from the szft section - haven't figured it out yet.

These are criteria for FLT #1 that have been passed:

- uzft.f reads in coefkdeq.dat correctly (5 radioelements, 10 rows & 9 parameters each)
- uzft.f calculates original number for log PCO2 correctly given base (10) and log
- uzft.f reads in sampled parameters from tpa.inp correctly
- uzft.f reads in values from coefkdeq.dat correctly
- uzft.f calculates kds correctly
- uzft.f calculates rds correctly
- uzft.f stores values (ph_uzsz and dlogco2pp_uzsz) correctly in common and szft.f gets stored values
- szft.f reads in sampled parameters from tpa.inp correctly
- szft.f calculates kds correctly
- szft.f calculates rds correctly (both fracture and matrix)
- formulas in code match Turner's table ????

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

1-29-2003 -

Continued with FLT #1. Szft passes as long as it's OK that the mixing zone layer is always UZ layer 6 (Bfw). Checked w/ Ron. Other questions:

- nradioelementsmax is set to 16, which is too small to include colloids; add to to do list that it should be bumped up to 20 at some later date
- conflicting notes in Sitakanta's notes for Michael about 'a' and what value it should be
- the flipping from descending to ascending of the logPCO2 column from coefkdeq.dat, along with the bubble sort (uzft.f) seems to be overkill, but no plans to change it
- there is no range check on aperture; if it is set to zero, could cause divide by zero
- uzft.f and szft.f both have extra reads of sampled parameters; the loop is by isotope while the data is by element, so where there are multiple isotopes of an element, the same data is read more than once
- the calculated values for some of the 5 elements (Pu, Th, and U) are outside the range that was specified in tpa.inp (Ron says they know about it, but that's the way they want it)

Need to check with Michael about differences in formulas in notes and the way they are coded.

1-30-2003 -

Michael Muller came over this morning and we discussed the leftover questions regarding the equations for Kd. The 'a' Sitakanta refers to as being a new sampled parameter was tabled for some future date. Unknown why the values are so different from what's in the coefkdeq.dat file now. Maybe it's supposed to be a multiplier. Sitakanta wanted it called CoefficientForResponseSurfaceUncertainty; it would be uniform ranging from 0.9 to 1.1. Regarding the missing divide by moisture content in the Rd equation: moisture content in the saturated zone is always 1.0 (fraction?) so the divide is unnecessary. The equation is only used to calculate saturated zone Rd. Michael didn't put in a check to prevent aperture from being entered as zero because that number makes no sense. I still think the check is necessary. I'll let Ron know and he can decide whether to forget it, fix it now, or fix it later. It could go on the list along with changing nradioelements max from 16 to 20.

I gave Michael a copy of the SCR for 371 for him to check, since I input his section. He said he'd get back to me tomorrow w/ any comments.

For SLT #1, confirmed that intermediate/output files are changed for case A. For case B, planned to show that putting nothing in coefkdeq.dat and uncommenting the info in tpa.inp produces the same results? But it didn't. Other changes went into the code between 4.2w and 4.3a, including colloids and mechfail changes. Even turning off colloids (modify tpa.inp and comment out the colloid nuclide section) and undoing SCR #385 (use 4.2w versions of

seismo2.f, mechfail.e, mechfail.def, and mechfail.inp) cannot recreate the same output. There must be other changes that haven't been identified yet. Just document part A of SLT #1.

1-31-2003 -

Made modifications to the test plan/report for SCR #371. Didn't finish.

Checked with Randy Fedors about SCR #346. He said he'd get to check the latest output after removing sample mode 2 sometime early next week.

Got response back from Michael Muller on mods for the SCR document. Will implement soon.
He raised a question about the use of correlated sampled parameters. Were LocCO2PartialPressure_AllUZ_SZLayers[atm] and pH_AllUZ_SZLayers[StandardUnits] supposed to be added to the correlated inputs section of tpa.inp? Ron suggested that I ask Sitakanta or David Turner.

Ron and I had talked to Sitakanta earlier in the day. I had two questions: was it correct to use Kd instead of Ka in the equation for fracture Rds? He said yes. What about the note about adding a sampled parameter called 'CoefficientForResponseSurfaceUncertainty' (uniform, 0.9, 1.1). The note says to use it for 'a' in the Ka equation. However, it wasn't done and the values for the sampled parameter are nowhere near the values already in coefkdeq.dat for column a. I thought maybe it was intended to be a multiplier for a. He confirmed that it was supposed to be a multiplier but that it should be implemented at a later date. He also said that at some time in the future, he would like to be able to vary all 6 parameters (columns a - f) in coefkdeq.dat. Will add to the TPA 'to do' list.

Ron said to add the check for zero fracture aperture to the list, too.

Having some problems with email. 'cscherer@swri.edu' no longer works; must enter 'cscherer@cnwra.swri.edu' now. Ron got an error when he tried to reply to one of my emails. Looking into it.

Tried to make new tpa.e give same results as old tpa.e (before SCR #371 added). Turned off the colloid code (comment colloid section out of tpa.inp), used old mechfail.e, mechfail.inp, and mechfail.def as well using values from tpa.inp instead of coefkdeq.dat, but I still couldn't get back to the old values. There must be some other code changes that I'm not aware of or maybe I didn't negate all of the colloid stuff. At this point, I don't know what I'd need to do.

Ron says that with Pickett's new numbers for the Kds and Rds, that TPA ran 1024 realizations w/o failing. Maybe that particular problem has disappeared.

2-3-2003 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Completed adding Michael's changes and additions to the SCR doc. for #371. Continued w/ test plan. Modified FLT #1 and added FLT #2. Ran FLT #2 and SLT # 2 - 6.

The system level tests, so far, are:

SLT #1: compare new code w/ base
SLT #2: remove coefkdeq.dat - error
SLT #3: remove one of the elements from coefkdeq.dat - error
SLT #4: remove one of the rows for an element from coefkdeq.dat - error
SLT #5: change number of elements from 5 to 3 in coefkdeq.dat - error
SLT #6: change number of rows from 10 to 11 in coefkdeq.dat - error

The code does not handle the errors - the run aborts. Talked to Ron. He wants me to add code under the auspices of SCR #392 - a cleanup scr.

Planned system level tests:

SLT #7: change pH value from tpa.inp
SLT #8: change CO2 value from tpa.inp
SLT #9: change FractureForceFactors (UZ & SZ) from tpa.inp
SLT #10: change values in coefkdeq.dat?
SLT #11: change values in coefkdeq.dat?
SLT #12: change values in coefkdeq.dat?

I'm still debating on 10 - 12; are they necessary. Need to get good values for the parameters in 7 - 9. What are the correct ranges? Also need for range checking in code. Need to ask Sitakanta, maybe Turner. Sitakanta is busy today and asked me to come back Thursday.

NEXT TESTS:

- remove all info in coefkdeq.dat w/o uncommenting data in tpa.inp - error (SLT #2)
- remove all info in coefkdeq.dat & uncomment data in tpa.inp - runs OK, using old values (FLT # 2)
- remove coefkdeq.dat - error (SLT # 4)
- remove one elements data from coefkdeq.dat but leave number elements = 5 (SLT #3)
- incorrect number of elements and/or number of radioelements in coefkdeq.dat - error (SLT #5 - 8)
- set values in coefkdeq.dat - pending
- set values in tpa.inp for new sampled parameters - pending

When I did the error tests above, it was clear that there is no error handling in place. Talked to Ron; he agreed that it needed to be addeded. Using SCR #392, a general cleanup scr, to cover the mods. I will be making the mods and then finishing the test plan.

2-4-2003 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Worked on adding the error handling. Code mods in uzft.f and szft.f. Uzft.f reads in coefkdeq.dat. Both uzft.f and szft.f user FractureAperture values from tpa.inp, so need to prevent value of 0. The subroutine calc_rd uses aperture as the denominator when calculating Fracture Rd.

2-5-2003 -

Continued adding error handling to uzft.f and szft.f for SCR #371/392. Checking for wrong number of lines, wrong number of elements, and missing coefkdeq.dat file. Need to make changes to latest version of the code, TPA 5.0betaM. So, will not include the tests that fail in test plan for SCR #371. Make local test plan for 392 (it will include many fixes from different programmers)??? Better yet, make a corrected version in scr392 and copy the executable over to scr371 to test the error handling!

Missing coefkdeq.dat file: code in place to handle missing file, but if used '.lt.' instead of '.ne.' in the condition statement in uzft.f. Made correction. added 'end=110,err=150' to read statement for parameters in coefkdeq.dat. Added check for zero value for aperture in both uzft.f. and szft.f.

Saved slt3 - slt6 directories as slt3old - slt6old. Changed slt3 - slt6 to slt5A - slt5D and added cases slt5E-I:

- slt5E - bad value for nradioelements; changed 5 to #5
- slt5F - bad value for nrows; changed 10 to lo (el-oh)
- slt5G - deleted nrows
- slt5H - FractureAperture_TSw set to 0 in tpa.inp (use tpa.inp from scr392)
- slt5I - FractureAperture_STFF set to 0 in tpa.inp (use tpa.inp from scr392)

SLT #5 was using tpa_mod392 from scr392 directory in scr371 directory with environment variables pointing to scr392. Ran all the error tests - all passed.

REMAINING TESTS PLANNED FOR SCR #371:

- set values in tpa.inp for new sampled parameters (SLT #3A-D) - change parameters from normal to constant and pick two values to compare
  - LogCO2PartialPressure_AllUZ_SZLayers[astm] (-4.0 & -2.0)
  - pH_AllUZ_SZLayers[StandardUnits] (7.0 & 9.0)

- set values in coefkdeq.dat (SLT #4A-B)
  - pH
  - logCO2

Set up the tpa.inp files for SLT #3. Ready to run.

2-6-2003 -

Decided not to run SLT #4. The values in coefkdeq.dat are intended to be fairly static and not be changed frequently. If I don't come up with another test 4, I'll have to rename 5's files and directories.

Ran SLT #3 tests, but the results are strange. In the output (screen display) files, the only differences are w/ Np, usually from szft. Reran using 100K years sim time. Compared nefiisz.inp and nefiiuz.inp files between A2 and B2 runs (vary LogCO2PartialPressure) and between C2 and D2 runs (vary pH). Looking at Mobil Rds and Immobile Rds only elements affected were the 5 in coefkdeq.dat (Am, Np, Pu, Th, and U). Good. Results:

- A/B: nefiiuz.inp - as LogCO2PartialPressure increases, Mobil Rd in all legs for Th, Np, and Pu increases while Mobil Rd for U and Am decreases
- A/B: nefiisz.inp - as LogCO2PartialPressure increases, Mobil Rd in leg 3 increases for Th, Np, and Pu while it decreases for U and Am
- A/B: nefiisz.inp - as LogCO2PartialPressure increases, Immobile Rd in leg 2 increases for Th, Np, and Pu while it decreases for U and Am. All other Rds remain the same
- C/D: nefiiuz.inp - as pH increases, Mobil Rd in all legs decreases for Am, Np, Th and U while it increases for Pu
- C/D: nefiisz.inp - as pH increases, Mobil Rd in leg 3 increases for Pu and decreases for Am, Np, Th, and U
- C/D: nefiisz.inp - as pH increases, Immobile Rd in leg 2 increases for Pu and decreases for Am, Np, Th, and U

Except where noted, values for Rds (both Mobil and (Immobile) were the same in both files compared.

Is this correct behavior? I can't say. Whom should I ask?

Spock 100% full. Went through all my directories and deleted what I could. Spock is now 98% full. I still need to go in and compress files, but that can wait a bit. In the meantime, somebody else is working on filling up spock.

2-7-2003 -

Took the Rds from SLT #3 (all 4 cases) and put the values into a spreadsheet.

Ron & I asked David Turner about correlating inputs for pH and LogCO2PP. He is working on a number and will send it to Ron.

Showed David the results from SLT #3. He gave us some characteristic graphs for the 5 elements. We're not sure if Pu is behaving properly in the code, so Ron wants me to generate the numbers necessary to plot the corresponding graphs to see if they're the same as the book. This is going to be SLT #4.

George has finished w/ SCR #346. Now I have to confirm the results and this one will be done. At last!

Ron wants to have a meeting 1st thing (9:00 or 9:30) Monday morning to discuss what we have to get done the next couple of weeks.

David Turner says to put in the correlated input for LogCO2PartialPressure and pH, using a value of -0.83.

Spock ran out of room a couple of times - totally down to 0 available space. Someone cleared up a huge chunk though and we're down to 95% used.

2-10-2003 -

Went to Ron's meeting at 9:00 a.m. Also there were George Adams, Brandi Winfrey, Bob Rogers, and Andy Jank (?). Andy is also from Div. 10 and is working w/ George. Delivery date for the code is 2/28. The week of Feb. 24 - 28 is set aside for QA issues. All SCRs must be signed off so Bruce can approve the delivery and sign off, too. Ron wants us to finish the SCRs this week and just test in general next week. We are supposed to meet again 2/17 - same time, same place.

My tasks are to finish SCR # 371 and SCR #346. Re: SCR #371: need to finish utility program to generate the Ka prime values to generate the plots to check against the ones given to us by David Turner. Then I need to update the .wpd file for the test plan and turn all the stuff in to Ron. After that, back to SCR #346, where I will need to modify my test plan and rerun system level test(s) on the changes made by Randy and George. Randy didn't like the looks of the results from my first system level test runs and removed sample mode 2, moving sample mode 3 up into its place. See no need to re-run the process level tests.

Ron added the correlated inputs for LogCO2PartialPressure and pH in tpa.inp.

Problems with the system. When I was in vi, I kept getting kicked into some mode I couldn't get out of and got an error message saying that device /var/tmp was full. Turned out it was. Perry cleaned it out and no further problems.

Worked on utility program for SLT #4: util_slt4.f. This program will write out the results when Ka' (kaprime) is calculated before it is used to calculate kds and rds. Using the code Michael added to uzft.f to read in and sort coefkdeq.dat and to calculate Ka'. Looping through all elements for all the $logPCO^2$ values in coefkdeq.dat and for pH values from 5.0 to 10.0. Will

write out Ka' for each set of pH and PCO^2 values to be used as input into Excel to create 3-D graphs that will, hopefully, look like the graphs David gave us from the book.

Found an error in the code. In coefkdeq.i, params is declared using the dimensions (nradioelementsmax, nrowsmax, nparams). But in uzft.f, the array params is used as if it were dimensioned (nparams, nrowsmax, nradioelementsmax). Luckily (?), there were only 5 elements and 10 rows, so never had an array bounds problem and only code written by Michael in uzft.f and szft.f accesses this data. Talked to Ron. We re-dimensioned params in coefkdeq.i to match the way it was being used in the code. While we were there anyway, Ron and I discussed whether to change nradioelementsmax from 16 to 20, to allow for the later addition of calculated Kds/Rds for the colloids. Ron thinks there is a very low probability of that ever happening so decided to leave it as is. Don't need to add to todo list, either.

2-11-2003 -

Generated data using my utility program and graphed Ka' for the 5 elements of interest (Am, Np, Pu, Th, and U). They don't look much like like Turner's reference graphs from the book (T.: **GeoChemistry of Soil Radionuclides**; Ch.: *Role Of Radionuclide Sorption*, p. 245, Fig. 10-20, Plot of DLM predictions of important HLW redionuclides at discrete PCO_2 values). Talked to Ron. We went to see Turner again. He agreed that the graphs don't look very close. He mailed me an Excel file which he used to generate the values in coefkdeq.dat. I'm going to check and see if they are different. If yes, then I'll make a corrected coefkdeq.dat and see if that makes the difference.

Using the equations that Dr. David Turner generated, I input new values into a new coefkdeq.dat file. Almost all the numbers were different (6 remained unchanged in the whole file). Ran util_slt4_newcoef.e & plan to graph the new results tomorrow. No one seems to know where the numbers in the current version (before TPA 5.0betaP - I know we have at least an O version.).

2-12-2003 -

Rechecked the numbers I put into coefkdeq.dat yesterday. There were a few typos, which I corrected. Reran util_slt4_newcoef.e and graphed the output using Excel. The graphs are still wacky. Checked my code to see if I did anything screwy; didn't find anything. Checked that the table remains correct through the flip and sort - it's AOK. Not sure what to do next. Talked to Ron and Turner. Came to the conclusion that we have a precision problem.

Tested Am at -3.5 at 4 and 8 decimal places instead of the default 2 and the answer was much closer to what was expected at 8, though still not right on. Generated numbers for all Am series and modified coefkdeq.dat.

Talked briefly with Michael Muller. He didn't have any new insights for the current problem.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

2-13-2003 -

Checked the numbers I modified in coefkdeq.dat. Ran util_slt4.e and graphed the results. Am looks very good now. (Eureka!) Now, do it for Np, Pu, Th and U, too.

Used Turner's spreadsheet and set all the 5th order equations for all elements to 8 decimal places. (This involved changing the number resolution in each of 50 graphs.) Isolated the coefficients and transferred them to a modified version of coefkdeq.dat. (To do this, I copied the 50 equations to a separate area. Then I copied the equation to six cells per element. In each of these cells, I deleted everything but the coefficient I needed for specific columns in coefkdeq.dat. Then I copied these values to a text file. Converted them from dos2unix. Removed tabs. Inserted the new values into coefkdeq.dat and did a little cleanup formatting in coefkdeq.dat..) Ran the utility program for all 5 elements. Transferred the data to Excel and graphed the elements. Am, Pu, Th, and U look acceptable (the trends match the target graphs, although the numbers are not right on - how close do they have to be). Np has problems though. It is still not very smooth. More precision? 12? 14? Showed the results to Ron. He wants me to try Np (only, so far) at 12 decimal places of precision.

2-14-2003 -

Redid the coefficients for Np up to 12 decimal places. Updated coefkdeq.dat and reran util_slt4.e. Pulled the results into Excel and graphed them. Np looks basically the same as yesterday, so this is not a precision problem.

Is it a fit problem? Looking at Turner's graphs, it looks like Np doesn't fit the generated curve as well as most of the others. (Pu also has some outlyers) Should it be a different order equation? At Ron's suggestion, tried Np as a 4th order equation (5th order coefficients set to 0 in coefkdeq.dat), but only for LogPCO2 at -1.5 and -2.0. These equations don't fit exactly either, but are they worse than the 5th order ones? Reran util_slt4.e and graphed the results. The series for those two pCO2 look somewhat better; overall, the result is a little better than before. Will have to get the expert opinion from Turner. He doesn't seem to be here today. We haven't checked the values Turner has in his spreadsheet that he used to generate the equations.

Ron says to stop working on this problem and finish off SCR #371. More can be done later under the auspices of SCR #427. Need to get the code ready to ship now. Will have to notate slt4 that the test is OK, but the results show that a change is needed to get more accurate results. Then reference SCR #427, which will be an SCR for a data change. In tpatest/scr371/tpa43a, the updated data file is called coefkdeq_mod427.dat.

SCR # for updating coefkdeq.dat is 427. Should copy my work on coefkdeq.dat over to new subdirectory for SCR #427.

Started cleaning up the directory and started on modifying the tp_scr371.wpd

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Set up directory tpatest/scr427 and copied files from scr371 related to modifying coefkdeq.dat.
Copied scr371.wpd and tp_scr371.wpd to scr427/docs to modify for this SCR. Also brought in latest version of TPA code, 5.0betaP. Taking copies of docs home with me.

2-17-2003 -

Ron called a meeting this morning to discuss status of SCRs and validation of the TPA code. (Besides Ron and me, attendees were Bob Rogers, Andy Jank, George Adams, and Brandi Winfrey.)
I'm finishing off SCR #371 and SCR #346. Ron added a new one for me SCR #419.

Made corrections to the SCR document for SCR #427 and emailed copy to Ron; also gave him a hard copy. Brandi has been assigned to go over the numbers and make sure I got them in right.
Implementation of coefkdeq.dat w/ correct and more precise coefficients is complete unless we want to make some more tweaks to make Np look better.

Cleaned up directory contents for SCR #371. Had to re-run a couple of the system level tests to make sure the correct output files were available. Went through the test plan document and modified it through most of SLT #3.

Created directory tpatest/scr419 and brought over TPA 5.0betaR. Got SCR hard copy from Ron and he emailed me the electronic copy.

2-18-2003 -

All tests for SCR #371 passed except SLT #4, which spawned SCR #427. Noted same in tp_scr371.wpd. Signed off.

Finished modifying the test plan for SCR #371 and proofread it. Generated hard copy for Ron, and asked him to OK my conclusion of SLT #4 (spawns SCR #427, but passes). He wants it to read FAILED w/ my note of explanation. OK. Made change. Made the CD of all the test directories (scr371 and below). Generated directory listing and ran hard copy for Ron. Turned in SCR #371. The interesting graphs are contained in .xls files in scr371 and scr427.

Updated the "to do" list and sent copy to Ron.

Moved on to SCR #346. Compared George's latest test plan with the one I had started before he and Randy decided to get rid of sample mode 2. Will need to replace my system level tests with the new ones and re-run. I need to get results that are the same as George's; Randy Fedors has already looked at his results and verified that they are correct and reasonable.

Copied test plan to new directory scr346B. My test plan has a part A and a part B with different baseline and test versions, so I can keep the process level tests and results I already had

(A) and replace the system level tests (B) with the new ones. I have the report and CD from George and Randy's efforts. Began editing and setting up to run the system level tests in scr346B.

Brandi called. Need to bring her up to speed for testing SCR #427. Showed her the files and where the numbers came from. Right now, she is only going to verify the numbers in coefkdeq.dat, not test it.

Cleaned up directories for scr371 and scr346.

2-19-2003 -

Continued with testing SCR #346B. Run-time error in mechfail having to do with **. Removed -r8 (replace with xtypemap). Recompiled. Now it runs. Using version R tpa.inp. Doesn't have correct comments for this SCR, but the rest is OK and I needed the new parameters.

Started running the system-level tests. SLT #1 passed. SLT #2 passed. SLT #3 part A passed; part B bombed - error; array out of bounds, fixed by scr #425 according to Ron. Will need to start up again tomorrow using 5.0betaU.

Started testing SCR #419 copied TPA 5.0betaT, R & Q. Ran diffs on the code mods. No test plan exists - started test plan. Updated scr.

2-20-2003 -

Brought over newest version of TPA (V) to scr346B after I moved the current stuff into a subdirectory called tpa50betaR. Recompiled using 4.2 compiler. Started re-running the system-level tests starting with slt3b this time.

SLT #3b -

Brought over the latest version. When I compiled using Makefile4.2, it caused problems. TPA bombed in snllhs and mechfail. Don't seem to have the correct version(s) of the Makefiles. Finally, got around the problem by modifying codes/Makefile4.2. Use the other line for snllhs.e and replace -r8 in mechfail with xtypemap.

Ran slt3b in scr346C and slt3a in 346B. However, used wrong *maydtbl.dat files for a. Must redo.

2-21-2003 -

Re-started the run for slt3a. Used the exisiting spreadsheet to enter and compare the new results. Also re-ran slt1 and slt2 with version 5.0betaV. All tests passed today!

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Finished running system level tests for SCR #346. Cleaned up directories scr346 (no system level tests) and scr346B (no process level tests). Cleaned out unnecessary files from subdirectories, too.

Updated scr and test plan documents. Made CD. Generated directory listing for CD. Printed out directory listing, test plan, and scr for Ron. Turned it in.

Moved on to SCR #419. Brought over latest version - V. Start with slt1 and compare before and after versions. Won't be able to finish today.

Also set up a fresh version of Q in a subdirectory. Recompiled w/ 'make -f Makefile4.2' after correcting the mechfail problem in codes/Makefile4.2. Compiled V using 4.2.

Talked w/ Ron about tests, and these are the current ones planned:

- flt1: don't know for sure, but can check that values read in correctly and that math works appropriately
- slt1: before and after mods (what files change & what stay the same)
- slt2: set colloid release fractions (already there) to 0 from 0.5 (match no colloids at all in upper section - what about Irreversible Colloid Model?)
- slt3: set PermanentLossColloidFilterFractions to 0 (from 0.1)
- slt4: set PermanentLossColloidFilterFractions to 1 (from 0.1)

Look at nefii*.inp files and ebsnef*.dat files and nefii*.src files. Lets see what happens.

2-22-2003 -

Ran diff on all the output files from slt1/testa (50betaQ) and slt1/testb (50betaV). Used command file to help. Added more to the test plan.

Initial check of results: there are differences in lhs.out and samplpar.out showing that lhs is choosing different numbers. Not supposed to. They're only off in least signigicant digits. Ron wants me to run again, recompiling, re-inserting the -r8 switch in snllhs. Also wants me to use version R instead of V. V has added a fix that might affect the numbers slightly. R is the first version with the new colloid code.

In general, peak mean dose and overall release goes down. When looking at individual elements/isotopes, most go down for release/dose/Ci for the colloid-added code. U is different, it goes up but it doesn't have a colloid form. See if it still happens using R instead of V.

Going to make scr419B to avoid compilation-related problems. Actually, the Makefile4.2 in codes is the same in version R and Q. There is a difference for snllhs.e by the time we get to version V. So I had complied with slightly different options before. Used the default command for snllhs and change the options for compiling mechfail so that xtypemap replaces -r8. Recompiled both versions R & Q) in scr419B.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

2-23-2003 -

Checked the run from yesterday. The numbers in lhs.out still are off between the runs. I double-checked and it looks like gentpa in version R was compiled at a different time than everything else, so re-compiled and re-ran R. Sigh. Same problem.

Made directory scr419C containing TPA 50betaR and subdirectory containing TPA 50betaQ. Compiled this time using default compiler for each and re-ran. Still problems, except I didn't have the environment variables correct (forgot to add the B to 419), so re-running again. Eureka! This time it worked correctly and I lost more than just lhs.out and samplpar.res from the difference list!.

Ron found an error in ebsrel that will make one of my tests for SCR #419 fail. The corrected code is in version Z, SCR #440. He wants me to test it, too. Shouldn't be a problem, once I finish this one.

Printed out some of the difference files for SLT #1. Went on to set up and run other system level tests:

SLT #2: set it up (IrreversibleColloidModel = 0), ran it, aborted in ebsrel reading ebsfilt.inp; ref. SCR #440.
SLT #3: set it up (A: ColloidReleaseFractions = 0 & ColloidFilterFactors = 0; B: CRF = 1 & CFF = 0; C: CRF = 0 & CFF = 1; D: CRF = 1 & CFF = 1)

Plan to leave out a functional level test - will use the output in SLT #1 to comment on functionality.

2-24-2003 -

Tried to run tpa.e Q w/ CRFs set to 1 and it bombed, so instead of running Q with CRFs of 0 and 1 instead of .5, ran R w/ .5 in slt3b instead of 1 to see if it will match Q. Still doesn't match exactly. May not be possible to do. Leave this test out.

Looking at slt3d, expect no colloids to get through but they are. Because: ebsfilt.f - does test which, if it fails, bypasses the CFF stuff. Made functional level test on ebsfilt - manipulated ebsfilt.inp; ran test, passed once I made it fail the test in ebsfilt by setting ksat greater than flux in ebsfilt.inp. Except that it is processing the filter factors backwards. Ebsfilt multiplies release by filter factor instead of 1-filter factor. SCR #443 to correct error. Error in use of PermanentLossColloidFilter Factors in ebsfilt.f (r*flfc instead of r * (1-flfc).

Permanent Loss Colloid Filter Factors are the percentage of colloids that are stopped or filtered out. Colloid Release Fractions are the percentage of regular elements that are converted/commuted to colloids.

Still want test where set CFF to 1 in invert but zero in uz layers. BUT shouldn't matter what UZ layers have because it was all stopped in the invert. Slt3f CFF = 0 in invert and 1 in Chnv and Chnz., all else = 0.

Tomorrow, bring over latest version ZA to scr443 and see if can pass other tests now (like 4 - CRFs = 0.5, all CFFs = 0.0).

2-25-2003 -

Set up scr443 with latest version (TPA 5.0betaZa) for testing of SCR #443.

If fracture flow, so no filtering is done! Need to ensure that layers are matrix to test the colloid filters. Colloids in fracture flow bypass the filters per test in ebsfilt.

Ron and I tested together for awhile and we're both satisfied that code is correct. Now need to finish the test plan/reports for 419, 440, and 443 for tomorrow afternoon. Set AAIO to 4.0 so invert will use filters and hardcoded lowestmatfraction in uzft.f to 1 so all layers are matrix and filtering will be used all the time. Re-run test 3 cases a-d in scr443 directory as slt1.

There will be no colloids showing up on the screen display capture for SZ because they are recombined with the regular element before is is printed. They will only appear in the Highest release listings from subarea # and from UZ.

Ran all the test runs this afternoon for SCR #433 so I can concentrate on writing up the test plans tomorrow. Made directory tpa50betaQ. Made a run version of Q where I hardcoded lowestmatfraction = 1. Changed ArealAverageMeanAnnualInfiltrationAtStart[mm/yr] to constant 4.0 in the tpa.inp for Q and ran it . Hopefully, it will match one of my runs. Although it occurs to me, I need CRFs set to 1 or 0. Make slt2 and use a Za version copying tpa_slt1b.inp to tpa_slt2.inp and compare with Q.

2-26-2003 -

Finished the test Plan for SCR #419. Updated the SCR. Made CD of the test directories.. Printed directory listing.

Finished the test Plan for SCR #440. Updated the SCR. Made CD of the test directories. Printed directory listing.

Finished the test Plan for SCR #443. Updated the SCR. Made CD of the test directories. Printed directory listing.

Turned in all the above to Ron.

George is doing SCR #441.

Still one bug left on PC running Lahey90 4.5. George and Ron are working the problem. This one is also a show stopper as regards delivery.

2-27-2003 -

George and Ron fixed the bug last night. Only happens on the PC version running Lahey90. Required that you specify blocksize so code could handle large lhs binary file.

Officially at TPA 5.0. No more beta.

Ron wants me to make a list of the flags & modes that can be set in tpa.inp. Then, start testing flipping flags. Also, before I leave today, start a 1024-realization run at 100,000 years. Should be done by Monday.

Run is on Idaho. Sent flags.xls to Ron containing a list of flags/modes/switches & number of possible values.

3-3-2003 -

Return to validation testing.

Updated flags.xls. Needed to use PRODUCT instead of COMBIN.

The 1024-realization run stopped when Guardian rebooted, for whatever reason, over the weekend. Will start again, in the background, this time. Made new directory cscherer/valid/reals. Run on Idaho instead of Spock. Disk space at 99% capacity.

Started testing flags.Will run for 5 realizations at 100,000 years for awhile.

Set up flags_test.xls containing the flags/modes/switches and test values. Today, started the 1024-realization run in /var/tmp on idaho w/ environment variables set to spock (valid/reals). Also ran tpa_flags1 - 4.

3-4-2003 -

The 1024-realization run bombed in realization 277 in NEFTRAN, exceeded dimensions of array BF. Again. Ron wants me to run it again at 20,000 years and set StartAtRealization and StopAtRealization at 277. Did. It ran OK. The lhs.out files are identical. Then, Ron wanted me to do it again for subarea 1. These files can be used to compare with the original run. Should have run subarea 2, so I did.

Tpa_flags4 ran to completion. Re-ran and hit ctrl-C in the 3rd realization. Same results in checkpnt4.out and tpa_flags4.out.

Ran tpa_flags5 and tpa_flags6. These passed.

Re-ran tpa_flags2 interrupting the run and then continuing.

Talked with Ron about the 1024-realization run for 100,000 years. Decided to try increasing the time steps after the compliance period (finer grid into NEFMKS - decrease range of change in individual cell). Ran w/o aborting at realization 277, subarea 1! Started run on idaho for the full run increasing NumberOfTimeStepsAfterCompliancePeriod.

Created spreadsheet test_summary.xls to track my testing and test results.

3-5-2003 -

The 1024-realization was still going, around realization 240.

As I run the flag tests, I've been matching each test run w/ a duplicate test run where all parameters are the same except for GenerateRestartFiles, so I can check the check.pnt file processing. So, I get the same results from tpa_flags3 and tpa_flags4 as well as tpa_flags5 and tpa_flags6. But I have a discrepancy for tpa_flags1 and tpa_flags2. I've rechecked the tpa.inp files for each. They seem to be in order. I already re-ran the flags1 run and flags2 w/ no interruption. In the process of re-running flags2 w/ interruption.

Spock is running around 99% -100% full today. When it fills up completely, the TPA run is aborted (along with everything else) so I have to start over. Plan to free up as much space as I can and hope it's enough to leave some room for running (if nobobdy else gobbles it up this time). Got CDs to make backups of my old scr directories. I can delete most of them from the system once I get them reliably backed up.

3-6-2003 -

Spock hit zero again, interrupted runs. Went through directories and removed files that aren't needed or can be regenerated.

Ron working on a code fix for the NEFTRAN problem

Made backups today of SCR #s: 344, 346, 346B, 371, 373, and 374 on CD. Then did a tar of the directories (UNIX NOTE: tar cvf directoryname.tar directoryname). When that was finished, I zipped the tar file (UNIX NOTE: bzip2 directoryname.tar - results in directoryname.tar.bz2) These files are considerably smaller. This is clearing up some usable space. Once I have all the directories tarred and zipped, I will also backup the compressed files on CD.

3-7-2003 -

Made backups today of SCR #s: 375, 376. 377, 379, 379B, 382, 390, 392, 393, 394, 395, 396, 399, 408, 412, 414, 416, 419, 427, 440 and 443 ,as well as directories: checkpoint, comp_study, invert, neftran_bomb, and neftran_glitch ,all on 3 CDs. The directories were special

projects or tests that I've run over the past year. Ron also has copies of all the SCRs in the individual folders. Then, I tarred and zipped the directories. When I finished with all of them, I backed up the zipped files on 1 CD (instead of 4). These files will stay on the system so I have something I can delete in a hurry in an emergency. All CDs were written by Nero at 12x. This effort freed up a significant amount of space - down to 96% instead of 100%

3-10-2003 -

Ron is freeing up some space on spock, too.

Continued with more flag testing - see spreadsheet

Met with Ron at 4:00 about validation testing. Brandi was also there. New priority: mods to files for validation testing. Pick a routine w/ a lot of variables and format per Ron's example by next Monday. Put off testing until this done.

People working on validation:
Bob Rogers - off for a while.
Andy - 40%
Brandi - 40%
George - 50%
Carol - 100%

Rob Rice will be back, coming in April, will work from El Paso; 80%.

3-13-2003 -

My files to validate are ebsfail.f, dcags.f, uzft.f, invent.f, uzflow.f, and subarea.f

My first routine will be prenefmks in uzft.f since it should be one of the worst and cover most eventualities.

Made format subdirectory under validate. Brought over 5.0 and made executable w/ Makefile4.2. Then made uzft_prenefmks subdir to work in for subroutine prenefmks. When done, it will contain the files prenefmks.f, prenefmks.h, and prenefmks.t.

Removed implicit statements, replaced them with implicit none (which is in the .h file). Declared all variables and formatted according to Ron's directions. Made file fmt.txt to bring in the necessary headers. Re-formatted prenefmks. Brought mods into uzft.f (see uzft_fmt.f) and re-made. Went on to resolving compile errors, including:

invent_.i & maxinvnuc - need to be in the .h file; used to dimension parameter
undeclared variables - all of them have to be declared now

3-14-2003 -

Got uzft to compile with new files and format.

Made subdirectory under format for each files. Then, separated all files into individual routines/functions, like prenefmks. Made templates for the .h and .t files (hfmt.txt and tfmt.txt). Copy them to working area and convert to files for specific routine.

3-15-2003 -

Working on coming up with tests for prenefmks.

Edited files for uzft subroutines (uzftmain, opnfil, aftnefmks, getvertlayers, putgwtt, trapint) and the function ucljs..

3-17-2003 -

Copied files from the weekend onto spock. Ran TPA. OK.

Included opnfil in uzft & recompiled. Ran TPA. OK

Included ucljs in uzft & recompiled - function invoked from other places than uzft. Tried to run TPA, but am getting strange errors.

3-18-2003 -

Met w/ Ron and the rest of the team (Brandi, Andy, George) at 9:00 to go over our experiences with re-formatting files.

Problems: function - code change required to make Ron's format work, due to assignment statement required in all functions - must match name of function. So, must change functionname = in the code to functionname_t =.

Right, now all 3 files I've worked on (prenefmks, opnfil, ucljs) are in uzft, but a zero-length leg comes out in nefii.inp for the repository leg (REPLEG) that bombs NEFTRAN. Just what I needed, another way to break NEFTRAN.

3-19-2003 -

Talked to Ron about function problem; hold off on functions for now until he decides how he wants to handle them. He doesn't want to have to do code changes. He will work on another way to do functions.

Added uzftmain and runnefmks, aftnefmks, getvertlayers,

bomb in run REPLEG set to 1.0 but writes out 0.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

3-20-2003 -

Ran diff on uzft_fmt.f and uzft_orig.f - can't see the problem.

maxnsuba.i

writevelocities, writesource, getelements, putgwtt, trapint, calc_kd, calc_rd, part of ljs and ljs2 (just *.h) added in

calc_ssarea commented out - didn't do test cases

Flaky REPLEG problem:

1) don't have the problem with the default compiler, just solapps 4.2
2) REPLEG declared as integer instead of double precision, still set to 1, though
   so, maybe it's some weird interaction between the solapps compiler and format statement?????

Declared REPLEG and SPASTP as double precision. Recompiled.

Got the same results from original tpa.e compiled using default compiler and tpa.e modified w/ uzft changes compiled using solapps 4.2 compiler; also, same results using original tpa.e compiled w/ solapps 4.2 compiler & modified uzft tpa using default compiler. YAY!!!!

3-21-2003 -

Took out extra print statements for REPLEG problem. Did some more general reformatting.
Ran TPA a couple of times. Saved tpa_fmt1.out for comparison.

Went back to working on cases for opnfil, prenfmks, getverlayers. Have a plan of attack for next week.

Need to turn in Electronic Scientific Notebook to Ron Janetkze by next Tuesday. Should be from where we left off last time through today.

They had to reboot Spock today. Having trouble with Guardian. Word is giving me weird messages, some about space, but this file is small and there is still room on C. D drive is basically full, none of it is mine. Most seems to belong to DFarrell. So I will back up my ESN files to floppy before I try to get them ready for Ron to have.

Ron says he has a new scheme for doing the functions. He'll show me when he has some time - there have been a lot of meetings, and what not.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

3-24-2003 -

Turned in Scientific Notebook to Ron last Friday for October, 2002, through March 21, 2003.

Continued writing tests for uzft.

Ron gave me SCR #445 to test on TPA 5.0a. This change imposes some limits that prevent TPA from bombing when extreme values for are sampled for some parameters (repository leg velocity, Rds, SAV leg length). Ron wants it finished this week.

Set up test directory scr445 under tpatest. Made two subdirectories: tpa50 and tap50a. Re-compiled everything using Makefile4.2. Started tp_scr445.wpd and scr445.wpd.

3-25-2003 -

Available space on spock became an issue again today. It went down to zilch and aborted everyone's programs. Perry Seely sent out an email to the biggest users asking them to clear up space. Some seem to be doing it, but not the BIGGEST users yet. I'm at about 840 M since I archived so much several weeks ago.

Unix note: use /usr/local/bin/du -sh * in /net/spock to get listing of all users disk space usage. Or insert specific path and directory instead of * to get a specific dir usage.

Team meeting at 9:00 in B237. (Me, Brandi, Ron, George, and Andy.) New protocol for functions to try.

Worked on SCR #445. Ran all three specific realization numbers on TPA 5.0 and TPA 5.0a. Aborted on the former and ran on the latter. Left a full 1024-realization run running in the background. Will check tomorrow. Hopefully, there will be enough space. About 5% has been cleared up during the day. Plan to finish 445 tomorrow and turn it all in to Ron if I can.

3-26-2003 -

Completed the test plan for #445, except that Ron says when he added Andy's latest change, it bombed NEFTRAN at realization 5. He wants me to add this one to the test. TPA 5.0c should be available tomorrow so I can finish. Ron says not to wait for the full 1024-realization run, although it may well be done sometime tomorrow.

Back to validation testing, formatting files, and writing test routines.

3-28-2003 -

TPA 5.0c wasn't available yet to finish testing SCR #445. Back to formatting files and creating test cases.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

3-29-2003 -

Worked on formatting subarea.f, ebsfail.f, dcags.f, invent.f, and uzflow.f.

3-31-2003 -

The 1024-realization run completed Sunday morning. It ran successfully.

Tested Ron's new format for functions on ucljs. It works fine.

Accidentally deleted uzft.f (was trying for uzft.o). Will need to rebuild it from all the subroutine/function files and probably fix some compile errors where I didn't copy the corrections back to the small files.

Worked on formatting and creating test cases in uzft.f.

4-01-2003 -

Met w/ Ron at 9:00 a.m.. Rob Rice is here for the week. Also met Eric Peterson (?sp) from Div. 07 who has joined the team. George and Brandi were also there. Andy was absent. Ron handed out the writeup of what we're doing. Eric is doing sampler instead of George. Ron wants a Word Perfect table containing the subroutine/function names, test names, and test descriptions from each of us.

Finished testing SCR #445 with the new test added. Gave CD, directory listing, test plan/report, and SCR to Ron. Started a 1024-realization run using TPA50c - pid 10602.

Rob has some archived test cases from when he did utility testing in 1997. This may help me w/ invent and subarea.

Worked on test cases for uzft.f.

4-02-2003 -

Opnfil test cases completed.

4-03-2003 -

Completed test cases for ucljs, ljs, and ljs2.

4-04-2003 -

The run started Tuesday is still going. Probably won't complete until sometime Sunday, if it doesn't abort. Expect that it will run to completion.

Rob going home today, will be back in mid-May. Got his phone numbers and email address.

In uzft.f, worked on test cases for getelements, getvertlayers, putgwtt, trapint, writevelocities, and writesource. Remaining: aftnefmks, prenefmks, runnefmks, uzft, calc_kd and calc_rd. Completed: opnfil, ucljs, ljs, and ljs2. Working on the idea for a generic routine to include that will help generalize display messages for my validation tests.

4-07-2003 -

The 1024-run finally completed this morning very early (around 2:00 a.m.). Zipped up the files for this run; some are very large.

Some of George's routines are dead code, so he needs some more to work on. Ron is giving him uzflow.f. If I get finished with everything else, I'll check to see if I need to take any back.

Continued with test cases (on-site) and formatting files (laptop). Have all of uzft test cases on paper - just need to finish implementing them.

4-08-2003 -

Continued with test cases (on-site) and formatting files (laptop).

4-09-2003 -

Continued with test cases (on-site) and formatting files (laptop).

Finished the validation print utilities (uzft_util.t and uzft_util.h). Hope Ron likes them. I think it'll make my test cases easier to implement - less overall typing. Added statement at bottom of uzft.f : 'include uzft_util.t. I may rename it vt_display_util.t to make it more generic. I can extern them in other routines (include external statements in use_vt_display_util.h). Currently, can grep on display output file for 'VT' and get all the messages printed out using the utilities. Should be easy to change to any format Ron wants.

I think I'll put my utility to check last file access (& any other necessary file-related utilities) in a separate file (vt_file_util.t). Vt_file_util.t contains subroutine lastfileaccess and function fileupdated. Added calls to file utilities in prenefmks.t. The include for vt_file_util.t is at the end of uzft.f.

4-10-2003 - 4 -11- 2003 -

Continued with uzft files, but took time out to finish up utilities.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Ron passed out handout with the format he'd like to see in the output. Small changes to print utilities will put me code in compliance. Made mods to vt_print_utilities.f and vt_file_utilities.f.

4-15-2003 -

Finished making the print utilities more generic. Distributed vt_print_utilities.f to the others.

Updated *.t and *.t1 to use the print utilities and match Ron's format for output.

Worked on final re-formatting mods to the uzft files and completing outstanding test cases.

4-16-2003 -

Worked on file utilities (vt_file_utilities.f) and prenefmks.t. George has added two routines to vt_print_utilties.f, although he still needs to comment them. Added testcases 3-5 to prenefmks.t. Need to debug 5 and finish tc6. Output is looking good and like Ron's example.

4-17-2003 - 4-18-2003 -

Continued w/ uzft.

Made two new print utilities: print_dparray2d and print_dparray1d - used George's utilities floattochar and inttochar. The new utilities could use some fine tuning in aligning the columns, but that can wait for later.

4-21-2003 -

Copied prenefmks.t to checkin directory. Test case 7 will cause the TPA run to abort w/ a divide by zero flag raised. In this test case, qm3peryrinsag is set to all 0's. This does cause a trap for too low infiltration for the Van Genuchten curve to be tripped, but the run is not stopped, which is puzzling, since there is a STOP statement in the code in uzft.??!!!! (Oh, I get it. The stop does occur; the divide by zero is just a warning message. Check array infil in uzft.f:prenefmks.) The else-if block for tc7 is commented out. Sent email to Ron.

Continued w/ test cases. Trapint.t ready for testing.

4-22-2003 -

Ran trapint tests. Problem: Code says that n <= 0.0 is an error. Prints out error message but doesn't stop run. Returns value of zero. Is that what they want or should there be a STOP? Need to keep a list of such errors/bugs/questions until Ron gets back from travel.

He left today and should be back 5/5/03. Brandi is gone for two weeks (to England for reserves) also and should be back 5/5/03.

There will be a meeting 5/19 for PA staff to talk about the consequence modules. We're supposed to attend and listen. Ron gave me a copy of the "Software Validation Test Plan For the Total-System Performance Assessment Version 5.0 Code" to review. The experts have written about how they expect to test the models used for the code. Ron wants us to integrate anything we can in the test cases we're currently developing.

Finished test cases and ran them before putting them in checkin directory for: trapint.t and getvertlayers.t. Worked on getelements.t

Read Validation Test Plan once.

Re-checked formatting of subarea.f; modified to meet current output format.

4-23-2003 - 4-25-2003 -

Went through validation test plan again so see what I could start adding to my test cases.

Continued with test cases in uzft.f. Getelements.t, putgwtt.t, and updated uzft.f sent to checkin/uzft subdirectory. Added subdirectories in checkin for each file; it was getting too full and unorganized.

Updated formatting in invent.f, dcags.f, and ebsfail.f.

Updated part of tpatestcase.wpd.

4-28-2003 -

Finished updating tpatestcase.wpd with tests finished so far.

Continued with test cases in uzft.f. More test cases designed for aftnefmks, calc_rd, calc_kd, and runnefmks.

4-29-2003 - 5-9-2003 -

Continued with test cases in uzft.f. More test cases designed for writevelocities, writesource, and uzft. When these are implemented, uzft will be complete until more tests requested by Ron or KTIs. When these are implemented, uzft will be complete until more tests requested by Ron or KTIs.

Did final (hopefully) pass through subarea.f, dcags.f, ebsfail.f, and invent.f for formatting - ready to finish test cases.

Trying to locate an error ???. The Peak Mean Dose has changed, but not sure when it first changed. Tested first by commenting out single include *.t lines, but didn't find it. Then, commented out all include *.t lines for subroutines (left functions) and got original PMD. Then, started un-commenting include *.t lines, one at a time. Opnfil.t, getelements.t, getvertlayers.t, prenefmks.t, putgwtt.t, trapint.t, uzft.t, aftnefmks.t, runnefmks.t, writesource, writevelocities, calc_kd, calc_rd, ljs.t, ljs2.t, and ucljs.t are all OK. The error didn't show up again. Oh well. Anyway, I moved two test cases from prenefmks towritesource and writevelocity - they're done. Finished off calc_rd and calc_kd so I can get back to subarea.

For test cases that cause early termination or that abort, set iEntryCount in if condition to 0, so they won't be included in run. Ron says to get rid of testcase5 for getelements - don't need to test for number of isotopes less than 27.

Completed going through the uzft files and copying them up to checkin.

Worked on new test cases for ebsfail.f.

5-12-2003 -

Worked on new test cases for ebsfail.f, dcags.f, invent.f, and subarea.f.

5-13-2003 -

Worked on new test cases for ebsfail.f, dcags.f, invent.f, and subarea.f.

Met with Ron, George, Andy, and Brandi to discuss progress and continuing plans for validating version 5.0 of the TPA code. Ron says NRC is considering version 5.1. George will finish uzflow and work with uzflow KTI on validation.

Probably won't start on the experts' validation tests for at least another week. Those tests are listed in validation plan and take top priority. Work on automatic tests when not busy with experts (KTIs or SMEs).

Having trouble with run-time error in invent since adding in formatted invent.f.

5-14-2003 - 5-16-2003 -

Finished integrating in all formatted files and got the code to compile cleanly. Tests stubbed for all functions and routines. Working on filling in specific tests.

5-19-2003 - 5-23-2003 -

PA meeting Monday from 9:00 a.m. until 5:00 p.m. attended by Ron, George, Andy, Eric, Brandi, Rob Rice, me, Sitakanta, Roland, Michael Smith, Osvaldo, Oleg, Gordon, Lane Howard. Overview of the code preparing for validation testing.

Went through dcags and ebsfail files again. All ebsfail files are at a point where they can be compiled and included in a test run. Still need to finish up some of the dcags .t files before they can be included. Same case for invent and subarea. Dcags, invent, and subarea are all ready for the "final" pass before they can be included in a test run. Including a minimal number of tests; others can be added.

Meeting next Tuesday with Gordon and Oleg about validation tests for SZFT (5 tests) and UZFT (3 tests). Need to set something up with Osvaldo for ebsrel, ebsfail, and weldfaild; with Michael Smith for dcags; and with Ron for invent (5 tests) and subarea (3 tests).

5-27-2003 -

Met with Gordon and Oleg in Gordon's office and with Tim McCartin on the phone. Made plans for who will do what. Draft test plans due in a week. I'm backup for Tim. Thursday or Friday, Tim plans to outline what goes on in uzft and szft. Will wait until we see what we get from Tim.

Set up meeting with Osvaldo for tomorrow at 1:30 a.m. for ebsfail (5 tests), and weldfail (3 tests). There is no separate weldfail module - it's all ebsfail.

Met with Ron about invent and subarea. Made plans for the tests I'll be doing..

Waiting to hear back from Michael Smith for dcags (3 tests) about when he wants to meet. Talked to him briefly. He expects to meet this week and probably will be out next week.

All these tests are supposed to be ready to run by Friday, June 13, 2003.

5-28-2003 -

Met with Osvaldo Pensado and guys from NRC (telecon) for testing ebsfail and weldfail.

5-29-2003 -

Working on automated tests and validation tests for subarea.

5-30-2003 -

Set up meeting with Michael Smith, Brandi, and LaPlante for next Monday afternoon to discuss dcags (& dcagw, volcano, ashplumo) testing.

Continued work on subarea tests.

6-2-2003 - 6-6-2003 -

Continued working on subarea and invent.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Met with Michael Smith and Pat Laplante on validation testing for dcags. I'm supposed to do some code inspection and hand checking to make sure that the sampled parameters from tpa.inp are used to update some input files (gdefaults.inp/def, ggeniis.inp/def, and gftranss.inp/def) for gentpa.

Ron released a new version of TPA (5.0f). I need to merge changes in invent and uzft into my formatted files. May affect testing (hopefully not the tests I've already done).

6-6-2003 -

All space on Spock is used up as of this morning. Makes it very difficult to work.

Gordon asked me to assist Sitakanta with some validation testing. Andy is supposed to work with Sitakanta on ebsrel, but he isn't here. I will need to do some similar process for ebsrel as I need to do for dcags.

6-9-2003 -

Spock full again. Perry Seely has asked the managers to ask their big users to free up some space. Set up directory on texas to use as interim place to work, but it's on a scratch drive '/tscr1/cscherer' so I will have to make sure everything I do is copied off by the weekend. So far, I haven't had to use it because Brandi was able to delete 5G of old stuff on spock.

Finished merging in changes from TPA 5.0f into my formatted files for uzft.f and invent.f.

Started looking at how to do Sitakanta's tasks.

6-10-2003 - 6-11-2003 -

Ran tpa (5.0f) with base case tpa.inp to get tpameans.out. Used it as tpa.inp for a mean value run. Ran again. Used ebsrel portion of tpa.inp (& some of ebsfilt, nfenv, and ebsfail) to see which sampled parameters were used to modify default files into input files for other codes (e.g., ebsfilt, releaset). Printed out files id'd in code: ebsrel.def, ebsrel.inp, ebsnuc.pac, ebsfilt.def, ebsfilt.inp, wpflow.def, and wpflow.dat. Compared files and mean values.

Started similar process with dcags. Michael/Pat and dcagw.

At one point today, spock had 14% available. Wow. I've never seen it that empty.

E-mail to Ron re:invent, etc.

6-12-2003 - 6-13-2003 -

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Looked at files for dcags and dcagw. Ran TPA twice, once base case and once with FractionOfRepositoryWasteInGlassForm set to .24 instead of zero to get glass. Sitakanta wants files from ebsrel, ebsfilt, failt, and releaset. Printed out files (base case): ebsfilt.def, ebsfilt.inp, ebsflo.dat, ebspac.nuc, ebsrel.def, ebsrel.inp, ebstrh.dat, wpflow.def, and wpflow.dat. Also looking at ebsnef.dat and ebsnef2.dat - didn't print them because they are 100 pages long. Waiting to hear from Sitakanta if he wants more than these files.

Talked to Ron re: invent and which tests to run.

6-19-2003 - 6-20-2003 -

Went through the diagrams of decay chains that Ron gave me. Determined what chains needed to be added to tpa.inp as well as additional sampled parameters for new elements/isotopes. Started modifying tpa.inp. Started running TPA. Each run found additional parameters that I needed to add to tpa.inp. Once I got past that problem, discovered that NEFMKS only allows decay chains with 6 members. The biggest "real" chains that I'm testing is 7. Needed to change MAXMEM in nefmks.f from 6 to 7. Info from runs before the change are in subdirectory svt1a.

6-23-2003 - 6-27-2003

Finished modifying code and tpa.inp for the 1st validation test (.svt1). Ran TPA. Results are in subdirectory svt1. Modified tpa.inp for 2nd validation test (different chains - svt2). Ran TPA. Results are in subdirectory svt2.

Printed out the invent portions of tpa_svt1.inp and tpa_svt2.inp. Used them to make changes for tpa_svt3.inp. Still need x???? runs to cover all the chains.

Ron wanted me to add a new subroutine (printinspectmsg) to vt_print_utilities. Also, he wanted more information on each line about which test was being run; the output can be confusing with nested tests. Changed outputstring to accept the TPA module name, the software routine name, and the test number. Then modified all my *.t and *.t1 files to pass the correct parameters to outputstring. This seems to be working OK. Sent the changed code and sample output to Ron to see if there are still other changes he would like.

Had meeting with Ron, Brandi, George, and Andy 6/26 re: validation testing.

Made another round of mods to vt_print_utilities.f so old outputstring stays around and the new one is called outputstring4. Had to go back into all my *.t and *.t1 files and change calls from outputstring to outputstring4. Updated files in my checkin directory for Ron.

6-30-2003 - 7-5-2003 -

Finished testing for invent. Just need to finish writing up SVTRs.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Finished testing for data transfer test for ebsrel-related files. Just need to finish writing up SVTR.

Andy is supposed to take over the subarea test that Ron was going to do where we test 11-14 subarea and then test > 14 subareas. He hasn't yet.

Brandi has taken over the data transfer test for dcags/dcagw.

Eric Peterson will be doing invent test relating to checking if inventory calculations are correct.

Helping Gordon with uzft test. Running version i.

Ron needs all the Software Validation Test Reports (SVTRs) either by COB July 3rd or email them to him July 6th.

Version j is available now.

7-7-2003 - 7-12-2003 -

Sitatkanta: E7-1 - sent draft SVTR to Ron and Sitakanta; responses back; talked wi/ Sitakanta about other runs to do;

Gordon: C9-3 - finished running 7/11; finished analyzing 7/12 (all OK), updated SVTR-sent to ron and gordon do CD after Gordon OKs.

C10-5 - started runs 7/11; worked on reconciling tpa.inps

E3-1, E3-2, E3-3, E3-5 (E3-4 by Eric Peterson and Rob Rice)

7-14-2003 -

Worked on C10-5. Problem: test failed for 0.5 infil when WastePackageFlowMultiplicationFactor was set to a constanct of 1000.0 instead of a sampled parameter (changed the numbers selected for Sps). Turned out to be because a) total effective filter factor includes PermanentLossColloidFilterFactors for legs/layers that aren't retained and b) doesn't exclude PFLCFFs where the layer is 0.0 meters thick. Also, we don't print out anywhere a list of layers whose PFLCFFs are part of the total effective filter factor (toteffplflfc). So, Ron will make changes (SCR #46?) and add a new file that will list all layers considered for filter factors, matrix or fracture and why a layer is tossed out and not included in nefii.inp.

TPA version 5.0k just came out today. Will need l to finish C9-3 and C10-5.

7-15-2003 - 7-16-2003 -

Finished work on C9-3 and C10-5. Turned in hard copies to Ron. Still need to burn CDs. Ron decided to have me do the test by adding print statements to uzft.f. He will update the code later, so that 0.0 m thick layers are not used in the colloid filtering process.

Back to C7-1 (EBSREL).

7-21-2003 -

Brandi is testing the SCR where MXMEM in SIZES.INC is increased from 6 to 7. I lent her a tpa.inp that has 7-chain members for the Nuclides and Chains section.

Gordon wants me to check the SVTR and tpa.inp for C10-2.

C7-1 (EBSREL): setting IModel = 5 in tpa.inp is invalid. The results are bogus because both spent fuel (non-glass) and glass form waste are run through releaset using imodel = 5. User should not set IModel to 5 in tpa.inp. Ebsrel resets Imodel to 5 if wasteform is equal to 2 (1=spentfuel; 2=glass). Ron released version5.0m today which included a change to ebsrel related to the glass wasteform processing. Re-ran all tests using 5.0m.

George is going to work on subarea instead of Andy.

7-22-2003 -

New deadline; Gordon wants to deliver early. Ron wants all SVTRs by COB Thursday.

Finished C7-1 EBSREL SVTR and turned it in to Ron.

Andy is working on SCR 450? He was confused on what to look for. I showed him C9-3 and C10-5 and explained Gordon's testing. I think he'll be able to use the same kind of test for his test plan.

7-23-2003 - 7-24-2003 -

Finished E3-1, E3-2, E3-3, and E3-5. Turned in hard copy reports to Ron.

7-28-2003 -

Burned CDs for E E3-1, E3-2, E3-3, E3-5, C7-1, C9-3, and C10-5. Printed directory listings for the CDs and turned them in to Ron.

Started SCR #471. Changed equation to calculate Rd - add divide by porosity. Changed way dispersion calculated for STFF leg.

Made directory scr471 under cscherer. Brought over TPA 5.0m and TPA 5.0n. Ran both base cases for comparison.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

7-29-2003 -

Finished SCR #471.

Question popped up while I was analyzing the results of the runs. In nefiisz.inp, there are porosities listed for each leg; one is Mobil and one is Immobil. In the case of the Mobil porosities, it works out that for the Rd calculations, the following holds true:

$$\frac{\text{(Mobil Rd value version m - 1)}}{\text{(Mobil Rd value version n - 1)}} = \text{porosity} \quad \text{(i.e., Immobile Poros.)}$$

The same is not true for the Immobile porosities, because, in szft.f, the ImmobilePorosity_STFF is multiplied by the ImmobilePorosityPenetrationFraction_STFF and stored in the pori array. This value is written out in nefiisz.inp as the Immobile Porosity value. However, when szft calls calc_rd, it passes porosity which is the same as ImmobilePorosity_STFF. So, in nefiisz.inp, the RDs reported were calculated w/o the PenetrationFactor while the Immobile Porosity does include Penetration Factor. The following relationship does hold, however, for Immobile Porosity:

$$\frac{\text{(Immobile Rd value version m - 1)}}{\text{(Immobile Rd value version n - 1)}} = \text{porosity} \quad \text{(i.e., ImmobilePorosity_STFF)}$$

Ron sent an email to Tim McCartin for resolution. Tim replied back that the way it is now (inconsistent use of porosity within nefiisz.inp) is correct. I'm still not sure I understand why two different values for porosity should be used, but if the expert says so, then that's the way it will be programmed. See following email:

```
From: Timothy McCartin [TJM3@nrc.gov]
Sent: Wednesday, July 30, 2003 6:31 AM
To: RJANETZKE@cnwra.swri.edu
Subject: Re: SZFT

Ron:

I am not sure I fully understand the question - we may need to talk this one through.
Generally, the Kd and Rd are properties of the rock and should use the rock porosity
(if we use the immobile porosity we would need to further revise the equation that use
the density of the rock to that portion of the rock that is represented by the
immobile porosity) -  my verification tests used 2 of Turner's 5 and the values will
come out correct if the matrix porosity is used.   The immobile porosity is used not
for the retardation but for the matrix diffusion calculation - i.e., immobile porosity
should not affect the determination of Kds and Rds.  I can discuss this later today if
necessary - I have ACNW most of the day.


Tim

>>> ron janetzke <rjanetzke@cnwra.swri.edu> 07/29/03 09:35PM >>>
Tim,
```

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

```
We have a question (related to the STFF Immobile Rds of the five
nuclides calculated by the Turner equations) we would like to resolve
before we package the code on Wednesday.

When you suggested the change to line 4495 of uzft.f to insert a
division by porosity,  did you want that value to be
ImmobilePorosityPenetrationFraction_STFF * ImmobilePorosity_STFF as seen
at about line 1340 of szft.f?   It appears that the Immobile Porosity
[pori()] in the Migration Path Properties Array section of nefii.inp is
equal to this product, whereas the rds of the Turner 5 are determined in
the calc_rd routine near the top of  szft.f  (about line 323) using the
ImmobilePorosity_STFF  porosity value and the rds values are
subsequently  written in  the Element Properties Array section of
nefii.inp.  The question is, do these two uses of the porosity need to
be consistent, or is one ignored by NEFTRAN?  You would not have noticed
a difference unless you were running the numbers on one of the five
Turner nuclides.

thanks,

ron j
```

7-30-2003 -

Added Excel spreadsheet to SCR #471 to demonstrate the test I did to pass the test.

The printer problem that kept me from printing the past two days has been cleared up.

Burned CD and printed out directory listing for Ron. He wants me to go and backfill some .t files w/ automatic tests. Got the subarea files from George Adams. He modified 3 .t files for validation testing. Waiting to hear from Eric Peterson on what he did for invent validation tests.

Validated code for TPA 5.0 has been delivered.

7-31-2003 - 8-1-2003 -

Back to working on the .t (& .t1) files. As far as I can tell by looking at Ron's .t subdirectory, George worked on qchitsa.t, qlhitsa.t, and qphitsa.t from subarea leaving gsaxym.t, linehitline.t, quadrilteral.t, root.t, solve2x2.t, solvenxn.t, ssadb.t, ssadba.t, and triangle.t to complete. I had already done fcn.t1, checkin.t, gnsa.t, gsamtu.t, gsanwp.t, gsanwpglass.t, gsarea.t, and gsaxy.t. For the invent files, it appears that Eric worked on allchains.t, chains.t, chainsolver.t, decay43mol.t, decay43molglass.t, decayremove43mol.t, newinventdb.t, and setage.t. That leaves activityperiso.t1, alambdaperiso.t1, amolepermtuatemplaceperiso.t1, cipermtuatemplaceperiso.t1, dmyexp.t1, epalimperiso.t1, halflifeperiso.t, indexofcolloidparent.t1, nameelem.t1, qpermtu.t1, and wmoleperiso.t1 for me to complete. I also need to finish checkvalue.t, copylinesgs.t, dcags.t, gentodcfgs.t, and gentpags.t in dcags.

8-4-2003 -

Finished gsaxym.t and copied to checkin/subarea. Worked on inventfiles/activityperiso.t1 and subareafiles/linehitline.t.

8-5-2003 -

Finished subareafiles/linehitline.t and copied to checkin/subarea.

8-6-2003 - 8-8-2003 -

Worked on subareafiles/quadrilateral.t.

8-11-2003 -

Finished quadrilateral.t. Rechecked fcn.t1 and checkin.t. Started root.t and triangle.t.

8-12-2003 -

Finished triangle.t. Started solve2x2.t and ssadb.t.

8-13-2003 -

Worked on root.t and solve2x2.t. Started ssadba.t and solvenxn.t. Finished ssadb.t and ssadba.t. Copied up to checkin/subarea. Continued working on remaining subarea .t files: root.t, solvenxn.t, and solve2x2.t.

There is a TPA_ERROR showing up in uzft:prenefmks, writevelocities, and writesource having to do with input files for NEFMKS being created. They used to work; will have to check this out.

8-14-2003 -

Had email forwarded to me from Gordon. He wanted me to add a disclaimer to all files in TPA and standalone codes that used alogrithms or code examples from Numerical Recipes. NRC wants it ASAP. Both version 5.0o and 4.1j. See text:

```
From: Gordon Wittmeyer [gwitt@cnwra.swri.edu]
Sent: Wednesday, August 13, 2003 5:27 PM
To: Carol Scherer
Cc: Ronald Janetzke; Sitakanta Mohanty
Subject: FW: Disclaimer

Carol:

NRC is very anxious for us to add the disclaimer cited below to the TPA
Version 5.0 as well as the TPA Version 4.1j source code.  Since Ron is out
on vacation until next Monday, August 22, 2003, I would ask that you please
```

do this as expeditiously as possible. Please work with Sitakanta and Bob Brient to determine if the addition of these comments to the numerical recipes subroutine should result in a new release.

I will back in the office on Friday, August 15, 2003.

Thanks,
Gordon

-----Original Message-----
From: Gordon Wittmeyer [mailto:gwitt@cnwra.swri.edu]
Sent: Tuesday, August 05, 2003 9:15 AM
To: Ronald Janetzke
Subject: Disclaimer

Ron:

Please modify the disclaimer I've prepared based on directions from Numerical Recipes Software direction (/net/spock/home/gwitt/NumRec/disclaimer) as needed and insert into every Numerical Recipes routine we used in the TPA code and associated standalones. This needs to be done ASAP.

Gordon

My reply:

From: Carol Scherer [cscherer@cnwra.swri.edu]
Sent: Thursday, August 14, 2003 2:29 PM
To: Gordon Wittmeyer
Cc: Sitakanta Mohanty; Ronald Janetzke
Subject: Numerical Recipes disclaimer

I found 19 subroutines in six different source code files (both versions) that looked like they needed the disclaimer. They are:

in condxyzt.f:
xgauleg (1)

in numrecip.f:
qgaus (2)
beta (3)
gammln (4)
gauleg (5)

in ran.f:
gasdev (6)
betapdf (7)

in codes/ashplume.f:
qromb4 (8)
trapzd4 (9)
trapzd3 (10)
trapzd2 (11)
trapzd1 (12)

```
      polint4          (13)
      polint3          (14)
      polint2          (15)
      polint1          (16)

in codes/failt.f
      qsimp            (17)
      rkqc             (18)

in codes/releaset.f:
      odeint           (19)

The modified files for TPA 5.0 are in /net/spock/home/cscherer/NumRec5.0o
and the files for TPA 41j are in /net/spock/home/cscherer/NumRec4.1j.
```

Finished solve2x2.t and solvenxn.t and copied them to the checkin/subarea directory.

8-15-2003 -

Numerical Recipes disclaimer:

example of disclaimer:

```
      c*********************************************************************
      c*** This routine, beta, is based on the routine beta                ***
      c*** function from the book                                          ***
      c*** "Numerical Recipes in FORTRAN (Cambridge University Press,      ***
      c*** Copyright (C) 1986-1992 by Numerical Recipes Software. Used by  ***
      c*** permission. Use of this routine other than as an integral part  ***
      c*** of the TPA code requires an additional license from Numerical   ***
      c*** Recipes Software.  Further distribution is prohibited.          ***
      c*********************************************************************
```

Mail from Gordon:

```
From: Gordon Wittmeyer [gwitt@cnwra.swri.edu]
Sent: Friday, August 15, 2003 11:48 AM
To: 'Carol Scherer'
Cc: 'Ronald Janetzke'; 'Sitakanta Mohanty'
Subject: RE: additional routines for Numerical Recipes disclaimer

Ron:

I will be leaving for NRC on Monday morning around 10:00 AM CDT.  Please
prepare tapes of the TPA Version 4.1j and TPA Version 5.0o codes that have
the disclaimers for the Numerical Recipes routines and ship them to NRC via
overnight mail.

Hope you had a nice break.

Thanks,
Gordon

-----Original Message-----
From: Carol Scherer [mailto:cscherer@cnwra.swri.edu]
```

Sent: Friday, August 15, 2003 11:37 AM
To: 'Carol Scherer'; 'Gordon Wittmeyer'
Cc: 'Ronald Janetzke'; 'Sitakanta Mohanty'
Subject: RE: additional routines for Numerical Recipes disclaimer

Gordon, I replaced the disclaimer from derivs in releaset.f with a statement saying it's a user-supplied routine that is called by routines that are based on routines from Numerical Recipes, etc. All files are ready for Ron to send out a new version Monday.

-----Original Message-----
From: Carol Scherer [mailto:cscherer@cnwra.swri.edu]
Sent: Friday, August 15, 2003 11:04 AM
To: Gordon Wittmeyer
Cc: Ronald Janetzke; Sitakanta Mohanty
Subject: additional routines for Numerical Recipes disclaimer

Gordon, the files you listed for me were:

trapzd	*- found it in failt.f; added disclaimer*
tred2	*- found references to tred2 in snllhs.f, but no such routine; there is a tred3 that is already attributed to another source; didn't add disclaimer*
tqli	*- couldn't find it; closest was routine in snllhs.f named IMTQL2 that is already attributed to another source; didn't add disclaimer*
derivs	*- found it in releaset.f; don't really think this is from Numerical Recipes; added disclaimer but couldn't find routine to reference; version of Numerical Recipes I'm looking at refers to a user-supplied routine called derivs that should return derivatives*
ranf	*- found a ranfe in ran.f; already attributed to another source; didn't add disclaimer*
rkqs	*- found it in releaset.f; added disclaimer*
ran1	*- found it in ran.f; also routine ran1seis; both routines already attributed to another source; didn't add disclaimer*

The updated files for both 4.1j and 5.0o are in NumRec4.1j and NumRec5.0o, respectively.

Finished root.t and copied it to the checkin/subarea directory. That's all of subarea .t files.

Started work on dcagsfiles/checkvalues.t. Also need to finish dcags.t, copylinesgs.t, gentodcfgs.t, and gentpags.t from dcagsfiles.

8-18-2003 - 8-19-2003 -

Spock down.

8-20-2003 -

Spock is back up with a lot more space available. Should come in very handy.

Ron looked at the Numerical Recipes disclaimer files. He said that we should add rk4 to the list. He took care of it. The disclaimers will be part of TPA 5.0p along with changes to dcags by Michael Smith.

8-21-2003 - 8-22-2003 -

Worked on .t files in dcagsfiles. Finished checkvalues.t and dcags.t, and copied them to checkin. Started copylines.t. Still have gentodcfgs.t and gentpags.t to complete before dcags is done.

8-26-2003 - 8-27-2003 -

Finished copylinesgs.t, gentodcfgs.t, and gentpags.t. There is a code problem with copylinesgs (in dcags) and also copylines (in dcagw). No error checking is done for nlines = 0. If nlines is greater than number available, error is caught, but processing continues, causing abort. Second problem can be solved with STOP in error trap. First problem will require a few lines of new code. Could also do away with one copy of the subroutine. Found small error in vt_file_utilities.f. Fixed it and made some code more generic. Copied all code mods to checkin. Emailed Ron that the files are ready for him.

8-28-2003 -

Ron wants me to add the Numerical Recipes disclaimer text to the same routines in tpa4.1jpd. Copied over tpa41jpd from Ron's area. Made subdirectory NumRec4.1jpd.

8-29-2003 -

Inserted disclaimers into 22 subroutines/functions in 6 source code files: condxyzt/xgauleg, numrecip/qgaus, numrecip/beta, numrecip/gammln, numrecip/gauleg, ran/gasdev, ran/betapdf, ashplume/qromb4, ashplume/trapzd4, ashplume/trapzd3, ashplume/trapzd2, ashplume/trapzd1, ashplume/polint4, ashplume/polint3, ashplume/polint2, ashplume/polint1, releaset/odeint, releaset/rkqs, failt/qsimp, failt/rkqc, failt/rk4, and failt/trapzd. The six modified files were placed in ~cscherer/NumRec4.1jpd, ready for Ron to copy over.

Back to finishing .t1 files for invent.f. There are 14 files to complete:

```
activityperiso.t1
alambdaperiso.t1
amolepermtuatemplaceperiso.t1
cipermtuatemplaceperiso.t1
dmyexp.t1
epalimperiso.t1
halflifeperiso.t1
indexofcolloidparent.t1
indexperiso.t1
```

```
indexperisonostop.t1
nameelem.t1
nameiso.t1
qpermtu.t1
wmoleperiso.t1
```

Nine of the functions are similar since they are "per iso", so the tests can be cut and pasted once one is done. Finished activityperiso.t1, halflifeperiso.t1, epalimperiso.t1, indexperiso.t1, indexperisonostop.t1, and wmoleperiso.t1. Finished files were copied up to the checkin directory.

There is an erroneous error message in the subroutine newinventdb that says an error occurred in cipermtuatemplaceperiso (line 2137); should be cleaned up. Have discovered that the following functions are never called:

alambdaperiso
amolepermtuatemplaceperiso
cipermtuatemplaceperiso

9-2-2003 -

Continued with .t1 files for invent. Finished indexofcolloidparent.t1, qpermtu.t1, namelem.t1, nameiso.t1, and dmyexp.t1. qpermtu.t1, nameelem.t1, and nameiso.t1 files were copied to checkin There are questions for Ron to resolve on indexofcolloidparent and dmyexp.

While working on indexofcolloidparent.t1, found interesting problem. My 2nd testcase didn't work because it appears that the value of cpname was retained from the previous call to the function. Since cpname was not saved, I didn't expect that to occur. I was testing a made up colloid, so cpname should have been blank. If I init cpname to '' and skip the search loop if cpname is blank, then the testcase works as expected. Need to check with Ron about this one.

Line 966 is "dmyexp = 1.0d0 + 99" when it should be "dmyexp = 1.0d99".

9-3-2003 -

Redid copylinesgs.t so that a new subroutine (createfile) is called that copies a specified number
of lines from tpa.inp into fromfile.txt for use in the tests. Don't need to have the file around ahead of time now. Copied the .t file back up to checkin/dcags.

Ron gave the dmyexp error SCR # 473. He says the indexofcolloidparent problem isn't really an error. Reader.f does an error check on all isotopes that are input, so one as obviously wrong as Ju235 won't happen.

C. Scherer SCIENTIFIC NOTEBOOK No. 170-4e
INITIALS:_____

Sent scr473.wpd to Ron. This latest form doesn't have a "Tested By" line. It has "Implemented By" and "Completed Date".

9-4-2003 -

Found 2 files for the form. One is right and one isn't. Corrected SCR and sent to Ron.

9 -16-2003 -

Final entry. This notebook is being closed. See #612-4e for future entries.

No original text entered into this Scientific Notebook has been removed.

Carol S. Scherer 9-15-2005

I have reviewed this scientific notebook and find it in compliance with QAP-001. There is sufficient information regarding methods used for conducting tests, acquiring and analyzing data so that another qualified individual could repeat the activity.

11-09-05