6.3.6 Unsaturated Zone Transport

The proposed repository at Yucca Mountain is located in the unsaturated zone about 300 m above the current water table. The unsaturated zone is composed of layers of welded and unwelded tuff that form a sequence of hydrologic units. The hydrologic and geochemical properties of the units and the infiltration flux control radionuclide transport through the unsaturated zone. The unsaturated zone transport component of the total system model tracks the movement of radionuclides released from the engineered barrier system to the water table. In general, radionuclides can migrate through the unsaturated zone as dissolved species or attached to colloids. Physical and chemical processes that can affect the transport of radionuclides in the unsaturated zone include: advection, diffusion, dispersion, sorption, matrix diffusion, colloid-facilitated transport, radioactive decay, and climate change. The impact of those processes on system performance is studied by running multiple realizations with different parameter values.

Performing multiple realizations (numbering in the hundreds) for such a complex system requires that the software used for simulating radionuclide transport in the system be efficient while also being able to handle complex physical and chemical processes with sufficient accuracy. FEHM (Finite Element Heat and Mass Transfer Code) (Zyvoloski et al. 1997 [100615]) was selected for simulating radionuclide transport in the system because of the efficiency of the particle tracking method and its ability to handle advection, dispersion, sorption, matrix-diffusion, and multiple-species radionuclide decay/ingrowth in the system (see Section 5.3.6 for assumptions) (CRWMS M&O 2000 [141418], Section 1).

The FEHM particle tracking model is based on the "residence time/transfer function" (RTTF) method (CRWMS M&O 2000 [141418], Section 1). The particle-tracking method in FEHM views the fluid flow computational domain as an interconnected network of fluid storage volumes. There are two steps in the particle-tracking approach: 1) determine the time a particle spends in a given cell (residence time); and 2) determine which cell the particle travels to next. Since particles travel only from cell to cell, requiring no greater resolution of the particle pathways, the computation burden is greatly reduced compared to the conventional particle tracking method that requires interpolation of the velocity field. This method enables large-scale transport simulations using millions of particles in a dual permeability system with complex source terms.

The following sections discuss the inputs and implementation of FEHM in the TSPA model.

6.3.6.1 UZ Transport Model Components and Input Parameters

The UZ transport model is based on the dual permeability model to address the importance of fracture flow and fracture-matrix interactions on radionuclide transport. Influence of lateral flow on radionuclide transport is studied through the three-dimensional flow model. Since within the projected system performance period there can be climate changes, the impact of climate change is addressed by switching one steady-state flow field to another steady-state flow field for the changed climate and studying the repose of the system to changes in flow field.

In the current TSPA model 26 species are traced through the UZ (Table 6-93). The following processes are simulated with FEHM to reflect radionuclide transport processes in UZ: advection, diffusion, dispersion, sorption, matrix diffusion, colloid-facilitated transport, radioactive decay/ingrowth, and climate change. Detailed information on the numerical model can be found in CRWMS M&O 2000 [123913] and CRWMS M&O 2000 [141418].

The following sections describe the component of the UZ transport model (numerical grid, flow fields, source term release from EBS, and outflow boundary at water table) and transport parameters.

Numerical Grid

The FEHM numerical grid is the same as the TOUGH2 UZ flow grid, the only difference is that the TOUGH2 grid is transferred into the format FEHM can read (CRWMS M&O 2000 [123913], Section 6.1). There are a total of 47,664 physical nodes in the grid (CRWMS M&O 2000 [123913], Section 6.1.2). For the dual permeability model, the number of actual nodes in the model is doubled. Grid blocks within the repository region are refined in the vertical direction for better resolution of the flow behavior around the repository. The following grid file is used in TSPA simulations for different infiltration scenarios (see Table 6-70).

Table 6-70. Grid File used in TSPA Simulations

Grid File Name	Comments
fm_pchm1.grid	Numerical grid used in all TSPA simulations

DTN: SN9910T0581699.002 [126110] (Unqualified, TBV# 3946)

Flow Fields Used in FEHM

In the TSPA simulations FEHM does not simulate the flow fields, rather, FEHM directly reads in pre-generated flow fields containing saturation, pressure, and flux data. In current TSPA simulations three climate states are considered and within each state there are three infiltration scenarios (e.g., low infiltration, medium infiltration, and high infiltration). Thus, there are a total of nine flow fields to reflect possible climate changes (as described in Section 6.3.1.1) and infiltration scenarios. The generation of the flow fields is documented in a separate report (CRWMS M&O 2000 [123913], Section 6).

FEHM Stiffness Matrix File

The FEHM stiffness matrix file (*.stor) contains connectivity arrays and control volumes for the grid. Information inside the stiffness matrix file is used by FEHM during particle tracking simulations for calculating particle residence time and for determining the movement of particles in the field. The stiffness matrix data in current TSPA simulations are directly derived based on TOUGH2 inputs (CRWMS M&O 2000 [123913], Section 6.1.2) and are read in by FEHM at run time. Table 6-71 lists the stiffness matrix files used in TSPA simulations.

Table 6-71. FEHM Stiffness Matrix Files used in TSPA Simulations

Stiffness Matrix File Name	Comments
fm_pchm1.stor	Contains connectivity arrays and control volume for the grid.

DTN: SN9910T0581699.002 [126110] (Unqualified, TBV# 3946)

Repository Release Bins and Water Table Collecting Bins

The following section describes the process of assigning nodes to bins for use in FEHM particle tracking. The bins are assigned for radionuclide source nodes at the EBS/UZ interface (i.e., within the repository) and for nodes at the UZ/SZ interface (i.e., nodes at or beneath the water table). These bins are defined so that radionuclides can be released at nodes that share similar prescribed infiltration ranges and so that particles can be captured in one of four quadrants defined for the SZ. Table 6-72 lists the data used in deriving the repository release bins and water table collect bins.

Table 6-72. Input Data used for Generating Repository Release Bins and Water Table Collect Bins

Title	Data Tracking Number (DTN)	Comments
3-D Uz Model Grids For Calculation Of Flow Fields For Pa For Amr U0000, "Development Of Numerical Grids F Or Uz Flow And Transport Modeling."	LB990701233129.001 [106785] (Unqualified, TBV# 3678)	Used dual-permeability mesh file containing the ELEME and CONNE cards for perched-water model #1.
Tspa Grid Flow Simulations For Amr U0050,"Uz Flow Models And Submodels."	LB990801233129.007 [118710]	Used glacial-transition, low infiltration TOUGH2 input file for perched model #1 that contains ROCKS card for hydrologic parameters and GENER card for infiltration rates.
Tspa Grid Flow Simulations For Amr U0050,"Uz Flow Models And Submodels."	LB990801233129.009 [118717]	Used glacial-transition, medium infiltration TOUGH2 input file for perched model #1 that contains ROCKS card for hydrologic parameters and GENER card for infiltration rates.
Tspa Grid Flow Simulations For Amr U0050,"Uz Flow Models And Submodels."	LB990801233129.011 [118722]	Used glacial-transition, upper infiltration TOUGH2 input file for perched model #1 that contains ROCKS card for hydrologic parameters and GENER card for infiltration rates.
Heat Decay Data And Repository Footprint For Thermal-Hydrologic And Conduction-Only Models For Tspa-Sr (Total System Performance Assessment-Site Recommendation	SN9907T0872799.001 [111485]	Repository outline used to define repository nodes.
Post-Processed Flow Fields For Rip: Developed Data From Amr U0125 (Abstract Flow Fields For Rip)	SN9910T0581699.002 [126110]	Used file that lists fracture nodes in prescribed repository. Used grid file that contains coordinates for all FEHM nodes.

FEHM Repository Release Bins–Radionuclides will be released from nodes corresponding to the repository location. Although the exact design and layout of the repository has not been determined as of this analysis, nodes within the repository footprint of the Enhanced Design Alternative (EDA) II design are identified to provide a source region for radionuclide release to the Unsaturated Zone (UZ) for TSPA-SR calculations. Figure 6-154 shows the location of these nodes within the prescribed repository footprint. The total number of physical repository release nodes within the region is 275 (CRWMS M&O 2000 [142004] Attachment II).

It was desired to combine these nodes into groups (or bins) that shared common infiltration ranges. This would help to categorize release points according to high or low infiltration rates. The five bins that were chosen were 0-3, 3-10, 10-20, 20-60, and > 60 mm/year (See Figure 6-154). These values were based qualitatively on seepage distributions and infiltration values that were used in the "column" models in the thermal-hydrology studies (see Section 6.3.1.2).

The software routine T2_BINNING v. 1.0 was used to group all the repository fracture nodes into one of the prescribed bins based on data listed in Table 6–72. The routine first reads in the prescribed repository nodes and determines the appropriate node numbering. It then determines the surface-infiltration rate corresponding to each repository node and places the node into the appropriate bin. A file containing a listing of these bins and the associated nodes was created for three cases: (1) low infiltration, glacial-transition climate; (2) medium infiltration, glacial-transition climate; and (3) high infiltration, glacial-transition climate. Because the spatially variant infiltration is different for each case, each file contains different nodal assignments for the bins. Sample input and output files are provided in Attachment IV along with a listing of T2_BINNING v. 1.0. A sample hand-calculation to verify the correct performance of the routine is also included.

Radionuclide Collecting Bins At UZ/SZ Interface–For the UZ/SZ interface, all nodes at (or below) the water table of the UZ model were grouped into four regions (or bins). The total radionuclide mass flow rate in each of these four bins will be focused at a random point (within each of the four bins) in the SZ model to reduce the effects of artificial dilution between the model interfaces. The four regions (Figure 6-155) are defined by an east-west boundary at a NSP northing coordinate of 233,590 m and a north-south boundary at a NSP easting coordinate at 171,200 m (CRWMS M&O 2000 [139440], Figure 3). The location of the four bins is based on the breakthrough locations of particles at the water table as described in Section 6.3.7.1.

The software routine WT_BINNING v. 1.0 was developed to group all nodes at or beneath the water table in the UZ model according to one of the four regions based on data listed in Table 6-72. As recommended in CRWMS M&O 2000 [123913], Section 6.2, a conservative water-table elevation at 850 m is used for all future-climate scenarios in TSPA-SR simulations. Therefore, the routine identifies the coordinates of all nodes in the UZ model, and if the node is beneath the water-table elevation of 850 m, it groups it into one of the four bins in a file that can be used by FEHM. Attachment V provides input and output files and a listing of the software routine.

Assignment of Transport Properties to Nodes

The previous sections described how nodes were assigned to bins to facilitate the release of radionuclides into the UZ and the collection of radionuclides for the SZ. In FEHM (Zyvoloski et al. 1997 [100615]), the hydrologic parameters can be initially assigned to zones based on stratigraphic units (PTn, TSw, etc) or directly to individual node. In TSPA simulations both methods are used and described below.

FEHM Zone Files–In current TSPA simulations 5 zone files are used (Table 6–73). The first two files in Table 6–73 are from DTN: SN9910T0581699.002 [126110]. The last three files include the infiltration bins described above for the low, medium, and high infiltration scenarios of the glacial-transitional climate. The division of zones in each file for non-repository nodes is based on rock properties; while for repository nodes the division of zones is based on surface infiltration values (see Section 6.3.1.2 and Attachment IV). Hence each non-repository zone contains nodes that are in the same geological unit. In current TSPA model a total of 48 fracture material layers and 48 matrix material layers are defined. Rock properties can then be easily assigned to each node through the defined zones.

Table 6-73. Zone Files used in FEHM

Zone File	Comments
fm_pchm1.zone	Defined zones based on rock geological units
fm_pchm1.zone2	Defined zones based on rock geological units plus repository nodes.
fm_pchm1.zone2.0100	Defined zones based on rock geological units plus repository nodes for low infiltration scenario.
fm_pchm1.zone2.0200	Defined zones based on rock geological units plus repository nodes for medium infiltration scenario.
fm_pchm1.zone2.0300	Defined zones based on rock geological units plus repository nodes for high infiltration scenario.

Based on DTN: SN9910T0581699.002 [126110] (Unqualified, TBV# 3946)

Assign Node Properties Individually–Because the division of repository release bins was based on surface infiltration values, one repository bin could contain nodes from different geological units. Thus, it is impossible to assign transport parameters to those nodes through the defined release bins. In this case we just directly assign transport parameters to individual node. For simplicity a software routine (MAKEPTRK V.2.0) was developed to assign transport parameters based on rock geological units to each node in the FEHM ptrk macro. Attachment VI contains a listing of the routine, along with a sample output file.

Radionuclide Release from EBS to UZ

Due to corrosion waste packages in the repository eventually fail; this process varies both in space and time. At early times a few packages may fail, releasing radionuclides into the UZ. As time proceeds, a greater number of packages may fail. In VA (Viability Assessment) calculations (DOE 1998 [100550], Figure 4-10), releases from the EBS to UZ (Unsaturated Zone) were spread over the entire repository sub-regions. Such treatment of the EBS release could result in significant artificial dilution of the UZ transport source term in some circumstances. In reality, waste packages may not fail uniformly in space and time. Rather, a few waste packages may fail at early times while others may fail gradually over longer time periods. An EBS random release model was developed in FEHM to allow the model to simulate early failed packages and time- and spatially-variable radionuclide releases (CRWMS M&O 2000 [141418], Section 6.2.4). The EBS random release model allows a user to release radionuclides from individual nodes to study the impact of early failed packages on system performance or to release radionuclides from randomly selected nodes within repository release bins. The number of selected nodes corresponds to the number of packages failed during the current time step.

In current TSPA simulations the number of early failed packages are 0. The number of failed packages within each repository bin are determined by the WAPDEG model and passed to FEHM. FEHM then randomly selects the releasing nodes and carries out the UZ transport simulations.

In all TSPA simulations radionuclide particles are released from repository fracture nodes only so as to reflect the nature of flow in fractured media and be conservative.

Unsaturated Zone Transport Parameters

The transport process of radionuclide in the unsaturated zone is determined by flow fields, fracture and rock properties, the properties of the radionuclide, and water chemistry. As the flow fields are predefined and described in Section 6.3.1.2, this section focuses on transport parameters that affect radionuclide movement in the media.

Transport mechanisms that can affect radionuclide transport include: dispersion, matrix diffusion, adsorption, fracture surface retardation, radionuclide decay, and colloid-facilitated transport.

Parameters that directly or indirectly affect radionuclide transport in the UZ and are simulated in TSPA include:

- Dispersivity (m)
- Matrix porosity and rock density (kg/m3)
- Matrix adsorption coefficient
- Matrix diffusion coefficient (m2/sec)
- Fracture residual saturation and fracture gamma parameters
- Fracture porosity, fracture spacing(m), and fracture aperture (m)
- Fracture surface retardation factor
- Matrix pore size (micron)
- Colloid size distribution, colloid K_c, colloid R_c, and colloid filtration factor
- Radionuclide half lives and daughter products.

Among those parameters, matrix porosity and adsorption coefficient affect the movement radionuclide in the rock matrix. Fracture residual saturation and fracture gamma parameters determine the effective fracture saturation and the fracture-matrix contact area reduction factor in the active fracture flow model, which together with the matrix diffusion coefficient partially determines the strength of the matrix diffusion process. Matrix pore size and colloid size control the filtration process between rock matrix interface, which can significantly impact the transport process of irreversible colloids, especially at the interface where water-carrying colloids flow from rock matrix with a larger pore size into rock matrix that has pore size smaller than the size of the colloids. Colloid K_c determines the mass balance between radionuclide in the water and the radionuclide attached to the colloid. Colloid R_c affects the retardation of colloid. Colloid filtration factor f_c controls the size exclusion effects of colloids moving from fracture into matrix.

The range or distribution of those parameters were reported in different AMRs and supplied to the TSPA model. In the TSPA model multiple realizations are carried out to investigate the environmental impact of possible radionuclide release from the proposed repository. For each realization parameter values are sampled based on their corresponding statistical distributions using the Latin Hyper-Cubic model inside GoldSim. The use of those parameters in the TSPA model is documented in the following sections.

Dispersivity–Yucca Mountain site-scale flow models have indicated that flow in the fractured rock system is dominated by fast fracture flow (CRWMS M&O 2000 [134732], Section 6.2.2, Figure 8). Thus, in such a system radionuclide transport is primarily advection controlled. The use of the FEHM particle tracking method also requires that the system be advection dominated with a Peclet number greater than or equal to 1 (Zyvoloski et al. 1997 [100615]).

As there are few data available on dispersivity distributions at Yucca Mountain site, we base our choice of dispersivity on previous published research papers. Neuman 1990 [101464] showed that field dispersivity varied with the scale of study. As the proposed repository is about 200 m above the raised water table (850 m) (CRWMS M&O 2000 [123913], Section 6.2), at this scale the dispersivity could vary from 8 m to 70 m (Neuman 1990 [101464], Figure 3). Field tracer tests at C-holes at Yucca Mountain also showed that on a 100 m scale, field dispersivity had a range of 10 m to 50 m (CRWMS M&O 2000 [152773], Table 52). Thus, in current TSPA simulations the fracture and matrix dispersivities are both set at 20 m. Figure 6-156 and Figure 6-157 show that transport of Tc99 is not sensitive to variations in dispersivity within UZ. Use of a dispersivity over 20 m does not show any significant effect on radionuclide peak concentration or arrival time.

Since most of the grid blocks have a dimension of larger than 20 m (pchm1.grid, DTN: SN9910T0581699.002 [126110]), the grid Peclet number for most grid blocks should be bigger than 1, which should also satisfy the requirement of FEHM (Zyvoloski et al. 1997 [100615]). Table 6–74 lists the dispersivity values used in TSPA simulations.

Table 6-74. Fracture and Matrix Dispersivities used in TSPA Simulations

Dispersivity	Values(m)	Used in FEHM File
Fracture	20	ptrk.multrlz, ptrk.multrlz.0100, ptrk.multrlz.0200, and ptrk.multrlz.0300
Matrix	20	

Matrix Porosity and Rock Density–Matrix porosity is used to calculate the matrix pore volume associated with each matrix block and matrix porosity and rock density values are used to determine matrix retardation factors used to simulate the adsorption process. Values of matrix porosity and rock density are from the TDMS (DTN: LB997141233129.001 [104055]) and are listed in Table 6-75 below (rock densities for the perched water materials denoted with a "pc" in the first two characters are obtained from "pch1.rock" in DTN: SN9910T0581699.002 [126110].

Table 6-75. Matrix Rock Porosity and Density Values

Matrix Layer	Matrix Porosity	Rock Density (kg/m^3)	Used in FEHM File
Tcwm1	2.53E-01	2550.	pch1.rock For all TSPA Simulations
Tcwm2	8.20E-02	2510.	
Tcwm3	2.03E-01	2470.	
Ptnm1	3.87E-01	2380.	
Ptnm2	4.39E-01	2340.	
Ptnm3	2.54E-01	2400.	
Ptnm4	4.11E-01	2370.	
Ptnm5	4.99E-01	2260.	
Ptnm6	4.92E-01	2370.	
Tswm1	5.30E-02	2510.	
Tswm2	1.57E-01	2550.	
Tswm3	1.54E-01	2510.	
Tswm4	1.10E-01	2530.	
Tswm5	1.31E-01	2540.	
Tswm6	1.12E-01	2560.	
Tswm7	9.40E-02	2560.	
Tswm8	3.70E-02	2360.	
Tswm9	1.73E-01	2360.	
Ch1mv	2.73E-01	2310.	
Ch2mv	3.45E-01	2240.	
Ch3mv	3.45E-01	2240.	
Ch4mv	3.45E-01	2240.	
Ch5mv	3.45E-01	2240.	
Ch1mz	2.88E-01	2310.	
Ch2mz	3.31E-01	2350.	
Ch3mz	3.31E-01	2350.	
Ch4mz	3.31E-01	2350.	
Ch5mz	3.31E-01	2350.	
Ch6mz	2.66E-01	2440.	
Pp4mz	3.25E-01	2410.	
Pp3md	3.03E-01	2580.	
Pp2md	2.63E-01	2580.	
Pp1mz	2.80E-01	2470.	
Bf3md	1.15E-01	2570.	
Bf2mz	2.59E-01	2410.	
tr3md	1.15E-01	2240.	
Bf2md	2.59E-01	2240.	
Pcm38	3.70E-02	2240.	

Table 6-75. Matrix Rock Porosity and Density Values (Continued)

Matrix Layer	Matrix Porosity	Rock Density (kg/m^3)	Used in FEHM File
Pcm39	1.73E-01	2240.	pch1.rock For all TSPA Simulations (Continued)
Pcm1z	2.88E-01	2240.	
Pcm2z	3.31E-01	2240.	
Pcm5z	3.31E-01	2240.	
Pcm6z	2.66E-01	2240.	
Pcm4p	3.25E-01	2240.	

DTN: LB997 141233129.001 [104055]
SN9910T0581699.002 [126110]

Matrix Adsorption Coefficients–Matrix adsorption coefficients for different rock types are taken from the TDMS (DTN: LA0003AM831341.001 [148751]). Values of the adsorption coefficient are divided into three groups based on rock type (e.g., devitrified, vitric, and zeolitic). Table 6-76 lists the statistical distribution of matrix adsorption coefficient for different radionuclide types. It shows that Plutonium has a high average adsorption coefficient of 100 ml/g in vitric and zeolotic layers, which may suggest that movement of Plutonium particles in those layers can be significantly delayed.

Table 6-76. Sorption-Coefficient Distributions for Unsaturated Zone Units

Element	Rock Type	Min K_d (ml g^{-1})	Max K_d (ml g^{-1})	E[x]	COV*	Distribution type
Americium	Devitrified	100	2000			Uniform
	Vitric	100	1000	400	0.20	Beta
	Zeolitic	100	1000			Uniform
	Iron oxide	1000	5000			Uniform
Plutonium	Devitrified	5	70			Uniform
	Vitric	30	200	100	0.25	Beta
	Zeolitic	30	200	100	0.25	Beta
	Iron oxide	1000	5000			Uniform
Uranium	Devitrified	0	2.0	0.5	0.3	Beta
	Vitric	0	1.0	0.5	0.3	Beta
	Zeolitic	0	10.0	4.0	1.0	Beta(exp)
	Iron oxide	100	1000			Uniform
Neptunium	Devitrified	0	1.0	0.3	0.3	Beta
	Vitric	0	1.0	0.3	1.0	Beta(exp)
	Zeolitic	0	3.0	0.5	0.25	Beta
	Iron oxide	500	1000			Uniform
Protactinium	Devitrified	0	100			Uniform
	Vitric	0	100			Uniform
	Zeolitic	0	100			Uniform
	Iron oxide	500	1000			Uniform

Table 6-76. Sorption-Coefficient Distributions for Unsaturated Zone Units (Continued)

Element	Rock Type	Min K_d (ml g^{-1})	Max K_d (ml g^{-1})	E[x]	COV*	Distribution type
Carbon	Iron oxide	10	100			Uniform
Actinium, Niobium, Samarium, Thorium, Zirconium: see Americium						
Chlorine, Technetium, Iodine		0	0			

DTN: LA0003AM831341.001 [148751]

NOTE: *Coefficient of variation: COV = $\sigma[x]/E[x]$

To address the influence of adsorption uncertainty on system performance, the matrix adsorption coefficient of each species is pre-sampled for each rock type (based on the listed distribution values in Table 6-76) for each TSPA realization. The sampled data are stored in the file uz_params_multi.sr as shown in Table 6-77. At run time FEHM reads in the sampled values from the uz_params_multi.sr file.

Table 6-77. Sampled Matrix Adsorption Coefficient Data

Element	Corresponding column Position in File uz_params_multi.sr			Used in FEHM File
	Vitric	Zeolitic	Devitrified	uz_params_multi.sr
Am243	44	45	46	
C14	47	48	49	
I129	50	51	52	
Np237	53	54	55	
Columns left for future use	56	57	58	
Pa231, Pu239, Pu242	59	60	61	
Tc99	62	63	64	
Th229	65	66	67	
U233,U234,U235, U236, U238	68	69	70	

Matrix Diffusion Coefficient–It has been show that matrix diffusion combined with matrix adsorption can play an important role in slowing the movement of radionuclides in fractured rocks (CRWMS M&O 2000 [141418], Section 6.1.3). In current TSPA simulations, matrix diffusion coefficients are based on data from the TDMS (DTN: LA0003AM831341.001 [148751]). The distribution of matrix coefficients are categorized into anions and cations and are listed in Table 6-78.

The influence of uncertainty of matrix diffusion coefficient on radionuclide transport is investigated by randomly sampling values based on the given statistical data in Table 6-78 for each anion or cation category for each realization. The sampled data are stored in file uz_params_multi.sr as listed in Table 6-79 and read in at run time by FEHM.

Table 6-78. Matrix Diffusion Coefficient Distribution for Unsaturated Zone

Ion	Min $D(m^2/sec)$	Max $D(m^2/sec)$	Average (m^2/sec)	Std. Deviation (m^2/sec)	Distribution Type	Applied to Radionuclides
anion	0	1.0E-9	3.2E-11	1.0E-11	Beta	C14, I29, and Tc99
cation	0	1.0E-9	1.6E-10	5.0E-11	Beta	Am243, Np237, Pa231, Pu239, Pu240, Th229, U233, U234, U235, U236, and U238

DTN: LA0003JC831362.001 [149557]

Table 6-79. Sampled Matrix Diffusion Coefficient Data

Ion	FEHM File	Listed in Column	Applied to Radionuclides
Anion	uz_params_culti.sr	42	C14, I29, and Tc99
Cation	uz_params_culti.sr	43	Am243, Np237, Pa231, Pu239, Pu240, Th229, U233, U234, U235, U236, and U238

In all TSPA simulations, colloid matrix diffusion (diffusion of a colloid from the fracture to the matrix) is neglected because of lack of data; this is conservative with respect to the dose rates calculated by the TSPA model. Matrix diffusion is neglected for the colloid-facilitated transport of Am^{241}, Am^{243}, Np^{237}, Pu^{238}, Pu^{239}, Pu^{240}, and U^{234}.

Fracture Residual Saturation and Fracture γ Parameter–Fracture residual saturation and fracture γ parameter values are used in FEHM for calculating the fracture spacing of active fractures in FEHM matrix diffusion model (CRWMS M&O 2000 [141418], Section 6.2.1). In all flow models, the fracture residual saturation is fixed at 0.01 (DTN: LB997141233129.001 [104055]). Values of fracture γ parameter vary with infiltration rates in each rock layer. Table 6-80 through Table 6-82 lists the fracture γ parameter values used in TSPA-SR for different infiltration scenarios.

Table 6-80. Fracture γ Parameter for Low Day Infiltration Scenario

Rock Layer	Fracture γ Parameter	Rock Layer	Fracture γ parameter	Used in FEHM File
tcwF1	0.25	ch2Fz	0.12	Ptrk.multrlz and ptrk.multrlz.0100 for low infiltration scenario
tcwF2	0.25	ch3Fz	0.12	
tcwF3	0.25	ch4Fz	0.12	
ptnF1	0.10	ch5Fz	0.12	
ptnF2	0.10	ch6Fz	0.12	
ptnF3	0.10	pp4Fz	0.12	
ptnF4	0.10	pp3Fd	0.43	
ptnF5	0.10	pp2Fd	0.43	Ptrk.multrlz and ptrk.multrlz.0100 for low infiltration scenario
ptnF6	0.10	pp1Fz	0.12	
tswF1	0.60	bf3Fd	0.43	
tswF2	0.23	bf2Fz	0.12	
tswF3	0.23	tr3Fd	0.43	
tswF4	0.23	tr2Fz	0.12	

Table 6-80. Fracture γ Parameter for Low Day Infiltration Scenario (Continued)

Rock Layer	Fracture γ Parameter	Rock Layer	Fracture γ parameter	Used in FEHM File
tswF5	0.23	pcF38	0.00	
tswF6	0.23	pcF39	0.00	
tswF7	0.23	pcF1z	0.00	
tswF8	0.23	pcF2z	0.00	
tswF9	0.23	pcF5z	0.00	
ch1Fv	0.12	pcF6z	0.00	
ch2Fv	0.12	pcF4p	0.00	
ch3Fv	0.12	tcwFf	0.30	
ch4Fv	0.12	ptnFf	0.10	
ch5Fv	0.12	tswFf	0.50	
ch1Fz	0.12	chnFf	0.30	

DTN: LB9971412233129.003 [119940] and LB991091233129.004 [126111]

Table 6-81. Fracture γ Parameter for Medium Infiltration Scenario

Rock Layer	Fracture γ Parameter	Rock Layer	Fracture γ Parameter	Used in FEHM File
tcwF1	0.30	ch2Fz	0.10	ptrk.multrlz0200 for medium infiltration scenario
tcwF2	0.30	ch3Fz	0.10	
tcwF3	0.30	ch4Fz	0.10	
ptnF1	0.09	ch5Fz	0.10	
ptnF2	0.09	ch6Fz	0.10	
ptnF3	0.09	pp4Fz	0.10	
ptnF4	0.09	pp3Fd	0.46	
ptnF5	0.09	pp2Fd	0.46	
ptnF6	0.09	pp1Fz	0.10	
tswF1	0.06	bf3Fd	0.46	
tswF2	0.41	bf2Fz	0.10	
tswF3	0.41	tr3Fd	0.46	
tswF4	0.41	tr2Fz	0.10	
tswF5	0.41	pcF38	0.00	
tswF6	0.41	pcF39	0.00	
tswF7	0.41	pcF1z	0.00	
tswF8	0.41	pcF2z	0.00	
tswF9	0.41	pcF5z	0.00	
ch1Fv	0.13	pcF6z	0.00	ptrk.multrlz0200 for medium infiltration scenario
ch2Fv	0.13	pcF4p	0.00	
ch3Fv	0.13	TcwFf	0.30	
ch4Fv	0.13	PtnFf	0.10	
ch5Fv	0.13	TswFf	0.50	
ch1Fz	0.10	ChnFf	0.30	

DTN: LB997141233129.001 [104055] and LB991091233129.004 [126111]

Table 6-82. Fracture γ Parameter for High Infiltration Scenario

Rock Layer	Fracture γ Parameter	Rock Layer	Fracture γ Parameter	Used in FEHM File
tcwF1	0.31	ch2Fz	0.10	Ptrk.multrlz.0300 for high infiltration scenario
tcwF2	0.31	ch3Fz	0.10	
tcwF3	0.31	ch4Fz	0.10	
ptnF1	0.08	ch5Fz	0.10	
ptnF2	0.08	ch6Fz	0.10	
ptnF3	0.08	pp4Fz	0.10	
ptnF4	0.08	pp3Fd	0.56	
ptnF5	0.08	pp2Fd	0.56	
ptnF6	0.08	pp1Fz	0.10	
tswF1	0.09	bf3Fd	0.56	
tswF2	0.38	bf2Fz	0.10	
tswF3	0.38	tr3Fd	0.56	
tswF4	0.38	tr2Fz	0.10	
tswF5	0.38	pcF38	0.00	
tswF6	0.38	pcF39	0.00	
tswF7	0.38	pcF1z	0.00	
tswF8	0.38	pcF2z	0.00	
tswF9	0.38	pcF5z	0.00	
ch1Fv	0.10	pcF6z	0.00	
ch2Fv	0.10	pcF4p	0.00	
ch3Fv	0.10	tcwFf	0.30	
ch4Fv	0.10	ptnFf	0.10	
ch5Fv	0.10	tswFf	0.50	
ch1Fz	0.10	chnFf	0.30	

DTN: LB997141233129.002 [119933] and LB991091233129.004 [126111]

Fracture Porosity, Fracture Spacing, and Fracture Aperture–Fracture porosity is used in FEHM to calculate the fracture pore volume of the corresponding fracture node block for determining the resident time of radionuclides within each fracture block. Fracture porosity values used in TSPA simulations are based on data from CRWMS M&O 2000 [141418], Table 3 and are listed in Table 6-83. The fracture porosity data are read in by FEHM from data file afm_pch1.dpdp.

Table 6-83. List of Fracture Porosity Values used in TSPA Simulations

Fracture Layer	Fracture Porosity	Fracture Layer	Fracture Porosity	FEHM File
tcwF1	2.8E-2	ch2Fz	4.3E-4	afm_pch1.dpdp. Same data are used in all TSPA simulations
tcwF2	2.0E-2	ch3Fz	4.3E-4	
tcwF3	1.5E-2	ch4Fz	4.3E-4	
ptnF1	1.1E-2	ch5Fz	4.3E-4	
ptnF2	1.2E-2	ch6Fz	1.7E-4	
ptnF3	2.5E-2	pp4Fz	4.3E-4	
ptnF4	1.2E-2	pp3Fd	1.1E-3	
ptnF5	6.2e-3	pp2Fd	1.1E-3	
ptnF6	3.6E-3	pp1Fz	4.3E-4	
tswF1	5.5E-3	bf3Fd	1.1E-3	
tswF2	9.5e-3	bf2Fz	4.3E-4	
tswF3	6.6E-3	tr3Fd	1.1E-3	
tswF4	1.0E-2	tr2Fz	4.3E-4	
tswF5	1.1E-2	pcF38	3.70E-2	
tswF6	1.5E-2	pcF39	1.73E-1	
tswF7	1.5E-2	pcF1z	2.88E-1	
tswF8	1.2E-2	pcF2z	3.31E-1	
tswF9	4.6E-3	pcF5z	3.31E-1	
ch1Fv	6.9E-4	pcF6z	2.66E-1	
ch2Fv	8.9E-4	pcF4p	3.25E-1	
ch3Fv	8.9E-4	Fs_tcwf	4.4E-2	
ch4Fv	8.9E-4	Fs_ptnf	1.6E-2	
ch5Fv	8.9E-4	Fs_tswf	3.6E-2	
ch1Fz	1.7E-4	Fs_chnf	1.6E-3	

DTN: LB990501233129.001 [106787] and SN9912T0581699.003 [146903] (afm_pch1.dpdp for materials beginning with "pc")

In FEHM, the fracture spacing data are combined with fracture residual saturation data and fracture γ parameter values to calculate the fracture spacing of active fractures in the finite spacing matrix diffusion model (CRWMS M&O 2000 [141418], Section 6.2.1). Since radionuclide transport is not expected to be very sensitive to fracture spacing (CRWMS M&O 2000 [141418], Section 6.2.1), mean fracture spacing values are used in all TSPA simulations. For FEHM input, half fracture spacing values are used because FEHM uses the half fracture spacing as the matrix node block length scale. But, when calculating active fracture spacing for the matrix diffusion model FEHM automatically multiplies the half fracture spacing values by 2 to get the fracture spacing values. The half fracture spacing values are listed in Table 6-84. The source data are from DTN: LB990501233129.001 [106787] and the listed values are derived by dividing the fracture spacing values by 2 and rounded to three significant digits. The half fracture spacing values are stored in file afm_pch1.dpdp for FEHM input.

Table 6-84. Fracture Spacing Values used in TSPA Simulations

Fracture Layer	Half Fracture Spacing(m)	Fracture Layer	Half Fracture Spacing(m)	FEHM File
tcwF1	0.543	ch2Fz	3.571	afm_pch1.dpdp. Same data are used in all TSPA simulations
tcwF2	0.262	ch3Fz	3.571	
tcwF3	0.179	ch4Fz	3.571	
ptnF1	0.746	ch5Fz	3.571	
ptnF2	1.087	ch6Fz	12.500	
ptnF3	0.877	pp4Fz	3.571	
ptnF4	1.087	pp3Fd	2.500	
ptnF5	0.962	pp2Fd	2.500	
ptnF6	0.515	pp1Fz	3.571	
tswF1	0.230	bf3Fd	2.500	
tswF2	0.446	bf2Fz	3.571	
tswF3	0.617	tr3Fd	2.500	
tswF4	0.116	tr2Fz	3.571	
tswF5	0.158	pcF38	99.000	
tswF6	0.124	pcF39	99.000	
tswF7	0.124	pcF1z	99.000	
tswF8	0.115	pcF2z	99.000	
tswF9	0.521	pcF5z	99.000	
ch1Fv	5.000	pcF6z	99.000	
ch2Fv	3.571	pcF4p	99.000	
ch3Fv	3.571	Fs_tcwf	0.263	
ch4Fv	3.571	Fs_ptnf	0.926	
ch5Fv	3.571	Fs_tswf	0.294	
ch1Fz	12.500	Fs_chnf	3.846	

NOTE: Based on DTN: LB990501233129.001 [106787]. The listed values are derived by taking the inverse of the fracture frequency and dividing the value by 2. For materials beginning with "pc", the values are from file afm_pch1.dpdp of data package SN9912T0581699.003 [146903].

Fracture half-aperture values are used in FEHM matrix diffusion model for estimating the effect of matrix diffusion on radionuclide transport. In current flow models, fracture half-aperture has a log-normal distribution (see assumption in Section 5.3.6) with geometric mean varies in different rock layers. The average geometric standard deviation for all layers is 1.9 (CRWMS M&O 2000 [141418], Section 6.2.1). Because it is expected that matrix diffusion is sensitive to the fracture aperture (CRWMS M&O 2000 [141418], Section 6.2.1), the fracture aperture values in each layer are sampled using the Latin Hyper-Cubic method (GoldSim) based on the fracture aperture distribution data for each realization. The sampled half-fracture apertures for each layer are documented in FEHM input data file uz_params_multi.sr from column 1 to column 41, respectively, and are read in by FEHM at run time. Table 6-85 lists the fracture aperture distribution data.

Table 6-85. Fracture Aperture Distribution in Each Rock Layer

Rock Layer (column #)	Geometric Mean Fracture Half Aperture (m)	Rock Layer	Geometric Mean Fracture Half Aperture (m)	FEHM File
tcwF1(1)	1.79E-02	ch4Fv(22)	2.07E-03	Data stored in uz_params_multi.sr from column 1 to column 41. Note: The average geometric standard deviation of 1.9 was used in sampling fracture aperture data (CRWMS M&O 2000 [141418], Section 6.2.1.
tcwF2(2)	1.49E-03	ch5Fv(23)	2.07E-03	
tcwF3(3)	3.98E-03	ch1Fz(24)	1.55E-03	
ptnF1(4)	1.10E-02	ch2Fz(25)	1.00E-03	
ptnF2(5)	8.51E-03	ch3Fz(26)	1.00E-03	
ptnF3(6)	1.43E-03	ch4Fz(27)	1.00E-03	
ptnF4(7)	3.53E-02	ch5Fz(28)	1.00E-03	
ptnF5(8)	5.69E-03	ch6Fz(29)	1.55E-03	
ptnF6(9)	1.01E-03	pp4Fz(30)	1.00E-03	
tswF1(10)	1.42E-03	pp3Fd(31)	1.80E-03	
tswF2(11)	2.96E-03	pp2Fd(32)	1.80E-03	
tswF3(12)	1.49E-03	pp1Fz(33)	1.00E-03	
tswF4(13)	7.39E-04	bf3Fd(34)	1.80E-03	
tswF5(14)	1.14E-03	bf2Fz(35)	1.00E-03	
tswF6(15)	1.22E-03	tr3Fd(36)	1.80E-03	
tswF7(16)	1.22E-03	tr2Fz(37)	1.00E-03	
tswF8(17)	9.00E-04	Fs_tcwf(38)	3.38E-03	
tswF9(18)	1.56E-03	Fs_ptnf(39)	1.23E-02	
ch1Fv(19)	2.30E-03	Fs_tswf(40)	4.19E-03	
ch2Fv(20)	2.07E-03	Fs_chnf(41)	3.40E-03	
ch3Fv(21)	2.07E-03			

DTN: SN0005T0581699.005 [151514]

Fracture Surface Retardation Factor–Because few data are available on fracture surface retardation factors no fracture surface retardation is simulated in the TSPA model. In current TSPA simulations, all the fracture surface retardation factors are set to 1.0 (no fracture surface retardation) to be conservative (see Section 5.2). Values of fracture surface retardation factors are included in FEHM input data file ptrk.median, ptrk.mutlrlz, ptrk.mutlrlz.001, ptrk.mutlrlz.002, ptrk.mutlrlz.003 (see Table 6-86).

Table 6-86. Fracture Surface Retardation Factor used in TSPA

Fracture Surface Retardation Factor	Used in FEHM File
1.0	ptrk.median, ptrk.mutlrlz, ptrk.mutlrlz.001, ptrk.mutlrlz.002, and ptrk.mutlrlz.003

Matrix Pore Size Distribution–Matrix pore size distribution combined with colloid size distribution is used in FEHM for determining colloid filtration at the interfaces between matrix units. Each time step, at a matrix unit interface FEHM compares a colloid's size against the sampled pore size of the matrix unit it is entering. If the colloid size is bigger than the pore size, then the colloid can not enter the matrix and is removed from the simulation (permanently filtered). In TSPA simulations the cumulative probabilities for colloid transport between one matrix unit and another are taken from CRWMS M&O 2000 [141418], Section 6.2.5) and are listed in Table 6-87 (only colloid size data beneath the repository level are listed). In FEHM the matrix pore size date are sampled based on colloid size data in Table 6-87 and the sampled data are used in simulating colloid filtration at matrix interfaces.

Table 6-87. Cumulative Probabilities for Colloid Transport at Matrix Interfaces

Units	Colloid Size (nm)							FEHM File
	2000	1000	450	200	100	50	6	
TMN/TSW4	1.00	0.92	0.87	0.81	0.71	0.55	0.31	tsw4.txt
TLL /TSW5	1.00	0.80	0.79	0.70	0.61	0.51	0.19	tsw5.txt
TM2/TSW6	1.00	0.94	0.90	0.82	0.65	0.51	0.21	tsw6.txt
TMN1/TSW7	1.00	0.99	0.99	0.99	0.93	0.68	0.36	tsw7.txt
PV3 /TSW8	1.00	0.98	0.96	0.94	0.90	0.89	0.68	tsw8.txt
PV2/TSW9	1.00	0.72	0.57	0.47	0.39	0.35	0.22	tsw9.txt
BT1a/CH1	1.00	0.91	0.89	0.87	0.85	0.83	0.53	ch1.txt
CHV	1.00	0.58	0.49	0.43	0.39	0.36	0.07	chv.txt
CHZ	1.00	0.79	0.76	0.73	0.68	0.56	0.30	chz.txt
BT/CH6	1.00	0.95	0.94	0.92	0.92	0.85	0.40	ch6.txt
PP1	1.00	0.79	0.68	0.63	0.57	0.48	0.21	pp1.txt
PP2	1.00	0.91	0.86	0.81	0.65	0.53	0.22	pp2.txt
PP3	1.00	0.49	0.34	0.26	0.21	0.16	0.07	pp3.txt
PP4	1.00	0.99	0.99	0.98	0.98	0.96	0.32	pp4.txt
BF2	1.00	0.98	0.97	0.96	0.96	0.83	0.25	bf2.txt
BF3	1.00	0.97	0.94	0.83	0.74	0.66	0.14	bf3.txt

DTN: LA0003MCG12213.002 [147285]

Colloid Size Distribution, Colloid K_c, Colloid R_c, and Colloid Filtration Factor–Colloid size distribution is used by FEHM to get the interpolated colloid size of each colloid particle. The colloid size information is then combined with pore size data to simulate filtration effect at matrix unit interfaces. The colloid size range of 6 nm to 450 nm is based on CRWMS M&O 2000 [147505], DTN: LL000122051021.116 [142973]. However, because a specific distribution was not available, the following distribution (Table 6-88) was chosen (not developed) to be consistent with CRWMS M&O 2000 [147505], DTN: LA0007MCG12213.001 [153251].

Table 6-88. Colloid Size Distribution

Colloid Size (nm)	Cumulative Probability	FEHM File
1	0	Same data used in all TSPA simulations.
6	0.2	ptrk.median
50	0.4	ptrk.multrlz
100	0.6	ptrk.multrlz.0100
200	0.8	ptrk.multrlz.0200
450	1.0	ptrk.multrlz.0300

NOTE: Based on DTN: LL000122051021.116 [142973]

The same colloid size distribution data are used in all TSPA simulations. FEHM ptrk.multrlz, ptrk.multrlz.001, ptrk.multrlz.002, and ptrk.multrlz.003 contain the colloid size input data under the macro "size".

Colloid equilibrium sorption parameter K_c is defined as $K_c = C_{coll}/C_{fluid}$, where C_{coll} is the radionuclide concentration residing on colloids and C_{fluid} is the radionuclide concentration in fluid. Colloid K_c is used in FEHM as an input parameter for calculating the retardation factors for colloid facilitated radionuclide transport in the media.

Radionuclide sorption to colloid can be categorized into reversible and irreversible two categories. When sorption to colloid is treated as an irreversible process, a very large number (1.0E20) is assigned for K_c (see Table 6-89).

Table 6-89. K_c for Irreversible Colloid

Irreversible Colloids	K_c	Used in FEHM File
Am^{241}, Am^{243}, Np^{237}, Pu^{238}, Pu^{239}, Pu^{240}, and U^{234}	1.0E20	ptrk.median, ptrk.multrlz, ptrk.multrlz0100, ptrk.multrlz0200, and ptrk.multrlz0300

For reversible radionuclide sorption to colloid, the K_c values are calculated by multiplying the adsorption coefficient, K_d, of radionuclide to colloid by colloid concentrations in the water. In TSPA simulations, for conservatism, the highest observed or expected colloid concentration of 3.0E-2 mg/l (CRWMS M&O 2000 [125156], Figure 14; DTN: MO0003SPAHLO12.004 [147952]) was used in calculating the K_c values. The adsorption coefficients to colloids vary between species. For waste form colloids, Pu has a geometric mean adsorption coefficient of 1.0E4 ml/g with a log-normal distribution, and Am has a geometric mean of 1.0E5 ml/g also with a log-normal distribution (DTN: MO0004SPAKDS42.005 [148810]). But, to be conservative and consistent with SZ calculations (CRWMS M&O 2000 [147972], Section 6.14), the high geometric mean value of Am was used for all the reversible sorption of radionuclides.

For multiple realizations, the Latin Hyper-Cubic module in GoldSim was used to sample the colloid K_d values so as to reflect the influence of reversible colloid facilitated radionuclide transport on system performance. Table 6-90 lists the adsorption coefficient distribution used for calculating colloid K_c.

Table 6-90. Adsorption Coefficient for Reversible Sorption of Radionuclide to Waste Form Colloids

Radionuclide	Geometric Mean	Distribution
Am, Pa, Pu, and Th	1.0E5 ml/g	log-normal, std. dev.=10

NOTE: Radionuclide adsorption coefficient was based on DTN: MO0004SPAKDS42.005 [148810]

The sampled radionuclide adsorption coefficients were then multiplied by the maximum colloid concentration to calculate the colloid K_c values. The calculated values are then stored in file uz_params_multi.sr and are read in by FEHM at run time. Table 6-91 lists the location of colloid K_c values in FEHM file.

Table 6-91. Colloid K_c Values in FEHM File

Radionuclide	K_c Calculated by	Used in FEHM File
Am and Th	Multiply sampled K_d by the maximum colloid concentration.	uz_params_multi.sr, column 71
Pa and Pu		uz_params_multi.sr, column 72

Colloid retardation factor, R_c, is used in FEHM to study the impact of colloid retardation in the fractured media on radionuclide transport. However, because there are no data available on UZ colloid retardation factor, the effect of colloid retardation on UZ transport is neglected by setting R_c=1, which is conservative.

At the fracture-matrix interface, when a colloid's size is larger than the matrix pore size, this colloid will stay in the fracture. The colloid size exclusion effect in the current FEHM model is simulated with a filtration factor f_c based on the percentage of the pores that are greater than the expected colloid size of 100 nm (CRWMS M&O 2000 [141418], Table 5). Table 6-92 lists the values used in FEHM.

Table 6-92. Filtration Factor used in FEHM for Size Exclusion Calculation

Rock Units	Filtration Factor	FEHM File
TMN (TSW4)	0.29	Used in all TSPA simulations in files: ptrk.multrlz ptrk.multrlz.0100 ptrk.multrlz.0200 ptrk.multrlz.0300
TLL (TSW5)	0.39	
TM2 (TSW6)	0.35	
TMN1 (TSW7)	0.07	
PV3 (TSW8)	0.10	
PV2 (TSW9)	0.61	
BT1a (CH1)	0.15	
CHV	0.61	
CHZ	0.27	
BT (CH6)	0.08	
PP4	0.02	
PP3	0.79	
PP2	0.35	
PP1	0.43	
BF3	0.26	
BF2	0.04	

DTN: LA0003MCG12213.002 [147285]

Radionuclide Half-Life and Daughter Products–FEHM needs the radionuclide half life and daughter products information to simulate the influence of radionuclide decay and ingrowth on system performance. The radionuclide half life and daughter products (CRC 1991 [131202]) for the following species are used in FEHM as input parameters (see Table 6-93).

Table 6-93. Radionuclide Half-Life and Daughter Products used in TSPA

Radionuclide	Half Life (Days)	Daughter Product	Used in FEHM File
Am^{243}	2.6919E+06	Pu^{239}	ptrk.median
C^{14}	2.0874E+06		
I^{129}	6.2093E+09		ptrk.multrlz
Np^{237}	7.8164E+08	U^{233}	
Pa^{231}	1.1871E+07		ptrk.multrlz.0100
Pu^{239}	8.8062E+06	U^{235}	
Pu^{240}	2.3876E+06	U^{236}	ptrk.multrlz.0200
Tc^{99}	7.7798E+07		
Th^{229}	2.8855E+06		ptrk.multrlz.0300
U^{233}	5.8075E+07	Th^{229}	
U^{234}	8.9486E+07		
U^{235}	2.5714E+11		
U^{236}	8.5469E+09	Th^{232}	
U^{238}	1.6290E+12	U^{234}	
Pu^{242}	1.3733E+08	U^{238}	
Th^{230}	2.7540E+07		
Th^{232}	5.1135E+12		
Am^{241}	1.5786E+05		
Pu^{238}	3.2047E+04	U^{234}	
Irreversible Colloid facilitated radionuclide transport			
Pu^{242}	1.3733E+08		
Np^{237}	7.8164E+08		
Pu^{238}	3.2047E+04		
Pu^{239}	8.8062E+06		
Pu^{240}	2.3876E+06		
Am^{241}	1.5786E+05	Np^{237}	
Am^{243}	2.6919E+06	Pu^{239}	

The values listed in the table are based on the CRC 72nd edition half lives (in years) multiplied by 365.25 days.

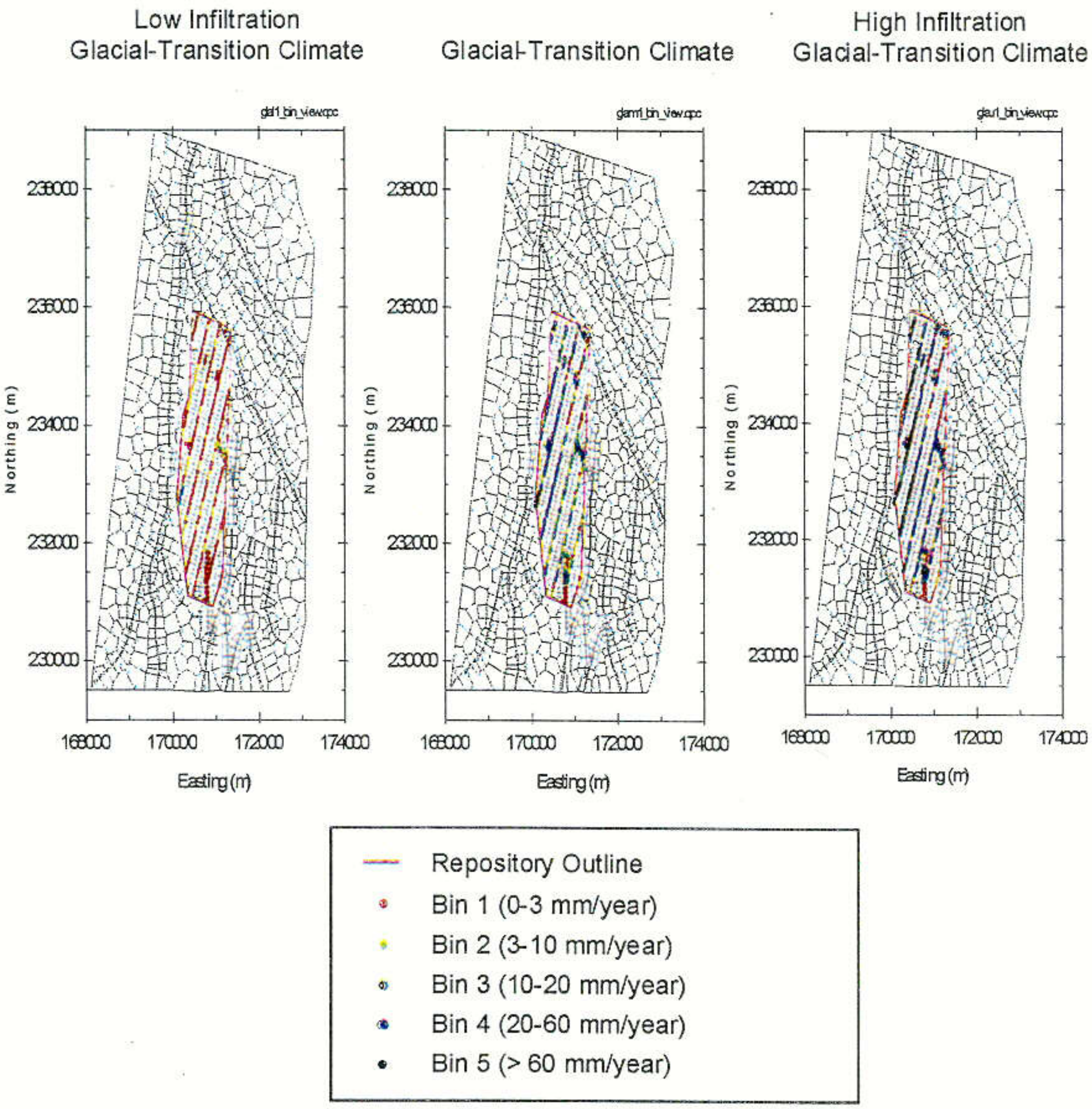


Figure 6-154. Location of Nodes within Each Infiltration Bin in the Repository Footprint

C01

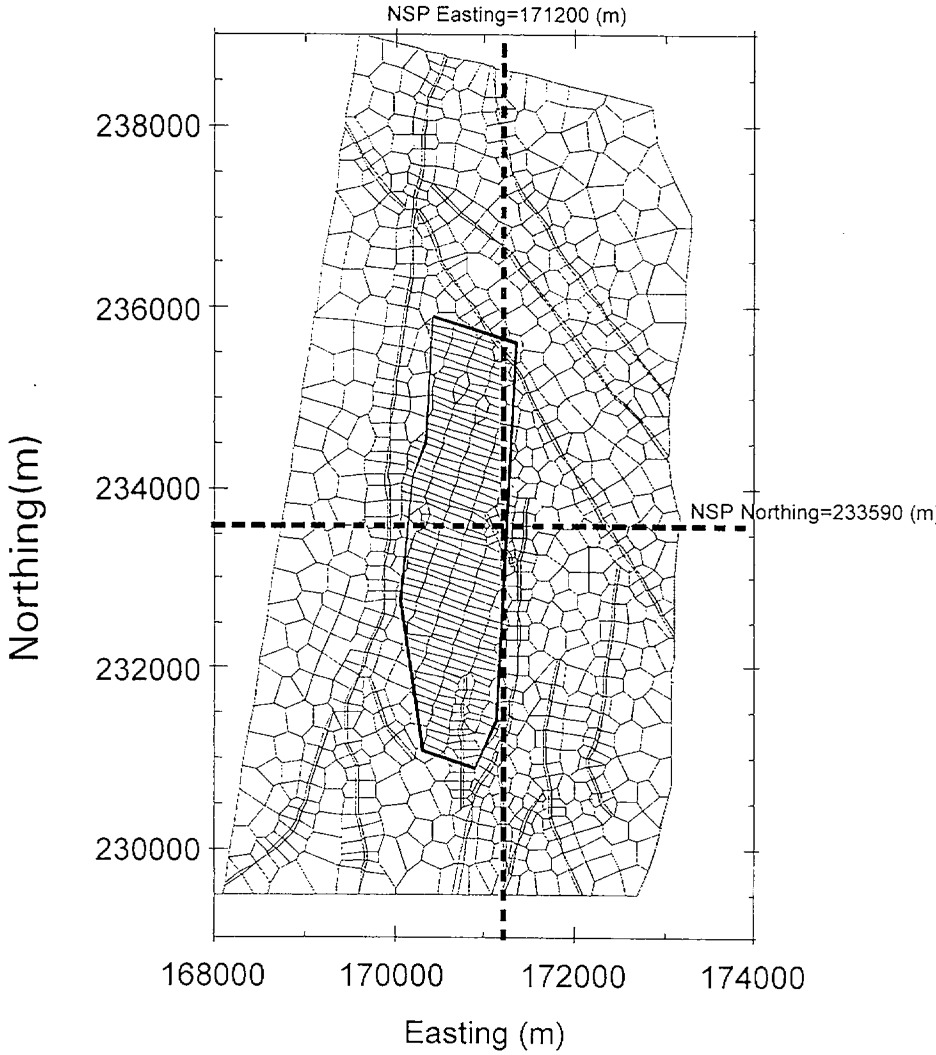


Figure 6-155. Location of Four Regions for UZ/SZ Interface

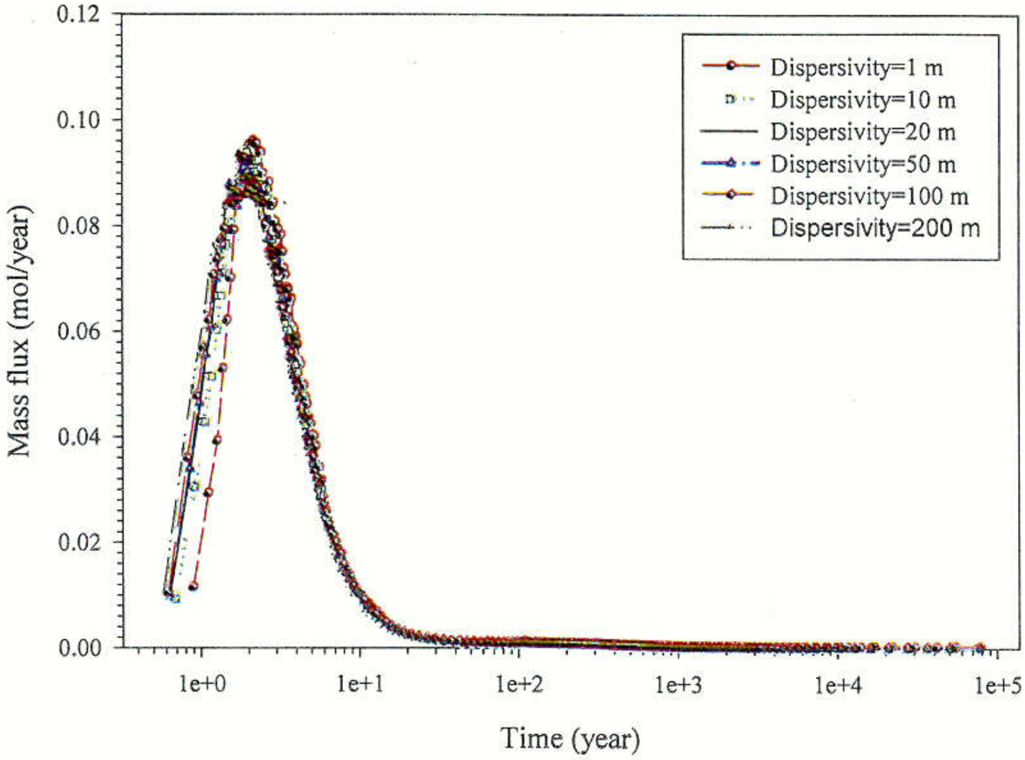


Figure 6-156. Influence of Dispersivity on the Transport of Tc99 in the UZ Under the Glacial Transitional Climate Condition. The Water Table was at 850 m

C02

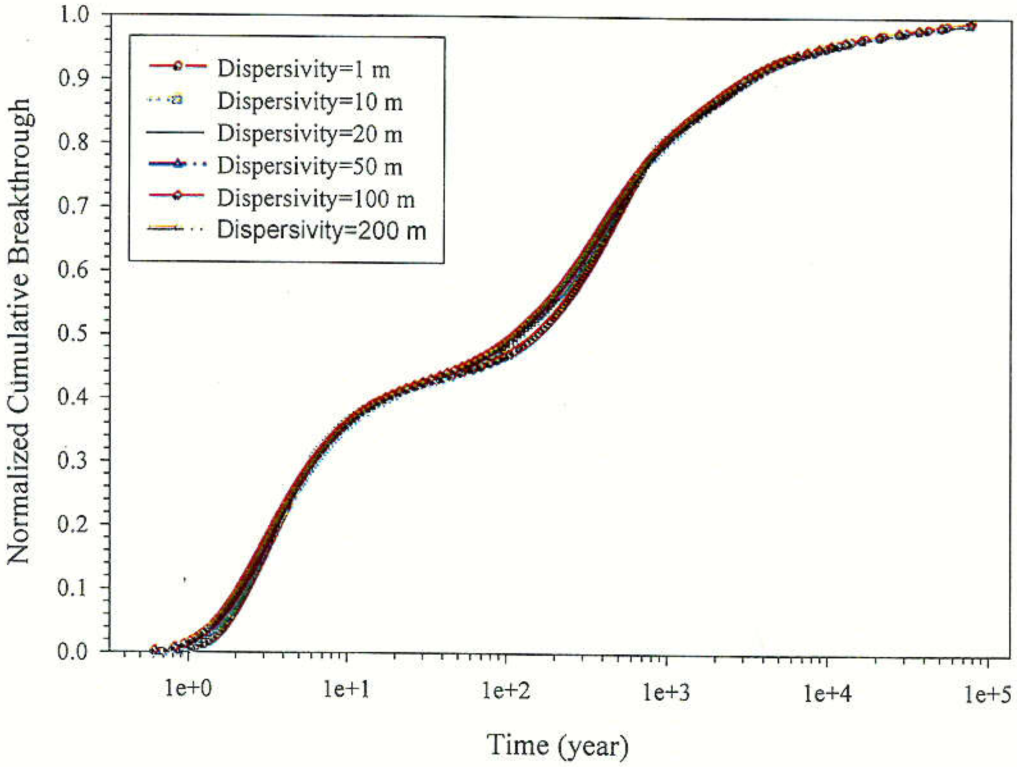


Figure 6-157. Influence of Dispersivity on Normalized Cumulative Breakthrough Curves of Tc-99 Under Glacial Transitional Climate Condition. The Water Table was Set at 850 m

C03

6.3.6.2 Assembling FEHM Input Data Files and FEHM Control File

Once the transport parameters are chosen the FEHM input data files are assembled based on the designated data structure of the FEHM files.

At run time GoldSim initiates a call to FEHM and FEHM reads in names of input and output data files in the default control file, fehmn.files. A sample of the control file is shown below.

```
fm_pchm1.dat       # input data file containing time step and ptrk file information.
fm_pchm1.grid      # numerical grid file
fm_pchm1.zone      # zone file containing property zone information
fm_pchm1.out       # output data file
ff0200.ini         # initial flow field data file
fm_pchm1.fin       # output file contains the final values and simulation time for the run
fm_pchm1.his       # time history data for pressure, temperature, flow and energy output
fm_pchm1.trc       # time history data for solute concentration at specified nodes
fm_pchm1.stor      #  data file containing finite element coefficients calculated by FEHM
fm_pchm1.chk       #  input check file generated by FEHM for parameter checking
None               #  screen output is turned off
                   # end of control file
```

Once all the data files are read in, FEHM starts UZ transport simulations and passes the simulated results back to GoldSim at the end of the simulation. The following section describes in detail the GoldSim-FEHM coupling process.

6.3.6.3 Coupling of FEHM and GoldSim for UZ Transportation Calculations

Radionuclide transport in the UZ is simulated with the FEHM particle tracking transport model and the three-dimensional flow fields. The advantage of using a three-dimensional model rather than an abstracted 1-D model is that complex flow and transport behavior at the mountain scale can be simulated. This is especially true at Yucca Mountain where the UZ flow model has shown lateral flow below the repository.

GoldSim is the repository integration program that can take inputs from external system models and integrate the results to generate system performance measures. The coupling of GoldSim and external system models are realized through GoldSim calling external system model DLLs (Dynamic Linked Libraries). Figure 6-158 shows the execution process of GoldSim within one iteration.

The GoldSim-FEHM coupling is done through the FEHM.DLL. At each time step, GoldSim takes the input from EBS and initiates a call to FEHM.DLL and passes input parameters to FEHM. FEHM then uses the passed parameters to start the UZ transport simulations. At the end of FEHM particle tracking simulations, FEHM passes the output results back to GoldSim as inputs to SZ simulations. Figure 6-159 demonstrates the coupling process between GoldSim and FEHM.

The interface between GoldSim and FEHM are control parameters, input parameters, and output results. At each call to FEHM, GoldSim pass the following control parameters, *method* and *state,* input parameter array *in[]*, and output array *out[]* to FEHM inside subroutine fehmn. The call to subroutine fehm inside FEHM is demonstrated below.

call fehmn(method, state, in, out)

Among the passed parameters, method and state, are used to control what functions FEHM should perform at each call as listed below.

method:

0, initialization. FEHM initialize arrays and corresponding variables.

1, call FEHM subroutine 'computefluxvalues' to calculate water flux through each predefined outflow region. The flux data are passed back to GoldSim through the *out[]* array.

2, return FEHM version number.

3, setup index parameters for use in loading output array *out[]* for particle tracking simulation results.

4, perform radionuclide transport simulation and send the simulation results back to GoldSim through the *out[]* array.

state: set for future use to pass error back to GoldSim

0, current setting, no error is passed back to GoldSim.

Subroutine fehmn then uses the passed control parameters to perform the designated functions.

At each call, the input parameters are passed to FEHM from GoldSim through the *in[]* array, which contains the flow index, transport parameter index, packages failed in each region, and mass release data, etc. The structure of the *in[]* array is shown in Figure 6-160 with detailed descriptions.

Parameters passed in the *in[]* array are:

time, current time passed to FEHM from GoldSim, in years. It is used by FEHM to determine the time step (current time - previous time) for particle tracking simulation.

FlowField_index, an index used by FEHM to select a flow field from a predefined flow field data base for use in the current transport simulation.

M_fine_failed_packages (corresponds to the data element *Fine_Groups*), number of early failed packages in the repository, which is used to simulate the impact of early failed packages on system performance.

transport_paramerter_index, used by FEHM to select a row of transport parameters from pre-generated transport parameter database.

(x,y)_coordinates_of_early_failed_packages, list of (x,y) coordinates of each early failed package, *(x,y)* for *i=1,M_fine_failed_packages*. FEHM uses the coordinates to find the closest grid nodes to those packages and release mass at the located nodes.

N_large (corresponds to the data element *Big_Groups*), number of repository sub-regions. In the TSPA-SR, the whole repository is sub-divided into 5 sub-regions based on surface infiltration values.

list_#_of_failed_packages_in_each_sub_region, list number of failed packages in each sub-region, a total of *N_large* values.

#_of_species, list the total number of radionuclide species simulated in the simulations.

mass_input_flag, a flag indicating mass release during the current time step.

0, no mass release during the current time step. FEHM by pass particle injection step and directly start particle tracking simulations.

1, mass will be released from the repository during the current time step. FEHM first calls subroutine set_mptr to inject mass into the system, then, starts particle tracking simulations.

#_of_input_buffers, the value is (*M_fine_failed_packages+N_large*) representing the number of mass release values for each species. It is used by FEHM to calculate the array index of mass release values for each species in the *in[]* array.

#_of_output_buffers, a value representing number of outflow regions at the water table. It is used by FEHM to calculate number of particles flow out through each outflow region and to load the results into the *out[]* array.

mass_release_for_each_species_during_the_current_time_step, list mass release values (in grams) for each species at the *M_fine_failed_packages* and the *N_large* sub_regions. The values are listed in the following order: start with the first early failed package, fill the *in[]* array with mass release values from the first species to the last species; repeat this process for the rest early failed packages. Once all the *M_fine_early_failed_packages* are done, then, fill the *in[]* array with mass release values from each sub-region starting with the first sub-region. Fill *in[]* array with values from the first species to the last species. Repeat this process until all the sub_regions values are put into the *in[]* array as shown in the program logic below.

for i=1 to #_of_input_buffers (M_fine_failed_packages+N_large){for j=1 to #_of_species{

in[]=mass_release at the specified package or sub-region
}

}

FEHM extracts the mass release values from the *in[]* array and converts the mass into number of particles by using a conversion factor. The particles for each species at each early failed package or sub-regions are injected into the system.

Once FEHM extracts the input parameters, reads in the transport parameters, and injects particles into the system, FEHM starts particle tracking simulations for each radionuclide species. At the end of each particle tracking simulation, FEHM calculates the number of particles flowing out of each designated outflow region at the water table and coverts number of particles back into mass (in grams) by using the inverse of the corresponding conversion factor. At the end of the FEHM simulations, FEHM loads the GoldSim passed *out[]* array and passes the simulation results back to GoldSim.

The *out[]* array is used twice in FEHN. (1) To pass particle tracking simulation results back to GoldSim (*method=4*); (2) To pass FEHM calculated flux values in each outflow region back to GoldSim (*method=1*).

Inside fehmn, after particle tracking simulations (*method=4*), FEHM calls the loadoutarray subroutine to load the particle tracking results into the *out[]* array. The structure of the *out[]* array for passing particle tracking simulation results back to GoldSim is shown in Figure 6-161. There are three group of values are passed back to GoldSim.

average_concentration_in_each_outflow_region, FEHM pack the *out[]* array with calculated average concentrations of each species in each outflow region (there are a total of *#_of_output_buffers* outflow regions). The sequence is show in the program logical below

```
for i=1 to #_of_output_buffers{
for j=1 to #_of_species{
out[]=average_concentration_of_j_species_in_i_region
}
if(is a dual permeability model){
for j=1 to #_of_species
out[]=average_matrix_concentration_of_j_species_in_i_region
}
}
```

maximum_concentration_in_each_outflow_region, the structure is similar to that for the average concentration except that the maximum radionuclide concentrations are loaded into the out[] array.

mass_at_each_outflow_region, the structure is similar to that for the average concentrations except that the total radionuclide mass of each species flow out of each outflow region is loaded into the *out[]* array.

FEHM also use the *out[]* array to pass calculated flux for each outflow region back to GoldSim, which is done through GoldSim calling FEHM with *method=1*. Inside FEHM, subroutine computefluxvalues is called to calculate flux in each outflow region and load the *out[]* array.

The flux values are calculated by summing up node flux values within each outflow region. For the single porosity model, FEHM just loads the *out[]* array with flux values from the first outflow region to the last outflow region. For dual porosity model, FEHM separate fracture flux values from matrix flux values and loads them into the *out[]* array next to each other for each outflow region. The program logic is

```
for i=1 to #_of_output_buffers{
out[]=flux_value_in_i_region
if(is a dual permeability model){
out[]=matrix_flux_value _in_i_region
}
}
```

Once the *out[]* array is loaded, FEHM returns program control back to GoldSim. GoldSim extracts FEHM output results from the *out[]* array and processes the data for use as input in SZ simulations. This coupling process repeats for each GoldSim iteration until simulation is complete.

Implementation

Most of the implementation of the UZ transport for the TSPA-SR model is accomplished by particle tracking using the external pathway *FEHM_External* called as a DLL during run time. Within GoldSim, the only requirements for UZ transport model implementation are to assure that GoldSim and FEHM are coupled correctly and that the transport pathways from the EBS to UZ and from the UZ to the SZ are implemented correctly. This section describes the pathways and inputs associated with the unsaturated zone (see Figure 6-162, Figure 6-163, and Figure 6-164).

At each time step, GoldSim passes time, flow field index, number of failed packages, and mass release from those failed packages to FEHM, then FEHM carries out the UZ simulation and passes results in terms of mass flux back to GoldSim as described in Section 6.3.6.2.

The parameter, *flow_field_index* is calculated within GoldSim as described in Section 6.3.1.2.

The number of failed packages and the amount of mass released from those failed packages are calculated by WEPDEG and GoldSim at run time.

Connections into and out of FEHM are specified to facilitate the mass release into the UZ from the EBS and the mass release out of the UZ at the water table. Five connections are specified from the EBS Bins to the UZ external pathway (see Figure 6-163). As described in the EBS section of this document, the pathways are defined with a water flux of 1e5 m^3/yr (*Collector_Flux*) in order to assure effectively no residence time in these collector cells. Radionuclide mass releases from these connections are converted into particles with an equivalent total mass and spread over the nodes in the UZ model associated with the bin locations. The initial failures (number equal to the value of *Fine_Groups*) are treated as point sources with particles released at a single node. For more details about the implementation of the initial failures for unsaturated zone transport see CRWMS M&O 2000 [141418], Section 6.2.4. The number of repository subregions is defined by *Big_Groups*.

Eight connections out of the UZ external pathway are specified for fracture and matrix release at the water table for each of four UZ-SZ interface regions. The water flux for these connections is arbitrarily set to 1×10^5 m^3/yr (*Collector_Flux*) (see assumption in Section 5.2). Note that the actual flux value is not important because it is not used in the calculation of the mass release of radionuclides for these connections. The outflow connections go to *UZ1OUT, UZ2OUT, UZ3OUT,* and *UZ4OUT*, cells used to connect the UZ and SZ transport models in GoldSim (see Figure 6-164). To assure effectively no residence time in the connector cells, a small volume ($1\text{x}10^{-6}$ m^3) with a large output flux (*Collector_Flux*) to the SZ external pathway is specified (see assumption in Section 5.2). An integrator function (*Cum_Output_UZ*) sums the mass release from *UZ1OUT, UZ2OUT, UZ3OUT,* and *UZ4OUT* to the SZ pathway. *Cum_Output_UZ* is used as the cumulative input for the pipe pathways that are used to calculate the SZ transport for selected radionuclides (see SZ transport section of this document for details).

Within the *Inventory_Boosting* container are calculation elements for adjusting the amounts of certain radionuclides to account for the SZ convolute not treating transport of daughter products (see Section 6.3.4.1, Inventory, for specific details).

Parameters within GoldSim for partition coefficients, fracture spacing, fracture aperture and matrix diffusion are defined in the container *UZ_Input_Parameters* but are not passed to the external pathway. Instead, FEHM obtains these input values from external files that are not directly associated with the GoldSim model. The parameters are defined within GoldSim so that correlations and other statistical tests can be performed between these stochastic parameters and other parameters within the GoldSim model. Note that the values that the parameters take within GoldSim match the realized values used within FEHM from external files. Previous subsections list all of the parameters with sources and explanations.

Partition coefficients (K_ds) are not correlated across UZ units. K_ds are largely determined by the rock matrix and secondarily by the water composition, though information on this topic is limited. The fact that pore water compositions in the UZ are quite variable vertically is another reason correlation of K_ds probably would not be justifiable.

Results and Verification

A test run was carried out to test the coupling between GoldSim, FEHM, and other coupling components. In this test run, FEHM tracked 21 species through the UZ for a period of 1 million years. There were three climate changes during the simulation period. The sequence of climate changes was present-day climate for the first 600 years, monsoonal climate from 600 years to 2,000 years and glacial-transition climate for times greater than 2000 years. The median transport parameter values were used. A maximum of 525,000 particles were used.

In order to get a clear view of the EBS release and UZ mass flux at the outflow boundary, we only selected seven species to plot.

Figure 6-165 and Figure 6-166 show the selected radionuclide mass releases from EBS and the corresponding mass fluxes at the UZ outflow boundary, respectively. It is clear from Figure 6-165 and that the FEHM UZ outflow mass flux curves trace the corresponding EBS release curves very well. These figures also demonstrate that the GoldSim-FEHM coupling worked as designed and that FEHM tracked the transport of radionuclides in the UZ correctly.

The validation and verification of software FEHM is documented in CRWMS M&O 2000 [141418], Section 6.3.

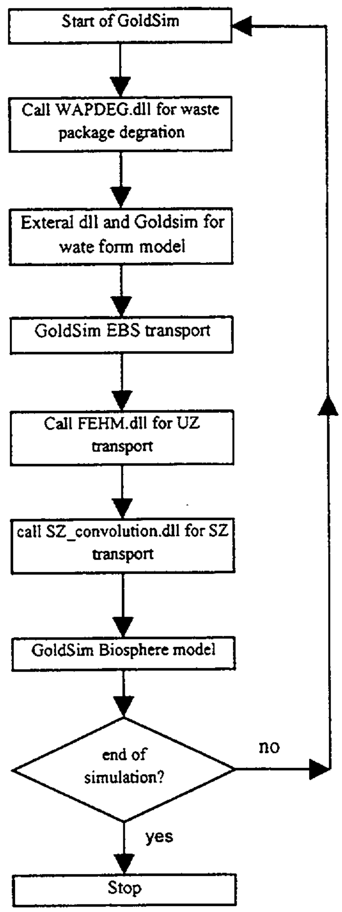


Figure 6-158. Flow Chart of FEHM-GOLDSIM Coupling

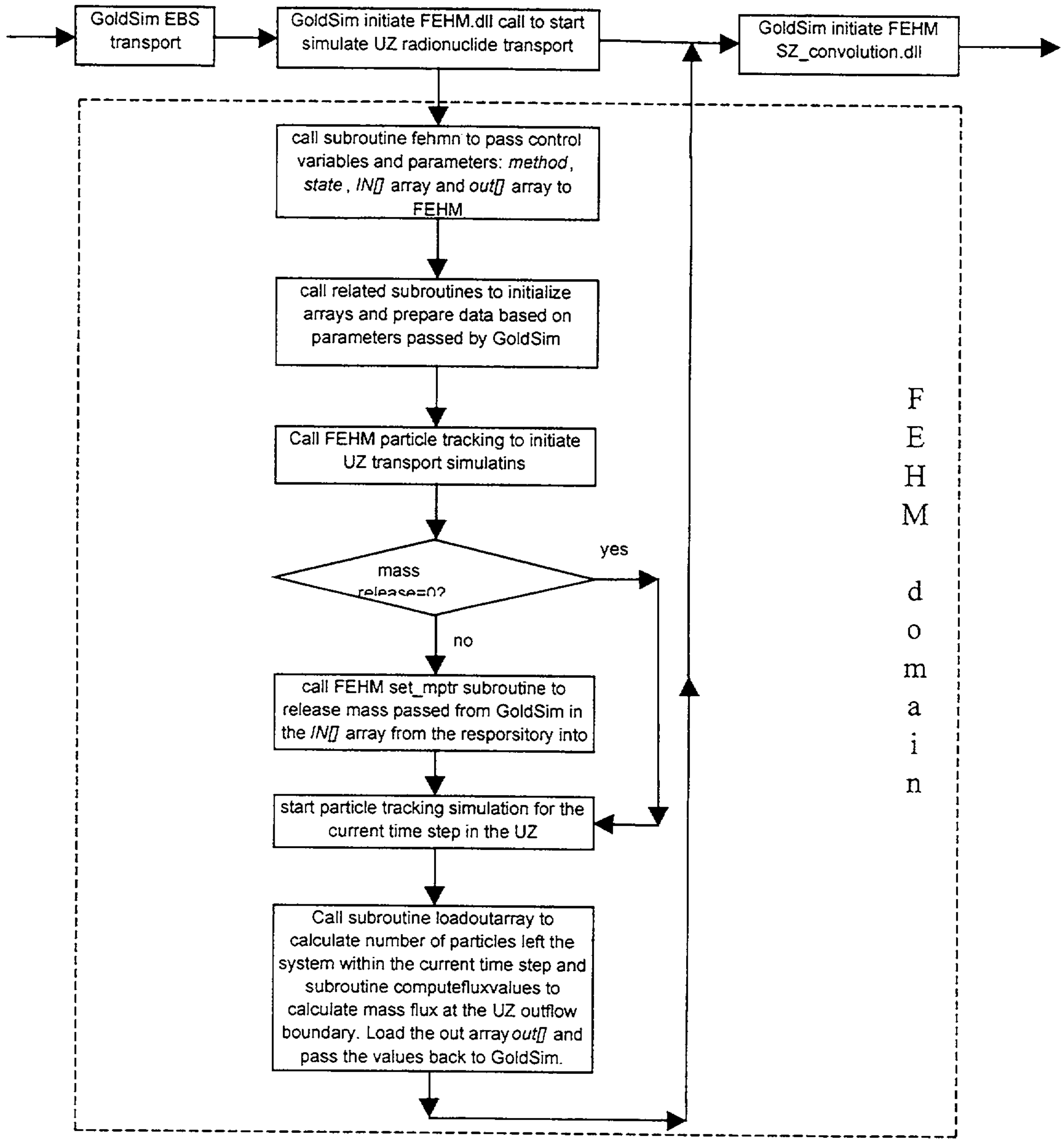


Figure 6-159. Flow Chart of GoldSim-FEHM Coupling and FEHM Simulation Processes

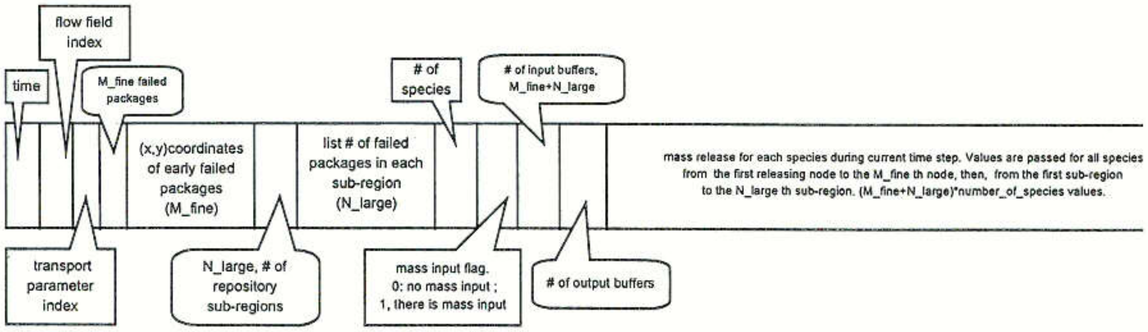


Figure 6-160. Structure of the GoldSim End Array

Average radionuclide concentrations for each radionuclide species in each outflow region.	Maximum radionuclide concentrations for each radionuclide species in each outflow region.	Total radionuclide mass of each species flow out of each outflow region.

Figure 6-161. Structure of out[] Array for Sending FEHM Simulation Results Back to GoldSim

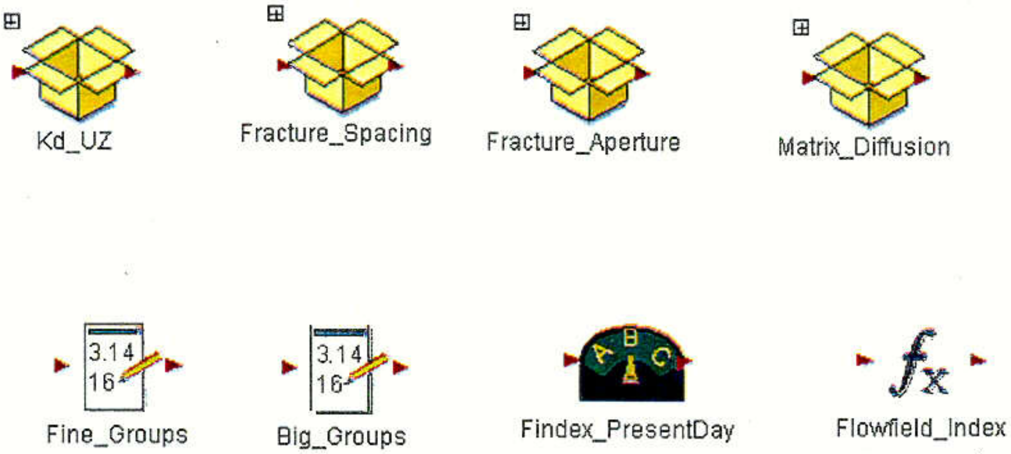


Figure 6-162. Input Parameters for UZ Transport Model

C04

Figure 6-163. Engineered Barrier Connection Cells for the Five Infiltration Bins

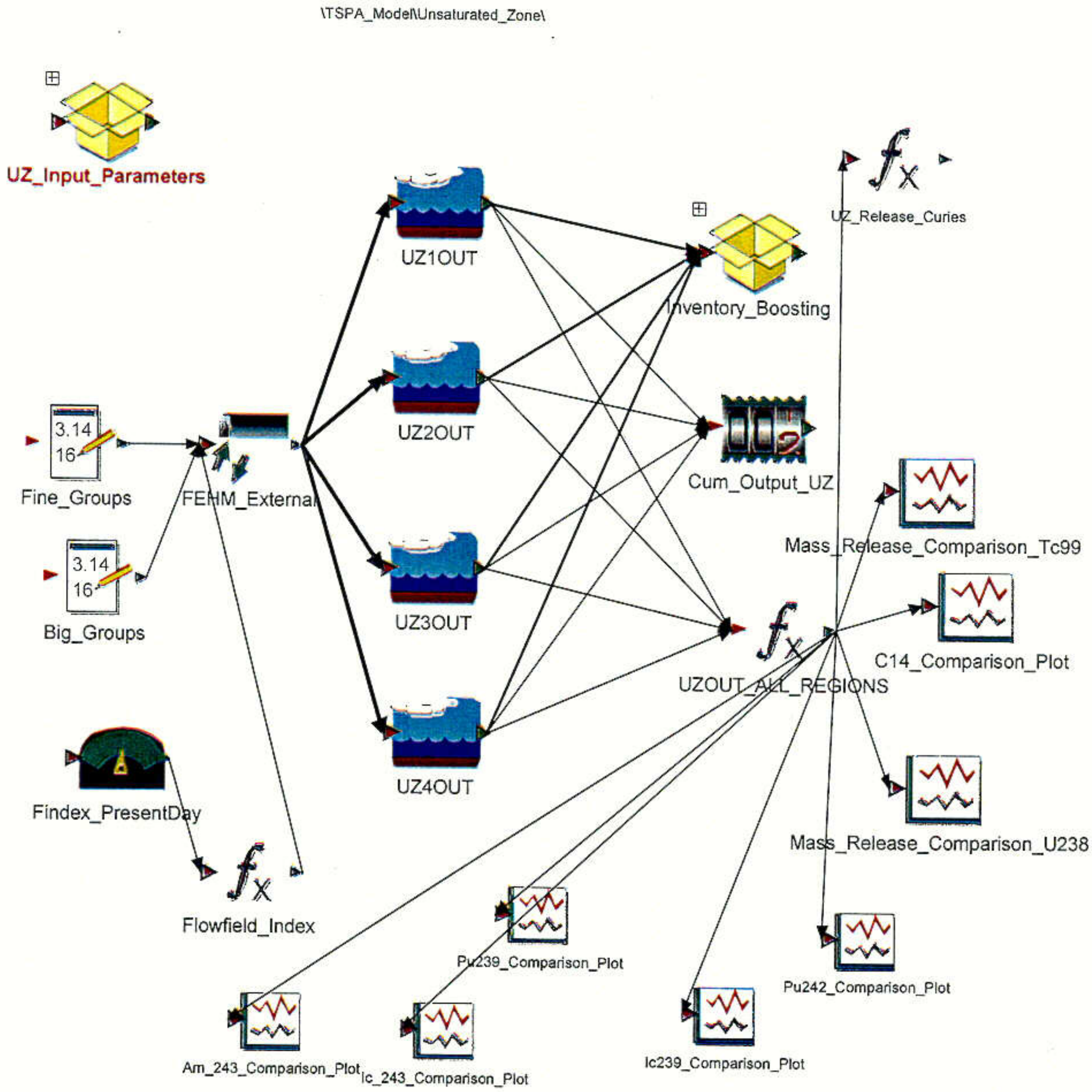


Figure 6-164. Cells and Parameters for Unsaturated Zone Transport

C05

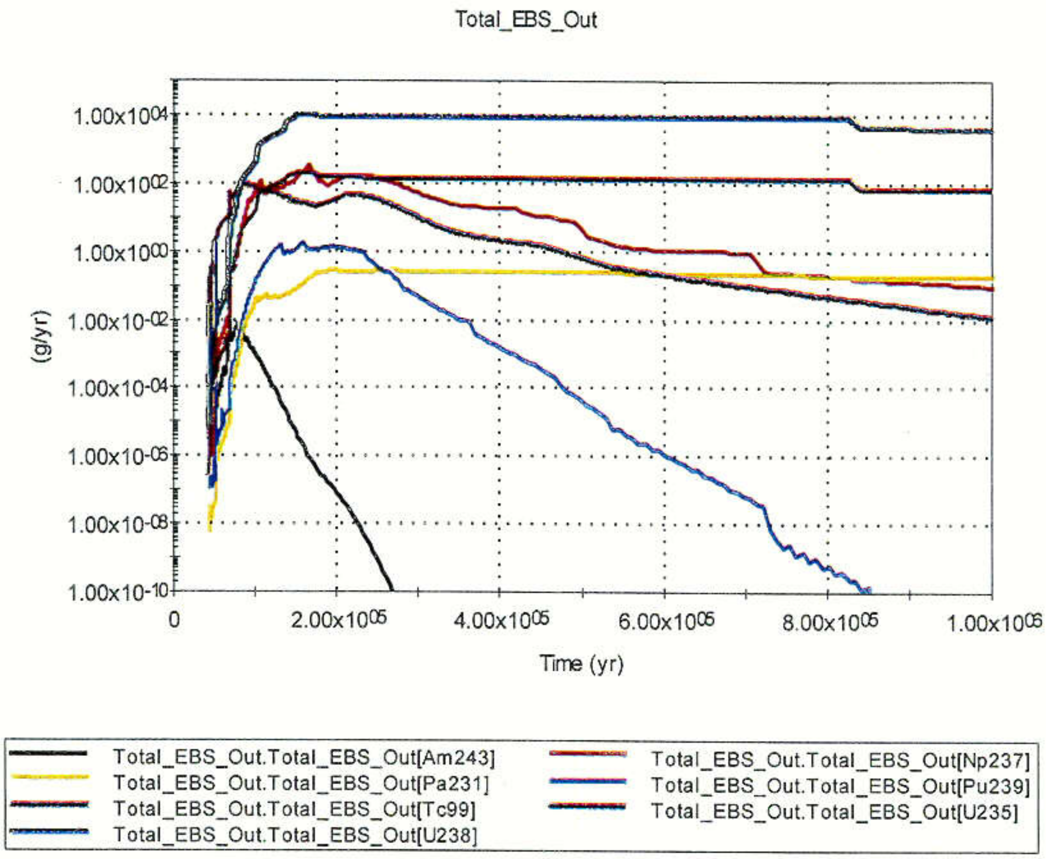


Figure 6-165. Radionuclide EBS Release for Selected Species

C06

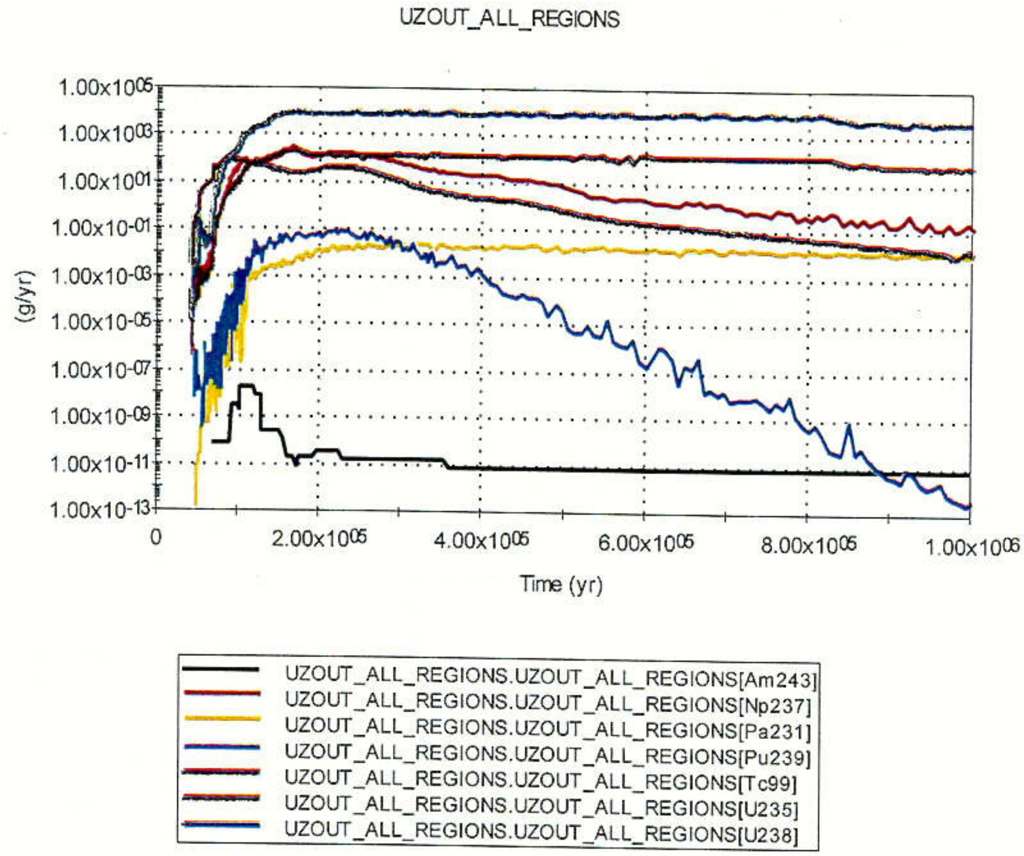


Figure 6-166. Radionuclide Mass Flux at the UZ Outflow Boundary for Selected Species

C07

6.3.7 Saturated Zone Transport

The saturated zone at Yucca Mountain is the region beneath the ground surface where rock pores and fractures are completely saturated with groundwater. The upper boundary of the saturated zone is called the water table. The flow-and-transport component of the TSPA-SR for the saturated zone evaluates the migration of radionuclides from their introduction at the water table below the repository to the release point to the biosphere (Figure 6-167). This component of the analysis receives input from the transport calculations for the unsaturated zone that describe the movement of contaminants in downward percolating groundwater from the repository to the water table (see Section 6.3.6). The input to the saturated zone flow-and-transport calculations is the spatial and temporal distributions of simulated mass flux at the water table. The saturated zone output (mass flux of radionuclides) is used within the biosphere analysis (see Section 6.3.8). The geosphere/biosphere interface is located 20 km (12 miles) from the location of the potential repository (Figure 6-168) (see Biosphere Section 6.3.8). Radionuclides reaching the 20 km boundary in the SZ waters are used in the biosphere component of the model to calculate average radiation dose rates received by the critical group (see Section 6.3.8). For the TSPA-SR model, the mass flux of radionuclides transported through the saturated zone per year is simulated. The saturated zone is a natural barrier responsible for the retardation of radionuclides released from the potential repository to the accessible environment. As such, the saturated zone is a fundamental component of the TSPA model.

6.3.7.1 Saturated Zone Transport Parameters

Overview

The TSPA-SR derives the input parameters for the saturated zone (SZ) transport model from the supporting abstraction, *Input and Results of the Base Case Saturated Zone Flow and Transport Model for TSPA*, (CRWMS M&O 2000 [139440], Section 6.2.1.2, Table 4). A set of radionuclide mass flux breakthrough curves (DTN: SN0004T0501600.004 [149288]) was generated from these input parameters. Additional input parameters were obtained from the AMRs, *In-drift Precipitates/Salts Analysis*, (CRWMS M&O 2000 [146857]) and *Waste Form Colloid-Associated Concentration Limits: Abstraction and Summary* (CRWMS M&O 2000 [125156], Section 4). The SZ abstraction generated 100 sets of curves for four saturated zone regions and eight representative radionuclide groups (CRWMS M&O 2000 [139440], Section 6.2.7). The breakthrough curves were produced by three-dimensional SZ transport simulations. They provide data for unit release of radionuclide mass transported from a point beneath the repository at the water table to the accessible environment. For the nominal scenario TSPA-SR base case the saturated zone breakthrough curves were generated for a distance of 20 km from the potential repository. The radionuclide mass flux curves produced by the three-dimensional model are used within the convolution integral routine (SZ_CONVOLUTE, STN: 10207-2.0-00 [153016]), described later in this section and in Section 3.1.8, to determine the radionuclide mass flux 20 km downgradient of the four SZ source regions.

Additionally, the TSPA-SR model also uses a one-dimensional transport model pipe pathways defined within the GoldSim code (see Section 3.1.1). The 1-D pipe pathways uses the Laplace transform solution for analytical solutions to advection dominated mass transport (Golder

Associates 2000 [143556]). The one-dimensional SZ is discretized into three smaller zones, or "pipe" segments along the 20-km flow path to increase computational accuracy and to incorporate variations and uncertainty in geology. As mentioned previously, the one-dimensional analysis is used for simulations requiring radionuclide chain decay and ingrowth. Although it is not anticipated that the decay products from these radioactive decay chains are significant contributors to the total radiological dose, regulations concerning groundwater protection may require explicit analysis of their concentrations in the water supply of the critical group. Input parameters for the one-dimensional model are used directly from the abstraction. The same radionuclide transport processes that are simulated in the three-dimensional SZ site-scale flow and transport model are analyzed in the one-dimensional "pipe" segments (e.g., sorption, matrix diffusion in fractured units, and colloid-facilitated transport), with the exception of transverse dispersion.

The radionuclide mass flux results of both the pipe model and convolution routine are combined with the estimated annual quantity of groundwater consumed by a hypothetical farming community to determine the average radionuclide concentrations in the water supply, from which a dose to a reasonably maximally exposed individual is calculated.

Inputs to the TSPA Model

The parameters used as input to the three-dimensional model to generate the breakthrough curves and used as input to the one-dimensional and three-dimensional models are listed in Table 6-94 and documented in "Uncertainty Distribution for Stochastic Parameters" CRWMS M&O 2000 [147972]. All of the data shown in Table 6-94 was specifically developed for the TSPA-SR model and, therefore is appropriate for use in the model. A list of parameter descriptions is given in Table 6-95.

Table 6-94. GoldSim Parameters used in TSPA-SR Saturated Zone Transport Model

GoldSim Parameter/ Source	Used In 1-D/3-D	Units	Data Type	Value				DTN
colloid_conc	1-D/3-D	mg/l	Scalar	3e-02				Section 6.3.4.6 *(This reference applies only to the maximum concentration.)
CORAL (CRWMS M&O 2000 [147972]), Table 14	1-D/3-D	None	Log-transformed Cumulative Distribution		Probability Level	Value		SN0004T0571599.005 [151515]
					0	0.01		
					0.1933	0.1273		
					0.3088	0.38086		
					0.5311	1.3958		
					0.7432	2.556		
					0.8244	3.013		
					0.9047	3.5105		

Table 6-94. GoldSim Parameters used in TSPA-SR Saturated Zone Transport Model (Continued)

GoldSim Parameter/ Source	Used In 1-D/3-D	Units	Data Type	Value				DTN
					0.9655	3.9989		
					0.9978	4.8795		
					1	6.3624		
CORVO (CRWMS M&O 2000 [147972]), Table 13	1-D/3-D	None	Log-transformed Cumulative Distribution		Probability Level	Value		SN0004T057159 9.005 [151515]
					0	0.02530 8		
					0.039	0.04139		
					0.08125	0.77815		
					0.2605	2		
					0.7605	2.4472		
					1	2.9031		
DCVO (CRWMS M&O 2000 [147972])	1-D/3-D	m^2/s	Log-transformed Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				-13.0		-10		
FISVO (CRWMS M&O 2000 [147972])	1-D/3-D	m	Log-transformed Normal Distribution	Min	Mean	Max	Std Dev	SN0004T057159 9.005 [151515]
				0	1.29	1e+10	0.43	
FPLAN (CRWMS M&O 2000 [147972])	1-D/3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
FPLAW (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
FPVO (CRWMS M&O 2000 [147972])	1-D/3-D	None	Log-transformed Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				-5.0		-1.0		
GWSPD (CRWMS M&O 2000 [147972])	1-D/3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
HAVO (CRWMS M&O 2000 [147972])	1-D/3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
I_hi_thresh_coll_gw	3-D	mol/l	Scalar	0.05				Section 6.3.4.6

Table 6-94. GoldSim Parameters used in TSPA-SR Saturated Zone Transport Model (Continued)

GoldSim Parameter/ Source	Used In 1-D/3-D	Units	Data Type	Value				DTN
I_lo_thresh_coll_gw	3-D	mol/l	Scalar	0.01				Section 6.3.4.6
KDIAL (CRWMS M&O 2000 [147972])	3-D	ml/g	Uniform Distribution	Min		Max		SN0004T0571599.005 [151515]
				0.32		0.63		
KDNPAL (CRWMS M&O 2000 [147972])	1-D/3-D	ml/g	Beta Distribution	Min	Mean	Max	Std Dev	SN0004T0571599.005 [151515]
				0	18.2	100	18.8	
KDNPVO (CRWMS M&O 2000 [147972])	1-D/3-D	ml/g	Beta Distribution	Min	Mean	Max	Std Dev	SN0004T0571599.005 [151515]
				0	0.5	2.0	0.5	
KDRN10 (CRWMS M&O 2000 [147972])	1-D/3-D	None	Uniform Distribution	Min		Max		SN0004T0571599.005 [151515]
				0.0		100		
KDRN9 (CRWMS M&O 2000 [147972])	1-D/3-D	None	Uniform Distribution	Min		Max		SN0004T0571599.005 [151515]
				0.0		50		
KDTCAL (CRWMS M&O 2000 [147972])	1-D/3-D	ml/g	Uniform Distribution	Min		Max		SN0004T0571599.005 [151515]
				0.27		0.62		
KDUAL (CRWMS M&O 2000 [147972])	1-D/3-D	ml/g	Uniform Distribution	Min		Max		SN0004T0571599.005 [151515]
				0		8		
KDUVO (CRWMS M&O 2000 [147972])	1-D/3-D	ml/g	Uniform Distribution	Min		Max		SN0004T0571599.005 [151515]
				0		4		
LDISP (CRWMS M&O 2000 [147972])	1-D/3-D	m	Log-transformed Normal Distribution	Min	Mean	Max	Std Dev	SN0004T0571599.005 [151515]
				0	2.0	1e+10	0.75	
Mcoll_gw_max	3-D	mg/l	Scalar	3e-2				Section 6.3.4.6
Mcoll_gw_min	3-D	mg/l	Scalar	3e-6				Section 6.3.4.6
NVF19 (CRWMS M&O 2000 [147972])	1-D/3-D	None	Truncated Normal Distribution	Min	Mean	Max	Std Dev	SN0004T0571599.005 [151515]
				0	0.18	1e+10	0.051	

Table 6-94. GoldSim Parameters used in TSPA-SR Saturated Zone Transport Model (Continued)

GoldSim Parameter/ Source	Used In 1-D/3-D	Units	Data Type	Value				DTN
NVF7 (CRWMS M&O 2000 [147972])	3-D	None	Truncated Normal Distribution	Min	Mean	Max	Std Dev	SN0004T057159 9.005 [151515]
				0	0.18	1e+10	0.051	
SRC1X (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
SRC1Y (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
SRC2X (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
SRC2Y (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
SRC3X (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
SRC3Y (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
SRC4X (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
SRC4Y (CRWMS M&O 2000 [147972])	3-D	None	Uniform Distribution	Min		Max		SN0004T057159 9.005 [151515]
				0.0		1.0		
Alluvium Density (CRWMS M&O 2000 [147972])	1-D	kg/m^3	Scalar	1270				SN0004T057159 9.005 [151515]
Factor1 (CRWMS M&O 2000 [147972])	1-D	None	Scalar	3.9				SN0004T057159 9.005 [151515]
Volcanic_Matrix_ Porosity (CRWMS M&O 2000 [147972])	1-D	None	Scalar	0.19				SN0004T057159 9.005 [151515]
Volcanic_Density (CRWMS M&O 2000 [139440]	1-D	kg/m^3	Scalar	1940				SN0004T057159 9.005 [151515]

Table 6-94. GoldSim Parameters used in TSPA-SR Saturated Zone Transport Model (Continued)

GoldSim Parameter/ Source	Used In 1-D/3-D	Units	Data Type	Value				DTN
Coating_Porosity (CRWMS M&O 2000 [139440]	1-D	None	Scalar	0.01				SN0004T057159 9.005 [151515]
Coating_ Thickness (CRWMS M&O 2000 [139440]	1-D	m	Scalar	0.00 01				SN0004T057159 9.005 [151515]

Table 6-95. GoldSim Parameter Descriptions for both 1-D and 3-D SZ Transport Models

Goldsim Parameter	Parameter Description
Colloid_conc	Expected mass of groundwater colloids per unit volume or mass of water
CORAL	Retardation factor for colloids in the alluvium units for the irreversible sorption model of colloid-facilitated transport
CORVO	Retardation factor for colloids in the fractured volcanic units for the irreversible sorption model of colloid-facilitated transport
DCVO	Effective diffusion coefficient in the fractured volcanic hydrogeologic units
FISVO	Flowing interval spacing in the fractured volcanic hydrogeologic units
FPLAN	Parameter to determine the northern boundary of the alluvial uncertainty zone
FPLAW	Parameter to determine the western boundary of the alluvial uncertainty zone
FPVO	Flowing interval porosity in the fractured volcanic hydrogeologic units
GWSPD	Parameter determining the groundwater flux case
HAVO	Parameter determining horizontal anisotropy case
I_hi_thresh_coll_g w	Ionic strength above which groundwater colloids are unstable
I_lo_thresh_coll_g w	Ionic strength below which groundwater colloids are unstable
Ionic_Strength_S Z	SZ Water Ionic Strength
KDIAL	Sorption coefficient for iodine in alluvium units
KDNPAL	Sorption coefficient for neptunium in alluvium units
KDNPVO	Sorption coefficient for neptunium in fractured volcanic hydrogeologic units
KDRN10	Sorption coefficient of strongly sorbing radionuclides for the reversible sorption model of colloid-facilitated transport
KDRN9	Sorption coefficient of moderately sorbing radionuclides for the reversible sorption model of colloid-facilitated transport
KDTCAL	Sorption coefficient for technetium in alluvium units
KDUAL	Sorption coefficient for uranium in alluvium units
KDUVO	Sorption coefficient for uranium in fractured volcanic units
LDISP	Longitudinal dispersivity
K_c-Am-Gw-Colloid	K_c parameter for equilibrium colloid-facilitated radionuclide transport
NVF19	Effective porosity of the valley fill hydrogeologic unit and the alluvial uncertainty zone

Table 6-95. GoldSim Parameter Descriptions for both 1-D and 3-D SZ Transport Models (Continued)

GoldSim Parameter	Parameter Description
NVF7	Effective porosity of the undifferentiated valley fill hydrogeologic unit
SRC1X	Parameter defining the east-west location of the radionuclide source in source region 1
SRC1Y	Parameter defining the north-south location of the radionuclide source in source region 1
SRC2X	Parameter defining the east-west location of the radionuclide source in source region 2
SRC2Y	Parameter defining the north-south location of the radionuclide source in source region 2
SRC3X	Parameter defining the east-west location of the radionuclide source in source region 3
SRC3Y	Parameter defining the north-south location of the radionuclide source in source region 3
SRC4X	Parameter defining the east-west location of the radionuclide source in source region 4
SRC4Y	Parameter defining the north-south location of the radionuclide source in source region 4
Alluvium Density	Alluvium bulk density
Factor1	Climate factor
Volcanic_Matrix_ Porosity	Porosity of volcanic matrix
Volcanic_Density	Density of volcanic matrix
Coating_Porosity	Porosity of coating on fracture surface
Coating_ Thickness	Thickness of coating on fracture surface

NOTE: CRWMS M&O 2000 [139440] Section 6.2.1.2 and Section 6.5.

Implementation

Two separate radionuclide SZ transport models were developed for the TSPA-SR model, a pipe model and a convolution integral solution (see Figure 6-169). The pipe SZ model was developed for direct implementation with the GoldSim code and is used for simulation of radionuclide decay and ingrowth in four decay chains in the TSPA-SR model. Although it is not anticipated that the decay products from these radioactive decay chains are significant contributors to the total radiological dose, regulations concerning groundwater protection may require explicit analysis of their concentrations in the water supply of the hypothetically exposed community. The results of the pipe transport modeling are only used for the daughter radionuclides. The model input parameters produced in the supporting AMR (CRWMS M&O 2000 [139440], Section 6.2.1.2, Table 4) are used as input parameters for both the pipe model and the convolution integral model within the TSPA-SR model (see Figure 6-170). Mass released at the UZ-SZ interface (see Section 6.3.6 UZ Transport), is passed to the SZ convolution through an external pathway. The four UZ outputs (see Section 6.3.6) correspond to the four saturated zone regions modeled for the SZ convolution. The same mass that is passed through a GoldSim external pathway to the SZ convolution, is passed through the pipe model via an accumulator (Cum_Output_UZ), that sums the total release from the four UZ out cells. The pipe model only uses one region for the entire SZ to approximate the SZ transport rather than four as the convolution integral model does. Of the 26 radionuclides transported through the UZ model (see Section 6.3.6), 13 are transported through the SZ convolution integral model, while 6 are taken from the pipe model for a total of 19 radionuclides that are calculated contributors to the total dose at the accessible environment (see Biosphere Dose Section 6.3.8.1).

Saturated Zone Pipe Model

The one-dimensional model is set up using the Pipe Pathway component of the Contaminant Transport Module in the GoldSim code (Golder Associates 2000 [143556]). The pipe component is able to simulate advection, longitudinal dispersion, retardation, decay and ingrowth, and matrix diffusion (Golder Associates 2000 [143556]). The 20-km flow path is discretized into three smaller zones called pipes (See *pipe_5km*, *pipe_12km*, and *pipe_20km* in Figure 6-169) that are used to represent the mass transport paths in the saturated zone. The pipes are not actually modeled at only 5 km, 12 km, and 20 km but are modeled at varying distances from the repository along the flow path and radionuclide mass is calculated at each distance. A mass flux loading, (*Cum_Output_UZ*), at the beginning of the first pipe, *pipe_5km*, is the source of the radionuclides that were transported along the connected pipes. Each pipe represents a one-dimensional mass transport model with a unique and uniform set of flow characteristics throughout its length. The *pipe_5km* segment is the transport pathway from the repository to a 5 km distance away from the repository boundary. The *pipe_12km* segment is the transport pathway from the *pipe_5km* segment distance to the boundary between the fractured volcanic hydrogeologic units and the alluvium unit. The *pipe_20km* segment is the transport pathway from the boundary between the volcanic units and the alluvium unit to the 20-km distance. Hydraulic properties are generally grouped into two categories: fractured volcanic unit for the pipe pathways pipe_5km and pipe_12km, and porous medium alluvium unit for the pipe pathways pipe_20km. The properties of the alluvium and volcanic fractures are in the Alluvium_Properties and Volcanic_Properties containers, respectively (see Figure 6-173 and Figure 6-174).

The input parameters correspond to those in the three-dimensional model and are generated by the parameter sampling module in the GoldSim code. The sorption coefficients of Np^{237}, U^{233}, U^{234}, U^{235}, U^{236}, and U^{238} (KDNPVO and KDUVO) are used directly in the matrix of the fractured volcanic units. There is no sorption within the fracture for these radionuclides. The sorption coefficients in the alluvium (KDNPAL and KDUAL) for these radionuclides are modified by a factor of the effective alluvium porosity (NVF19) to maintain the appropriate retardation factor (CRWMS M&O 2000 [139440], Section 6.5.1).

Values of specific discharge for segments represented by pipe pathways in the one-dimensional radionuclide transport model vary along the flow path from the repository. The uncertainty of the flow path length through the alluvium is represented by the stochastic parameter, FPLAN. Specific discharge, represented by ratio of the volumetric outflow rate to the cross-sectional area of each pipe pathway (CRWMS M&O 2000 [139440], Section 6.5.1), is determined by horizontal anisotropy (HAVO) and groundwater flux (GWSPD) (CRWMS M&O 2000 [139440], Section 6.5.1). Specific discharge (*Specific_Discharge5km* and *Specific_Discharge20km*) is used in the model to the calculate geometry's of *pipe_12km*, and *pipe_20km*. Fracture_Area and Fracture_Perimeter define the geometry of *pipe_5km*. Considerations for the glacial-transition climate state are incorporated to account for changes in groundwater flux due to climate. This is implemented by scaling *Discharge_5km* by *Factor* as shown in Figure 6-175. This scaling operation takes place in the Flow_Properties container (see Figure 6-171).

There is no matrix diffusion in fractured media for irreversible colloids. Consequently, there is no sorption in the matrix for irreversible colloids. This is simulated by specifying an arbitrarily small value of available matrix porosity and zero sorption coefficients for these species in the volcanic matrix. The retardation in fractures for irreversible colloids (CORVO) is accounted for as sorption onto coating on the fracture surface is using the fracture coating option in the GoldSim code (CRWMS M&O 2000 [139440], Section 6.5.1).

The parameters SRC4X, SRC4Y, SRC3X, SRC3Y, SRC2X, SRC2Y, SRC1X, and SRC1Y are used to linearly scale the x and y location within the rectangular area of each source region in the three-dimensional SZ flow and transport model (CRWMS M&O 2000 [139440], Section 6.2.1.3). The sorption coefficient in the alluvium for irreversible colloids is calculated as a function of the alluvium bulk density (Alluvium_Density), alluvium porosity (NVF19), and retardation factor (CORAL) in the alluvium (CRWMS M&O 2000 [139440], Section 6.5.1).

Radionuclide sorption for the reversible colloids is effected by the reduced availability of the radionuclide in the aqueous phase of the fractures. This process is simulated by reducing the diffusion coefficient in the matrix (CRWMS M&O 2000 [139440], Section 6.5.1).

There is no sorption in the volcanic fractures for reversible colloids. In the alluvium, the sorption coefficient for reversible colloids is calculated as a function of the original sorption coefficient, the effective porosity of the alluvium, and the radionuclide affinity for attachment to colloids (K_c) (CRWMS M&O 2000 [139440], Section 6.5.1).

The diffusion coefficient in the pipe model is defined by the tortuosity since the reference diffusivity may be used by other processes in the TSPA model. It is assumed that reference diffusivity is an arbitrary value of 1 m^2/s and the tortuosity is assigned a value that results in the correct effective diffusion coefficient (CRWMS M&O 2000 [139440], Section 6.5.1).

SZ Convolution Integral

The SZ convolution integral method is simulated during the TSPA-SR model run via an external dll (SZ_CONVOLUTE, STN: 10207-2.0-00 [153016]) linked to the GoldSim code through an external pathway element. The mass released from the UZ component model (see Section 6.3.6) is linked from the UZ Out cell pathways for regions 1 through 4, to the *SZ_External* Pathway. All the mass that exits the UZ transport code (FEHMN) is passed to the SZ convolution integral component model. The *SZ_External* element in the TSPA model, passes the mass release in (g/yr) from the UZ (per region), the time step, the climate state, and *SZ_Index* parameter to the external dll. Additionally, the parameter Cum_Inven_Boost is passed to the convolution model via the Cumulative Input field within the External Pathway element (*SZ_External*). The mass released from the UZ is increased for species where parent radionuclides decay to daughter species tracked through the SZ model. As the convolution integral method does not account for ingrowth from parent species to daughter species, the daughter's mass is 'boosted' or increased by calculation of simple decay of the parent mass (released from the UZ) to the daughter over a time interval equal to the simulation time (etime) minus the total simulation time (e.g., 100,000 or 1,000,000 years). For example, Uranium 234 is the daughter of Uranium 238, the parameter *UZ1_Boost* (vector by species) calculates the amount of mass of U^{234} that would be generated from the mass of U^{238} if it were decayed over the remainder of the simulation as follows: (UZ1OUT.Water_to_SZ_External[U238] * (234 / 238) * (1 - exp (-Decay_Rate.Decay_Rate [U238] * (Run_Time-ETime)))). Therefore the daughter mass is increased equal to the amount that would eventually by created through ingrowth over the interval of time from the current time step until the end of the model simulation. The mass of the parent remains constant and equal to the mass released from the UZ over each time step.

For the convolution, pre-generated saturated zone breakthrough curves are necessary for the model to simulate the mass transport through the SZ. The TSPA-SR model parameter SZ_Index is used to pass the appropriate index number to the SZ dll that is used to select the correct breakthrough curve. The index number is fixed to a value of '1' for a median value simulation, and varies with realization number for a probabilistic simulation. The saturated zone breakthrough curves are supplied from FEHMN 3-D saturated zone simulations (see CRWMS M&O 2000 [139440]), and are referenced from DTN: SN0004T0501600.004 [149288]. There are two sets of breakthrough curves, one set for a median value simulation (with 1 breakthrough curve for each species) and one set (with 100 breakthrough curves for each species) for a probabilistic simulation. During the simulation the SZ convolution takes the data input (mass release from the UZ per region in g/yr, *etime*, *climate_state*, and *sz_index*), the dll applies the appropriate breakthrough curve for each radionuclide (see Figure 6-169 and text, SZ_CONVOLUTE VERSION 2.0 Users Manual CRWMS M&O 2000 [153016]), applies the correct dilution factor for the current climate state, and returns the mass release per region per radionuclide species over the time step interval. The SZ convolution input file, szconvolute2.dat, holds the information concering which breakthrough curve to use for each radionuclide, the species' half lives, the dilution factors, duration of the model run, climate switches, and total number of radionuclides that will be passed to the convolution model. The szconvolute2.dat file is described in detail in the users manual for SZ_CONVOLUTE 2.0 (Users Manual CRWMS M&O 2000 [153016]). Table 6-96 lists the radionuclides in the order they are passed to the SZ model (radionuclide species list) and the appropriate saturated zone breakthrough curve used for each. (Note there are only eight specific breakthrough curves as they vary by element or type (e.g., colloid or Kc species) rather than one for each individual isotope.). For a detailed discussion of the convolution method and SZ_CONVOLUTE 2.0 dll, please refer to the supporting documentation: *Input and Results of the Base Case Saturated Zone Flow and Transport Model for TSPA*, (CRWMS M&O 2000 [139440]).

Table 6-96. SZ Convolution Model Breakthrough Curves for each Radionuclide in the TSPA-SR Model

Species	SZ Curve Number
Am^{241}	SZ_03
C^{14}	SZ_01
I^{129}	SZ_02
Ic^{242}	SZ_05
Ic^{237}	SZ_05
Ic^{238}	SZ_05
Ic^{239}	SZ_05
Ic^{240}	SZ_05
Ic^{241}	SZ_05
Ic^{243}	SZ_05
Np^{237}	SZ_04
Pa^{231}	0

Table 6-96. SZ Convolution Model Breakthrough Curves for each Radionuclide in the TSPA-SR Model (Continued)

Species	SZ Curve Number
Pu^{239}	SZ_03
Pu^{240}	SZ_03
Tc^{99}	SZ_06
Th^{229}	0
U^{233}	0
U^{234}	SZ_07
U^{235}	0
U^{236}	SZ_07
U^{238}	SZ_07
Pu^{242}	SZ_03
Th^{230}	0
Th^{232}	0
Am^{241}	SZ_03
Pu^{238}	SZ_03
Ac^{227}	0
Cs^{137}	0
Pb^{210}	0
Ra^{226}	0
Ra^{228}	0
Sr^{90}	0
U^{232}	0
Col	0

NOTE: Where SZ Curve Number = 0; species is not tracked through the SZ 3-D model

Table 6-97. Model.Median Value Saturated Zone Breakthrough Curves

Type	Date	Time	Size (bytes)	File Name	Source
Median	3/8/2000	03:53p	98,153	SZ_01_01.0000.z	SN0004T0501600.004 [149288].
Median	3/8/2000	03:55p	98,153	SZ_01_02.0000.z	SN0004T0501600.004 [149288].
Median	3/8/2000	03:55p	98,153	SZ_01_03.0000.z	SN0004T0501600.004 [149288].
Median	3/8/2000	03:56p	98,153	SZ_01_04.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	09:33p	98,153	SZ_02_01.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	09:33p	98,153	SZ_02_02.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	09:34p	98,153	SZ_02_03.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	09:34p	98,153	SZ_02_04.0000.z	SN0004T0501600.004 [149288]

Table 6-97. Model.Median Value Saturated Zone Breakthrough Curves (Continued)

Type	Date	Time	Size (bytes)	File Name	Source
Median	3/8/2000	03:57p	196,153	SZ_03_01.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:57p	196,153	SZ_03_02.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:58p	196,153	SZ_03_03.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:58p	196,153	SZ_03_04.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:58p	245,153	SZ_04_01.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:58p	245,153	SZ_04_02.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:58p	245,153	SZ_04_03.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:58p	245,153	SZ_04_04.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:59p	98,153	SZ_05_01.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	03:59p	98,153	SZ_05_02.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	04:00p	98,153	SZ_05_03.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	04:00p	98,153	SZ_05_04.0000.z	SN0004T0501600.004 [149288]
Median	3/15/2000	09:49a	245,153	SZ_06_01.0000.z	SN0004T0501600.004 [149288]
Median	3/15/2000	09:49a	245,153	SZ_06_02.0000.z	SN0004T0501600.004 [149288]
Median	3/15/2000	09:49a	245,153	SZ_06_03.0000.z	SN0004T0501600.004 [149288]
Median	3/15/2000	09:49a	245,153	SZ_06_04.0000.z	SN0004T0501600.004 [149288]
Median	4/12/2000	12:44p	245,153	SZ_07_01.0000.z	SN0004T0501600.004 [149288]
Median	4/12/2000	12:44p	245,153	SZ_07_02.0000.z	SN0004T0501600.004 [149288]
Median	4/12/2000	12:44p	245,153	SZ_07_03.0000.z	SN0004T0501600.004 [149288]
Median	4/12/2000	12:45p	245,153	SZ_07_04.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	04:09p	196,153	SZ_08_01.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	04:10p	196,153	SZ_08_02.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	04:10p	196,153	SZ_08_03.0000.z	SN0004T0501600.004 [149288]
Median	3/8/2000	04:10p	196,153	SZ_08_04.0000.z	SN0004T0501600.004 [149288]

Table 6-98. Saturated Zone Breakthrough Curves for Probabilistic Simulations

Type	Date	Time	Size (bytes)	File Name	Source
Multi	1/28/2000	04:15p	7,454,385	SZ_01_01.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:15p	7,454,385	SZ_01_02.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:15p	7,413,715	SZ_01_03.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:15p	7,413,715	SZ_01_04.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:16p	7,475,137	SZ_02_01.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:16p	7,475,137	SZ_02_02.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:16p	7,475,137	SZ_02_03.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:16p	7,475,137	SZ_02_04.z	SN0004T0501600.004 [149288]
Multi	4/12/2000	10:31a	16,447,812	SZ_03_01.z	SN0004T0501600.004 [149288]
Multi	4/12/2000	10:40a	16,447,812	SZ_03_02.z	SN0004T0501600.004 [149288]
Multi	4/12/2000	10:45a	16,447,812	SZ_03_03.z	SN0004T0501600.004 [149288]
Multi	4/12/2000	10:49a	16,447,812	SZ_03_04.z	SN0004T0501600.004 [149288]
Multi	3/8/2000	03:31p	7,531,121	SZ_04_01.z	SN0004T0501600.004 [149288]
Multi	3/8/2000	03:32p	7,531,121	SZ_04_02.z	SN0004T0501600.004 [149288]
Multi	3/8/2000	03:36p	7,531,121	SZ_04_03.z	SN0004T0501600.004 [149288]
Multi	3/8/2000	03:36p	7,531,121	SZ_04_04.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:20p	10,449,584	SZ_05_01.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:20p	10,625,985	SZ_05_02.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:20p	9,743,980	SZ_05_03.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:20p	9,743,980	SZ_05_04.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:21p	7,476,265	SZ_06_01.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:21p	7,476,265	SZ_06_02.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:21p	7,476,265	SZ_06_03.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:21p	7,476,265	SZ_06_04.z	SN0004T0501600.004 [149288]

Table 6-98. Saturated Zone Breakthrough Curves for Probabilistic Simulations (Continued)

Type	Date	Time	Size (bytes)	File Name	Source
Multi	4/12/2000	12:44p	6,963,312	SZ_07_01.z	SN0004T0501600.004 [149288]
Multi	4/12/2000	12:44p	6,963,312	SZ_07_02.z	SN0004T0501600.004 [149288]
Multi	4/12/2000	12:45p	6,963,312	SZ_07_03.z	SN0004T0501600.004 [149288]
Multi	4/12/2000	12:45p	6,963,312	SZ_07_04.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:17p	19,909,441	SZ_08_01.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:17p	20,076,777	SZ_08_02.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:17p	20,076,777	SZ_08_03.z	SN0004T0501600.004 [149288]
Multi	1/28/2000	04:17p	20,076,777	SZ_08_04.z	SN0004T0501600.004 [149288]

Results and Verification

The median value case will be examined for the purposes of this subsection. The element SZ_Index (see Figure 6-169) is the index for selecting the appropriate breakthrough curves and is determined by *Realz_Number* in Simulation_Settings (see Section 6.4). Verification of the correct selection by SZ_Index is demonstrated by comparison of the value of 1 to the value in *Realz_Number*. The expression contained in Realz_Number,

if (Median_Value_Run==1,1, MasterClock.Realization)

indicates that realization 1 for the median value case should be selected. The parameter *Median_Value_Run* is a switch contained within the simulation settings container and discussed within Section 6.4. *Median_Value_Run* will equal 1 for median value simulations. Input parameters for the pipe model include selected values used to model SZ flow and transport that are identical to the pre-sampled values used in the convolution (see supporting documentation: *Input and Results of the Base Case Saturated Zone Flow and Transport Model for TSPA* ([CRWMS M&O 2000 [139440]) for a discussion of each parameter and its selected range of values). Verification that the pipe model is using appropriate values for its input parameters for the median value case is demonstrated in Table 6-99. Additional confirmation of correct selection of median value parameters is established by examination of the selector switch element logic. An example of the logic is shown below:

If Median_Value_Run==1 then -10.49

else DCVO

The switch is instructed to choose the median value, -10.49, which is the actual value selected as shown in Table 6-99. Verification of all input parameters shown in Figure 6-172 was carried out in the same manner. The median values chosen by the selector switches are equal to the median values used in the convolution and defined in the AMR abstraction, all of these values fall within the required parameter ranges in Table 6-94. The comparison or the input parameter ranges provided in the abstraction and the selected values is shown below in Table 6-99.

Table 6-99. Comparison of Abstraction Prescribed Input Parameter Values to those Selected for Median Value Simulation

TSPA-SR Model Parameter	Data Type	Model Value	Specified Value			
colloid_conc	Scalar		3e-02			
CORAL	Cumulative Distribution			**Probability Level**	**Value**	
		1.24		0	0.01	
				0.1933	0.1273	
				0.3088	0.38086	
				0.5311	1.3958	
				0.7432	2.556	
				0.8244	3.013	
				0.9047	3.5105	
				0.9655	3.9989	
				0.9978	4.8795	
				1	6.3624	
CORVO	Cumulative Distribution			**Probability Level**	**Value**	
		2.21		0	0.025308	
				0.039	0.04139	
				0.08125	0.77815	
				0.2605	2	
				0.7605	2.4472	
				1	2.9031	
DCVO	Uniform Distribution		**Min**		**Max**	
		-10.49	-13.0		-10	
FISVO	Truncated Normal Distribution		**Min**	**Mean**	**Max**	**Std Dev**
		1.32	0	1.29	1.0E+10	0.43
FPLAN	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
FPVO	Uniform Distribution		**Min**		**Max**	
		-3	-5.0		-1.0	
GWSPD	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	

Table 6-99. Comparison of Abstraction Prescribed Input Parameter Values to those Selected for Median Value Simulation (Continued)

TSPA-SR Model Parameter	Data Type	Model Value	Specified Value			
HAVO	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
I_hi_thresh_coll_gw	Scalar		0.05			
I_lo_thresh_coll_gw	Scalar		0.01			
Ionic_Strength_SZ	Scalar		2.5 e-3			
Kc_Pu_gw_Colloid	Expression (colloid_conc)*(Kd_Pu_coll_wf_rev)	0.00021679				
Kc_Am_gw_Colloid	Espression (colloid_conc)*(Kd_Am_coll_wf_rev)	0.01	-			
KDIAL	Uniform Distribution		**Min**		**Max**	
		0.47	0.32		0.63	
KDNPAL	Beta Distribution		**Min**	**Mean**	**Max**	**Std Dev**
		18.2	0	18.2	100	18.8
KDNPVO	Beta Distribution		**Min**	**Mean**	**Max**	**Std Dev**
		0.5	0	0.5	2.0	0.5
KDRN10	Uniform Distribution		**Min**		**Max**	
		50*	0.0		100	
KDRN9	Uniform Distribution		**Min**		**Max**	
		25*	0.0		50	
KDTCAL	Uniform Distribution		**Min**		**Max**	
		0.45	0.27		0.62	
KDUAL	Uniform Distribution		**Min**		**Max**	
		4	0		8	
KDUVO	Uniform Distribution		**Min**		**Max**	
		2	0		4	
LDISP	Truncated Normal Distribution		**Min**	**Mean**	**Max**	**Std Dev**
		2	0	2.0	1.E+10	0.75
Mcoll_gw_max	Scalar		3e-2			

Table 6-99. Comparison of Abstraction Prescribed Input Parameter Values to those Selected for Median Value Simulation (Continued)

TSPA-SR Model Parameter	Data Type	Model Value	Specified Value			
Mcoll_gw_min	Scalar		3e-6			
NVF19	Truncated Normal Distribution		**Min**	**Mean**	**Max**	**Std Dev**
		0.18	0	0.18	1.0E+10	0.051
NVF7	Truncated Normal Distribution		**Min**	**Mean**	**Max**	**Std Dev**
		0.18	0	0.18	1.0E+10	0.051
SRC1X	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
SRC1Y	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
SRC2X	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
SRC2Y	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
SRC3X	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
SRC3Y	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
SRC4X	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	
SRC4Y	Uniform Distribution		**Min**		**Max**	
		0.5	0.0		1.0	

The values for the contents of the Alluvium_Properties, Volcanic_Properties, Flow_Properties, and Pipe_Length containers as well as pipe segments Pipe_5km, Pipe_12km, and Pipe_20km were verified by the comparison of the values given in the abstraction with those used in the model. Parameters that exist in the model as a function were verified with hand calculations. Input values were found to be correct and the equations for the pipe model were found to give the correct result or be within the range of round off error. The comparisons are shown in Table 6-100 through Table 6-104.

Table 6-100. Comparison of Abstraction Prescribed Input Parameter Values for Alluvium_Properties to Those Selected for Median Value Simulation

TSPA-SR Model Parameter	Model Equation (if applicable)	Model Value	Abstraction Value or Calculated Value
Alluvium_Density	Direct input	1270 kg/m^3	1270 kg/m^3
Kd_Allu_Irr	NVF19_Median*(10^CORAL_Median-1)/Alluvium_Density	0.00232129 m^3/kg	0.002321 m^3/kg
Kd_Allu_Rev	(KDRN10_Median*(NVF19_Median/0.35)/(1.0+KC_AM_GW_COLLOID_Median))	0.0254597 m^3/kg	0.0254597 m^3/kg
Length_Alluvium	(FPLAN_Median*7+1)	4.5 km	4.5 km

Table 6-101. Comparison of Abstraction Prescribed Input Parameter Values for Volcanic_Properties to those Selected for Median Value Simulation

TSPA-SR Model Parameter	Model Equation (if applicable)	Model Value	Abstraction Value or Calculated Value
Kd_Vol_Rev	KDRN10_Median	0.05 m^3/kg	0.05 m^3/kg
Volcanic_MatrixPorosity	Direct input	0.19	0.19
Volcanic_Density	Direct input	1940 kg/m^3	1940 kg/m^3
Fracture_Perimeter	Direct input	2.0 m	2.0 m
Fracture_Area	((10^(FPVO_Median*1{m-1}))*(10^(FISVO_Median*1{m-1}))*1 m^2	0.020893 m^2	0.020893 m^2
Coating_Thickness	Direct input	0.0001 m	0.0001 m
Coating_Porosity	Direct input	0.01	0.01
Kd_Vol_Irr	((10^CORVO_Median-1)*Fracture_Area/Fracture_Perimeter/Coating_Thickness-Coating_Porosity)/Volcanic_Density	8.67924 m^3/kg	8.67926 m^3/kg
Dispersivity	(10^(LDISP_Median*1{m-1}))*1 m	100 m	100m

Table 6-102. Comparison of Abstraction Prescribed Input Parameter Values for Flow_Properties to those Selected for Median Value Simulation

TSPA-SR Model Parameter	Model Equation	Model Value	Abstraction Value or Calculated Value
Discharge_5km	If (HAVO_Median<0.5 then if (GWSPD_Median<0.24 then 0.066 else if(GWSPD_Median>=0.24 and GWSPD_Median<=0.76 then 0.66 else 6.6)) else if (GWSPD_Median<0.24 then 0.075 else if (GWSPD_Median>=0.24 and GWSPD_Median <= 0.76 then 0.75 else 7.5)))	0.75	0.75
Factor	Direct input	3.9	3.9

Table 6-102. Comparison of Abstraction Prescribed Input Parameter Values for Flow_Properties to those Selected for Median Value Simulation (Continued)

TSPA-SR Model Parameter	Model Equation	Model Value	Abstraction Value or Calculated Value
Specific_Discharge 5km	Factor*Discharge_5km{m/yr}	2.925 m/yr	2.925 m/yr
Flow	(10^(FISVO_Median*1{m-1}))*1.0{m2}* Specific_Discharge5km	61.1119 m^3/yr	61.1119 m^3/yr
Discharge_20km	If (HAVO_Median<0.5 then if (GWSPD_Median<0.24 then 0.23 else if(GWSPD_Median>=0.24 and GWSPD_Median<=0.76 then 2.3 else 23)) else if (GWSPD_Median<0.24 then 0.29 else if (GWSPD_Median>=0.24 and GWSPD_Median <= 0.76 then 2.9 else 29.0)))	2.9	2.9
Specific_Discharge 20km	Factor*Discharge_20km{m/yr}	11.31 m/yr	11.31 m/yr

Table 6-103. Comparison of Abstraction Prescribed Input Parameter Values for Pipe_Length to those Selected for Median Value Simulation

TSPA-SR Model Parameter	Model Equation (if applicable)	Model Value	Abstraction Value or Calculated Value
Pipe_Length_5km	Direct input	6 km	6 km
Pipe_Length_12km	15{km}-Length_Alluvium	10.5 km	10.5 km
Pipe_Length_20km	Length_Alluvium	4.5 km	4.5 km

Table 6-104. Comparison of Abstraction Prescribed Input Parameter Values to those Selected for Median Value Simulation

TSPA-SR Model Parameter	Model Equation	Model Value	Abstraction Value or Calculated Value
Pipe_12km	Fracture_Area/Specific_Discharge20km*Specific_ Discharge5km	0.005403355 m^2	0.0054034 m^2
Pipe_20km	Flow/Specific_Discharge20km	5.40335 m^2	5.40335 m^2
	2.0*(1.0{m}+Flow/Specific_Discharge20km/1.0{m})	12.8067 m	12.8067 m

The saturated zone component model yields mass of radionuclides released at a 20 km boundary point per year. This mass per year is passed to the Biosphere model and diluted with a volume of water to produce a concentration used to calculate dose (see Section 6.3.8). Both the SZ convolution and the SZ pipe model yield mass release of radionuclides in grams/year at the 20 km for select species. Figure 6-176 through Figure 6-181 display the median value results for select radionuclide species, for both the SZ convolution integral model and the SZ pipe model. For comparison, the mass release from the UZ is plotted on the same graph. As can be seen in all cases, either convolution integral or pipe model results, the mass transferred from the UZ is appropriately passed through to the SZ model. Additionally, the mass released at the 20 km boundary, mass release from the SZ, is commensurate with the UZ input. The SZ convolution integral model results are shown for Np^{237}, Pu^{239}, Tc^{99}, and Ic^{239} (Figure 6-176, Figure 6-177, Figure 6-180 and Figure 6-181). For Np^{237} and Pu^{239} the mass release is appropriately retarded through the SZ model as defined by the SZ AMR. Figure 6-177 displays the greater retardation of Pu^{239} as compared to Np^{237}, as expected by the greater sorption of Pu in the SZ alluvium. In contrast, both Tc^{99} and Ic^{239}, pass through the SZ convolution integral model with negligible delay (compared versus the UZ mass released at the water table) as expected from non-sorbing species and colloidal transported species. The results of the convolution integral model are consistent with the SZ AMR abstraction (see supporting documentation: *Input and Results of the Base Case Saturated Zone Flow and Transport Model for TSPA* (CRWMS M&O 2000 [139440]). SZ pipe model results are displayed in Figure 6-178 and Figure 6-179, as represented by Th^{229} and its parent species U^{233}. For comparison the UZ mass release at the water table for Th^{229} and U^{233} is displayed. Species are tracked through the SZ pipe model to capture the ingrowth from parent species for decay chains. As displayed in both Figure 6-178 and Figure 6-179, the mass release from the SZ at the 20 km boundary exceeds the mass input from the UZ model. Ingrowth from the decay U^{233} to Th^{229} is evident from the results of the SZ pipe model. Given that the SZ mass release at 20 km is reasonable compared to the SZ abstraction described in the source AMR, the SZ parameter input values were demonstrated to be selected correctly within the TSPA model, and the SZ parameter values were correctly input from the source AMR (DTNs), it can been concluded that the SZ component model is verified.

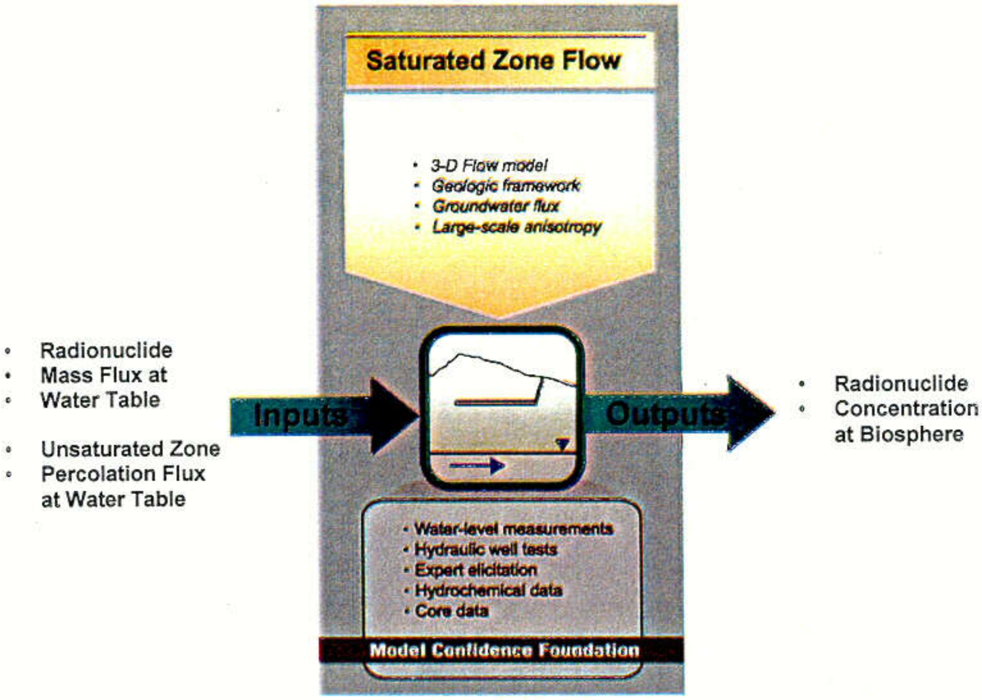


Figure 6-167. Evaluation of the Migration of Radionuclides from Their Introduction at the Water Table below the Repository to the Release Point to the Biosphere

C08

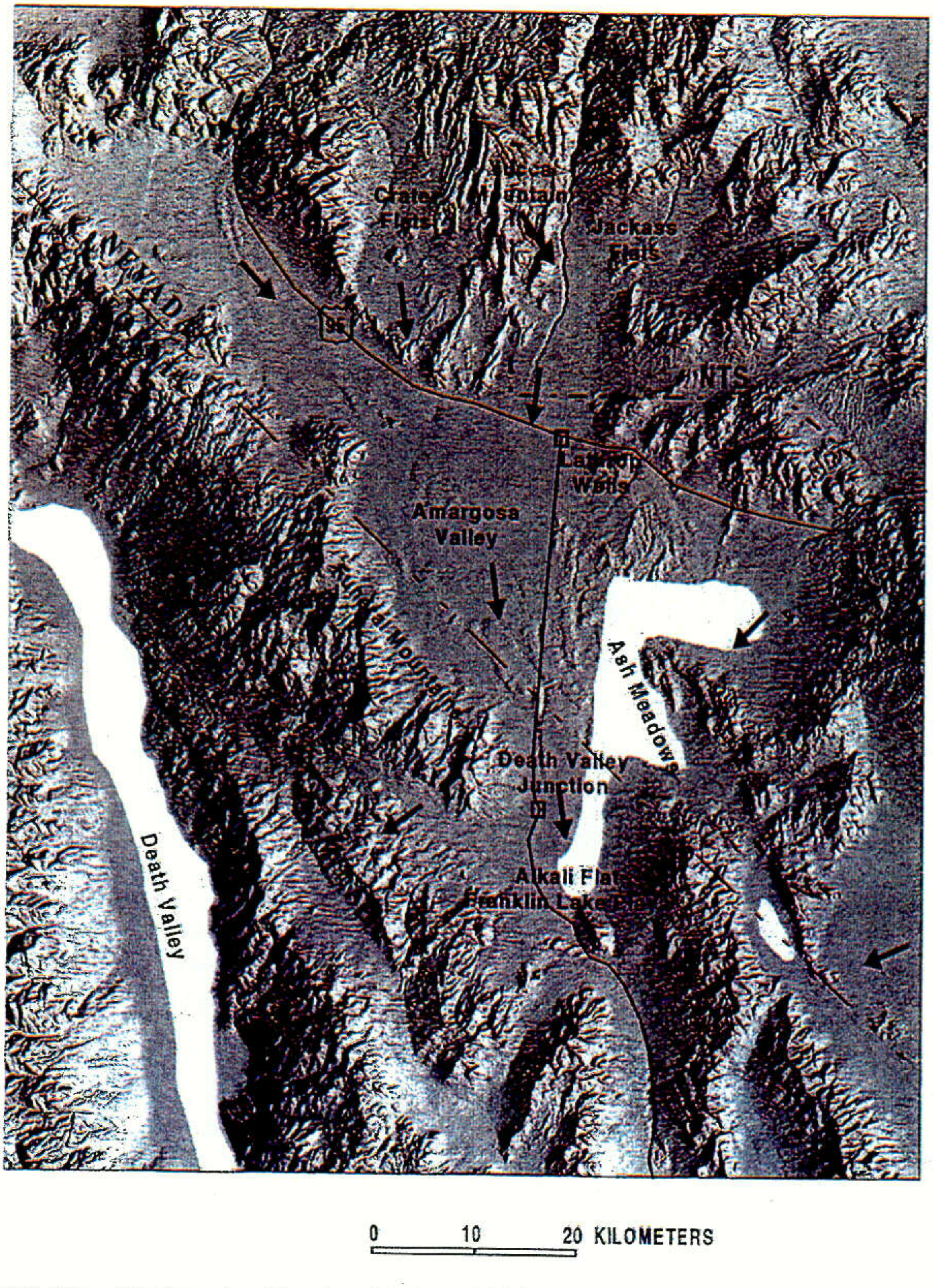


Figure 6-168. The Geosphere/Biosphere Interface which is assumed to be located 20 km from the Repository

C09

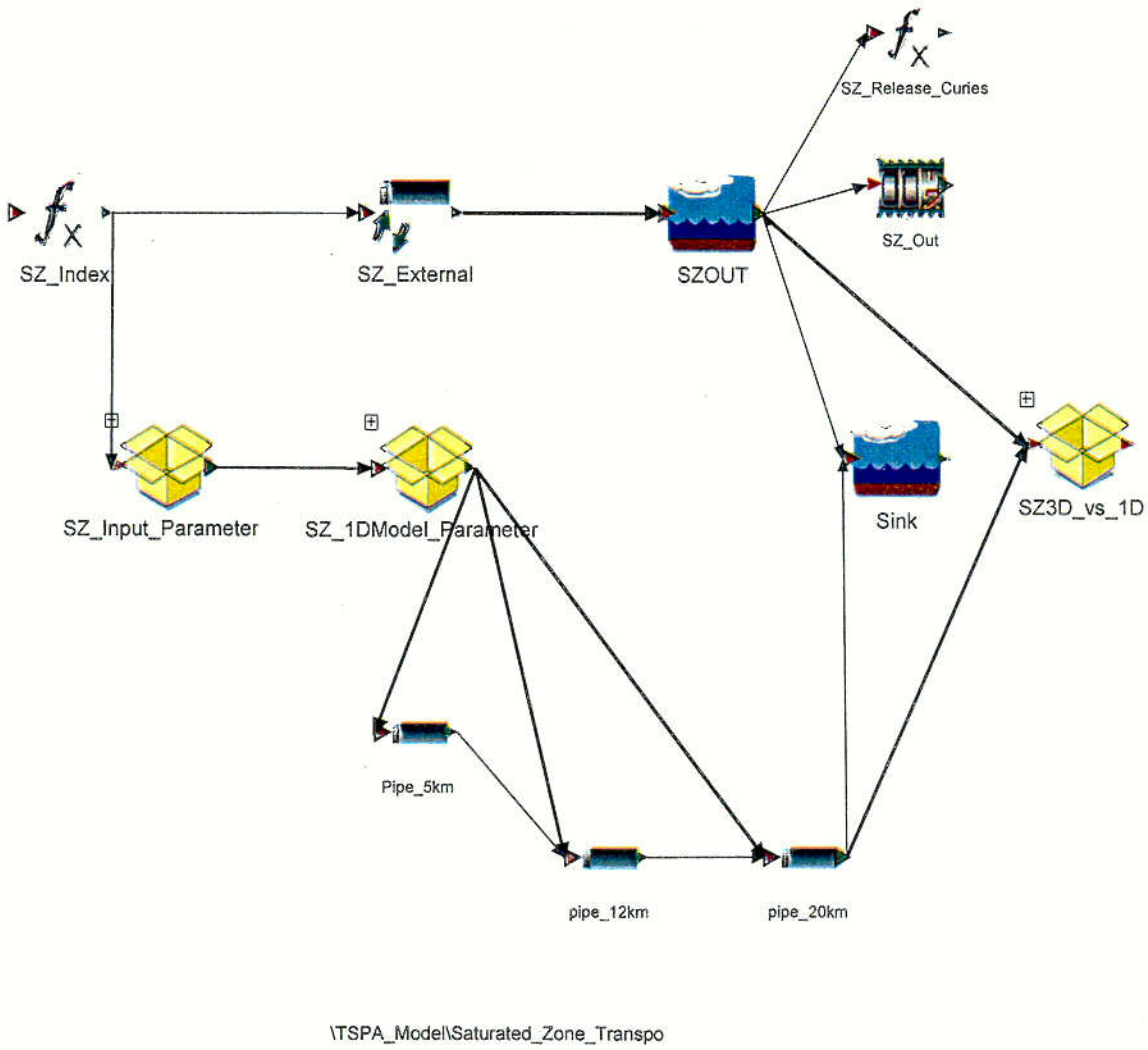
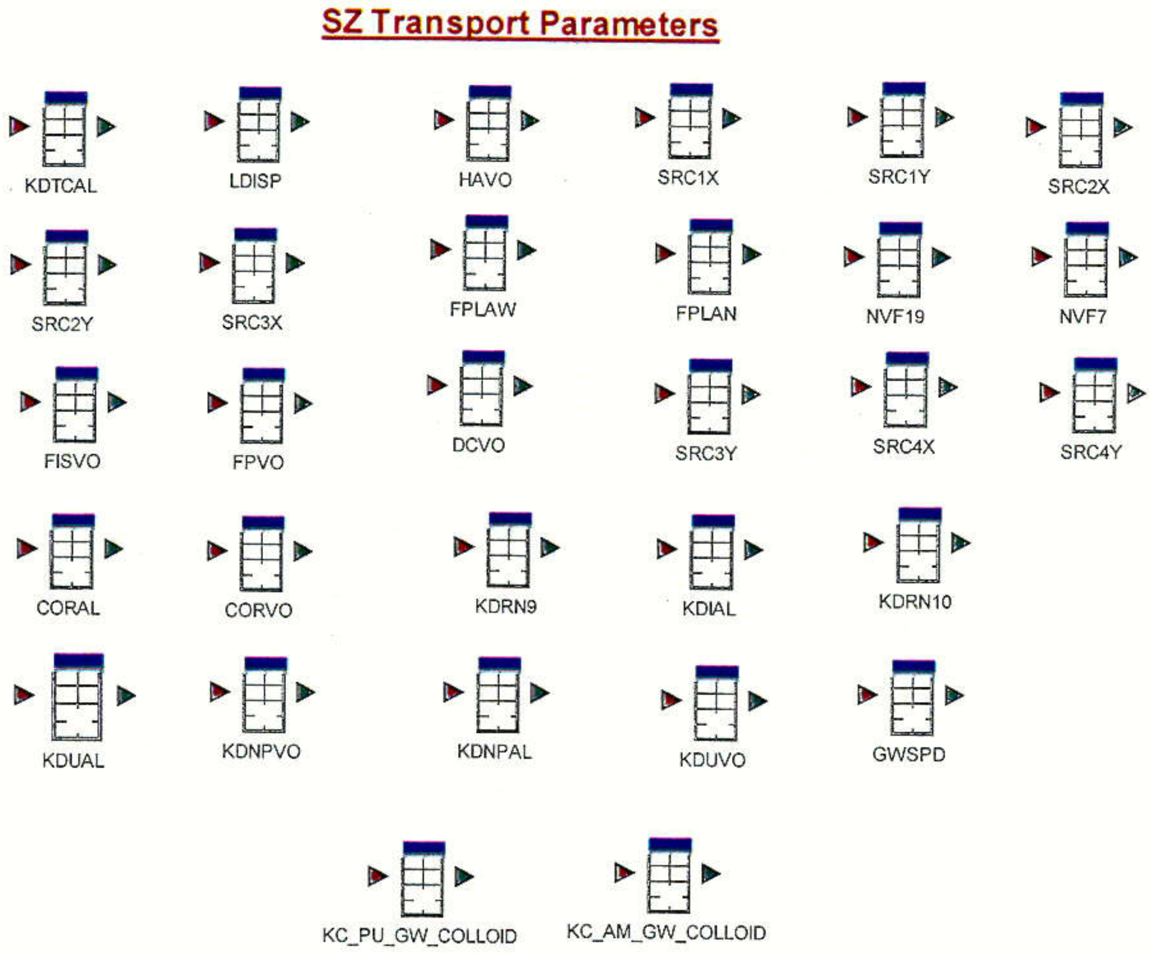


Figure 6-169. Illustration of Saturated Zone One-dimensional Pipe Model and Convolution Integral Model Implementation

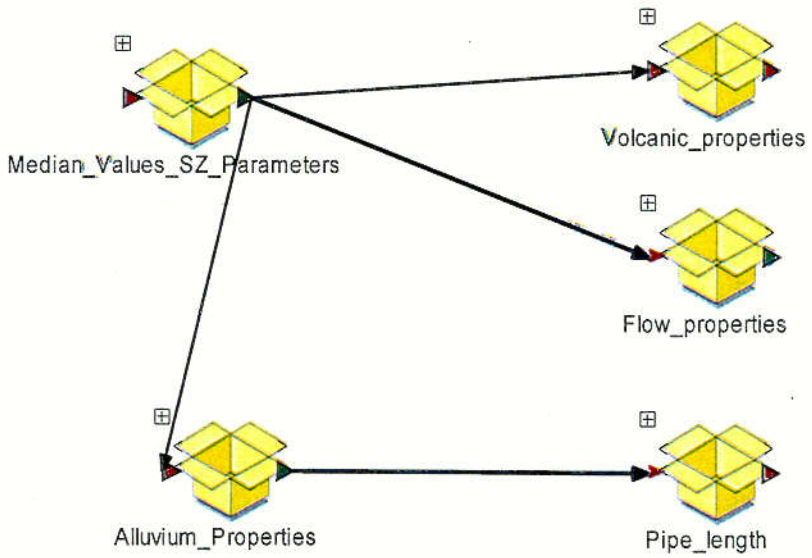
C10



\TSPA_Model\Saturated_Zone_Transport\SZ_Input_Parameters\

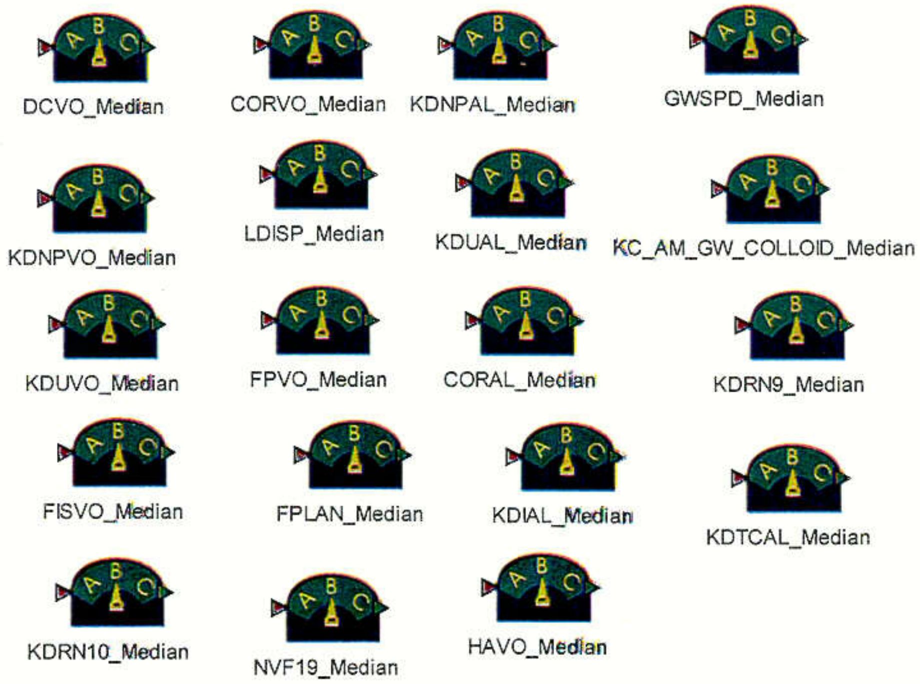
Figure 6-170. Look-up Tables for Multiple Realization Simulation Residing in SZ_Input_Parameters Container

C11



\TSPA_Model\Saturated_Zone_Transport\SZ_1DModel_Parameters\

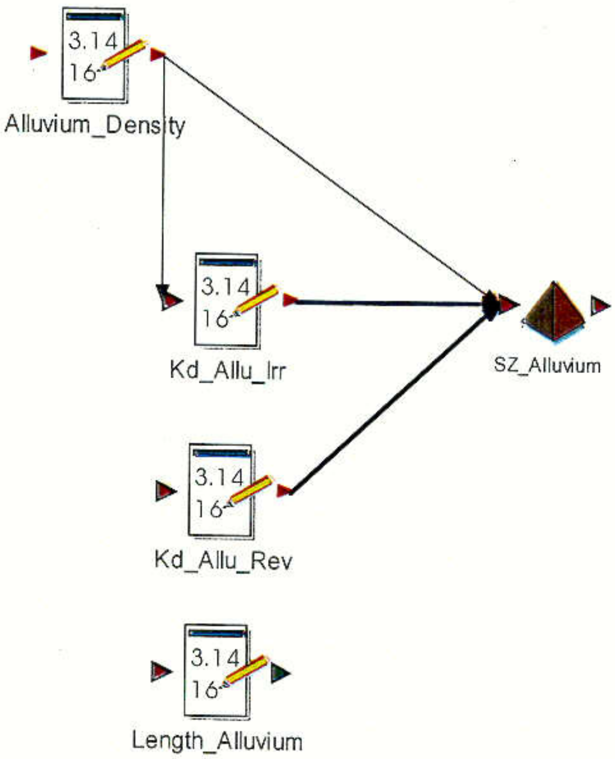
Figure 6-171. Contents of SZ_1DModel_Parameters Container



\TSPA_Model\Saturated_Zone_Transport\SZ_1DModel_Parameters\Median_Values_SZ_Parameters\

Figure 6-172. Contents of Median_Values_SZ_Parameters Container

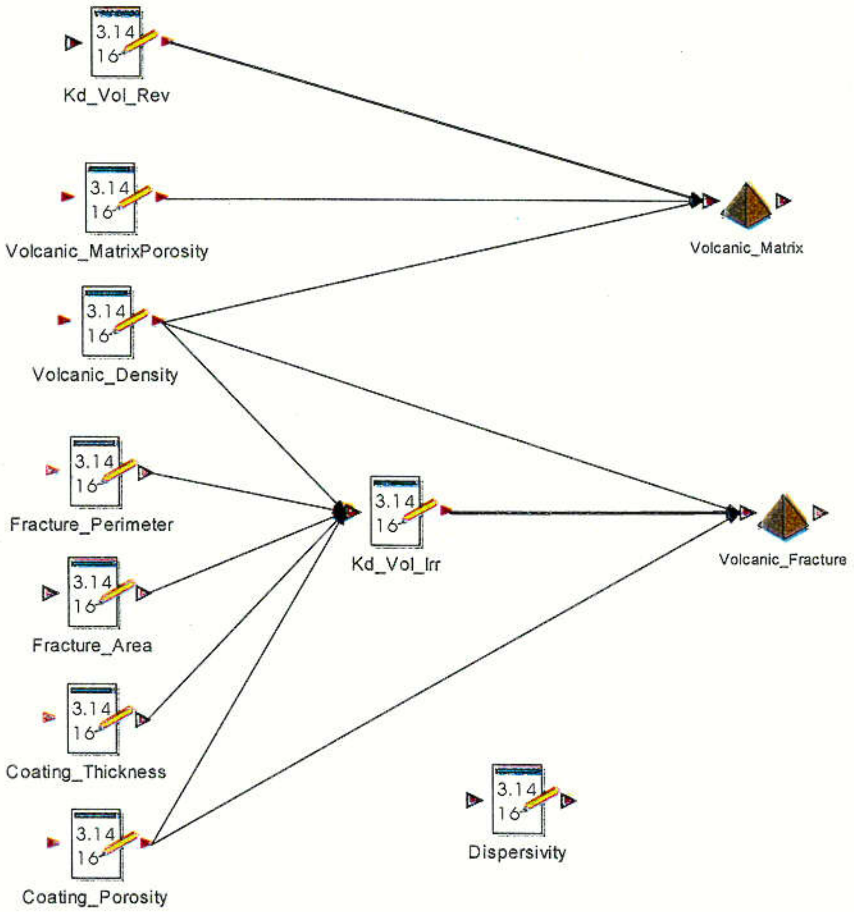
C12



\TSPA_Model\Saturated_Zone_Transport\SZ_1DModel_Parameters\Alluvium_Properties\

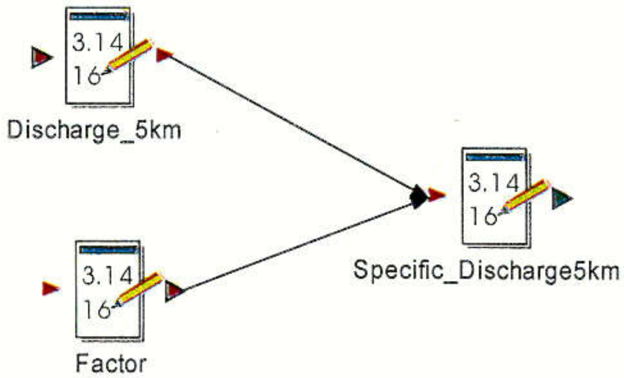
Figure 6-173. Contents of Alluvium_Properties Container

C13



\TSPA_Model\Saturated_Zone_Transport\SZ_1DModel_Parameters\Volcanic_properties\

Figure 6-174. Contents of Volcanic_Properties Container



\TSPA_Model\Saturated_Zone_Transport\SZ_1DModel_Parameters\Flow_properties\Distance_5Km\

Figure 6-175. Scaling of Discharge_5km by Factor1 to Account for Changes in Groundwater Flux Due to Climate

C14

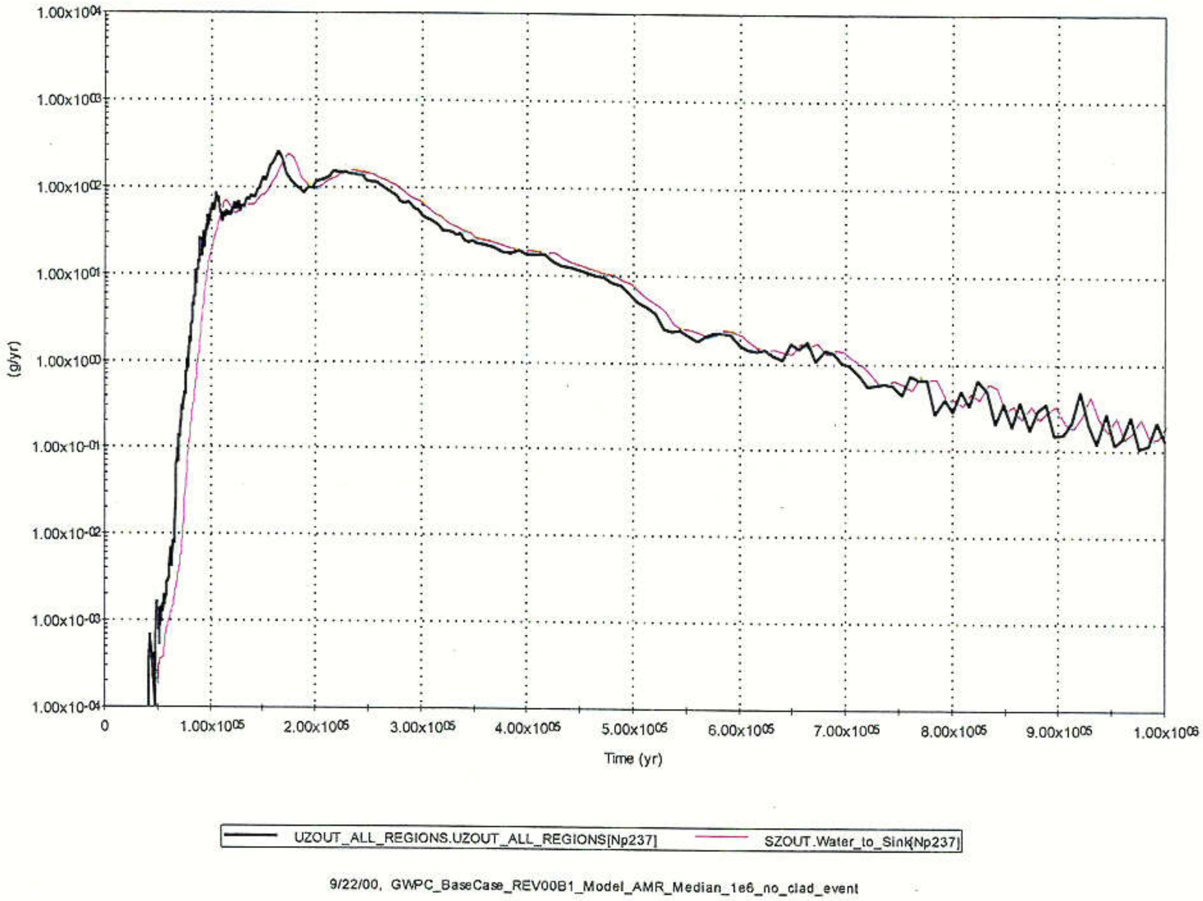


Figure 6-176. Np^{237} Transport through the SZ Convolution Integral Model in Grams/Year Compared to Total UZ Release in Grams/Year

C15

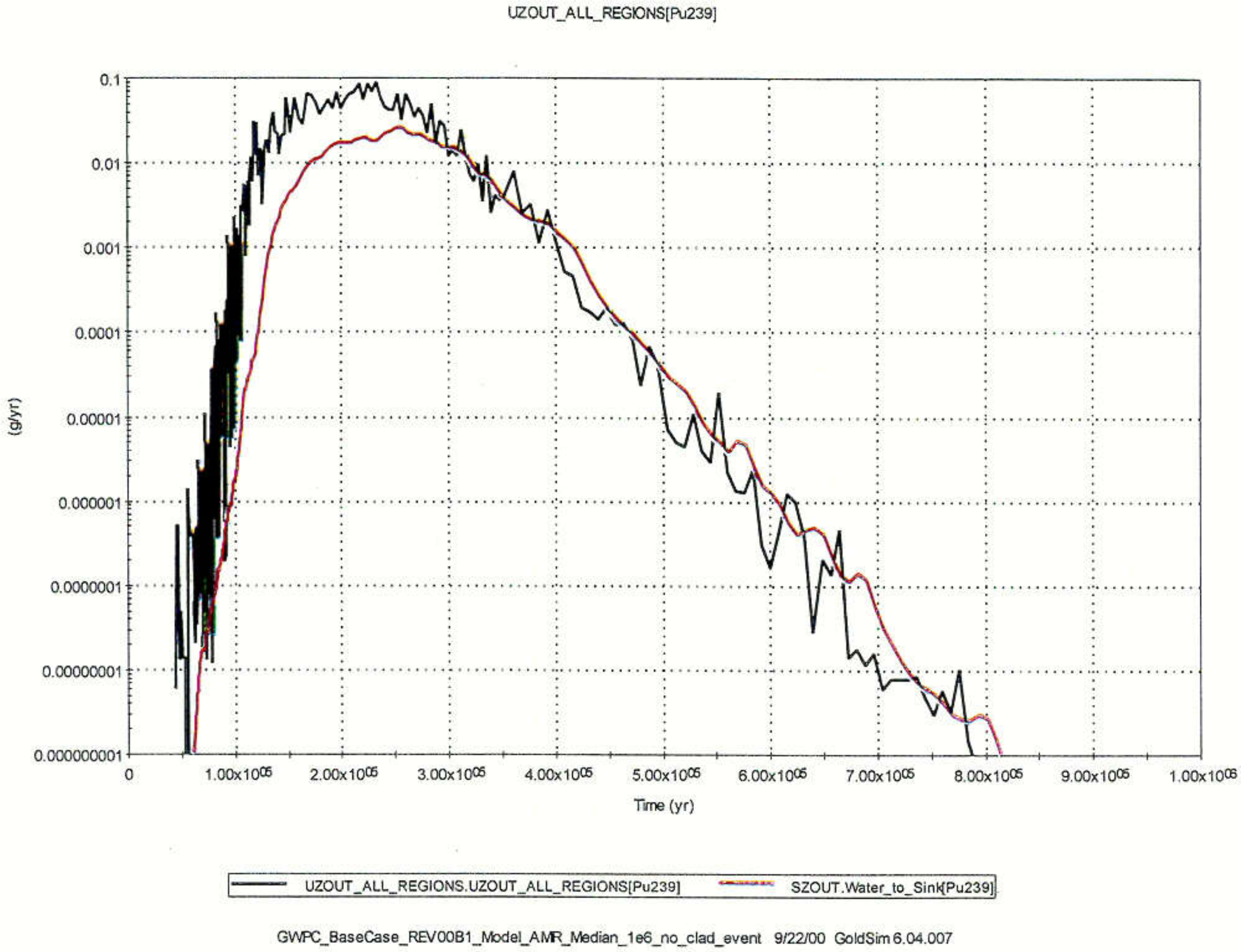


Figure 6-177. Time History Result for SZ Convolution Integral Model Mass Release in grams/year Compared to Total UZ Mass Release for Pu^{239}

C16

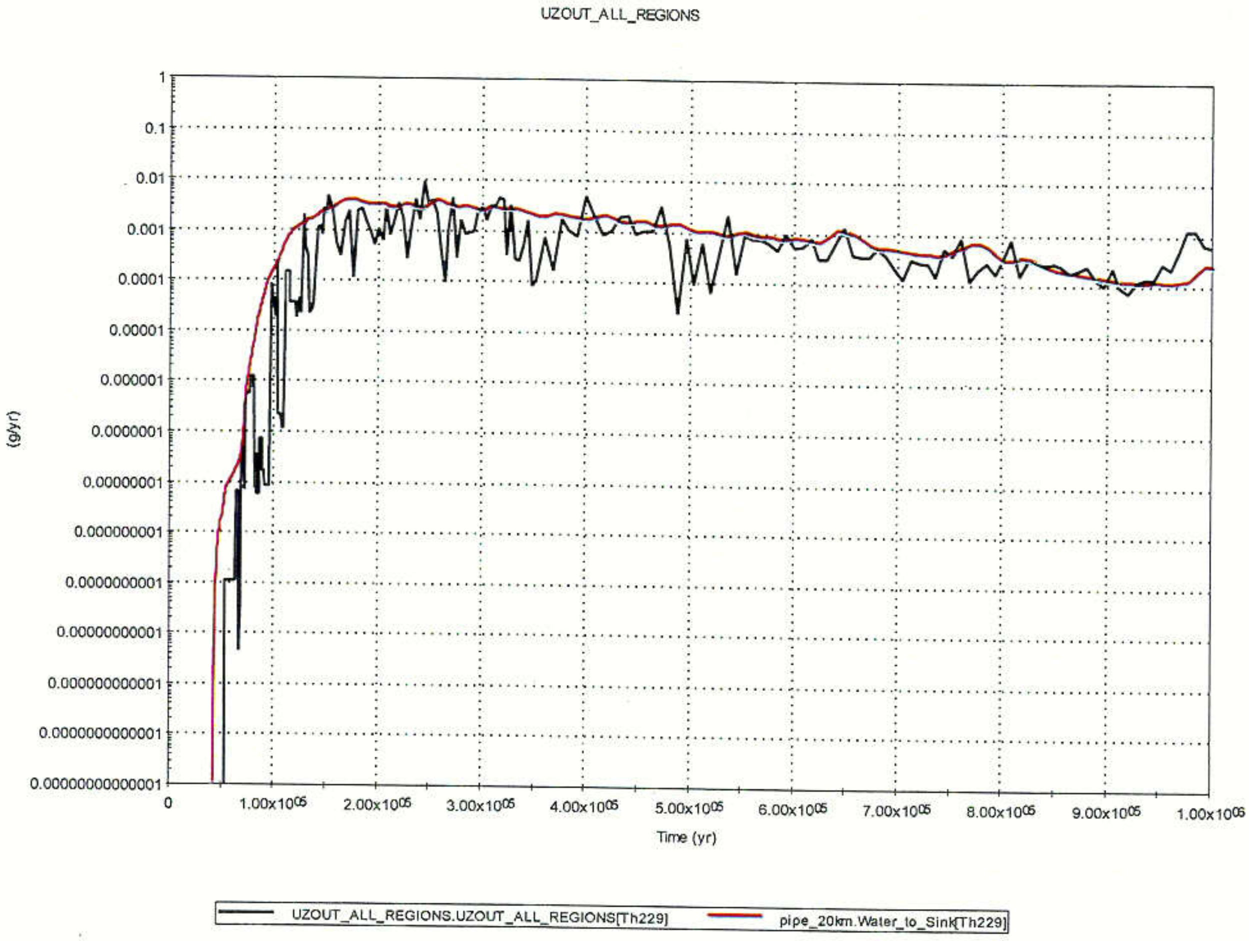


Figure 6-178. Time History Result for SZ Pipe Model Mass Release in grams/year Compared to Total UZ Mass Release for Th^{229}

C17

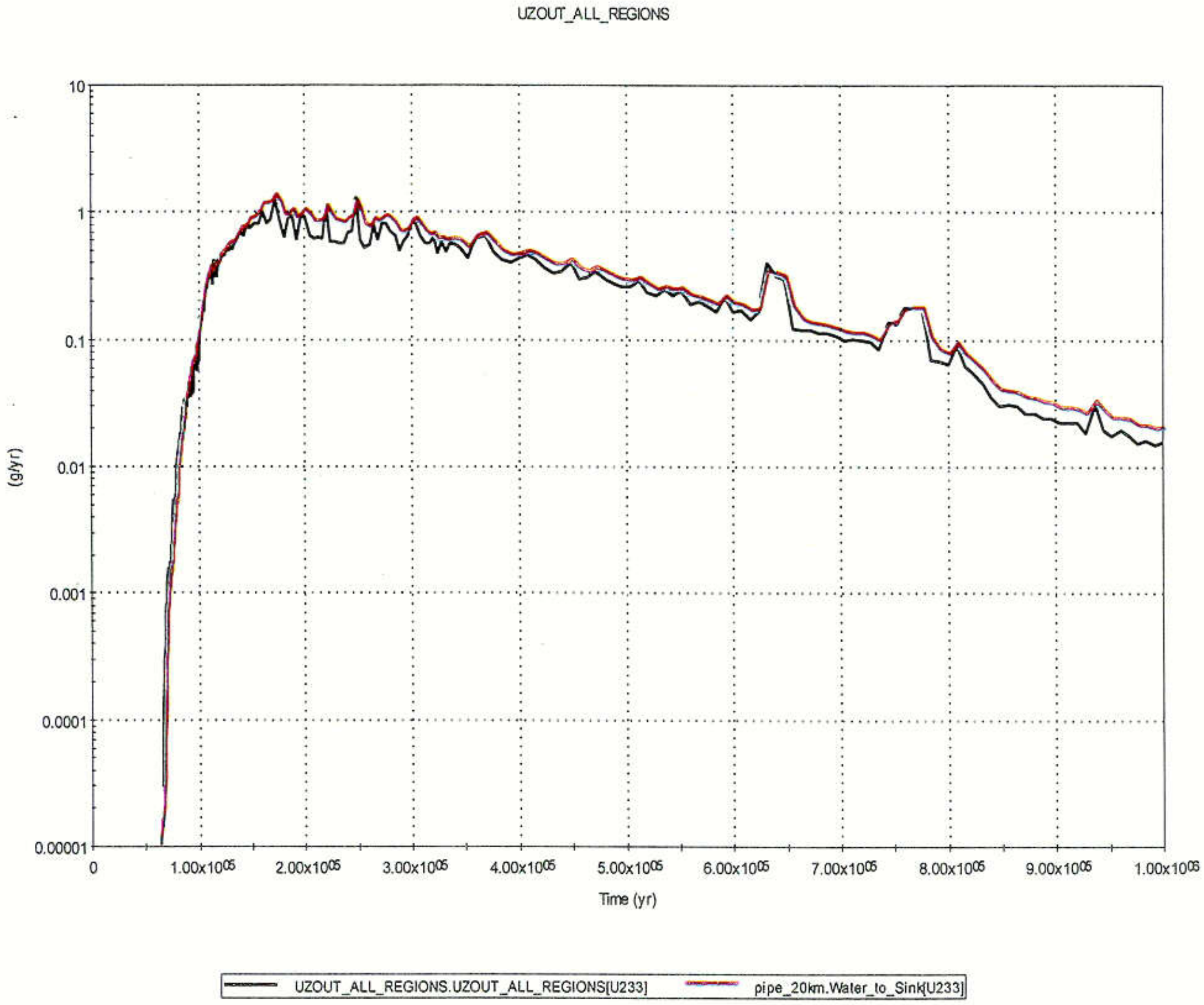


Figure 6-179. Time History Result for SZ Pipe Model Mass Release in Grams/Year Compared to Total UZ Mass Release for U^{233} (Parent Species for Th^{229})

C18

Figure 6-180. Time History Result for SZ Convolution Integral Model Mass Release in Grams/Year Compared to Total UZ Mass Release for Tc^{99}

C19

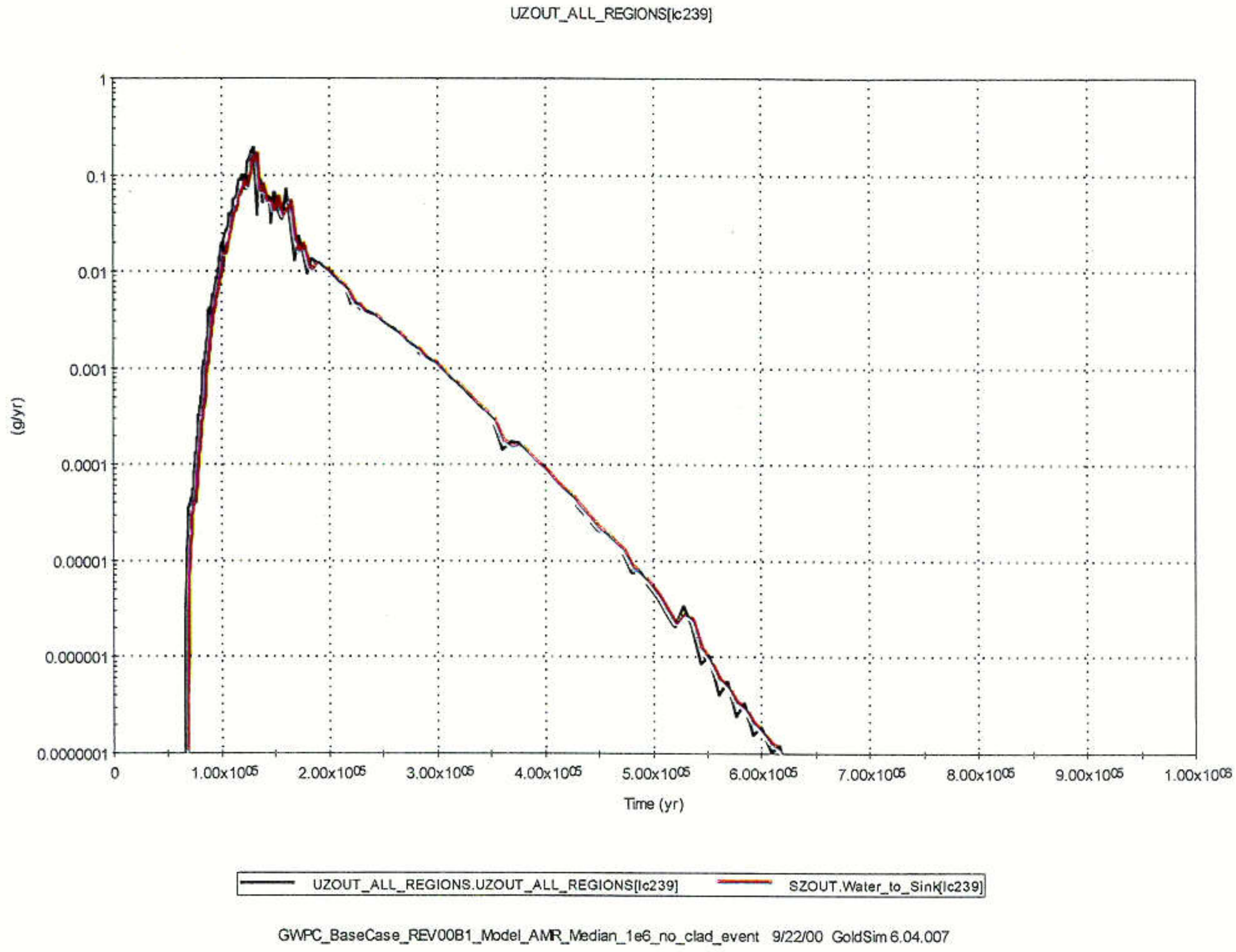


Figure 6-181. Time History Result for SZ Convolution Integral Model Mass Release in Grams/Year Compared to Total UZ Mass Release for Ic^{239} (Irreversible Pu colloid)

6.3.8 Biosphere

After the permanent closure of the repository, the engineered barrier systems within the repository may eventually lose their ability to fully contain the radionuclide inventory. Then the radionuclides could migrate through the geosphere and eventually enter the biosphere—chiefly via the local groundwater used for crop irrigation in areas of human habitation. The resultant migration may eventually be manifested as a potential radiation exposure to individuals consuming crops and groundwater directly or via animal products derived from the groundwater.

The primary measure of the repository performance is the annual dose of radiation that potentially would be received by an average member of the critical group living in the region (Dyer 1999 [105655], Section 113(b)). In the base-case human exposure scenario, the critical group resides in a farming community located approximately 20 km south from the proposed facility and is the group that is expected to receive the greatest exposure. The reference receptor is representative of the small group of people who are most at risk within the community. This receptor, the average member of the critical group, is an adult who lives year-round at this

C-20

location, uses a local well as the primary water source, and otherwise has habits, such as the consumption of local foods, that are similar to those of inhabitants of the region but in a range that increases potential radiation exposure (Dyer 1999 [105655], Section 115).

Except for releases associated with eruptive volcanism (Section 6.3.9.1), contaminated well water is the only pathway by which radionuclides from the repository can reach the reference biosphere. The primary human exposure scenario to radionuclides in groundwater is a well for domestic use, livestock drinking water, and irrigation of food chain crops and other household grasses and plants, resulting in a dose to the reference receptor in the proposed community via various biosphere pathways, as described in Section 3.9 of *Total System Performance Assessment For the Site Recommendation* (CRWMS M&O 2000 [143665]).

Once the radionuclide exposure pathways (e.g., ingestion, inhalation, and external exposure) of the average member of the critical group are established, the annual dose rate per unit groundwater concentration (its BDCF) is established for each radionuclide identified as being of potential concern. These biosphere dose conversion factors (BDCFs) are used to calculate the annual dose of radiation to the average member of the critical group.

6.3.8.1 Biosphere Dose Conversion Factors

Overview

The GENII-S computer code, a code for statistical and deterministic calculations of radiation doses to humans from radionuclides in the environment, was used to model radionuclide transport through biosphere pathways to the specified receptor, the average member of the critical group (CRWMS M&O 2000 [144692], Section 3, p. 8). Model parameters were quantified using site-specific data and other accepted information. GENII-S biosphere pathway and consumption inputs were scalar data or stochastic distributions. GENII-S uses these inputs and a unit concentration of the radionuclide in water to produce a biosphere dose conversion factor (BDCF) for that radionuclide. The BDCF includes the effects of various pathways through the environment (e.g., irrigation and uptake of the contaminant by vegetables, then ingestion by the receptor), as well as various pathways through the reference person (e.g., the fraction of the ingested radionuclide taken up by the reference person, where it is accumulated in the body, and its retention time). The AMR *Non-Disruptive Event Biosphere Dose Conversion Factors* (CRWMS M&O 2000 [136285], Section 4.1.1, Table 1) details the selection criteria and operable parameters for deriving the BDCFs.

The GENII-S code was used to generate a discrete distribution of BDCFs (with 130 values of for the groundwater release scenarios and 160 for the eruptive igneous release scenario) for 16 radionuclides for each of 6 prior irrigation time periods—the number of years that the land has been irrigated before intake occurs. Because many stochastic distributions were used as inputs, 130 realizations (160 for the eruptive scenario) with GENII-S were simulated for each radionuclide and the results recorded. Six different irrigation periods were investigated to evaluate the effects of contaminant build-up in the soil resulting from contaminant partitioning. The data was compiled and a statistical distribution was fit to each of the data sets. The AMRs *Distribution Fitting to the Stochastic BDCF Data* (CRWMS M&O 2000 [144055], Sections 6.1 and 6.2) and *Abstraction of BDCF Distributions for Irrigation Periods* (CRWMS M&O 2000

[144054], Sections 6.1 and 6.2) describe the statistical analysis and abstraction in detail. The χ^2 test for goodness of fit was used to determine whether a distribution provided an acceptable fit to the BDCF data. The stochastic distribution parameters were optimized using the Excel solver routine to minimize χ^2 and determine the distribution parameters providing the best fit. The soil build-up for five radionuclides, ^{90}Sr, ^{137}Cs, ^{229}Th, ^{243}Am, and ^{232}U, was significant enough to warrant soil erosion concerns, a radionuclide removal mechanism that could impact the amount of build-up expected. Additional studies were performed to determine the significance of soil erosion for radionuclides susceptible to soil build-up. The BDCF data given to TSPA were those that represented the justifiable but reasonable upper limits for the soil build-up process.

Inputs to the TSPA Model

BDCF relationships were abstracted from the aforementioned statistical evaluations and incorporated into the TSPA-SR model as stochastic distributions. The final BDCF distributions, along with their reference sources, are presented in Table 6-105. This data was input to the TSPA model container *BDCF*, as shown in Figure 6-182 and Figure 6-183. The output arrows in Figure 6-183, which go from the stochastic element for the Np-237 BDCF to all other stochastic BDCFs, indicate that all BDCFs are correlated to the Np-237 BDCF. The stochastic elements for the BDCFs of Th-230, Ra-226, and Pb-210 are not shown in Figure 6-183 because they are in another container in the model file. In fact, they are in the very top container because they were added at a later date and it was necessary to put them in the top container to maintain the same sampling sequence for all of the other stochastic variables.

Table 6-105. Best Fit Parameters for BDCF Distributions

Description	TSPA Parameter	Parameter value	Reference/DTN
BDCF for ^{227}Ac	BDCF_Ac227 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=18.011 Geom.S.D.=1.1623	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{241}Am	BDCF_Am241 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=5.0118 Geom.S.D.=1.1562	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{243}Am	BDCF_Am243 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=5.0294 Geom.S.D.=1.1631	CRWMS M&O 2000 [144054], Table 7, DTN: MO0003SPAABS07.006 [148876]
BDCF for ^{14}C	BDCF_C14 (mrem/yr)/(pCi/L)	Shifted Lognormal Geom.Mean=0.0005536 Geom.S.D.=1.5177 Off Set=0.0034675	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{129}I	BDCF_I129 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=0.35619 Geom.S.D.=1.1874	CRWMS M&O 2000, [144055] Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{237}Np	BDCF_Np237 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=6.7382 Geom.S.D.=1.1625	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{231}Pa	BDCF_Pa231 (mrem/yr)/(pCi/L)	Lognormal Distribution True Mean=13.7 True S.D.=2.32	CRWMS M&O 2000 [144700], Table 5, DTN:MO0002SPACRI02.002 [150040]

Table 6-105. Best Fit Parameters for BDCF Distributions (Continued)

Description	TSPA Parameter	Parameter value	Reference/DTN
BDCF for ^{210}Pb	BDCF_Pb210 (rem/yr)/(pCi/L)	Cumulative Distribution Min 5.363E-03 5.0% 5.961E-03 10.0% 6.116E-03 15.0% 6.231E-03 20.0% 6.365E-03 25.0% 6.451E-03 30.0% 6.548E-03 35.0% 6.696E-03 40.0% 6.870E-03 45.0% 6.990E-03 50.0% 7.081E-03 55.0% 7.201E-03 60.0% 7.474E-03 65.0% 7.755E-03 70.0% 7.994E-03 75.0% 8.438E-03 80.0% 8.729E-03 85.0% 9.059E-03 90.0% 9.626E-03 95.0% 1.031E-02 Max 1.276E-02	CRWMS M&O 2000 [153034] DTN:MO0006SPABDC01.007 [152837]
BDCF for ^{238}Pu	BDCF_Pu238 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=4.1087 Geom.S.D.=1.1607	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{239}Pu	BDCF_Pu239 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=4.9759 Geom.S.D.=1.1505	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{240}Pu	BDCF_Pu240 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=4.9525 Geom.S.D.=1.1511	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{242}Pu	BDCF_Pu242 (rem/yr)/(pCi/L)	Cumulative Distribution Min 3.073E-03 5.0% 3.688E-03 10.0% 3.814E-03 15.0% 3.881E-03 20.0% 3.932E-03 25.0% 3.983E-03 30.0% 4.027E-03 35.0% 4.078E-03 40.0% 4.112E-03 45.0% 4.267E-03 50.0% 4.364E-03 55.0% 4.440E-03 60.0% 4.498E-03 65.0% 4.604E-03 70.0% 4.683E-03 75.0% 4.913E-03 80.0% 5.076E-03 85.0% 5.289E-03 90.0% 5.464E-03 95.0% 5.750E-03 Max 7.897E-03	CRWMS M&O 2000 [153034] DTN:MO0006SPABDC01.007 [152837]

Table 6-105. Best Fit Parameters for BDCF Distributions (Continued)

Description	TSPA Parameter	Parameter value	Reference/DTN
BDCF for ^{226}Ra	BDCF_Ra226 (rem/yr)/(pCi/L)	Cumulative Distribution Min 4.166E-03 5.0% 5.018E-03 10.0% 5.325E-03 15.0% 5.991E-03 20.0% 6.479E-03 25.0% 6.987E-03 30.0% 7.498E-03 35.0% 8.031E-03 40.0% 8.614E-03 45.0% 9.598E-03 50.0% 1.055E-02 55.0% 1.146E-02 60.0% 1.253E-02 65.0% 1.380E-02 70.0% 1.599E-02 75.0% 1.860E-02 80.0% 1.973E-02 85.0% 2.544E-02 90.0% 3.037E-02 95.0% 4.468E-02 Max 1.161E-01	CRWMS M&O 2000, [153034] DTN:MO0006SPABDC01.007 [152837]
BDCF for ^{99}Tc	BDCF_Tc99 (mrem/yr)/(pCi/L)	Shifted Lognormal Geom.Mean=0.001494 8 Geom.S.D.=1.8423 Off Set=0.0021631	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{229}Th	BDCF_Th229 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=5.3919 Geom.S.D.=1.1666	CRWMS M&O 2000 [144054], Table 7, DTN: MO0003SPAABS07.006 [148876]
BDCF for ^{230}Th	BDCF_Th230 (rem/yr)/(pCi/L)	Cumulative Distribution Min 5.245E-03 5.0% 6.673E-03 10.0% 7.127E-03 15.0% 8.052E-03 20.0% 8.676E-03 5.0% 9.541E-03 30.0% 1.012E-02 35.0% 1.106E-02 40.0% 1.176E-02 45.0% 1.354E-02 50.0% 1.456E-02 55.0% 1.645E-02 60.0% 1.791E-02 65.0% 1.985E-02 70.0% 2.267E-02 75.0% 2.715E-02 80.0% 2.872E-02 85.0% 3.679E-02 90.0% 4.471E-02 95.0% 6.645E-02 Max 1.734E-01	CRWMS M&O 2000, [153034] DTN:MO0006SPABDC01.007 [152837]
BDCF for ^{232}U	BDCF_U232 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=2.0645 Geom.S.D.=1.150	CRWMS M&O 2000 [144054], Table 7, DTN: MO0003SPAABS07.006 [148876]

Table 6-105. Best Fit Parameters for BDCF Distributions (Continued)

Description	TSPA Parameter	Parameter value	Reference/DTN
BDCF for ^{233}U	BDCF_U233 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=0.38478 Geom.S.D.=1.1611	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{234}U	BDCF_U234 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=0.37690 Geom.S.D.=1.1617	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{236}U	BDCF_U236 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=0.35644 Geom.S.D.=1.1644	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]
BDCF for ^{238}U	BDCF_U238 (mrem/yr)/(pCi/L)	Lognormal Distribution Geom.Mean=0.35121 Geom.S.D.=1.1586	CRWMS M&O 2000 [144055], Table 9, DTN: MO0003SPAABS08.004 [148453]

Implementation

To obtain the yearly radiation dose to human receptors for the nominal or igneous intrusion groundwater release scenarios, the concentration of each radionuclide in the groundwater 20 km downgradient of the repository is multiplied by the corresponding BDCF in Table 6-105.[1]

The annual mass flux (g/yr) of each radionuclide species 20-km downgradient of the repository is calculated in the Saturated Zone component of the TSPA SR model (see Section 6.3.7). The annual groundwater usage is determined in the Groundwater Usage component of the TSPA-SR model (see Section 6.3.8.2). From these two calculations, the mass concentration (g/m^3) of each radionuclide in the biosphere, *Conc_Grams_20km_add_RNs*, is calculated as the annual mass flux of each radionuclide from the saturated zone (either *SZOUT.Water_to_Sink[species]* or (*pipe_20km.Water_to_Sink[species]*) divided by the annual groundwater usage (m^3/yr), *Annual_Water_Usage*. The latter is in the container, *Groundwater_Usage* (see Figure 6-182). The parameter *Conc_Grams_20km_add_RNs* is a one-dimensional array element, which calculates and stores each radionuclide's annual mass per unit volume of water in the groundwater. Table 6-106 lists the calculation equations for each species in the one-dimensional array, *Conc_Grams_20km_add_RNs.*

Table 6-106. Calculation of the Annual Mass of Each Radionuclide in the Groundwater Consumed

Radio-nuclide	TSPA Parameter	Parameter Value
^{243}Am	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Am243]+SZOUT.Water_to_Sink[Ic243])/Annual_Water_Usage
^{14}C	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[C14]/Annual_Water_Usage
^{129}I	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[I129]/Annual_Water_Usage

[1] This section does not discuss the dose for the human intrusion scenario, which would only differ from the scenarios discussed here in that the ^{137}Cs and ^{90}Sr dose would be added in.

Table 6-106. Calculation of the Annual Mass of Each Radionuclide in the Groundwater Consumed (Continued)

Radio-nuclide	TSPA Parameter	Parameter Value
^{242}Ic*	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{237}Ic	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{238}Ic	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{239}Ic	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{240}Ic	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{241}Ic	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{243}Ic	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{237}Np	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Np237]+SZOUT.Water_to_Sink[Ic237])/Annual_Water_Usage
^{231}Pa	Conc_Grams_20km_add_RNs	pipe_20km.Water_to_Sink[Pa231]/Annual_Water_Usage
^{239}Pu	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Pu239]+SZOUT.Water_to_Sink[Ic239])/Annual_Water_Usage
^{240}Pu	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Pu240]+SZOUT.Water_to_Sink[Ic240])/Annual_Water_Usage
^{99}Tc	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[Tc99]/Annual_Water_Usage
^{229}Th	Conc_Grams_20km_add_RNs	pipe_20km.Water_to_Sink[Th229]/Annual_Water_Usage
^{233}U	Conc_Grams_20km_add_RNs	pipe_20km.Water_to_Sink[U233]/Annual_Water_Usage
^{234}U	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[U234]/Annual_Water_Usage
^{235}U	Conc_Grams_20km_add_RNs	pipe_20km.Water_to_Sink[U235]/Annual_Water_Usage
^{236}U	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[U236]/Annual_Water_Usage
^{238}U	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[U238]/Annual_Water_Usage
^{242}Pu	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Pu242]+SZOUT.Water_to_Sink[Ic242])/Annual_Water_Usage
^{230}Th	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[Th230]/Annual_Water_Usage
^{232}Th	Conc_Grams_20km_add_RNs	SZOUT.Water_to_Sink[Th232]/Annual_Water_Usage
^{241}Am	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Am241]+SZOUT.Water_to_Sink[Ic241])/Annual_Water_Usage
^{238}Pu	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Pu238]+SZOUT.Water_to_Sink[Ic238])/Annual_Water_Usage
^{227}Ac†	Conc_Grams_20km_add_RNs	(pipe_20km.Water_to_Sink[Pa231]/Annual_Water_Usage)* Species.Specific_Activity[Pa231]/Species.Specific_Activity[Ac227]

Table 6-106. Calculation of the Annual Mass of Each Radionuclide in the Groundwater Consumed (Continued)

Radio-nuclide	TSPA Parameter	Parameter Value
^{137}Cs	Conc_Grams_20km_add_RNs	0.0{g/m3}
$^{210}Pb^{\dagger}$	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Th230]/Annual_Water_Usage)* Species.Specific_Activity[Th230]/Species.Specific_Activity[Pb210]
$^{226}Ra^{\dagger}$	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Th230]/Annual_Water_Usage)* Species.Specific_Activity[Th230]/Species.Specific_Activity[Ra226]
$^{228}Ra^{\dagger}$	Conc_Grams_20km_add_RNs	(SZOUT.Water_to_Sink[Th232]/Annual_Water_Usage)* Species.Specific_Activity[Th232]/Species.Specific_Activity[Ra228]
^{90}Sr	Conc_Grams_20km_add_RNs	0.0{g/m3}
^{232}U	Conc_Grams_20km_add_RNs	0.0{g/m3}
Col	Conc_Grams_20km_add_RNs	0.0{g/m3}

* The element symbol "Ic" stands for "irreversible colloid" and is a fictitious element required in the model to transport the irreversibly sorbed colloidal form of a species. For example, ^{242}Ic means ^{242}Pu irreversibly sorbed onto colloids.

† The concentrations of ^{227}Ac, ^{228}Ra, ^{226}Ra, and ^{210}Pb are calculated using secular equilibrium as discussed in Section 6.3.4.1.

The concentrations of ^{235}U, ^{231}Pa, ^{233}U, ^{229}Th, ^{230}Th, and ^{232}Th are not calculated from the output of *SZ_External*. Each of the radionuclides is a second, third, or fourth generation daughter product of one or more parent radionuclides decaying in the model. The transport of these components is calculated through the GoldSim code 1-D pipe pathways (see Sections 6.3.4.1 and 6.3.7 for more details). Without pipe pathways, which include radionuclide ingrowth from radionuclide decay, the TSPA-SR model would not be able to account for the contribution of these radionuclides due to ingrowth because the *SZ_External* only computes simple decay, not ingrowth. However, *SZ_External* is used to compute the breakthrough of first generation daughters, e.g., ^{239}Pu, but in order to account for ingrowth of their parents, the mass input to *SZ_External* must be "boosted" at each timestep by the theoretical maximum amount of decay of the parent. This is equal to the total decay over the maximum time the parent particles could be retained in the SZ, which is the total simulation time, *Run_Time*, minus the current time, *Etime* (see Section 6.3.4.1). One granddaughter, ^{234}U is also boosted, but only by one of its parents, ^{238}U, not by either its grandparent ^{242}Pu, or by its other parent ^{238}Pu. The latter is irrelevant because of its very short half-life, and the former has little effect.

The concentrations of ^{137}Cs, ^{90}Sr, ^{232}U, and Col are set to 0.0 g/m^3. Col is the name given to a fictitious species added to account for irreversible colloid formation from the dissolution of the waste form glass (see Section 6.3.4.6). The other three radionuclides do not contribute significantly to groundwater dose in the nominal or igneous intrusion scenarios because of their short half lives. This is consistent with the discussion in Section 6.3.4.1.

The activity concentration (Ci/m^3) present in the groundwater, *Conc_Curies_20km_add_RNs*, is the product of the mass concentration of each radionuclide, *Conc_Grams_20km_add_RNs*, and its specific activity. The specific activity of each nuclide is defined in the *Species* list, where the properties of each of the radionuclides are identified.

For each radionuclide the annual dose to the average member of the critical group, *Avg_Individual_Dose_add_RNs*, is the product of its activity concentration in the groundwater, *Conc_Curies_20km_add_RNs*, and its biosphere dose conversion factor, *BDCF_Nominal_GWPC*, which is within the container labeled *GWPC_Results* (see Figure 6-182 and Figure 6-184). Both *Conc_Curies_20km_add_RNs* and *BDCF_Nominal_GWPC* are one-dimensional arrays, as is the product array, *Avg_Individual_Dose_add_RNs*. The total dose rate, *Total_Dose_add_RNs*, is the sum of the individual radionuclide doses.

The parameter *Avg_Ind_Dose_with_Ash* in Figure 6-182 is the dose calculated for the scenario that includes volcanic disruptive events. The calculation is the sum of the dose to the receptor in the nominal scenario, *Avg_Individual_Dose*, plus the dose resulting from eruptive (direct) volcanic events, *Weight_Ash_Dose*. *Weight_Ash_Dose* is discussed in Section 6.3.9 of this document. The main difference between *Avg_Individual_Dose* and *Avg_Individual_Dose_add_RNs*, which was described above, is that the latter includes a few additional radionuclides whose dose is important over the 1,000,000-year time span (i.e., important to peak dose) but are not important to doses over the first 100,000 years, which is the time span used for most of the sensitivity cases in the TSPA-SR. These additional radionuclides are ^{230}Th, ^{226}Ra, and ^{210}Pb. The other difference between these two parameters is that *Avg_Individual_Dose* computes the doses of the irreversible colloids separately from the reversible colloids, e.g., ^{239}Ic and ^{239}Pu will each have a separate dose attributable to them. This computation is actually facilitated in the data element *Conc_Grams_20km* which separately computes the concentration of the irreversible and reversible species, unlike *Conc_Grams_20km_add_RNs* which adds them together. The model container *Colloid_Adjusted_Dose* then adds the contributions of the irreversible colloid species to their corresponding reversible colloid/solute species. Clearly, this is somewhat repetitious with *Conc_Grams_20km_add_RNs*, but these various data elements were created at different times for different reasons, the latter for the groundwater protection evaluation and the million-year dose and the former for the nominal and igneous scenarios for time spans of 100,000 years or less.

When the *Case_Selector* is set to 2, implying that the TSPA-SR model is being run for the igneous scenario, then the parameter *Intrusive_Event_Factor* is set to the *Event_Probability*. The latter is the annual frequency of an igneous intrusion times the total simulation time. In effect, the parameter *Intrusive_Event_Factor* reduces the dose by the scenario probability.

Table 6-107 summarized the calculated TSPA-SR model parameters in the Biosphere component of the TSPA-SR.

Results and Verification

The annual radiation dose from each radionuclide, *Avg_Individual_Dose_add_RNs*, for the nominal scenario, median-value simulation, without seismic cladding failure (SR00_038ne6), is presented in Figure 6-185.

Table 6-107. Details for TSPA Parameters in the Biosphere Component

Description	TSPA Parameter	Parameter Value/Other Inputs
radionuclide mass concentration in the biosphere, not including ^{230}Th, ^{226}Ra, ^{210}Pb, ^{232}Th, ^{228}Ra, and ^{238}Pu	Grams_Conc_20km	One-dimensional array of radionuclide mass concentration (g/m^3)
radionuclide activity concentration in the biosphere, not including ^{230}Th, ^{226}Ra, ^{210}Pb, ^{232}Th, ^{228}Ra, and ^{238}Pu	Conc_Curies_20km	One-dimensional array of radionuclide activity concentration (Ci/m^3)
Groundwater dose in mREM/yr, not including ^{230}Th, ^{226}Ra, ^{210}Pb, and ^{238}Pu	Avg_Individual_Dose	Curies_Conc_20km*BDCF (mREM/yr)
Sum of groundwater dose and dose from eruptive volcanic events in mREM/yr	Avg_Ind_Dose_with_Ash	Avg_Individual_Dose+ Weight_Ash_Dose
Sum of groundwater dose from all radionuclide species in mREM/yr (not including groundwater dose from ^{230}Th, ^{226}Ra, ^{210}Pb, and ^{238}Pu)	Total_Dose	Sumv(Avg_Individual_Dose)
Sum of groundwater dose and eruptive dose in mREM/yr from all species (not including groundwater dose from ^{230}Th, ^{226}Ra, ^{210}Pb, and ^{238}Pu)	Total_Dose_with_Ash	Sumv(Avg_Ind_Dose_with_Ash)
radionuclide mass concentration in the biosphere	Grams_Conc_20km_add_RNs	One-dimensional array of radionuclide mass concentration (g/m^3)
radionuclide activity concentration in the biosphere	Conc_Curies_20km_add_RNs	One-dimensional array of radionuclide activity concentration (Ci/m^3)
Groundwater dose in mREM/yr	Avg_Individual_Dose_add_RNs	Curies_Conc_20km*BDCF_Nominal_GWPC (mREM/yr)
Sum of groundwater dose and dose from eruptive volcanic events in mREM/yr	Total_Dose_add_RNs	Sumv(Avg_Individual_Dose_add_RNs)
Switch used for weighting groundwater doses by the event probability of igneous intrusion	Intrusive Event Factor	if(Case_Selector==2,Indirect_Release_Zone1.Event_Probability,1)

The Biosphere component consists of a series of calculations converting an annual mass flux of each radionuclide crossing the 20 km locus into an effective dose equivalent to the reference receptor. The first part of the conversion divides the annual mass flux of each radionuclide by the annual water usage to arrive at an average concentration of each radionuclides species in the groundwater. The concentration of each radionuclide species in the groundwater is then multiplied by the specific activity for each radionuclide. This calculation is verified here for two radionuclides, one with a mass flux returned from the 3-D model in *SZ_External* (from the cell *SZOUT* in the container *Saturated_Zone_Transport*) and one from the GoldSim code 1-D pipe pathway (from the pipe element *pipe_20km* in the container *Saturated_Zone_Transport*). Table 6-108 shows the calculation results for the concentration of radionuclides present in the groundwater for the median value simulation. The groundwater concentrations are extracted from the recorded results at 100,000 years, but they can be extracted for any model time step. The values in the column with the heading "Calculated Conc." are the manually calculated result using the appropriate parameter values for the 100,000 year time step. The values in the column with the heading "Model Result" are the values for *Conc_Grams_20km_add_RNs* extracted from the TSPA-SR model at 100,000 years.

Table 6-108. Mass Concentration of U^{238} and Th^{229} in Groundwater for the Median Value Simulation

TSPA Parameter	Radionuclide	Mass flux @ t=100,000 years (g/yr)	Groundwater Usage (m^3/yr)	Calculated Conc. (g/m^3)	Model Result, (g/m^3)
Conc_grams_20km _add_RNs	^{238}U	209.3553	2.39098E+06	8.7560E-05	8.7560E-05
Conc_grams_20km _add_RNs	^{229}Th	1.9730E-04	2.39098E+06	8.2517E-11	8.2517E-11

Once the mass concentration in the groundwater is calculated, the activity concentration in this groundwater can be determined. The radionuclide mass concentration in the groundwater multiplied by its specific activity yields the radionuclide activity concentration. The specific activity is calculated by the TSPA-SR model and recorded in the data element *Species*. Table 6-109 compares manual calculations with the model results. The values in the column with the heading "Calculated Conc." are the manually calculated result using the appropriate parameter values for the 100,000 year time step. The values in the column with the heading "Model Result" are the values for *Conc_Curies_20km_add_RNs* extracted from the TSPA-SR model at 100,000 years.

Table 6-109. Activity Concentration of U^{238} and Th^{229} in Groundwater for the Median Value Simulation

TSPA Parameter	Radionuclide	Conc_Grams_20k_ add_RNs @ t=100,000 years (g/m^3)	Species Activity (Ci/g)	Calculated Conc (Ci/m^3)	Model Result, (Ci/m^3)
Conc_Curies_20km _add_RNs	^{238}U	8.7560E-05	3.3679E-07	2.9489E-11	2.9489E-11
Conc_Curies_20km _add_RNs	^{229}Th	8.2517E-11	0.19761	1.6306E-11	1.6306E-11

Once the activity concentration in the groundwater is determined, the annual radiation dose from each radionuclide is calculated by multiplying the activity concentration by the appropriate BDCF. Table 6-110 compares the dose of ^{238}U and ^{229}Th calculated manually and the model result for the median value simulation.

Table 6-110. Annual Receptor Dose of U^{238} and Th^{229} for the Median Value Simulation

TSPA Parameter	Radionuclide	Conc_Curies_20km _add_RNs @ t=100,000 years (Ci/m^3)	Species BDCF (mrem/yr)/(pCi/l)	Calculated Value (mrem/yr)	Model Result (mrem/yr)
Avg_Individual_ Dose_add_RNs	^{238}U	2.9489E-11	0.35121	1.0357E-02	1.0357E-02
Avg_Individual_ Dose_add_Rns	^{229}Th	1.6306E-11	5.3919	8.7920E-02	8.7920E-02

The above calculations demonstrate that the Biosphere component of the TSPA-SR model receives and transmits data properly among different model components. In addition, the agreement between the manual calculations and model calculations also demonstrates that the Biosphere component of the TSPA-SR model is implemented correctly, and the receptor dose calculated is appropriate given the Saturated Zone output.

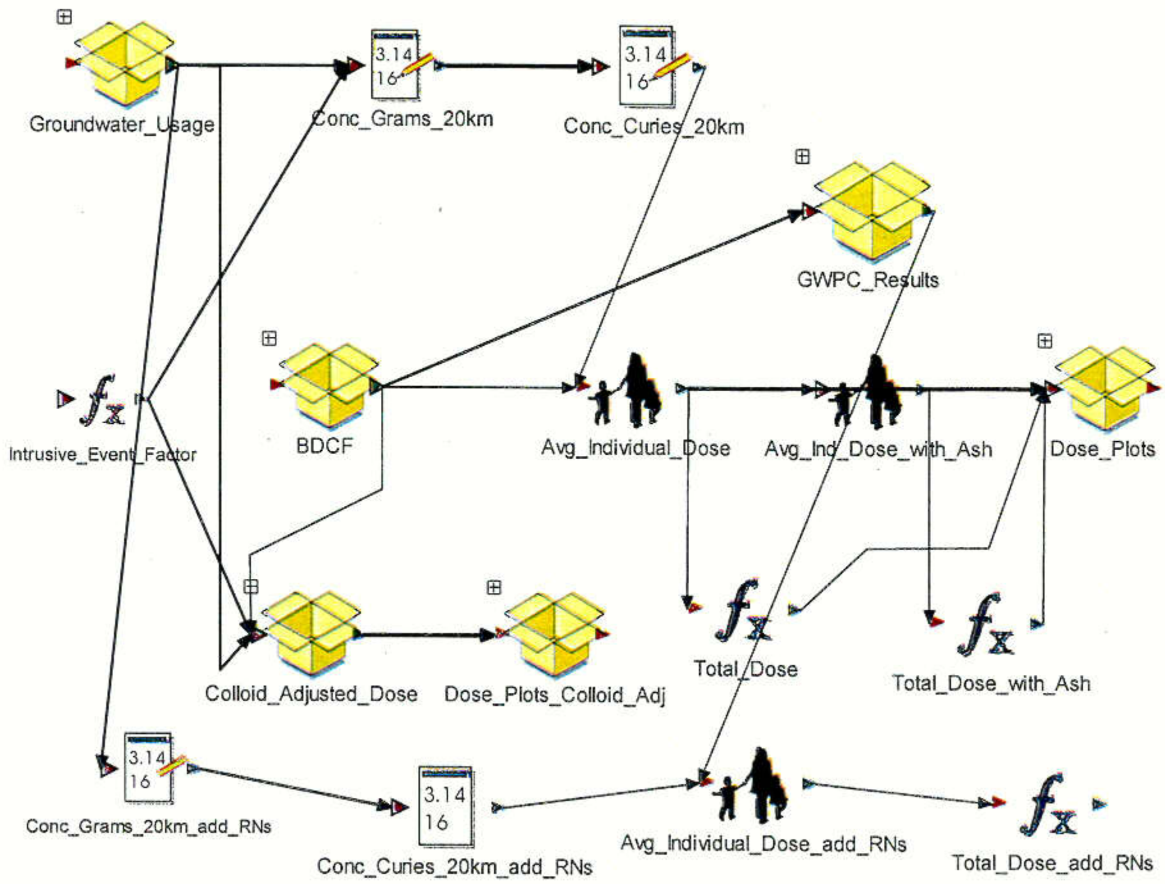


Figure 6-182. Graphical Illustration of the Biosphere Component

C21

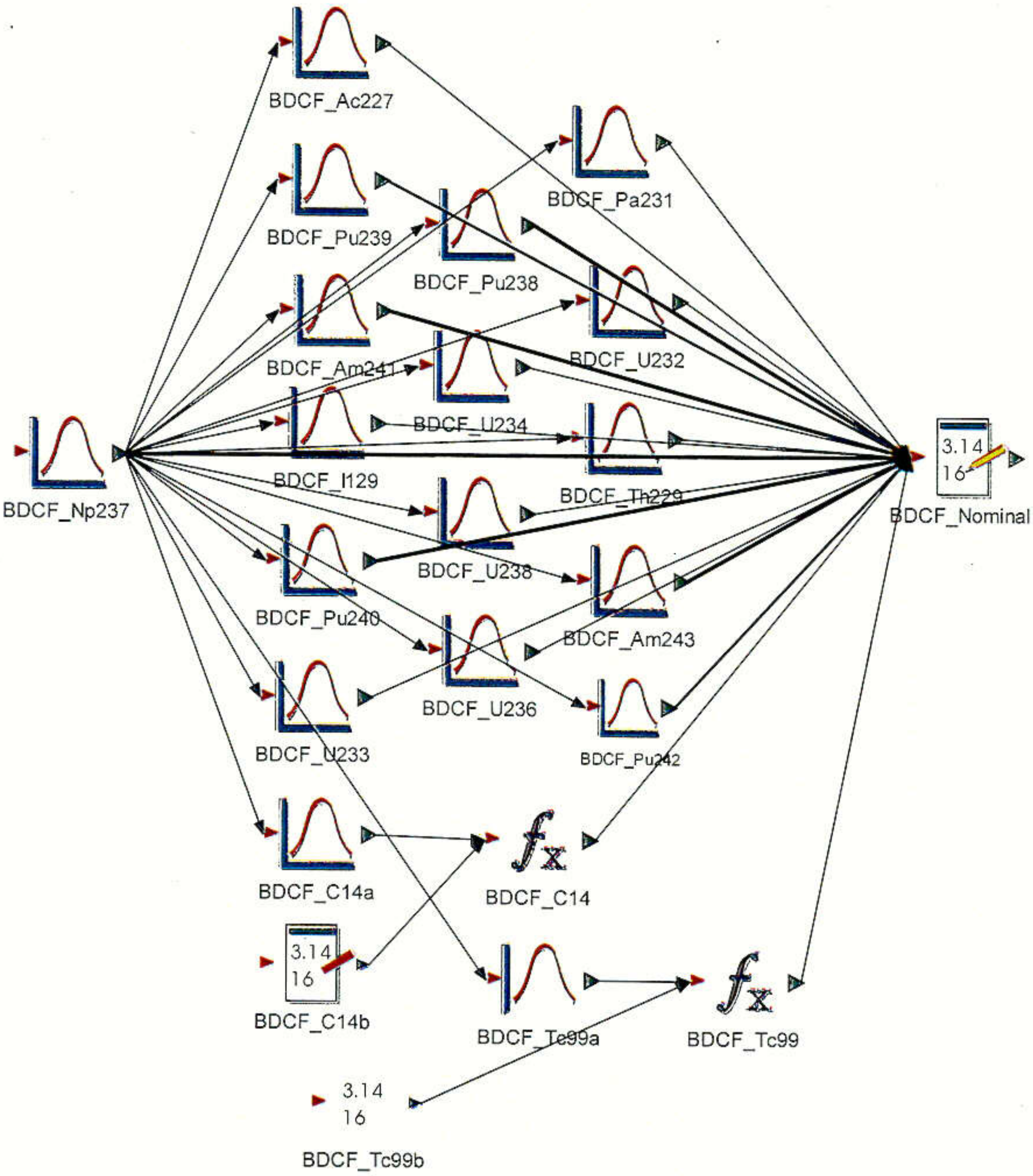


Figure 6-183. Biosphere Dose Conversion Factors in the TSPA-SR Model

C22

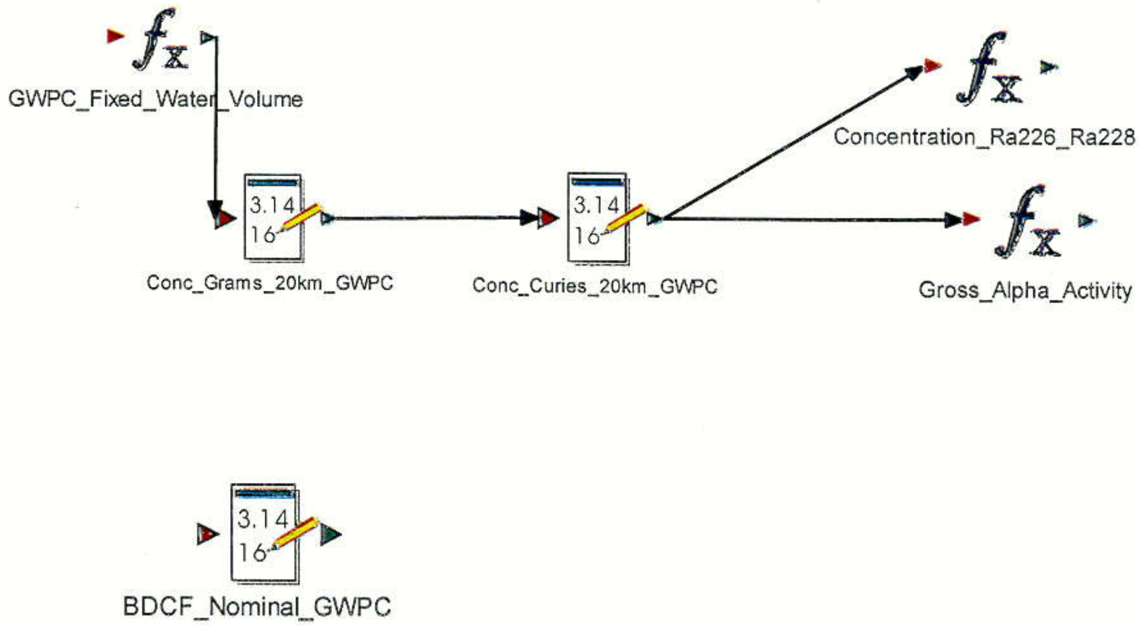


Figure 6-184. Graphical Illustration of the GWPC_Results Container: calculation of alpha_activity

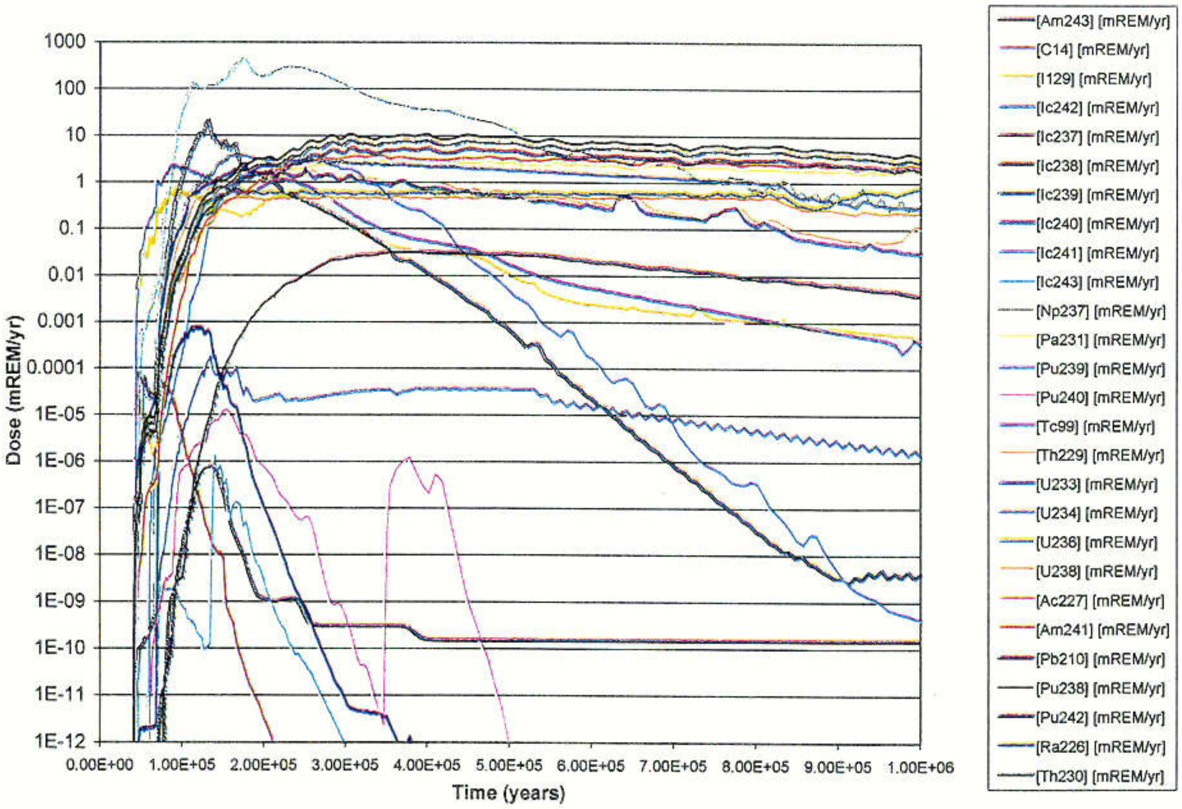


Figure 6-185. Dose Rate from Each Radionuclide for the Median Value of all Input Parameters

C23

6.3.8.2 Annual Groundwater Usage

Overview

The annual volume of groundwater used by the farming community containing the critical group is needed in the TSPA-SR analysis to determine radionuclide dilution. The radionuclide concentration in groundwater is derived by dividing the annual mass of radionuclides crossing the 20-km boundary by the annual volume of water used by the proposed farming community.

The reference receptor resides in a farming community of approximately 100 individuals, residing on 15 to 25 farms (CRWMS M&O 2000 [144056], Section 5.3.2). The analysis Groundwater Usage by the Proposed Farming Community (CRWMS M&O 2000 [144056], Section 7.2.2) concluded that total annual groundwater usage by the proposed farming community can be based on the number of farms specified to exist within the community and the current groundwater usage in the Amargosa Valley community.

Inputs to the TSPA Model

The groundwater usage AMR (CRWMS M&O 2000 [144056]) provides an estimate of the annual water usage that is a function of the number of farms. This estimate of groundwater usage is given in Table 6–111. The estimated number of farms ranges from 15 to 25 (DTN: MO0003SPASGU01.003 [151075]). A detailed discussion on the TSPA annual groundwater usage can be found in Groundwater Usage by the Proposed Farming Community (CRWMS M&O 2000 [144056]).

Table 6-111. Estimates of Annual Water Usage per Farm

Parameter	Value (acre-feet per year)
Mean Water Usage	96.92
Uncertainty In Mean Water Usage	37.77

DTN: MO0003SPASGU01.003 [151075]

Implementation

The groundwater usage AMR delineates the implementation of the groundwater usage analysis into the TSPA-SR model as follows (CRWMS M&O 2000 [144056], Section 7.2.3):

1. Select a random number ($R1$) distributed uniformly over the interval -1 to 1.

2. Determine the (estimated) average annual agricultural water usage (A) per farming unit for this realization ($A = mean + R1 \times uncertainty$) where the mean is 96.92 acre-feet/year, and the uncertainty is 37.77 acre-feet/year (mean and uncertainty values are from CRWMS M&O 2000 [144056], Table 6). This value represents an estimate of water usage over the 95-percentile confidence limit range of the mean value.

3. Select a random integer (R2) distributed uniformly from 15 to 25 representing the number of farms for this realization.

4. Determine (estimated) annual agricultural water usage (T) for the farming community by taking the product of *R2* and *A*. ($T = R2 \times A$). This total value *(T)* will now reflect the independent stochastic nature of both the individual farm water usage and the number of farms to be considered.

5. Convert *T* from acre-feet to m^3 to use in determining the average annual concentration of radionuclides in the groundwater used in the biosphere dose calculations.

Figure 6-186 illustrates this implementation in the TSPA-SR model.

Results and Verification

The result of the median value simulation for annual groundwater usage generated by the TSPA-SR model is 2.39×10^6 m^3/yr.

The groundwater usage AMR defines the equation for determining the annual groundwater usage for the Amargosa Valley farming community (CRWMS M&O 2000 [144056], Section 7.2.3) as $T = R2 \times A$ with $A = mean + R1 \times uncertainty$. *R1* is a uniform distribution ranging from -1 to 1, *R2* is a uniform distribution ranging from 15 to 25, the *mean* is 96.92 acre-feet/year, and the *uncertainty* is 37.77 acre-feet/year.

For the median value case, $R1 = 0$ and $R2 = 20$. Therefore, $A = 96.92 + 0 * 37.77 = 96.92$ acre-feet/year, and so $T = 20 * 96.92$ acre-feet/year = 1938.4 acre-feet/year. To convert acre-feet to m^3, multiply 1938.4 acre-feet/year by 1233.5 m^3/acre-feet, which gives the result 2.39E+06 m^3/yr. This is the same result calculated by GoldSim. The annual groundwater usage calculation does not receive input from any other models.

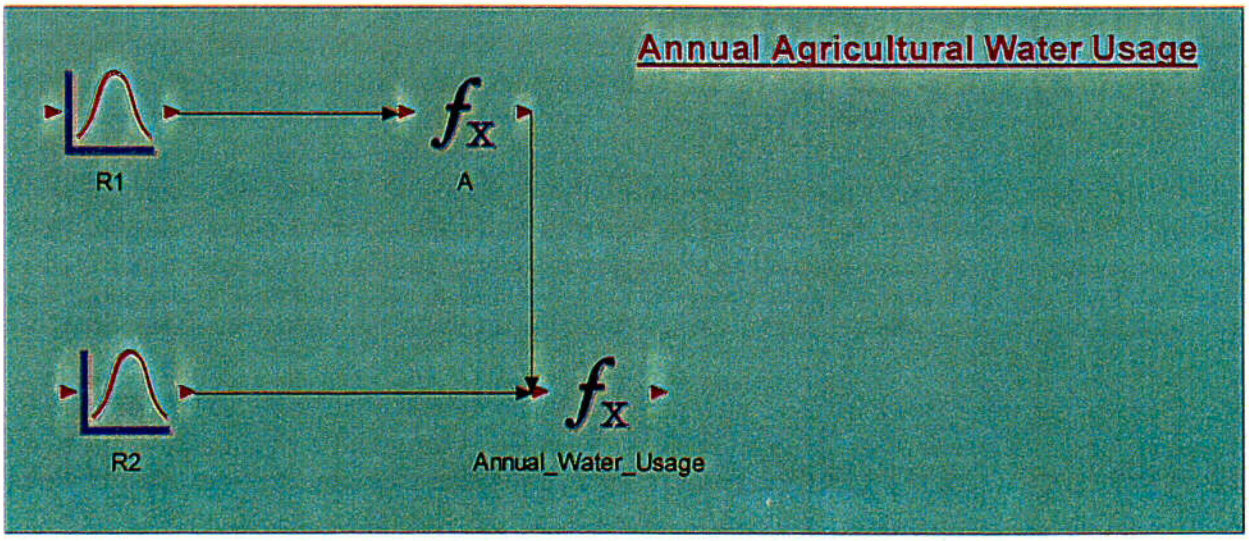


Figure 6-186. Overview of the TSPA Annual Groundwater Usage Model (Screen Capture from model file, refer to Attachment I for description of GoldSim Elements)

C24

6.3.9 Disruptive Events

The disruptive events analysis examines those events that can disrupt the repository and may occur sometime during the life of the repository, but their probability of occurring during the life of the repository is less than 100 percent. The disruptive events that are considered in the TSPA-SR are igneous events.

The igneous consequence models cover two different types of igneous events. The first event is a volcanic eruption through a section of the repository that entrains radionuclide-bearing waste in an ash plume that is then dispersed downwind and deposited on the ground. The second event is an igneous dike that intersects a section of the repository and partially or completely engulfs the intersected waste packages in magma. This waste is then available for transport in the groundwater. This event is known as an intrusive event. The volcanic release event is modeled using a code that has as inputs characteristics of the igneous event and the environment. The code then calculates the ash and waste dispersal and deposition. The intrusive event that results in groundwater transport of radionuclides is modeled using existing flow and transport models developed for analysis of the nominal performance scenario, using input assumptions and parameters specified in the AMR *Igneous Consequence Modeling for the TSPA-SR*. (CRWMS M&O 2000 [139563], Section 1.1)

Figure 6-187 illustrates the TSPA-SR representation of the disruptive events model. The following sections describe in detail the disruptive event models (volcanic release and intrusive release) and their implementation in the TSPA model.

6.3.9.1 Volcanic Release

Overview

The volcanic release event is a hypothetical volcanic eruption that intersects the repository. In this scenario an igneous dike rises to the repository level and intersects one or two waste containing drifts in the repository. The dike then continues to rise towards the surface and at some depth a conduit forms to the surface resulting in a volcanic eruption. Each conduit that reaches the surface contains a corresponding vent at the surface. Each dike may result in up to five conduits and corresponding vents being formed that could potentially intersect the waste containing drifts. The conduits erupt to the surface entraining any waste that was intersected in the eruptive rock and magma (ash) in the ash plume that is then dispersed downwind and deposited on the ground. (CRWMS M&O 2000 [139563], Section 4.1).

The atmospheric dispersion of ash and entrained waste is modeled using the ASHPLUME V1.4LV code. Characteristics about an eruptive volcanic event and the environment are input into ASHPLUME and ASHPLUME then models the atmospheric dispersion of the ash and waste until it settles on the ground. This code does not model the nature of the event, but only evaluates the ash and waste dispersal caused by an igneous event. The ASHPLUME code is run directly within GoldSim as a dynamic link library (DLL) (refer to Section 3). GoldSim is used to sample many of the ASHPLUME input parameters that are input into GoldSim as CDFs. The output of the ASHPLUME model is surface concentration (g/cm^2) of ash and waste at points on a defined grid. This result is used within GoldSim in conjunction with Biosphere Dose

Conversion Factors (BDCFs), the soil removal factor, and event probability to calculate the dose to the critical group from the volcanic release event.

The soil removal factor is the loss of soil due to normal erosion processes. The residence time of radionuclide contaminants in soils can have an influence on the relative contribution of the various contaminant exposure pathways to a receptor's total exposure. Therefore, the disruptive event analysis of the dose contribution attributable to contaminated volcanic ash considers the removal of ash and/or soil and associated radionuclides from the site due to erosion. A detailed discussion of soil erosion is given in the AMR *Evaluate Soil/Radionuclide Removal by Erosion and Leaching* (CRWMS M&O 2000 [136281]).

It is assumed that the ash from the eruption contributes its activity uniformly through the upper 15 cm (6 inches) of the ground surface (Section 5.3.9, BDCF assumptions 1, 2, and 3). It is further assumed that the groundwater is not contaminated (Section 5.3.9, BDCF assumption 3) in the eruptive scenario, so that any radionuclide accumulation is due to ash contaminated soils. The biosphere pathways will include ingestion, inhalation, and external exposure pathways, with the dominant pathway being the ingestion of contaminated foods.

The expected thickness of the ash is very small in comparison to the surface soil thickness, (Section 5.3.9, BDCF assumption #) and therefore no separate set of BDCFs for ash-like medium was required to support the TSPA. The ash-soil mixture was assumed to have properties of the bulk soil (Section 5.3.9, BDCF assumption 2). No radionuclide transport by air, surface water, biotic transport, or waste form degradation was considered (Section 5.3.9, BDCF assumptions 4, 5, and 7).

The AMR *Disruptive Event Biosphere Dose Conversion Factors Analysis* (CRWMS M&O 2000 [143378]) details the use of a comprehensive biosphere model, GENII-S, to describe the movement of radionuclides released to the environment to a receptor. This model was set up to calculate BDCFs for conditions following a volcanic eruption and to be consistent with the exposure scenario. The analysis showed that twelve radionuclides had sufficient releases to be considered relevant for disruptive event simulation. BDCFs for reasonable representation were obtained in statistical runs using Latin Hypercube sampling method and 160 trials (CRWMS M&O 2000 [143378], Section 6.0).

The Volcanic Release event is treated probabilistically. This implemented in the TSPA-SR model by having the Volcanic Release event occur on periodic basis, and then multiplying the event source term by probability of the event occurring over the period.

Inputs to the TSPA Model

The input parameters required for the volcanic release event model are listed below in Table 6-112 through Table 6-117. These input parameters were obtained from the AMR *Igneous Consequence Modeling for the TSPA-SR* (CRWMS M&O 2000 [139563], Section 6).

Table 6-112. Direct Volcanic Release Model Input Parameters

Parameter Name	Description	Parameter Value
Vent_Diameter	Vent Diameters	CDF (see Table 6-112)
Num_Pkgs_Hit_Drift_a	Number of Packages Hit per Drift	CDF (see Table 6-113)
Num_Drifts_Vent_a	Number of Drifts Hit per Vent	CDF (see Table 6-114)
Num_Drifts_Random	Random number used for selecting the number of drifts intersected per vent	uniform (0, 1)
Volcano_Period[a]	A code parameter used to specify the volcano period for simulations that include volcanic releases	31.25 years if running a simulation that includes volcanic release, otherwise 250 years
Nvents	Number of Vents Intersecting Waste	CDF (see Table 6-5)
EVENT_PROBABILITY[a]	Probability of an igneous intrusion into the repository	CDF (see Table 6-6)
Vent_1_5	Probability of >0 Vents	0.3643

DTN: SN0006T0502900.002 [150856]

NOTE: [a] This is not data, it is used to specify the timestep length the soilexp dll should use in determining the soil removal factor. This value must be equal to the smallest GoldSim timestep used.

Table 6-113. Vent Diameter and Initial Eruptive Velocity CDF

Vent Diameter (m)	Initial Eruption Velocity (cm/s)	CDF
15	2196	0
20	2940	0.0009
25	3685	0.0094
30	4429	0.0417
35	5174	0.1133
40	5918	0.2247
45	6662	0.3605
50	7407	0.5000
55	8151	0.6267
60	8895	0.7317
65	9640	0.8131
70	10384	0.8730
75	11128	0.9154
80	11873	0.9444
85	12617	0.9640
90	13362	0.9768
95	14106	0.9852
100	14850	0.9906
105	15595	0.9940
110	16339	0.9962
115	17083	0.9976
120	17828	0.9985
125	18572	0.9991
130	19316	0.9994
135	20061	0.9996

Table 6-113. Vent Diameter and Initial Eruptive Velocity CDF (Continued)

Vent Diameter (m)	Initial Eruption Velocity (cm/s)	CDF
140	20805	0.9998
145	21550	0.9999
150	22294	1

DTN: SN0006T0502900.002 [150856], Table 28 and 29

Table 6-114. Number of Packages Hit per Drift per Vent CDF Sampled on Vent Diameter

Conduit/Vent Diameter (m)	Number of Packages Hit per Drift	CDF
15	3	0
20	4	0.0009
25	5	0.0094
30	6	0.0417
35	7	0.1133
40	8	0.2247
45	9	0.3605
50	10	0.5000
55	11	0.6267
60	12	0.7317
65	13	0.8131
70	14	0.8730
75	15	0.9154
80	16	0.9444
85	17	0.9640
90	18	0.9768
95	19	0.9852
100	20	0.9906
105	21	0.9940
110	22.5	0.9962
115	22	0.9976
120	23	0.9985
125	24	0.9991
130	25	0.9994
135	26	0.9996
140	27	0.9998
145	28	0.9999
150	29	1

DTN: SN0006T0502900.002 [150856], Table 32

Table 6-115. Number of Drifts Hit per Vent CDF Sampled on Vent Diameter

Vent Diameter (m)	Number of Drifts Hit per Vent	CDF
15	1	1
20	1	1
25	1	1
30	1	1
35	1	1
40	1	1
45	1	1
50	1	1
55	1	1
60	1	1
65	1	1
70	1	1
75	1	1
80	1	0.94
85	1	0.874
90	1	0.808
95	1	0.742
100	1	0.675
105	1	0.609
110	1	0.543
115	1	0.477
120	1	0.411
125	1	0.344
130	1	0.278
135	1	0.212
140	1	0.146
145	1	0.079
150	1	0.013

DTN: SN0006T0502900.002 [150856], Table 33

Table 6-116. Number of Vents Intersecting Waste Drifts PDF

Number of Vents Intersecting Waste Drifts	PDF
1	0.8606
2	0.1232
3	0.0124
4	0.0019
5	0.0019

DTN: SN0006T0502900.002 [150856], Table 34

Table 6-117. Event Probability CDF

Frequency (yr^{-1})	CDF	Frequency (yr^{-1})	CDF
8.91300E-12	0	**1.51440E-09**	1.46238E-01
9.55500E-12	1.43798E-21	**1.69920E-09**	1.62961E-01
1.07200E-11	8.17170E-07	**1.90640E-09**	1.80037E-01
1.20280E-11	4.56989E-05	**2.13920E-09**	2.02377E-01
1.34960E-11	2.35820E-04	**2.40020E-09**	2.26202E-01
1.51440E-11	2.56313E-04	**2.69280E-09**	2.52160E-01
1.69920E-11	2.75967E-04	**3.02140E-09**	2.73543E-01
1.90640E-11	3.02403E-04	**3.39000E-09**	3.00125E-01
2.13920E-11	3.39549E-04	**3.80380E-09**	3.22626E-01
2.40020E-11	4.94913E-04	**4.26820E-09**	3.49793E-01
2.69280E-11	9.21993E-04	**4.78880E-09**	3.72131E-01
3.02140E-11	1.64580E-03	**5.37300E-09**	3.94744E-01
3.39000E-11	1.72342E-03	**6.02880E-09**	4.20510E-01
3.80380E-11	1.88771E-03	**6.76440E-09**	4.48491E-01
4.26820E-11	2.83633E-03	**7.58960E-09**	4.77451E-01
4.78880E-11	3.06944E-03	**8.51600E-09**	5.12031E-01
5.37300E-11	3.25478E-03	**9.55500E-09**	5.42171E-01
6.02880E-11	3.42150E-03	**1.07200E-08**	5.77740E-01
6.76440E-11	4.75419E-03	**1.20280E-08**	6.10988E-01
7.58960E-11	5.09198E-03	**1.34960E-08**	6.42040E-01
8.51600E-11	7.68157E-03	**1.51440E-08**	6.74726E-01
9.55500E-11	8.54944E-03	**1.69920E-08**	7.07644E-01
1.07200E-10	9.81512E-03	**1.90640E-08**	7.46418E-01
1.20280E-10	1.01521E-02	**2.13920E-08**	7.82949E-01
1.34960E-10	1.05781E-02	**2.40020E-08**	8.13209E-01
1.51440E-10	1.25944E-02	**2.69280E-08**	8.43987E-01
1.69920E-10	1.56189E-02	**3.02140E-08**	8.71226E-01
1.90640E-10	1.60594E-02	**3.39000E-08**	8.95031E-01
2.13920E-10	1.96527E-02	**3.80380E-08**	9.17926E-01
2.40020E-10	2.05899E-02	**4.26820E-08**	9.38355E-01
2.69280E-10	2.88359E-02	**4.78880E-08**	9.52171E-01
3.02140E-10	3.26450E-02	**5.37300E-08**	9.64017E-01
3.39000E-10	3.39195E-02	**6.02880E-08**	9.73003E-01
3.80380E-10	3.77270E-02	**6.76440E-08**	9.81574E-01
4.26820E-10	3.99210E-02	**7.58960E-08**	9.88703E-01
4.78880E-10	4.50245E-02	**8.51600E-08**	9.92176E-01
5.37300E-10	4.93939E-02	**9.55500E-08**	9.94948E-01
6.02880E-10	5.40941E-02	**1.07200E-07**	9.96120E-01
6.76440E-10	6.30893E-02	**1.20280E-07**	9.97221E-01
7.58960E-10	6.77652E-02	**1.34960E-07**	9.98862E-01
8.51600E-10	7.34717E-02	**1.51440E-07**	9.99479E-01

Table 6-117. Event Probability CDF (Continued)

Frequency (yr^{-1})	CDF	Frequency (yr^{-1})	CDF
9.55500E-10	8.77999E-02	**1.69920E-07**	9.99767E-01
1.07200E-09	9.41527E-02	**2.14860E-07**	9.99994E-01
1.20280E-09	1.14478E-01	**4.06574E-07**	1.00000E+00
1.34960E-09	1.32916E-01		

DTN: SN0006T0502900.002 [150856], Table 36 and 38

ASHPLUME requires a variety of input parameters that describe the characteristics of the eruption and of the environment. Given these inputs, ASHPLUME can then calculate the dispersion and the deposition of ash and waste. The input parameters required by ASHPLUME are defined in GoldSim, and when GoldSim calls the ASHPLUME DLL, the required input parameters are passed to ASHPLUME.

The ASHPLUME DLL was compiled to accept 47 input parameters. These input parameters and their values are listed in Table 6-118 through Table 6-125. As noted in Table 6-118 many of these inputs are only needed when ASHPLUME is run in stochastic mode. For the TSPA-SR model, the ASHPLUME DLL is run in deterministic mode, but GoldSim samples the input distributions during each timestep and passes them to ASHPLUME. Refer to the AMR *Igneous Consequence Modeling for the TSPA-SR* and the ASHPLUME user's manual for further information on these ASHPLUME input parameters (CRWMS M&O 2000 [139563], Section 6 and CRWMS M&O 1999 [150744], respectively).

The value of the parameter *Uran* is not a data value, this value is calculated by GoldSim based on the values sampled for the vent diameter, the number of vents intersecting the waste, the number of drifts hit per vent, and the number of packages hit per drift.

Table 6-118. Data Inputs used by ASHPLUME

Parameter Name	Description	Parameter Value
ichoice	Instructs ASHPLUME to stochastically sample numerous volcanoes (Input Value = 1) or one volcano (Input Value =2)	2
ipchar	Option to save particle size information at the dose points (1 for yes or 2 for no)	2
beta_dist	Ash dispersion controlling constant	CDF (see Table 6-118)
dmean	Ash mean particle diameter	CDF (see Table 6-119)
dsigma	Ash mean particle diameter standard deviation	CDF (see Table 6-120)
rhocut	Incorporation ratio (unitless)	0.3
Uran	Mass of waste released	calculated in GoldSim for each realization of a volcanic event
Udir	Wind Direction	PDF (see Table 6-121)
U	Wind Speed	CDF (see Table 6-122)
Erupt_Velocity	Initial Eruption Velocity	CDF (see Table 6-112)
Power	Event Power	CDF (see Table 6-123)

Table 6-118. Data Inputs used by ASHPLUME (Continued)

Parameter Name	Description	Parameter Value
Erupt_Volume	Event Volume	CDF (see Table 6-124)
denash	Ash settled density	1.0 g/cm^3
Xmin	Minimum grid location on x-axis	0 km
Xmax	Maximum grid location on x-axis	0 km
Ymin	Minimum grid location on y-axis	-20 km
Ymax	Maximum grid location on y-axis	0 km
Numptsx	Number of grids on x-axis	1
Numptsy	Number of grids on y-axis	1
Vlog_min[a]	Minimum value for log of event volume (km^3)	-2
Vlog_max[a]	Maximum value for log of event volume (km^3)	0
Powlog_min[a]	Minimum value for log of event power (watts)	9 .41
Powlog_max[a]	Maximum value for log of event power (watts)	11.55
Blog_min[a]	Minimum value for log of beta (unitless)	-2
Blog_max[a]	Maximum value for log of beta (unitless)	-0.3
Dmean_min[a]	Minimum value of log of mean ash particle diameter (cm)	-2
Dmean_med[a]	Mode value of log of mean ash particle diameter (cm)	-1
Dmean_max[a]	Maximum value of log of mean ash particle diameter (cm)	1
Dsigma_min[a]	Minimum value of log of ash particle standard deviation	0.1
Dsigma_max[a]	Maximum value of log of ash particle standard deviation	1
Ashden_min[a]	Ash particle density at minimum particle size	2.08 g/cm^3
Ashden_max[a]	Ash particle density at maximum particle size	1.04 g/cm^3
Ashrho_low[a]	Minimum value of ash log-diameter for density calculation	-3
Ashrho_hi[a]	Maximum value of ash log-diameter for density calculation	0
Fshape	Particle Shape Factor	0.5
Airden	Air Density	0.001117 g/cm^3
Airvis	Air Viscosity	0.0001758 g/m-s
C	C-Constant Relating Eddy Diffusivity and Particle Fall Time	400 $cm^2/s^{5/2}$
Dmax	Maximum particle diameter for transport	10 cm
Fdmin	Minimum waste particle size	0.0001 cm
Fdmean	Median waste particle size	0.002 cm
Fdmax	Maximum waste particle size	0.05 cm
hmin	Minimum height on eruption column considered in transport	0.001 km
Acutoff	Threshold limit on ash accumulation	1E-10 g/cm^2
rhocut_a[a]	Incorporation ratio (unitless)	0.3
Uran_min[a]	Minimum value of mass of waste released	4.2E+06
Uran_max[a]	Maximum value of mass of waste released	1.66E+08
Wind_Direction	Wind direction for median value simulations (replaces Udir)	-90

DTN: SN0006T0502900.002 [150856].

NOTE: [a] These parameters are only used when running stochastic simulations within ASHPLUME, and therefore are not generally used in the TSPA-SR simulations.

Table 6-119. Ash Dispersion Controlling Constant CDF

Ash Dispersion Controlling Constant	CDF	Ash Dispersion Controlling Constant	CDF
0.010	0	0.086	0.55
0.012	0.05	0.105	0.60
0.015	0.10	0.127	0.65
0.018	0.15	0.155	0.70
0.022	0.20	0.188	0.75
0.027	0.25	0.229	0.80
0.032	0.30	0.278	0.85
0.039	0.35	0.338	0.90
0.048	0.40	0.411	0.95
0.058	0.45	0.500	1
0.071	0.50		

Table 6-120. Ash Mean Particle Diameter CDF

Ash Mean Particle Diameter (cm)	CDF	Ash Mean Particle Diameter (cm)	CDF
0.0010	0	0.0126	0.55
0.0013	0.05	0.0158	0.60
0.0016	0.10	0.0200	0.65
0.0020	0.15	0.0251	0.70
0.0025	0.20	0.0316	0.75
0.0032	0.25	0.0398	0.80
0.0040	0.30	0.0501	0.85
0.0050	0.35	0.0631	0.90
0.0063	0.40	0.0794	0.95
0.0079	0.45	0.1000	1
0.0100	0.50		

Table 6-121. Ash Mean Particle Diameter Standard Deviation CDF

Ash Mean Particle Diameter Standard Deviation	CDF	Ash Mean Particle Diameter Standard Deviation	CDF
1.00	0	2.10	0.55
1.10	0.05	2.20	0.60
1.20	0.10	2.30	0.65
1.30	0.15	2.40	0.70
1.40	0.20	2.50	0.75
1.50	0.25	2.60	0.80
1.60	0.30	2.70	0.85
1.70	0.35	2.80	0.90
1.80	0.40	2.90	0.95
1.90	0.45	3.00	1
2.00	0.50		

Table 6-122. Wind Direction PDF

Wind Direction (Blowing Towards)	Wind Direction (ASHPLUME Degrees)	PDF
West-South	-150	0.073
South-West	-120	0.092
South	-90	0.109
South-East	-60	0.084
East-South	-30	0.047
East	0	0.063
East-North	30	0.101
North-East	60	0.218
North	90	0.126
North-West	120	0.037
West-North	150	0.027
West	180	0.023

Table 6-123. Wind Speed CDF

Wind Speed (cm/s)	CDF	Wind Speed (cm/s)	CDF
0.00	0	1131.78	0.8875
51.44	0.1190	1183.22	0.9097
102.89	0.1231	1234.67	0.9236
154.33	0.1329	1286.11	0.9324
205.78	0.1449	1337.56	0.9417
257.22	0.1718	1389.00	0.9505
308.67	0.2056	1440.45	0.9579
360.11	0.2403	1491.89	0.9634
411.56	0.2750	1543.33	0.9699
463.00	0.3208	1594.78	0.9755
514.44	0.3648	1646.22	0.9796
565.89	0.4194	1697.67	0.9833
617.33	0.4653	1749.11	0.9861
668.78	0.5157	1800.56	0.9889
720.22	0.5685	1852.00	0.9907
771.67	0.6208	1903.45	0.9921
823.11	0.6792	1954.89	0.9935
874.56	0.7250	2006.33	0.9949
926.00	0.7653	2057.78	0.9968
977.45	0.8060	2160.67	0.9986
1028.89	0.8352	2263.56	0.9991
1080.33	0.8653	2366.45	1

Table 6-124. Event Power CDF

Event Power (W)	CDF
1.000×10^{9}	0
7.943×10^{9}	0.143
1.259×10^{11}	0.286
3.162×10^{11}	0.429
5.012×10^{11}	0.572
1.000×10^{12}	0.715
6.310×10^{12}	0.858
6.310×10^{13}	1

Table 6-125. Eruptive Volume CDF

Eruptive Volume (km^3)	CDF	Eruptive Volume (km^3)	CDF
0.0020	0	0.0388	0.55
0.0026	0.05	0.0509	0.60
0.0034	0.10	0.0666	0.65
0.0045	0.15	0.0872	0.70
0.0059	0.20	0.1142	0.75
0.0077	0.25	0.1496	0.80
0.0101	0.30	0.1959	0.85
0.0132	0.35	0.2566	0.90
0.0173	0.40	0.3360	0.95
0.0227	0.45	0.4400	1
0.0297	0.50		

The annual soil depth reduction for the major soil series occurring in the vicinity of Lathrop Wells was estimated in the analysis and model report *Evaluate Soil/Radionuclide Removal by Erosion and Leaching* (CRWMS M&O 2000 [136281], Section 6.1.1). The calculated annual soil depth reduction rates are generally between 0.06 and 0.08 cm/yr. This range of soil depth reduction rates is used in the TSPA model as a uniform distribution ranging from 0.06 to 0.08 cm/yr, see Table 6-126. The soil removal rate is found by dividing the soil reduction rate by the soil depth (15 cm).

Table 6-126. Soil Reduction Rate

Parameter Name	Description	Parameter Value
Soil_Removal	Soil reduction rate	uniform (0.06, 0.08 cm/yr.)

DTN: SN9912T0512299.002 [136370]

The radionuclides that were determined to be important to dose during a direct volcanic release are listed in Table 6-127 along with their BDCFs. For further information on how these BDCFs were determined, refer to the AMR *Disruptive Event Biosphere Dose Conversion Factor Analysis* (CRWMS M&O 2000 [143378]).

Table 6-127. Biosphere Dose Conversion Factors for Direct Volcanic Release

	Volcanic Release Biosphere Dose Conversion Factors, rem/year per pCi/m^2					
	BDCF_Ash_Sr90	BDCF_Ash_Cs137	BDCF_Ash_Ac227	BDCF_Ash_Th229	BDCF_Ash_Pa231	BDCF_Ash_Pu238
Min.	6.33E-10	4.09E-10	1.07E-07	3.17E-08	3.89E-08	1.16E-08
5%	1.22E-09	4.62E-10	1.29E-07	3.86E-08	4.42E-08	1.33E-08
10%	1.80E-09	5.22E-10	1.42E-07	4.25E-08	5.07E-08	1.52E-08
15%	2.26E-09	5.47E-10	1.65E-07	5.02E-08	5.27E-08	1.60E-08
20%	2.56E-09	5.82E-10	1.91E-07	5.72E-08	5.94E-08	1.78E-08
25%	3.09E-09	6.02E-10	2.21E-07	6.81E-08	6.40E-08	1.92E-08
30%	3.61E-09	6.80E-10	2.56E-07	7.83E-08	7.42E-08	2.19E-08
35%	4.40E-09	7.03E-10	2.96E-07	9.15E-08	8.00E-08	2.40E-08
40%	4.85E-09	7.59E-10	3.37E-07	1.05E-07	9.14E-08	2.74E-08
45%	5.37E-09	8.13E-10	4.06E-07	1.27E-07	9.92E-08	3.00E-08
50%	6.06E-09	8.46E-10	4.63E-07	1.45E-07	1.12E-07	3.38E-08
55%	7.19E-09	9.38E-10	5.64E-07	1.76E-07	1.30E-07	3.94E-08
60%	8.28E-09	9.87E-10	6.55E-07	2.06E-07	1.45E-07	4.40E-08
65%	9.42E-09	1.09E-09	7.73E-07	2.44E-07	1.69E-07	5.11E-08
70%	1.14E-08	1.17E-09	9.06E-07	2.86E-07	1.94E-07	5.87E-08
80%	1.64E-08	1.64E-09	1.25E-06	3.96E-07	2.65E-07	7.93E-08
85%	2.02E-08	1.77E-09	1.50E-06	4.77E-07	3.13E-07	9.48E-08
90%	2.62E-08	2.24E-09	1.79E-06	5.67E-07	3.70E-07	1.12E-07
95%	3.90E-08	2.89E-09	2.13E-06	6.74E-07	4.35E-07	1.32E-07
100%	1.84E-07	1.59E-08	2.53E-06	8.02E-07	5.12E-07	1.55E-07
Min.	1.29E-08	1.29E-08	1.33E-08	1.33E-08	1.07E-08	2.23E-09
5%	1.47E-08	1.47E-08	1.51E-08	1.51E-08	1.37E-08	2.85E-09
10%	1.69E-08	1.68E-08	1.74E-08	1.73E-08	1.51E-08	3.17E-09
15%	1.77E-08	1.77E-08	1.80E-08	1.80E-08	1.66E-08	3.44E-09
20%	1.99E-08	1.99E-08	2.03E-08	2.03E-08	1.91E-08	3.91E-09
25%	2.13E-08	2.13E-08	2.19E-08	2.19E-08	2.21E-08	4.54E-09
30%	2.43E-08	2.43E-08	2.54E-08	2.54E-08	2.75E-08	5.62E-09
35%	2.66E-08	2.66E-08	2.71E-08	2.71E-08	3.05E-08	6.30E-09
40%	3.04E-08	3.04E-08	3.13E-08	3.12E-08	3.47E-08	7.24E-09
45%	3.33E-08	3.32E-08	3.40E-08	3.39E-08	4.07E-08	8.27E-09
50%	3.75E-08	3.74E-08	3.82E-08	3.81E-08	4.76E-08	9.67E-09
55%	4.37E-08	4.36E-08	4.46E-08	4.45E-08	5.67E-08	1.15E-08
60%	4.88E-08	4.87E-08	4.98E-08	4.97E-08	6.64E-08	1.35E-08
65%	5.66E-08	5.65E-08	5.78E-08	5.76E-08	7.73E-08	1.57E-08
70%	6.51E-08	6.50E-08	6.64E-08	6.62E-08	9.05E-08	1.83E-08
75%	7.96E-08	7.95E-08	8.13E-08	8.10E-08	1.12E-07	2.26E-08
80%	8.80E-08	8.78E-08	9.09E-08	9.06E-08	1.31E-07	2.64E-08
85%	1.05E-07	1.05E-07	1.07E-07	1.07E-07	1.50E-07	3.04E-08
90%	1.24E-07	1.24E-07	1.27E-07	1.27E-07	1.79E-07	3.62E-08
95%	1.46E-07	1.46E-07	1.49E-07	1.49E-07	2.12E-07	4.29E-08
100%	1.72E-07	1.72E-07	1.76E-07	1.75E-07	2.53E-07	5.11E-08

DTN: MO0008MWDVEB03.003 [151547]

Implementation

The implementation of the volcanic release scenario in the TSPA model is presented in this section. Figure 6-188 illustrates the basic implementation of the volcanic release scenario. The ASHPLUME DLL reads in information on the characteristics of the eruptive event, of the environmental conditions, and of the mass of the waste to be included in the event.

The container labeled *Input_Parameters* contains the data on the characteristics of the eruptive event (eruptive power, initial eruption velocity, etc.) and the environmental parameters (wind speed and direction, air density, etc.). These parameters are listed in Table 6-118 (ASHPLUME Input Parameters).

The *Input_Mass* container is used to calculate the mass of waste *(Uran)* involved in a direct release event. Figure 6-189 shows the contents of the *Input_Mass* container. The vent diameter *(Vent_Diameter)*, which is correlated to the eruption velocity, is used to determine the number of waste emplacement drifts that will be affected by the volcanic release event. This is shown in the upper part of Figure 6-189. Once the number of drifts per vent is determined, the total number of waste packages affected is calculated by multiplying the number of drifts per vent by the number of packages affected per drift and by the number of vents. The total inventory involved in the volcanic release event is calculated by determining the number of each type of waste package affected (CSNF and CDSP) and multiplying that result by their respective inventories. These calculations, which are shown schematically in Figure 6-189, provide to the ASHPLUME DLL the input value for the total mass of waste released (*Uran*).

Given the inputs described above, the ASHPLUME DLL then provides the waste deposition results (g/cm^2) at the location of interest. This is represented in Figure 6-190 as *XFuel*. The affected waste inventory is converted to a total mass of waste (*Uran*) for use by ASHPLUME (i.e., individual species are not tracked in ASHPLUME) and ASHPLUME returns the *XFuel* result. To determine dose from this result, the species composition of the *XFuel* result must be recovered so that the BDCFs can be applied. The function *Output_Mass* converts the ASHPLUME fuel surface concentration (g/cm^2) result *XFuel* to mass (g). The function *Output_Species* then converts back to the individual species amounts by multiplying the *Output_Mass* by the *Total_Inventory* and dividing by *Inventory_Sum* (*Total_Inventory* and *Inventory_Sum* were calculated earlier (refer to Figure 6-189. This result is then placed in a cell pathway (*Mass_Time*) where GoldSim will allow the inventory to decay with time.

The dose due the volcanic release event is then calculated within the *Direct_Release_Dose* container (shown in Figure 6-188). The graphical representation of this dose calculation is illustrated in Figure 6-191.

The dose due to an individual volcanic release event is calculated by multiplying the mass of each radionuclide in the cell pathway (*Mass_Time*) by its BDCF and by a soil removal factor. Since a volcano event is modeled to occur over each period (specified by the parameter *Volcano_Period*) the dose from all previous volcanic events has to be included in the calculation of the dose at the current time. The method for this is given as follows:

The peak concentration of waste in the ash changes with time due to radioactive decay and soil removal (erosion). The dose at time t_1 is equal to the peak concentration at time t_1 (C_1) times the BDCFs:

$$dose(t_1) = C_1 * BDCF \qquad \text{(Eq. 6-12)}$$

So, at some later time, t_2, the dose will have changed due to radioactive decay and soil removal effects. The dose at time t_2 can be calculated as follows:

$$dose(t_2) = BDCF * C_1 e^{-\lambda(t_2-t_1)} * e^{-k(t_2-t_1)} + C_2 * BDCF \qquad \text{(Eq.6-13)}$$

C_2 is the peak concentration at time t_2, λ is the radioactive decay constant (yr^{-1}), and k is the soil removal rate (yr^{-1}). Note that C_2 is defined in terms of C_1 as follows:

$$C_2 = C_1 e^{-\lambda(t_2-t_1)} \qquad \text{(Eq. 6-14)}$$

So Equation 6-2 can be written as:

$$dose(t_2) = BDCF * C_1 e^{-\lambda(t_2-t_1)} * e^{-k(t_2-t_1)} + BDCF * C_1 e^{-\lambda(t_2-t_1)} \qquad \text{(Eq. 6-15)}$$

If this is repeated for time t_3, the following equation results:

$$dose(t_3) = BDCF * C_1 e^{-\lambda(t_3-t_1)} * e^{-k(t_3-t_1)} + BDCF * C_1 e^{-\lambda(t_3-t_1)} * e^{-k(t_3-t_2)} + BDCF * C_3 \qquad \text{(Eq. 6-16)}$$

C_3 can be defined in terms of C_1 as follows:

$$C_3 = C_1 e^{-\lambda(t_3-t_1)} \qquad \text{(Eq. 6-17)}$$

Making this substitution for C_3 in Equation 6-5 and simplifying yields the following equation for dose at any time, t_n:

$$dose(t_n) = BDCF * C_1 e^{-\lambda(t_n-t_1)} \sum_{i=1}^{n} e^{-k(t_n-t_i)} \qquad \text{(Eq. 6-18)}$$

If equal sized time-steps are used, the above equation for dose at any time, t_n, can be written as follows:

$$dose(t_n) = BDCF * C_1 e^{-\lambda(t_n-t_1)} \sum_{i=1}^{n} e^{-ik\Delta t} \qquad \text{(Eq. 6-19)}$$

The dll element soilexp is called by GoldSim to calculate the soil removal factor (the summation portion of Equation 6-8). See Section 3 and Attachment III for more information on the soilexp dll.

The BDCFs for the radionuclides considered for direct volcanic releases were implemented in the TSPA-SR as a set of stochastic distributions (these are listed in Table 6-127). The probability-weighted dose is obtained by multiplying the dose by the probability of occurrence of a volcanic release event that intersects the repository, by the probability that more than zero vents occur, and by the volcano period (for simulations that include direct volcanic release, the volcano period is set to the smallest GoldSim timestep used in the simulation).

Results and Verification

The total mass of inventory affected by a volcanic release event (*Uran*, see Figure 6-189) for a median value simulation was calculated by GoldSim as 6.08E+07 g. To check this result, note that median value of the cumulative distribution defined for the vent diameter is 50 m. Due to drift spacing, for a 50 m vent only one drift can possibly be intersected by the vent (refer to Table 6-115), so *Num_Drifts_Vent_a*, *Num_Drifts_Vent_b*, and *Num_Drifts_Vent* are all equal to one. The cumulative distribution defined for the number of packages that are hit per drift (*Num_Pkgs_Hit_Drift_a*) has a median value of 10, and the number of vents intersecting the waste is one (see Table 6-114 and Table 6-116). Therefore, the total number of packages hit is 10. To find the number of CSNF packages hit, multiply the fraction of the total number of packages that are CSNF packages by the number of packages hit and round to the nearest integer (thus the remaining packages are CDSP packages). The 7,860 CSNF packages plus the 3,910 CDSP packages gives a total of 11,770 packages. This gives 7 CSNF packages hit, and 3 CDSP packages hit. Table 6-128 shows the inventory per CDSP package (HLW + DSNF) and per CSNF package and the total affected inventory (i.e., multiply the HLW and DSNF values by 3, and the CSNF values by 7 and add them) (refer to Section 6.3.4.1 for information on the radionuclide inventory). The total at the bottom of Table 6-128 matches the result given by GoldSim.

Table 6-128. Mass of Waste Released in a Direct Volcanic Release for a Median Value Simulation

Radionuclide	HLW (g/pkg)	DSNF (g/pkg)	CSNF (g/pkg)	Total (g)
Am243	1.55E+00	1.68E+00	1.29E+03	9.04E+03
C14	7.11E-03	6.63E-01	1.37E+00	1.16E+01
I129	4.41E+01	8.08E+01	1.80E+03	1.30E+04
Ic242	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Ic237	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Ic238	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Ic239	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Ic240	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Ic241	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Ic243	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Np237	1.78E+02	4.26E+02	4.74E+03	3.50E+04
Pa231	7.44E-01	3.02E-01	9.87E-03	3.21E+00
Pu239	3.52E+03	2.13E+03	4.38E+04	3.24E+05
Pu240	3.39E+02	4.55E+02	2.09E+04	1.49E+05
Tc99	7.01E+02	4.53E+02	7.68E+03	5.72E+04
Th229	3.79E-03	2.46E-02	0.00E+00	8.52E-02
U233	1.02E+01	1.98E+02	7.00E-02	6.25E+02

Table 6-128. Mass of Waste Released in a Direct Volcanic Release for a Median Value Simulation (Continued)

Radionuclide	HLW (g/pkg)	DSNF (g/pkg)	CSNF (g/pkg)	Total (g)
U233	1.02E+01	1.98E+02	7.00E-02	6.25E+02
U234	3.39E+01	2.77E+02	1.83E+03	1.37E+04
U235	1.56E+03	1.74E+04	6.28E+04	4.96E+05
U236	3.65E+01	5.27E+03	3.92E+04	2.90E+05
U238	7.86E+05	4.67E+05	7.92E+06	5.92E+07
Pu242	6.25E+00	1.15E+01	5.41E+03	3.79E+04
Th230	7.00E-03	1.75E-02	1.84E-01	1.36E+00
Th232	1.59E+04	1.38E+04	0.00E+00	4.14E+04
Am241	6.03E+01	1.13E+02	1.09E+04	7.68E+04
Pu238	5.69E+01	8.79E+01	1.51E+03	1.10E+04
Ac227	4.36E-04	1.05E-04	3.09E-06	1.64E-03
Cs137	4.04E+02	5.52E+02	5.34E+03	4.02E+04
Pb210	1.31E-07	1.38E-08	0.00E+00	4.34E-07
Ra226	1.52E-05	2.21E-06	0.00E+00	5.22E-05
Ra228	6.51E-06	6.46E-06	0.00E+00	1.94E-05
Sr90	2.67E+02	3.01E+02	2.24E+03	1.74E+04
U232	7.64E-04	1.37E-01	1.01E-02	4.84E-01
Col	1.00E+06	0.00E+00	0.00E+00	0.00E+00
Total				6.08E+07

Figure 6-192 shows the individual radionuclide doses (*Ash_Dose*) from a direct volcanic release for the first 5000 years of the simulation. Note that these doses are not probability weighted. Table 6-129 gives the dose for each of the radionuclides shown in Figure 6-192 for the first 1,000 years.

The doses given in Table 6-129 can be checked by noting that the dose can be calculated by multiplying the surface concentration (mass per unit area) of the radionuclide by its BDCF and by the soil removal factor at the timestep of interest. The BDCFs for those radionuclides important to dose following a direct volcanic release are given in Table 6-126. The results at the 500-year timestep will be checked to verify that the model is functioning correctly. The ASHPLUME dll is a qualified code, and therefore the results will not be checked here (refer to Section 3). Similarly, the Software Routine Report (SRR) for the soilexp dll is presented in Attachment II, and therefore the results will not be checked here.

The cell pathway *Mass_Time* (shown in Figure 6-189) contains the time histories of the mass of each radionuclide (this is actually the surface concentration because the ASHPLUME result *XFuel* was divided by 1 cm^2 prior to input into the *Mass_Time* cell). These time histories are shown in Table 6-130 for the timestep at 500 years. The soil removal factor at 500 years is given by the data element, *Soil_Removal_Factor,* as 6.5425. The BDCFs for a median value case are given by *BDCF_Ash,* and are listed in Table 6-131 (converted to (mrem/yr)/(g/m^2)). The conversion factor for m^2 to cm^2 is 10,000 cm^2 = 1 m^2.

Table 6-129. Dose from Direct Volcanic Release for the First 1,000 years (not probability weighted)

	Dose (mrem/year)											
Time (yr)	**Am243**	**Pa231**	**Pu239**	**Pu240**	**Th229**	**U233**	**Am241**	**Pu238**	**Ac227**	**Cs137**	**Sr90**	**U232**
0.00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
31.25	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
62.50	1.95E+01	4.87E-03	2.13E+02	3.58E+02	2.05E-03	1.67E-02	2.60E+03	1.11E+03	1.96E-02	1.99E+02	9.18E+02	7.87E-02
93.75	3.62E+01	9.12E-03	3.97E+02	6.66E+02	5.08E-03	3.11E-02	4.61E+03	1.61E+03	3.74E-02	1.82E+02	8.17E+02	1.07E-01
125.00	5.06E+01	1.28E-02	5.56E+02	9.29E+02	8.87E-03	4.36E-02	6.14E+03	1.76E+03	5.29E-02	1.25E+02	5.46E+02	1.10E-01
156.25	6.29E+01	1.61E-02	6.93E+02	1.16E+03	1.33E-02	5.45E-02	7.28E+03	1.72E+03	6.64E-02	7.69E+01	3.25E+02	1.00E-01
187.50	7.35E+01	1.89E-02	8.11E+02	1.35E+03	1.81E-02	6.38E-02	8.11E+03	1.57E+03	7.82E-02	4.42E+01	1.81E+02	8.56E-02
218.75	8.26E+01	2.14E-02	9.13E+02	1.51E+03	2.33E-02	7.20E-02	8.69E+03	1.39E+03	8.85E-02	2.45E+01	9.75E+01	7.05E-02
250.00	9.03E+01	2.35E-02	1.00E+03	1.66E+03	2.87E-02	7.90E-02	9.07E+03	1.19E+03	9.74E-02	1.32E+01	5.10E+01	5.65E-02
312.50	1.03E+02	2.71E-02	1.14E+03	1.88E+03	3.99E-02	9.04E-02	9.38E+03	8.30E+02	1.12E-01	3.63E+00	1.33E+01	3.45E-02
375.00	1.11E+02	2.98E-02	1.25E+03	2.04E+03	5.15E-02	9.90E-02	9.27E+03	5.54E+02	1.24E-01	9.58E-01	3.30E+00	2.01E-02
437.50	1.18E+02	3.20E-02	1.32E+03	2.16E+03	6.30E-02	1.06E-01	8.92E+03	3.60E+02	1.32E-01	2.46E-01	8.00E-01	1.14E-02
500.00	1.22E+02	3.37E-02	1.38E+03	2.24E+03	7.45E-02	1.10E-01	8.43E+03	2.30E+02	1.39E-01	6.20E-02	1.90E-01	6.38E-03
562.50	1.26E+02	3.50E-02	1.42E+03	2.29E+03	8.57E-02	1.14E-01	7.87E+03	1.45E+02	1.45E-01	1.54E-02	4.47E-02	3.52E-03
625.00	1.28E+02	3.61E-02	1.45E+03	2.33E+03	9.68E-02	1.17E-01	7.29E+03	9.05E+01	1.49E-01	3.81E-03	1.04E-02	1.92E-03
687.50	1.29E+02	3.70E-02	1.47E+03	2.36E+03	1.08E-01	1.19E-01	6.70E+03	5.62E+01	1.53E-01	9.35E-04	2.41E-03	1.04E-03
750.00	1.30E+02	3.77E-02	1.49E+03	2.37E+03	1.18E-01	1.21E-01	6.14E+03	3.48E+01	1.56E-01	2.28E-04	5.55E-04	5.64E-04
812.50	1.30E+02	3.83E-02	1.50E+03	2.37E+03	1.29E-01	1.23E-01	5.60E+03	2.14E+01	1.59E-01	5.56E-05	1.27E-04	3.04E-04
875.00	1.31E+02	3.89E-02	1.51E+03	2.37E+03	1.39E-01	1.24E-01	5.10E+03	1.32E+01	1.61E-01	1.35E-05	2.92E-05	1.64E-04
937.50	1.30E+02	3.93E-02	1.51E+03	2.37E+03	1.49E-01	1.25E-01	4.64E+03	8.10E+00	1.63E-01	3.28E-06	6.68E-06	8.78E-05
1000.00	1.30E+02	3.98E-02	1.52E+03	2.36E+03	1.59E-01	1.26E-01	4.21E+03	4.97E+00	1.65E-01	7.93E-07	1.53E-06	4.71E-05

Table 6-130. Surface Concentration of Radionuclides Following a Direct Volcanic Release (at the 500-year timestep)

Radionuclide	Surface Concentration (g/cm^2)
Am243	2.42E-10
Pa231	9.56E-14
Pu239	8.94E-09
Pu240	3.95E-09
Th229	3.92E-14
U233	1.77E-11
Am241	9.65E-10
Pu238	5.98E-12
Ac227	6.27E-17
Cs137	1.29E-14
Sr90	3.51E-15
U232	8.97E-17

Table 6-131. BDCFs for a Median Value Simulation

Radionuclide	BDCF (mrem/yr)/(g/m^2)
Am243	7.75E+06
Pa231	5.38E+06
Pu239	2.36E+06
Pu240	8.66E+06
Th229	2.90E+07
U233	9.52E+04
Am241	1.34E+08
Pu238	5.87E+08
Ac227	3.40E+10
Cs137	7.32E+07
Sr90	8.29E+08
U232	1.09E+09

Calculating the dose from each radionuclide at the 500-year timestep as described above gives the results listed in Table 6-132. Multiplying the values in Table 6-132 by 0.3643 (the probability of having greater than zero vents; parameter *Vent_1_5*), 8.7972E-09 (parameter *EVENT_PROBABILITYa*), 31.25 years (parameter *Volcano_Period*), and 1/yr (appropriate unit conversion factor) yields the values in Table 6-133. Comparing the results in Table 6-132 with those calculated in GoldSim from Table 6-129 at 500 years (see Figure 6-193), as well as the comparison in Table 6-133 verifies that the model is calculating the direct volcanic release dose correctly.

Table 6-132. Dose from Direct Volcanic Release at 500 Years (not probability weighted)

Radionuclide	Dose (mrem/year)
Am243	1.22E+02
Pa231	3.37E-02
Pu239	1.38E+03
Pu240	2.24E+03
Th229	7.45E-02
U233	1.10E-01
Am241	8.43E+03
Pu238	2.30E+02
Ac227	1.39E-01
Cs137	6.20E-02
Sr90	1.90E-01
U232	6.38E-03

Table 6-133. Dose from Direct Volcanic Release at 500 Years (probability weighted)

Radionuclide	Dose (mrem/year) hand calculation	Dose (mrem/year) from GoldSim
Am243	1.23E-10	1.23E-10
Pa231	3.37E-14	3.37E-14
Pu239	1.38E-09	1.38E-09
Pu240	2.24E-09	2.24E-09
Th229	7.46E-14	7.46E-14
U233	1.11E-13	1.11E-13
Am241	8.44E-09	8.44E-09
Pu238	2.30E-10	2.30E-10
Ac227	1.40E-13	1.40E-13
Cs137	6.21E-14	6.21E-14
Sr90	1.90E-13	1.90E-13
U232	6.39E-15	6.39E-15

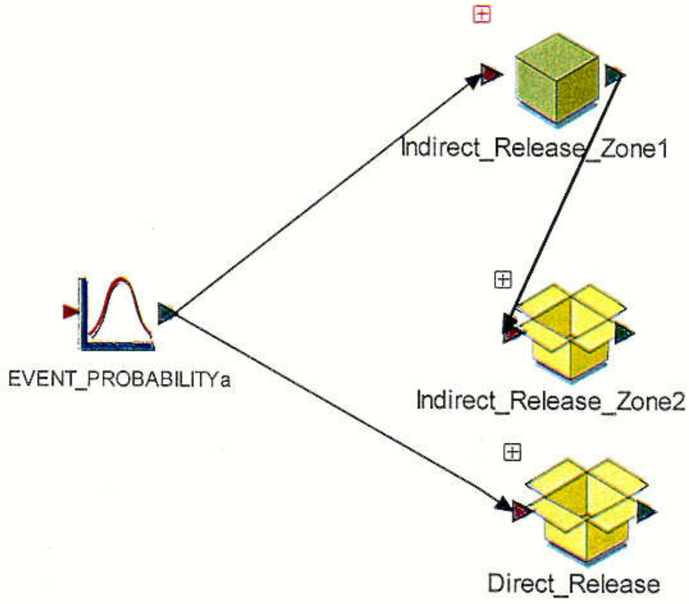


Figure 6-187. Disruptive Events Model within GoldSim

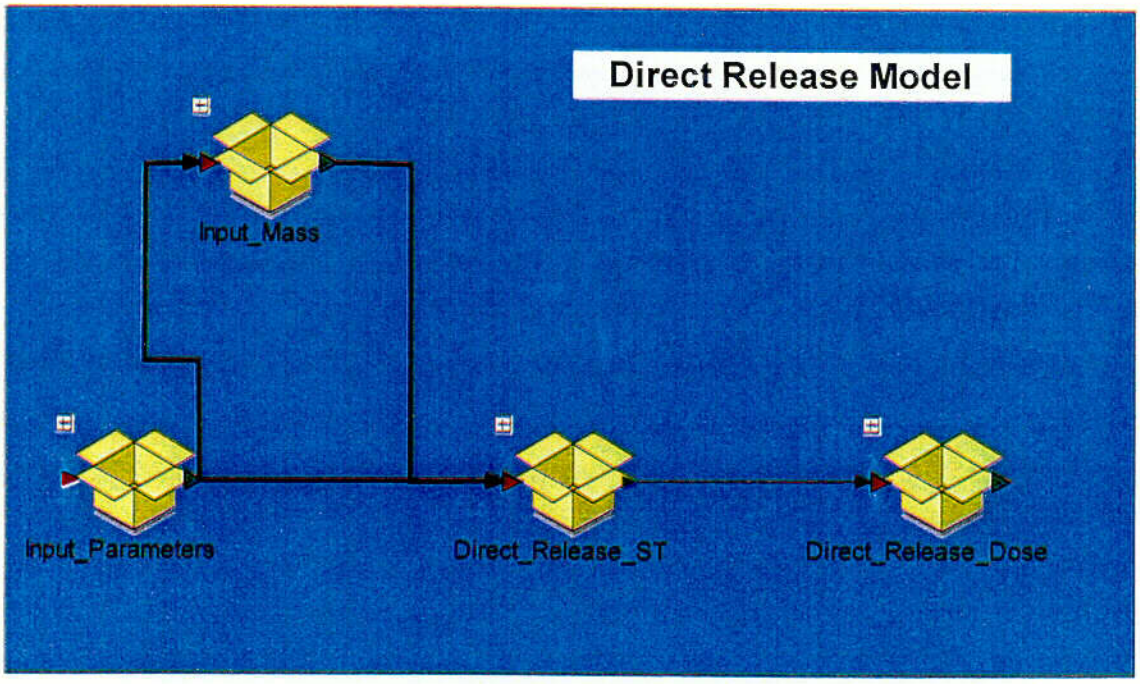


Figure 6-188. Implementation of the Volcanic Release Event in the TSPA Model

C25

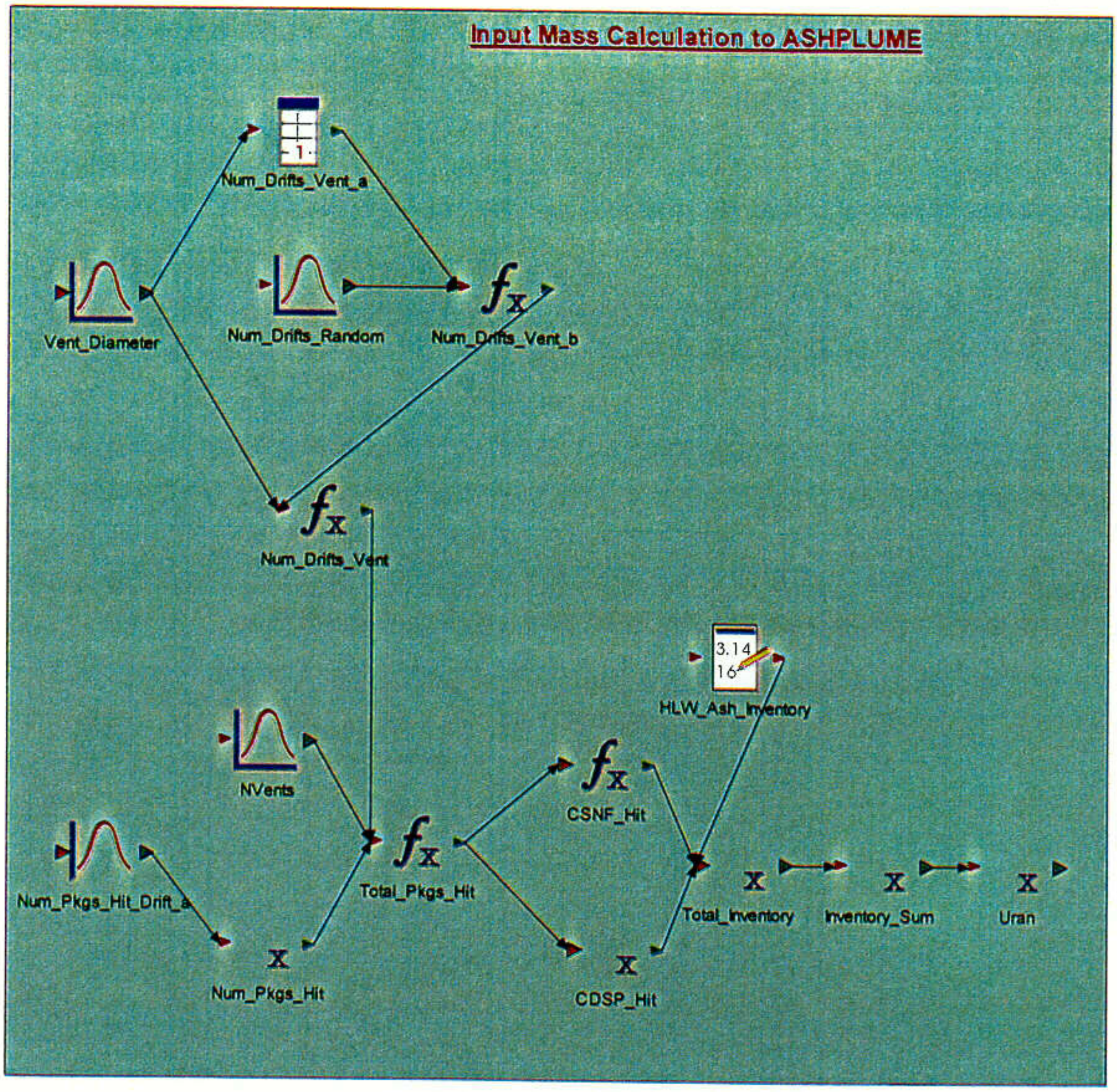


Figure 6-189. Calculations for Determining the Total Waste Inventory Affected by a Volcanic Release Event

C26

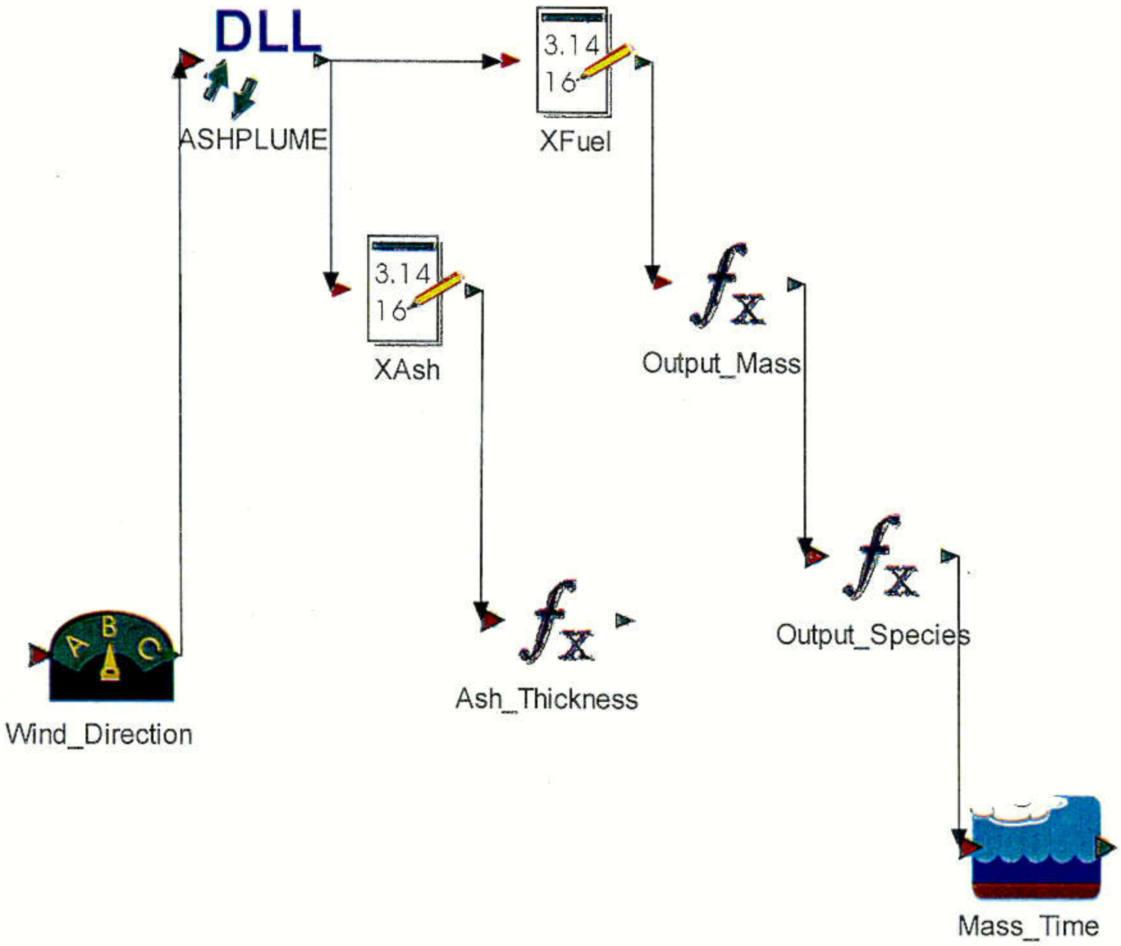


Figure 6-190. Calculation of the Source Term from Direct Volcanic Release

C27

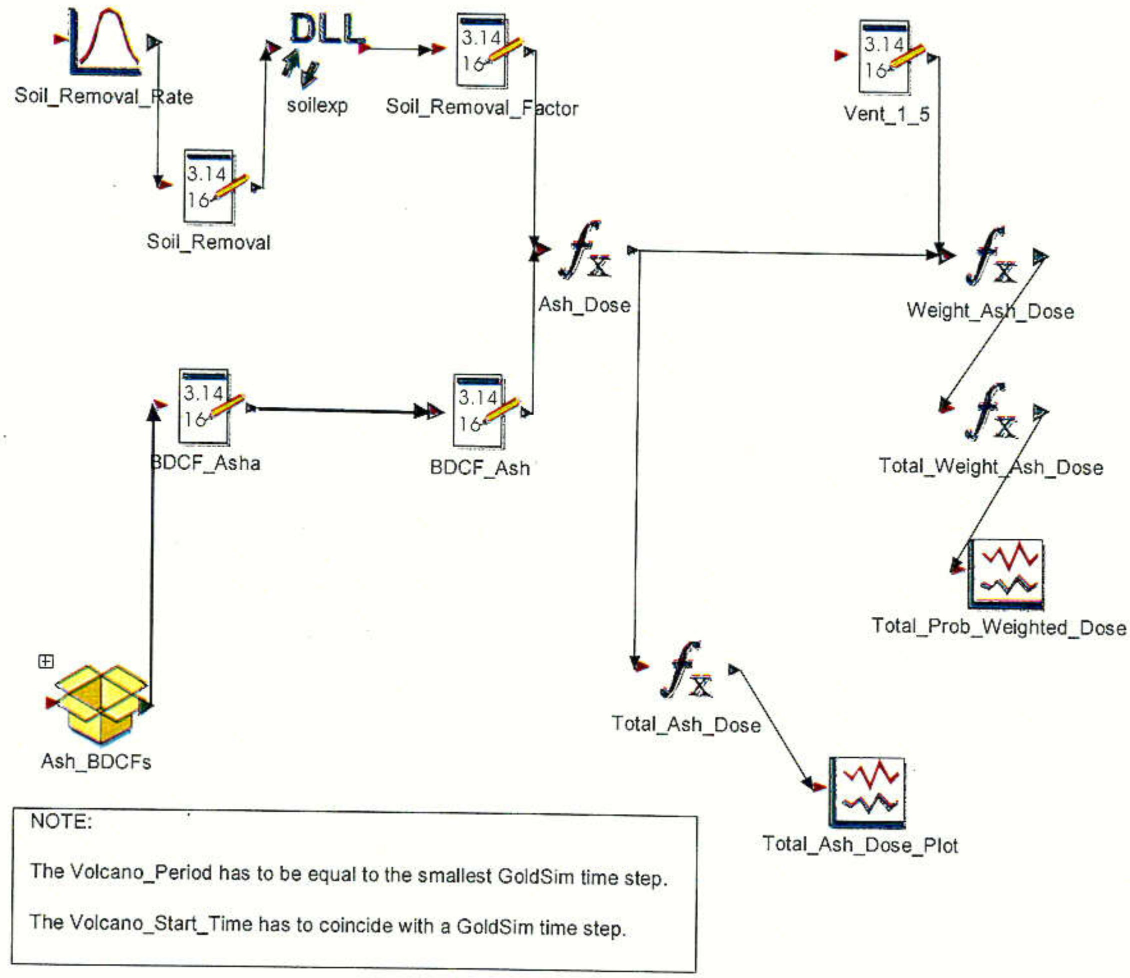


Figure 6-191. Volcanic Direct Release Dose Calculation

C28

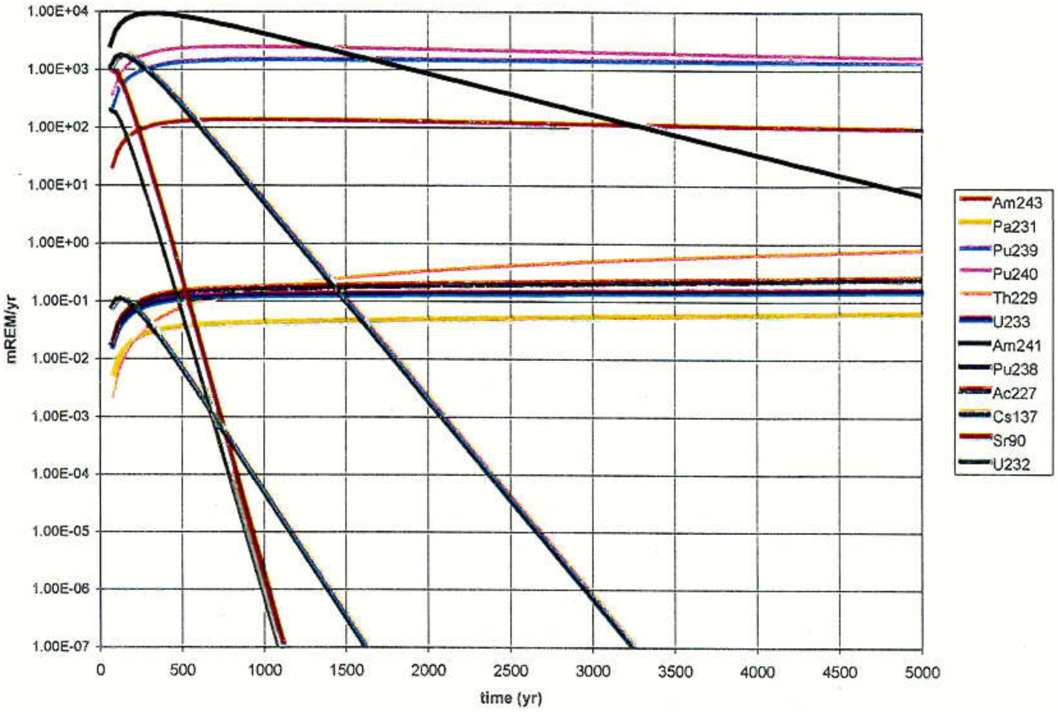


Figure 6-192. Dose From Direct Volcanic Release, *Ash_Dose* (not probability weighted)

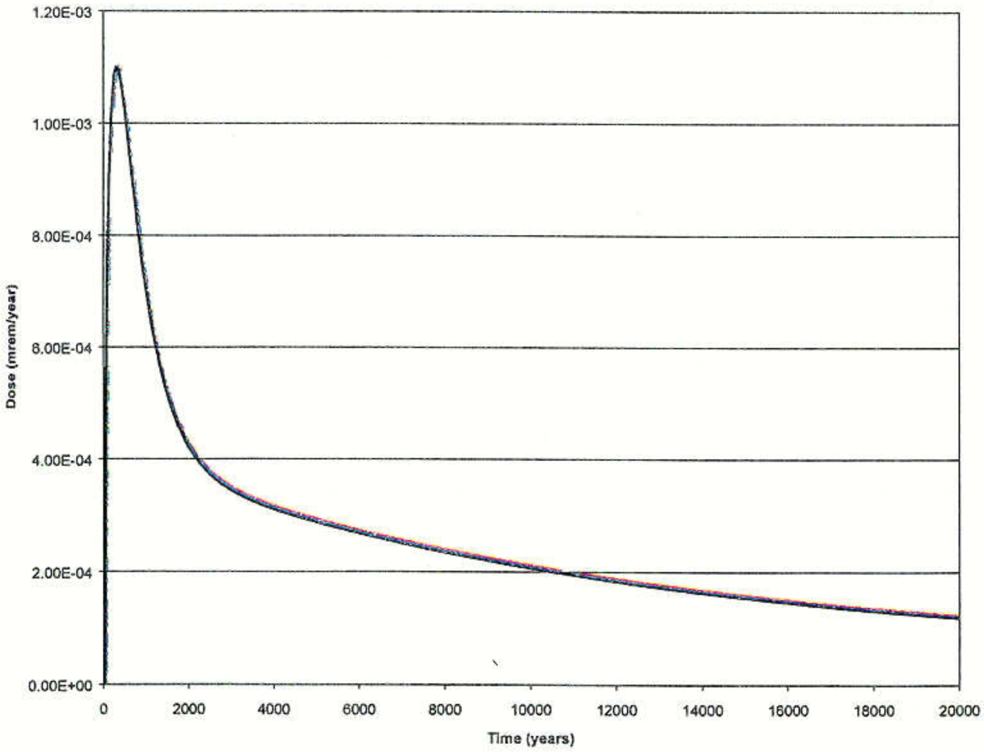


Figure 6-193. Probability Weighted Total Dose From Direct Volcanic Release

C29

6.3.9.2 Intrusive Indirect Release

Overview

The proposed geologic repository at Yucca Mountain could potentially be impacted by volcanic activity. Volcanic activity includes both extrusive and intrusive indirect events. The intrusive indirect event considered in the TSPA-SR model simulates a hypothetical igneous intrusion that results in exposing the waste for groundwater transport away from the repository. This event is characterized by an igneous dike rising to the repository level and intersecting one or more waste containing drifts in the repository (CRWMS M&O 2000 [139563], Section 4.1). The magma from the dike damages the waste packages in the intersected drifts. These affected waste packages are breached and the contents are then available for transport in groundwater. An igneous event is defined here to be an igneous intrusion that intersects the repository footprint at the repository elevation. The probability for both an intrusive and extrusive event occurring is reported in DTN: SN0006T0502900.002 [150856].

The probability distribution is used as reported in DTN: SN0006T0502900.002 [150856]. This reported distribution utilizes probabilities for the full repository layout, including the primary and contingency blocks (CRWMS M&O 2000 [142321], Section 6.5.2.1). This has the effect of slightly overestimating the probabilities than if only the primary block was used.

In the event of an intrusive indirect event, magma flowing into a repository drift will cause waste packages to fail, freeing the contents of the waste packages and making them available for transport to the unsaturated zone (CRWMS M&O 2000 [139563], Section 6.2). It is also assumed that there is a continuous flow of seeping water into the drift that can transport the radionuclides from the EBS to the UZ. The concentration of radionuclides released to the groundwater from an intrusive indirect event is only limited by the dissolution of the waste form, the solubility of the radionuclides, and the inventory present.

The number of packages damaged by an intrusive indirect event is used as reported in DTN: SN0006T0502900.002 [150856]. Probability distributions were developed for backfill and no backfill conditions. The distributions were developed in a spreadsheet by combining all possible combinations of dike lengths and azimuth angles for each set of dike widths and number of dikes combinations. The resulting number of packages hit for each "dike length/number of dikes" combination is a weighted average number of packages hit for that realization. This means that for every "dike width/number of dikes" combination all possible azimuth angles/dike lengths are considered. The number of packages hit by each of these possible azimuth angle dike length pairs is coupled with the probability that that azimuth angle, dike length occurs. The number of packages hit for the "dike width/number of dikes" combination is then calculated as the weighted average over all the azimuth angles and dike lengths. This has the effect of providing a median value for the number of packages hit and eliminating the high and low end tails from the distribution for the number of packages hit (CRWMS M&O 2000 [139563], Section 6.2.4). Furthermore, 100 percent of the packages hit by the intrusive indirect event fail rendering all of the waste material in the damaged package available for transport to the Unsaturated Zone (see Assumptions). No credit is taken for water diversion by the remnants of the drip shield or waste package, and cladding is assumed to be fully degraded. No credit is taken for the chilled pyroclast that will fill all or part of the drift space, and which will form a rind coating all surfaces it contacts in the drift. This last assumption may be slightly conservative, in that it neglects completely the extent to which pyroclast may encapsulate the waste and waste package shells, slowing or preventing water from reaching the waste (CRWMS M&O 2000 [139563], Section 5.3.4).

For more information on intrusive indirect events, the reader is referred to the AMR *Igneous Consequence Modeling for the TSPA-SR* (CRWMS M&O 2000 [139563]).

Inputs to the TSPA-SR Model

The cumulative distribution for the intrusive indirect event probability reported in DTN: SN0006T0502900.002 [150856] is used. The data were incorporated into the TSPA-SR model and checked against the tabulated data in the AMR attachment. The cumulative distribution is shown in Figure 6-194. The TSPA-SR model parameter containing the distribution is *EVENT_PROBABILITYa*.

The cumulative distributions of the number of packages hit by an intrusive indirect event for backfill and no backfill conditions reported in DTN: SN0006T0502900.002 [150856] are used. The data were incorporated into the TSPA-SR model and checked against the tabulated data in the DTN. The cumulative distributions are shown in Figure 6-195 and Figure 6-196. The TSPA-SR model parameters containing the distributions are *NUM_PKGS_INTR_ZONE1* and *NUM_PKGS_HIT_INTR_TOTAL*.

The In-Package Chemistry sub-component of the Indirect Intrusive Event (IIE) component uses the pH and ionic strength of the seepage water at the drift wall instead of using the relationships developed for in-package chemistry (see Section 6.3.4.2 In-Package Chemistry) for the nominal case. The latter relationships were developed for conditions inside a failed waste package, taking into account the chemical reactions of the seepage water with the package and waste form materials. For the IIE component there are no waste package materials to alter the chemical composition of the seepage water and the developed relationships are therefore inappropriate. The pH and ionic strength of the seepage water at the drift wall are shown in Table 6-134. The ionic strength values are not directly reported in the TDMS. The ionic strength values are calculated using the equation (CRWMS M&O 2000 [127818], Section 6.3.2):

$$I = C_{Na} + C_K + 4(C_{Ca} + C_{Mg}) \qquad \text{(Eq. 6-20)}$$

The values for C_{Na} C_K, C_{Cl}, and C_{Mg} are tracked by DTN: MO9912SPAPAI29.002 [148596].

Table 6-134. In-Package Chemistry Parameters for Intrusive Indirect Releases

TSPA Parameter	Time Range (yr)	Parameter Value
pH_Period2	<=1000 yr	8.1
pH_Period3	<=2000 yr	7.8
pH_Period4	>2000 yr	7.3
Ionic_Str_Period2	<=1000 yr	0.004
Ionic_Str_Period3	<=2000 yr	0.0069
Ionic_Str_Period4	>2000 yr	0.0099

DTN: MO9912SPAPAI29.002 [148596]

Implementation

The presence of backfill in the repository drifts has an impact on the number of WPs failed and the extent to which the failed WPs are damaged. In the backfill case a limited number of WPs are completely failed by the IIE. In the no backfill case a larger number of WPs are failed, however the damage to many of those failed WPs is less extensive. This differentiation in failed packages is implemented in the TSPA-SR model by modeling completely failed CSNF and CDSP WPs as separate source terms in the *Indirect_Release_Zone1* container (see Figure 6-197). The number of partially failed packages is calculated in the *Indirect_Release_Zone2* container. The partially failed packages are then modeled using EBS structure of the TSPA-SR model.

The Intrusive Indirect Event (IIE) component of the TSPA-SR model is part of the Disruptive Events component. Figure 6-198 shows the Zone 1 portion of the Intrusive Indirect Event component of the model. Figure 6-199 illustrates the Zone 2 portion of the Intrusive Indirect component of the model.

The contents of the *Indirect_Switches* container, shown in Figure 6-198, are used to initiate an intrusive indirect event in Zone 1. Figure 6-200 shows the contents of the container.

The model parameter *indirect_event_failure_switch* is a uniform distribution with a minimum equal to zero and a maximum equal to one. The selector switch *Indirect_Failure_Switch* sets the time of the intrusive indirect event. According to the model logic if the median value simulation is run, the event occurs at 50 years, otherwise it randomly occurs some time between 50 years from the beginning and end of the simulation. The random selection is based on the value of *indirect_event_failure_switch* and follows the logic:

*indirect_event_failure_switch**(*Run_Time*-50{yr})+50{yr}

where

Run_Time = the simulation duration in years.

The expression element *Failure_Fraction* represents the fraction of the hit waste packages that are failed by an intrusive indirect event. The value of the expression element is conditional on the value of *Indirect_Failure_Switch* and the elapsed time for the simulation, *Etime*. If the elapsed time exceeds the value of *Indirect_Failure_Switch* in years, then the event has occurred and the failure fraction should switch from zero to one. This effectively fails all of the Zone 1 waste packages once the event has occurred.

The expression element *Event_Probability* determines the probability that an intrusive indirect event will occur at any time during the simulated duration. The value of the parameter is the product of the annual probability of an event occurrence, *EVENT_PROBABILITYa*, and the simulation duration, *Run_Time,* in years. In a TSPA-SR simulation that includes disruptive events, an intrusive indirect event occurs regardless of the probability of that event occurring. The mass of radionuclides released from that event at each time step is then multiplied by the event probability, *Event_Probability*, to incorporate the low probability of occurrence and retain the statistical significance.

The contents of the *Bin_Selector_Parameters* container shown in Figure 6-198 are shown in Figure 6-201. The model logic in this sub-component of the IIE model determines which infiltration rate bin receives the Zone 1 radionuclides released from an intrusive indirect event. The bin selector selects one of the five infiltration rate bins, based on a probabilistic distribution, and places all of the Zone 1 waste packages impacted by an intrusive indirect event into this bin.

The three parameters *BIN_Probability_Low, BIN_Probability_Mean,* and *BIN_Probability_High* are discrete stochastic distributions used to place the impacted packages into one of the five infiltration rate bins. The discrete values used in each distribution are the whole numbers 1 through 5 and each value represents one of the five infiltration rate bins, the bin with the same numerical value. The probability associated with each of the discrete values is equal to the fractional area of the repository assigned to each bin (CRWMS M&O 2000 [149860], Table 5). For the low infiltration scenario, the fraction of the waste packages in bin 1 is equal to 0.597. The probability of placing the failed waste packages from a intrusive indirect event in bin 1 under the low infiltration scenario, the probability assigned to the discrete value "1" in the distribution *BIN_Probability_Low*, is also equal to 0.597. Table 6-135 shows the bin placement probabilities based on the infiltration scenario assigned at the initiation of the model.

Table 6-135. Indirect Release Bin Placement Probabilities

Bin Number	Infiltration Scenario	Fraction of Repository Area	IIE Bin Placement Probability	Discrete Value
1	Low	0.597	0.597	1
2	Low	0.403	0.403	2
3	Low	0	0	3
4	Low	0	0	4
5	Low	0	0	5
1	Mean	0.01607	0.01607	1
2	Mean	0.13154	0.13154	2
3	Mean	0.3212	0.3212	3
4	Mean	0.5285	0.5285	4
5	Mean	0.00269	0.00269	5
1	High	0	0	1
2	High	0.0123	0.0123	2
3	High	0.134	0.134	3
4	High	0.548	0.548	4
5	High	0.3057	0.3057	5

At the initiation of the simulation a value is assigned to each of the three probability distributions. If the median value simulation is run, the median value will be assigned to each stochastic, otherwise a value sampled from the discrete distribution will be assigned to each parameter. The selector switch *BIN_Selector* receives a value from the TSPA-SR parameter *Infiltration_Scenario* (see Section 6.3.1.1 Climate and Infiltration) and based upon this value, chooses which of the three distribution results to retain. If the value assigned to *Infiltration_Scenario* equals 1, the *BIN_Selector* value equals the value assigned to *BIN_Probability_Low*. If the value assigned to *Infiltration_Scenario* equals 2, the *BIN_Selector* value equals the value assigned to *BIN_Probability_Mean*. If the value assigned to *Infiltration_Scenario* does not equal 1 or 2, the *BIN_Selector* value equals the value assigned to *BIN_Probability_High*. The *BIN_Selector* value will be 1, 2, 3, 4, or 5, the discrete values of the three distributions. The inventory from the waste packages failed due to an intrusive indirect event will be placed into the bin with the same numerical value. The model logic for placing impacted packages into a bin will be discussed later in this section.

The number of Zone 1 packages impacted by an intrusive indirect event is determined in the *Failed_Packages* container shown in Figure 6-198. The contents of this container are shown in Figure 6-202.

The stochastic distribution *NUM_PKGS_INTR_ZONE1* is assigned the CDF distribution in Figure 6-195. At the initiation of the simulation a value is sampled from this distribution. For the median value simulation the number of packages impacted by an intrusive indirect event is 195. For all other simulations the number of impacted packages will be between 104 and 227. The two expression elements, *CSNF_Pkgs_Intr* and *CDSP_Pkgs_Intr* divide the number of Zone 1 packages between CSNF and CDSP package types. The number of the packages which are the CSNF type, *CSNF_Pkgs_Intr*, equals the number of Zone 1 packages multiplied by the fraction of all packages in the repository which are the CSNF type (i.e., the total number of CSNF packages divided by the total number of CSNF and CDSP packages). This value is rounded to yield a whole number. The number of impacted packages that are the CDSP type, *CDSP_Pkgs_Intr*, is calculated using the same logic but uses the fraction of repository packages which are the CDSP type.

The *Failed_Packages_Per_Bin* container shown in Figure 6-198 contains a set of selector switches which are used by FEHM in the Unsaturated Zone Transport component of the TSPA-SR model (see Section 6.3.1.3 Unsaturated Zone Flow). The contents are shown in Figure 6-203.

The expression element *Fraction_Failed_Packages* calculates the Zone 1 fraction of all the repository packages that are impacted by the intrusive indirect event. The five selector switches *Failed_Pkgs_Intr_Bin1*, *Failed_Pkgs_Intr_Bin2*, *Failed_Pkgs_Intr_Bin3*, *Failed_Pkgs_Intr _Bin4*, and *Failed_Pkgs_Intr_Bin5* calculate the number of FEHM nodes dedicated to intrusive releases in the UZ transport component of the TSPA-SR model. The number of FEHM nodes, calculated by these five selector switches is the rounded product of *Fraction_Failed_Packages* and *FEHMN_Nodes*. The value of *FEHMN_Nodes* equals 275 and represents total the number of nodes used in the FEHM analysis. The model logic is implemented similarly for all five selector switches. The model parameter *BIN_Selector*, discussed earlier, determines which of the five infiltration rate bins get the FEHM nodes. If the value of *BIN_Selector* equals 1 then Bin 1 gets the number of calculated nodes and the other bins get zero nodes. If the value of *BIN_Selector* equals 2 then Bin 2 gets the nodes and the other four bins get zero nodes. This logic is repeated for the other three bins. For more information regarding the use of nodes in FEHM see Section 6.3.1.3 Unsaturated Zone Flow and the FEHM Users Manual (Zyvoloski et al. 1997 [100528]).

The release of radionuclides from Zone 1 waste packages impacted by an intrusive indirect event is implemented separately for each package type, CSNF and CDSP. Figure 6-198 illustrates this separation. The ability of the engineered barrier system to contain radionuclide releases from CSNF packages impacted by an intrusive indirect event is calculated within the *Intrusive_Events_CSNF_Packages* container. The model logic implemented for the intrusive indirect event release is nearly identical to the model logic implemented for assessing the performance of the engineered barrier system under nominal conditions. A few modifications were made to accommodate the conceptual differences. These modifications are discussed next.

For the nominal case the model implementation is broken into five infiltration rate bins with three dripping environments, Always Drip, Intermittent Drip, and Never Drip, within each bin. For the intrusive releases, all of the impacted waste packages are exposed to the thermal hydrological conditions associated with the Always Drip environment within one of the infiltration rate bins. The one infiltration rate bin is selected by the model parameter *BIN_Selector* discussed earlier. If the value of *BIN_Selector* equals 1, the nominal mass released from the EBS from Bin 1 to the UZ is supplemented with all of the mass released from the intrusive indirect event. If the value of *BIN_Selector* equals 2, the nominal mass released from the EBS from Bin 2 to the UZ is supplemented with the mass released from the intrusive indirect event. If the value of *BIN_Selector* equals 3, the nominal mass released from the EBS from Bin 3 to the UZ is supplemented with all of the mass released from the intrusive indirect event. If the value of *BIN_Selector* equals 4, the nominal mass released from the EBS from Bin 4 to the UZ is supplemented with all of the mass released from the intrusive indirect event. And similarly, if the value of *BIN_Selector* equals 5, the nominal mass released from the EBS from Bin 5 to the UZ is supplemented with all of the mass released from the intrusive indirect event. Similar logic is used any time the value of *Bin_Selector* is referenced in the IIE component of the TSPA-SR model.

The calculation of Zone 1 radionuclide releases after a simulated intrusive indirect event impacting CSNF waste packages is illustrated in Figure 6-204. The discussion that follows details the implemented logic for simulating the event.

The in-drift chemistry implementation for IIE is located in the *In-Drift_Chemistry* container shown in Figure 6-205. The inputs to this sub-component are the thermal hydrology temporal profiles for the invert evaporation rate, the invert liquid flux rate, and the invert relative humidity (see Section 6.3.2.1 Thermal Hydrology). The temporal profiles for these parameters are defined specifically for each infiltration rate bin and package type in the NFE component of the TSPA-SR model. The IIE component of the TSPA-SR model selects the proper set of profiles (bin 1, bin 2, bin 3, bin 4, or bin 5) based on the value of *BIN_Selector*. Figure 6-205 graphically illustrates the implementation of the In-Drift Chemistry Component within the IIE model component. The selector switches *QEvap_Num*, *QFlux_Denom,* and *RH_Invert* select the appropriate CSNF thermal hydrology profiles based on the value assigned to *BIN_Selector*. The model calculations are then performed with these profiles. The model logic describing the calculations in this sub-component is identical to the model logic described in Section 6.3.2.2 In-Drift Geochemical Environment.

The container *Invert_Properties* shown in Figure 6-204 uses the value of *BIN_Selector* and model logic to appropriately assign the correct TH profile for the invert temperature, *Invert_Temperature*, and invert liquid saturation, *Invert_Saturation* (see Section 6.3.2.1 Thermal Hydrology) to the IIE component. The model logic for assigning the proper profiles is similar to the model logic used in the *In_Drift_Chemistry* container discussed earlier. The model logic is explicitly stated in Table 6-136. The model logic for the remainder of the calculations in this sub-component is identical to the model logic described in Section 6.3.5.2 EBS Transport Parameters. A graphical representation of the model logic for calculating invert properties is shown in Figure 6-206.

Although the cladding of a CSNF waste package is assumed to be immediately failed after an intrusive indirect event, the release of radionuclides is still hindered by the CSNF matrix degradation rate. The model logic implemented for the calculation of the CSNF degradation rate is located in the *Matrix_Degradation_Rate* container shown in Figure 6-204. The model logic is identical to the model logic implemented for the nominal case, with one minor change. In the nominal case, the waste package surface temperature profile used for the rate calculation is specific to each bin, as defined in the TH component. In the IIE component the temperature profile, *Temp_WP,* is not specified per bin, instead the selector switch *Temp_WP* selects the proper bin temperature profile based on the value assigned to *BIN_Selector*. The model logic for selecting the correct profile is similar to that discussed previously for the In-Drift Chemistry properties. The model logic for selecting the appropriate waste package temperature is listed in Table 6-136. The model logic for the remainder of the calculations in this sub-component is identical to the model logic described in Section 6.3.4.3 Cladding Degradation Model. A graphical representation of the model logic for calculating CSNF matrix degradation rates is shown in Figure 6-207.

The seepage flux through the failed waste packages is implemented in the *Flux_DS_WP* container shown in Figure 6-204. The contents of this container are shown in Figure 6-208. In the nominal case the seepage flux through a failed waste package is a fraction of the seepage flow calculated by the Seepage DLL (see Section 6.3.5.1 EBS Flow and Transport and Section 6.3.1.2 Seepage into Drifts). Water entering the failed waste package must flow through the failed drip shield and waste package openings, reducing the volume which contacts the waste form. In the Zone 1 IIE component, flow diversion by a drip shield is neglected and there is no outer barrier waste package to limit the contact between the seeping water and the waste form. The result is that all the water flowing into the bin contacts the waste form. The seepage flow into the bin is selected by the selector switch *QFlux*. *QFlux* selects the appropriate seepage flow rate, calculated per bin by the Seepage DLL, based on the value assigned to *BIN_Selector*. The model logic for selecting the appropriate seepage rate is listed in Table 6-136.

The model logic of the In-Package Chemistry component for Zone 1 waste packages impacted by IIE differs significantly from the model logic implemented for the nominal case. In the nominal case the pH inside a CSNF waste package is a function of the cladding coverage and the flux through the waste package. Chemical reactions with the waste package contents would change the pH of the seepage water. In the intrusive indirect event component, there is no waste package to consider and the pH of the water contacting the waste form is assigned the pH of the drift at the drift wall. The ionic strength implementation, analogous to the in-package ionic strength in the nominal case, also uses the chemical conditions at the drift wall. The in-package chemistry component is shown in Figure 6-209. The values for the data elements *pH_Period2, pH_Period3, pH_Period4, Ionic_Str_Period2, Ionic_Str_Period3* and *Ionic_Str_Period4* are listed in Table 6-134. The pH data were generated for the TSPA-SR model and are tracked by

DTN: MO9912SPAPAI29.002 [148596]. The values used to calculate the ionic strength in Equation 6-1 are tracked with the same DTN.

The selector switches *pH* and *Ionic_Strength* change the value of the pH and ionic strength according to the time schedule in Table 6-134. The expression elements *pH_CSNF* and *pH_CDSP* track the *pH* selector switch and are used in the IIE component in the calculation of U, Np, and Am solubilities, waste form degradation rates, colloïd formation, and the total carbonate concentration as they would be in the nominal case simulation. The model logic implementing these calculations is identical to the nominal case implementation. The ionic strength parameters *Ionic_Str_CSNF* and *Ionic_Str_CDSP* equal the value of *Ionic_Strength.* The ionic strength is used in the Colloid Model sub-component of the IIE component. The model logic in the Colloid Model sub-component is identical to the model logic in the nominal case simulation discussed in Section 6.3.4.6 Colloids.

The remainder of the EBS implementation for CSNF waste packages within the IIE component of the TSPA-SR model is identical to the nominal case implementation. The number of Zone 1 packages failed by the intrusive indirect event, discussed previously, is determined by the stochastic element *NUM_PKGS_INTR_ZONE1*. The number of CSNF waste packages impacted by the intrusive indirect event is the number of intrusive indirect event packages, *NUM_PKGS_INTR_ZONE1* multiplied by the fraction of total packages which are CSNF. This number of packages, *CSNF_Pkgs_Intr* is entered in the source element *CSNF_Always_Drip*. The source element *CSNF_Always_Drip* is further characterized as a CSNF waste package that has a failed fraction equivalent to 1, once the intrusive indirect event occurs. This condition is implemented with the parameter *Fraction_Failed* discussed earlier. Once the intrusive indirect event occurs, the transport through the EBS, although unhindered by cladding degradation, proceeds as previously discussed in Section 6.3.5.1 EBS Flow and Transport Pathways.

Once the radionuclides are released from the EBS, they are added to the UZ transport model component. The calculations for this sub-component are performed in the *CSNF_Indirect_Output* container shown in Figure 6-204. This model implementation is graphically shown in Figure 6-210.

The mass of each radionuclide exiting the EBS is accumulated in the quantitative element, *CSNF_Intr_Release*. *CSNF_Intr_Release* tracks all the mass which leaves the EBS as a result of an intrusive indirect event impacting Zone 1 CSNF waste packages. This mass is placed within one of the CSNF infiltration rate bins. The selection of which bin receives the mass is determined by the value of *Bin_Selector* as described earlier. The selector swithes *CSNF_Bin1_Input, CSNF_Bin2_Input, CSNF_Bin3_Input, CSNF_Bin4_Input, and CSNF_Bin5_Input* contain model logic which, based on the value of *Bin_Selector*, assign appropriate output, the cummulative radionuclide mass released or zero mass, to each bin. The vector parameter *Zero_Mass* passes zero mass for each species to the bins which were determined to receive no mass from the event.

This next section describes the logic changes which were implemented for Zone 1 CDSP waste packages impacted by an intrusive indirect event. The model logic changes for Zone 1 CDSP waste package releases is similar to the model logic changes discussed above for Zone 1 CSNF waste packages. The bin-specific CDSP thermal hydrological profiles are used as appropriate.

The profile selection is determined by the value of *Bin_Selector* and uses the same model logic principles as discussed for Zone 1 CSNF waste packages.

The appropriate waste form degradation model is used to model the degradation of the Zone 1 CDSP waste form. The implementation is nearly identical to the nominal case implementation (see Section 6.3.4.4 Dissolution Rate Model). A few modifications were necessary to evaluate the model equations with the appropriate thermal hydrological profiles. As is the case for the CSNF implementation, the local pH and temperature parameters used by the Glass Dissolution Model component do not apply. Instead the pH of the drift and selected bin waste packages' temperatures apply. The pH of the drift is equal to the pH describe above for the Zone 1 CSNF waste package failures. The temperature for the thermal dependence of the dissolution equation is calculated using the appropriate Thermal Hydrological dataset, *Temp_CDSP_WP*. The bin-specific profile is determined by the value assigned to *Bin_Selector* by the parameter *Temp_WP*. The implemented logic is graphically shown in Figure 6-211. The logic for selecting the appropriate waste package temperature is listed in Table 6-136.

The number of waste packages that are partially failed by the IIE is calculated in the *Indirect_Release_Zone2* container. This model implementation is graphically shown in Figure 6-199.

The stochastic parameter *NUM_PKGS_HIT_INTR_TOTAL* determines the total number of Zone 1 (completely failed) and Zone 2 (partially failed) waste packages in an IIE with no backfill. The number of Zone 2 packages in an IIE is calculated by the parameter NUM_PKGS_INTR_ZONE2, which subtracts the number of Zone 1 WPs (parameter *NUM_PKGS_INTR_ZONE1*) from *NUM_PKGS_HIT_INTR_TOTAL*. If calculated difference is less than zero, the number of Zone 2 waste packages is set equal to zero. The number of CSNF and CDSP Zone 2 packages is determined by the parameters *NUM_CSNF_PKGS_ZONE2* and *NUM_CDSP_PKGS_ZONE2*, respectively. In each case the number of packages of the given waste type is determined by multiplying the total number of Zone 2 packages by the ratio of the total number of packages of that type (e.g., *Total_CSNF_Packages* or *Total_CDSP_Packages*) to the total number of packages in the repository.

The CSNF and CDSP Zone 2 packages are apportioned between the Infiltration Rate bins in container *PKGS_IN_BINS_ZONE2*. This model implementation is graphically shown in Figure 6-199. In mean infitration scenarios packages are first put in Bin 4 (parameter *CSNF_WP_BIN4_2* or *CDSP_WP_BIN4_2*). If the number of Zone 2 packages for a given fuel type is greater than the number of packages in Bin 4, the remaining packages are then put in Bin 3 (parameter *CSNF_WP_BIN3_2* or *CDSP_WP_BIN3_2*). If the number of Zone 2 packages is greater than the number of packages in Bins 4 and 3 , the remaining packages are put in Bin 2 (parameter *CSNF_WP_BIN2_2* or *CDSP_WP_BIN2_2*). Then if the total number of Zone 2 packages is greater than the number of packages Bins 4, 3, and 2 the remaining packages are put in Bin 1 (parameter *CSNF_WP_BIN1_2* or *CDSP_WP_BIN1_2*). Finally if the number of Zone 2 packages is greater than the number of packages in Bins 4, 3, 2, and 1, the remaining packages are put in Bin 5 (parameter *CSNF_WP_BIN5_2* or *CDSP_WP_BIN5_2*). Similar logic is used for the high and low infiltration scenarios. In the high infiltration scenario Zone 2 packages are apportioned to Bin 4, Bin 5, Bin 3, and Bin 2. In the low infiltration scenario Zone 2 packages are apportioned to Bin 2 and Bin 1. A set of switches for CSNF (*CSNF_WP_BIN1*,

CSNF_WP_BIN2, *CSNF_WP_BIN3*, *CSNF_WP_BIN4*, and *CSNF_WP_BIN5*) and CDSP (*CDSP_WP_BIN1*, *CDSP_WP_BIN2*, *CDSP_WP_BIN3*, *CDSP_WP_BIN4*, and *CDSP_WP_BIN5*) are used to select the appropriate number of packages for each bin based on the infiltration scenario.

The breach area of Zone 2 packages is given by the stochastic parameter *WP_Breach_Area_Zone2_Pkgs*. The fractional area of the breach is determined for each waste package type (*Zone2_Patch_Fraction_CSNF* or *Zone2_Patch_Fraction_CDSP*) by dividing the Zone 2 WP breach area by the respective package surface area (*WP_SA_CSNF* or *WP_SA_CDSP*). If the calculated fraction is greater than 1, it is set equal to 1.

Similar to the Zone 1 packages, the parameter *WP_Fraction_Failed_Zone2* is used to fail Zone 2 packages in the EBS. When ETime is less than the time of the IIE (parameter *Indirect_Failure_Switch*) *WP_Fraction_Failed_Zone2* is equal to 0, when the time of the IIE is equal to or greater than *ETime*, *WP_Fraction_Failed_Zone2* is equal to 1.

Table 6-136 lists the parameters that are added to implement the IIE component of the TSPA-SR model successfully. The use of these parameters has been discussed previously in the preceding paragraphs. Each of the parameters listed below can only be used within the IIE component of TSPA-SR model. Referencing nominal case parameters as appropriate is done routinely. The mass of each radionuclide released from the IIE component is passed to the nominal case component of the TSPA-SR model for further analysis.

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
Event_Probability	Probability of and IIE occurring during the simulated duration	if(ETime==0{yr},0,((EVENT_PROBABILITYa *Run_Time*1{1/yr})))	IIE Only
Indirect_event_failure_time	Random number generator used for initiating a random IIE	Uniform (Min=0, Max=1)	IIE Only
Indirect_Failure_Switch	Initiation time for the IIE occurrence	IF Median_Value_Run==1 THEN 50.0 {yrs} ELSE indirect_event_failure_switch*(Run_Time-50{yr})+50{yr}	IIE Only
Failure_Fraction	Fraction of IIE impacted waste packages that have failed.	if(ETime>=Indirect_Failure_Switch,1,0)	IIE Only
BIN_Probability_Low	Discrete distribution used to randomly assign the IIE released mass to a specific bin, low infiltration scenario	See Table 6-135	IIE Only

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
BIN_Probability_High	Discrete distribution used to randomly assign the IIE released mass to a specific bin, high infiltration scenario	See Table 6-135	IIE Only
BIN_Selector	Selector switch used to assign the IIE released mass to a specific bin based on the chosen infiltration scenario	IF Infiltration_Scenario==1 THEN BIN_Probability_Low ELSEIF Infiltration_Scenario==2 THEN BIN_Probability_Mean ELSE BIN_Probability_High	IIE Only
CSNF_Pkgs_Intr	Number of Zone 1 CSNF packages impacted by the IIE	round(NUM_PKGS_INTR_ZONE1*Total_CSNF_Packages/(Total_CSNF_Packages + Total_CDSP_Packages))	IIE Only
CDSP_Pkgs_Intr	Number of Zone 1 CDSP packages impacted the IIE	round(NUM_PKGS_INTR_ZONE1*Total_CDSP_Packages/(Total_CSNF_Packages + Total_CDSP_Packages))	IIE Only
Fraction_Failed_Packages	Fraction (Zone 1) of total inventory impacted by the IIE	NUM_PKGS_INTR_ZONE1/(Total_CDSP_Packages+Total_CSNF_Packages)	IIE Only
FEHMNN_NODES	Number of FEHMNN Nodes for UZ transport	275	IIE Only
Failed_Pkgs_Intr_Bin1	Number of FEHMNN Nodes to reserve for UZ transport of IIE releases to Bin 1	IF Bin_Selector==1 THEN round(Fraction_Failed_Packages*FEHMNN_Nodes) ELSE 0	IIE Only
Failed_Pkgs_Intr_Bin3	Number of FEHMNN Nodes to reserve for UZ transport of IIE releases to Bin 2	IF Bin_Selector==2 THEN round(Fraction_Failed_Packages*FEHMNN_Nodes) ELSE 0	IIE Only
Failed_Pkgs_Intr_Bin3	Number of FEHMNN Nodes to reserve for UZ transport of IIE releases to Bin 3	IF Bin_Selector==3 THEN round(Fraction_Failed_Packages*FEHMNN_Nodes) ELSE 0	IIE Only
Failed_Pkgs_Intr_Bin4	Number of FEHMNN Nodes to reserve for UZ transport of IIE releases to Bin 4	IF Bin_Selector==4 THEN round(Fraction_Failed_Packages*FEHMNN_Nodes) ELSE 0	IIE Only
Failed_Pkgs_Intr_Bin5	Number of FEHMNN Nodes to reserve for UZ transport of IIE releases to Bin 5	IF Bin_Selector==5 THEN round(Fraction_Failed_Packages*FEHMNN_Nodes) ELSE 0	IIE Only

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
Q_EVAP_NUM	Bin specific evaporation rate selected by Bin_Selector	IF BIN_Selector==1 THEN QEvap_CSNF_Inv.Qevap_CSNF_Inv[Bin_1] ELSEIF BIN_Selector==2 THEN QEvap_CSNF_Inv.Qevap_CSNF_Inv[Bin_2] ELSEIF BIN_Selector==3 THEN QEvap_CSNF_Inv.Qevap_CSNF_Inv[Bin_3] ELSEIF BIN_Selector==4 THEN QEvap_CSNF_Inv.Qevap_CSNF_Inv[Bin_4] ELSE QEvap_CSNF_Inv.Qevap_CSNF_Inv[Bin_5]	Local to CSNF waste packages
Q_EVAP_NUM	Bin specific evaporation rate selected by Bin_Selector	IF BIN_Selector==1 THEN QEvap_CDSP_Inv.Qevap_CDSP_Inv[Bin_1] ELSEIF BIN_Selector==2 THEN QEvap_CDSP_Inv.Qevap_CDSP_Inv[Bin_2] ELSEIF BIN_Selector==3 THEN QEvap_CDSP_Inv.Qevap_CDSP_Inv[Bin_3] ELSEIF BIN_Selector==4 THEN QEvap_CDSP_Inv.Qevap_CDSP_Inv[Bin_4] ELSE QEvap_CDSP_Inv.Qevap_CDSP_Inv[Bin_5]	Local to CDSP waste packages
QFlux_Denom	Bin specific liquid flux rate selected by Bin_Selector	IF BIN_Selector==1 THEN QFlux_CSNF_Inv.Qflux_CSNF_Inv[Bin_1] ELSEIF BIN_Selector==2 THEN QFlux_CSNF_Inv.Qflux_CSNF_Inv[Bin_2] ELSEIF BIN_Selector==3 THEN QFlux_CSNF_Inv.Qflux_CSNF_Inv[Bin_3] ELSEIF BIN_Selector==4 THEN QFlux_CSNF_Inv.Qflux_CSNF_Inv[Bin_4] ELSE QFlux_CSNF_Inv.Qflux_CSNF_Inv[Bin_5]	Local to CSNF waste packages
QFlux_Denom	Bin specific liquid flux rate selected by Bin_Selector	IF BIN_Selector==1 THEN QFlux_CDSP_Inv.Qflux_CDSP_Inv[Bin_1] ELSEIF BIN_Selector==2 THEN QFlux_CDSP_Inv.Qflux_CDSP_Inv[Bin_2] ELSEIF BIN_Selector==3 THEN QFlux_CDSP_Inv.Qflux_CDSP_Inv[Bin_3] ELSEIF BIN_Selector==4 THEN QFlux_CDSP_Inv.Qflux_CDSP_Inv[Bin_4] ELSE QFlux_CDSP_Inv.Qflux_CDSP_Inv[Bin_5]	Local to CDSP waste packages
RH_Invert	Bin specific invert relative humidity selected by Bin_Selector	IF BIN_Selector==1 THEN RH_CSNF_Inv.RH_CSNF_Inv[Bin_1] ELSEIF BIN_Selector==2 THEN RH_CSNF_Inv.RH_CSNF_Inv[Bin_2] ELSEIF BIN_Selector==3 THEN RH_CSNF_Inv.RH_CSNF_Inv[Bin_3] ELSEIF BIN_Selector==4 THEN RH_CSNF_Inv.RH_CSNF_Inv[Bin_4] ELSE RH_CSNF_Inv.RH_CSNF_Inv[Bin_5]	Local to CSNF waste packages

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
RH_Invert	Bin specific invert relative humidity selected by Bin_Selector	IF BIN_Selector==1 THEN RH_CDSP_Inv.RH_CDSP_Inv[Bin_1] ELSEIF BIN_Selector==2 THEN RH_CDSP_Inv.RH_CDSP_Inv[Bin_2] ELSEIF BIN_Selector==3 THEN RH_CDSP_Inv.RH_CDSP_Inv[Bin_3] ELSEIF BIN_Selector==4 THEN RH_CDSP_Inv.RH_CDSP_Inv[Bin_4] ELSE RH_CDSP_Inv.RH_CDSP_Inv[Bin_5]	Local to CDSP waste packages
Invert_Saturation	Bin specific invert saturation selected by Bin_Selector	IF BIN_Selector==1 THEN Sat_CSNF.Sat_CSNF[Bin_1] ELSEIF BIN_Selector==2 THEN Sat_CSNF.Sat_CSNF[Bin_2] ELSEIF BIN_Selector==3 THEN Sat_CSNF.Sat_CSNF[Bin_3] ELSEIF BIN_Selector==4 THEN Sat_CSNF.Sat_CSNF[Bin_4] ELSE Sat_CSNF.Sat_CSNF[Bin_5]	Local to CSNF waste packages
Invert_Saturation	Bin specific invert saturation selected by Bin_Selector	IF BIN_Selector==1 THEN Sat_CDSP.Sat_CDSP[Bin_1] ELSEIF BIN_Selector==2 THEN Sat_CDSP.Sat_CDSP[Bin_2] ELSEIF BIN_Selector==3 THEN Sat_CDSP.Sat_CDSP[Bin_3] ELSEIF BIN_Selector==4 THEN Sat_CDSP.Sat_CDSP[Bin_4] ELSE Sat_CDSP.Sat_CDSP[Bin_5]	Local to CDSP waste packages
Invert_Temperature	Bin specific invert temperature selected by Bin_Selector	IF BIN_Selector==1 THEN Temp_CSNF_INV.Temp_CSNF_INV[Bin_1] ELSEIF BIN_Selector==2 THEN Temp_CSNF_INV.Temp_CSNF_INV[Bin_2] ELSEIF BIN_Selector==3 THEN Temp_CSNF_INV.Temp_CSNF_INV[Bin_3] ELSEIF BIN_Selector==4 THEN Temp_CSNF_INV.Temp_CSNF_INV[Bin_4] ELSE Temp_CSNF_INV.Temp_CSNF_INV[Bin_5]	Local to CSNF waste packages
Invert_Temperature	Bin specific invert temperature selected by Bin_Selector	IF BIN_Selector==1 THEN Temp_CDSP_INV.Temp_CDSP_INV[Bin_1] ELSEIF BIN_Selector==2 THEN Temp_CDSP_INV.Temp_CDSP_INV[Bin_2] ELSEIF BIN_Selector==3 THEN Temp_CDSP_INV.Temp_CDSP_INV[Bin_3] ELSEIF BIN_Selector==4 THEN Temp_CDSP_INV.Temp_CDSP_INV[Bin_4] ELSE Temp_CDSP_INV.Temp_CDSP_INV[Bin_5]	Local to CDSP waste packages

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
Temp_WP	Bin specific waste package temperature selected by Bin_Selector	IF BIN_Selector==1 THEN Temp_CSNF_WP.Temp_CSNF_WP[Bin_1] ELSEIF BIN_Selector==2 THEN Temp_CSNF_WP.Temp_CSNF_WP[Bin_2] ELSEIF BIN_Selector==3 THEN Temp_CSNF_WP.Temp_CSNF_WP[Bin_3] ELSEIF BIN_Selector==4 THEN Temp_CSNF_WP.Temp_CSNF_WP[Bin_4] ELSE Temp_CSNF_WP.Temp_CSNF_WP[Bin_5]	Local to CSNF waste packages
Temp_WP	Bin specific waste package temperature selected by Bin_Selector	IF BIN_Selector==1 THEN Temp_CDSP_WP.Temp_CDSP_WP[Bin_1] ELSEIF BIN_Selector==2 THEN Temp_CDSP_WP.Temp_CDSP_WP[Bin_2] ELSEIF BIN_Selector==3 THEN Temp_CDSP_WP.Temp_CDSP_WP[Bin_3] ELSEIF BIN_Selector==4 THEN Temp_CDSP_WP.Temp_CDSP_WP[Bin_4] ELSE Temp_CDSP_WP.Temp_CDSP_WP[Bin_5]	Local to CDSP waste packages
QFlux	Bin specific flow exposed to waste package selected by Bin_Selector	IF BIN_Selector==1 THEN if(SeepFlux_AI_CSNF_1==0{m3/yr},SeepFlux_In_CSNF_1,SeepFlux_AI_CSNF_1) ELSEIF BIN_Selector==2 THEN if(SeepFlux_AI_CSNF_2==0{m3/yr},SeepFlux_In_CSNF_2,SeepFlux_AI_CSNF_2) ELSEIF BIN_Selector==3 THEN if(SeepFlux_AI_CSNF_3==0{m3/yr},SeepFlux_In_CSNF_3,SeepFlux_AI_CSNF_3) ELSEIF BIN_Selector==4 THEN if(SeepFlux_AI_CSNF_4==0{m3/yr},SeepFlux_In_CSNF_4,SeepFlux_AI_CSNF_4) ELSE if(SeepFlux_AI_CSNF_5==0{m3/yr},SeepFlux_In_CSNF_5,SeepFlux_AI_CSNF_5)	Local to CSNF waste packages
QFlux	Bin specific flow exposed to waste package selected by Bin_Selector	IF BIN_Selector==1 THEN if(SeepFlux_AI_CDSP_1==0{m3/yr},SeepFlux_In_CDSP_1,SeepFlux_AI_CDSP_1) ELSEIF BIN_Selector==2 THEN if(SeepFlux_AI_CDSP_2==0{m3/yr},SeepFlux_In_CDSP_2,SeepFlux_AI_CDSP_2) ELSEIF BIN_Selector==3 THEN if(SeepFlux_AI_CDSP_3==0{m3/yr},SeepFlux_In_CDSP_3,SeepFlux_AI_CDSP_3) ELSEIF BIN_Selector==4 THEN if(SeepFlux_AI_CDSP_4==0{m3/yr},SeepFlux_In_CDSP_4,SeepFlux_AI_CDSP_4) ELSE if(SeepFlux_AI_CDSP_5==0{m3/yr},SeepFlux_In_CDSP_5,SeepFlux_AI_CDSP_5)	Local to CDSP waste packages

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
pH	pH at the drift wall	IF Etime<= 950{yr} THEN pH_Period2 ELSEIF ETime<=1950{yr} THEN pH_Period3 ELSEIF ETime<=99950{yr} THEN pH_Period4 ELSE pH_Period5	IIE Only
Ionic_Strength	Ionic strength at the drift wall	IF Etime<= 950{yr} THEN Ionic_Str_Period2 ELSEIF ETime<=1950{yr} THEN Ionic_Str_Period3 ELSEIF ETime<=99950{yr} THEN Ionic_Str_Period4 ELSE Ionic_Str_Period5	IIE Only
CSNF_Intr_RelRate	Amount (g/yr) of each radionuclide species released from an IIE	Collector.Water_to_Sink[RN]	Local to CSNF waste packages
CSNF_Intr_Release	Cumulative mass of each radionuclide species released from an IIE	Initial Mass: Zero_Mass Accumulation Rate: CSNF_Intr_RelRate	Local to CSNF waste packages
Zero_Mass	Vector for assigning zero mass to bins other than that selected by Bin_Selector	0.0 g	Local to CSNF waste packages
CSNF_Bin1_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==1 THEN CSNF_Intr_Release ELSE Zero_Mass	Local to CSNF waste packages
CSNF_Bin2_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==2 THEN CSNF_Intr_Release ELSE Zero_Mass	Local to CSNF waste packages
CSNF_Bin3_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==3 THEN CSNF_Intr_Release ELSE Zero_Mass	Local to CSNF waste packages
CSNF_Bin4_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==4 THEN CSNF_Intr_Release ELSE Zero_Mass	Local to CSNF waste packages
CSNF_Bin5_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==5 THEN CSNF_Intr_Release ELSE Zero_Mass	Local to CSNF waste packages
CDSP_Intr_RelRate	Amount (g/yr) of each radionuclide species released from an IIE	Collector.Water_to_Sink[RN]	Local to CDSP waste packages
CDSP_Intr_Release	Cumulative mass of each radionuclide species released from an IIE	Initial Mass: Zero_Mass Accumulation Rate: CDSP_Intr_RelRate	Local to CDSP waste packages
Zero_Mass	Vector for assigning zero mass to bins other than that selected by Bin_Selector	0.0 g for each of the 32 species in the species list	Local to CDSP waste packages

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
CDSP_Bin1_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==1 THEN CDSP_Intr_Release ELSE Zero_Mass	Local to CDSP waste packages
CDSP_Bin2_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==2 THEN CDSP_Intr_Release ELSE Zero_Mass	Local to CDSP waste packages
CDSP_Bin3_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==3 THEN CDSP_Intr_Release ELSE Zero_Mass	Local to CDSP waste packages
CDSP_Bin4_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==4 THEN CDSP_Intr_Release ELSE Zero_Mass	Local to CDSP waste packages
CDSP_Bin5_Input	Mass contribution to EBS release from IIE	IF BIN_Selector==5 THEN CDSP_Intr_Release ELSE Zero_Mass	Local to CDSP waste packages
NUM_PKGS_HIT_INTR_ZONE1	Number of Zone 1 packages failed by the IIE	CDF See DTN: SN0006T0502900.002 [150856]	IIE only
NUM_PKGS_HIT_INTR_TOTAL	Number of Zone 1 and Zone 2 packages failed by the IIE	CDF See DTN: SN0006T0502900.002 [150856]	IIE only
NUM_PKGS_INTR_ZONE2	Number of Zone 2 packages failed by the IIE	if((NUM_PKGS_HIT_INTR_TOTAL – Indirect_Release_Zone1.NUM_PKGS_INTR_ZONE1) >0,(NUM_PKGS_HIT_INTR_TOTAL – Indirect_Release_Zone1.NUM_PKGS_INTR_ZONE1), 0)	IIE only
NUM_CSNF_PKGS_ZONE2	Number of CSNF Zone 2 packages	round((Total_CSNF_Packages/(Total_CDSP_Packages+Total_CSNF_Packages))*NUM_PKGS_INTR_ZONE2)	IIE only
NUM_CDSP_PKGS_ZONE2	Number of CDSP Zone 2 packages	NUM_PKGS_INTR_ZONE2-NUM_CSNF_PKGS_ZONE2	IIE only
WP_Breach_Area_Zone2_Pkgs	Zone 2 Package Breach Area	Truncated Log-Normal Distribution: true mean: 10{cm2} true S.D.: 1{cm2} minimum: 1{cm2} maximum: 1.9e+4{cm2}	IIE only
Zone2_Patch_Fraction_CSNF	Zone 2 CSNF WP patch fraction	if((WP_Breach_Area_Zone2_Pkgs/WP_SA_CSNF)<= 1,WP_Breach_Area_Zone2_Pkgs/WP_SA_CSNF,1)	IIE only
Zone2_Patch_Fraction_CDSP	Zone 2 CSNF WP patch fraction	if((WP_Breach_Area_Zone2_Pkgs/WP_SA_CDSP)<=1,WP_Breach_Area_Zone2_Pkgs/WP_SA_CDSP,1)	IIE only
WP_Fraction_Failed_Zone2	Outer Barrier Failure Fraction for Zone 2 WPs	if(ETime >= Indirect_Release_Zone1.Indirect_Failure_Switch, 1, 0)	IIE only

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
CSNF_WP_BIN5_2	Number of Zone 2 CSNF Bin 5 WPs in the mean infiltration scenario	if(((NUM_CSNF_PKGS_ZONE2 – Num_CSNF_WP.Num_CSNF_WP[Bin_4] – Num_CSNF_WP.Num_CSNF_WP[Bin_3]) > Num_CSNF_WP.Num_CSNF_WP[Bin_2]), (NUM_CSNF_PKGS_ZONE2 – Num_CSNF_WP.Num_CSNF_WP[Bin_4] – Num_CSNF_WP.Num_CSNF_WP[Bin_3] – Num_CSNF_WP.Num_CSNF_WP[Bin_2]), 0)	IIE only
CSNF_WP_BIN5_3	Number of Zone 2 CSNF Bin 5 WPs in the high infiltration scenario	if((CSNF_WP_BIN4 == Num_CSNF_WP.Num_CSNF_WP[Bin_4]) , (if (((NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4) > Num_CSNF_WP.Num_CSNF_WP[Bin_5]) , Num_CSNF_WP.Num_CSNF_WP[Bin_5] , (NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4))) , 0)	IIE only
CSNF_WP_BIN4_2	Number of Zone 2 CSNF Bin 4 WPs in the mean infiltration scenario	if((NUM_CSNF_PKGS_ZONE2-Num_CSNF_WP.Num_CSNF_WP[Bin_4]) > 0,(Num_CSNF_WP.Num_CSNF_WP[Bin_4]),NUM_CSNF_PKGS_ZONE2)	IIE only
CSNF_WP_BIN4_3	Number of Zone 2 CSNF Bin 4 WPs in the high infiltration scenario	if((NUM_CSNF_PKGS_ZONE2-Num_CSNF_WP.Num_CSNF_WP[Bin_4]) > 0,(Num_CSNF_WP.Num_CSNF_WP[Bin_4]),NUM_CSNF_PKGS_ZONE2)	IIE only
CSNF_WP_BIN3_2	Number of Zone 2 CSNF Bin 3 WPs in the mean infiltration scenario	if((CSNF_WP_BIN4 == Num_CSNF_WP.Num_CSNF_WP[Bin_4]) , (if (((NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4) > Num_CSNF_WP.Num_CSNF_WP[Bin_3]) , Num_CSNF_WP.Num_CSNF_WP[Bin_3] , (NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4))), 0)	IIE only
CSNF_WP_BIN3_3	Number of Zone 2 CSNF Bin 3 WPs in the high infiltration scenario	if((CSNF_WP_BIN5 == Num_CSNF_WP.Num_CSNF_WP[Bin_5]) , (if (((NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4 - CSNF_WP_BIN5) > Num_CSNF_WP.Num_CSNF_WP[Bin_3]) , Num_CSNF_WP.Num_CSNF_WP[Bin_3] , (NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4 – CSNF_WP_BIN5))), 0)	IIE only
CSNF_WP_BIN2_1	Number of Zone 2 CSNF Bin 2 WPs in the low infiltration scenario	if((NUM_CSNF_PKGS_ZONE2-Num_CSNF_WP.Num_CSNF_WP[Bin_1]) > 0,(NUM_CSNF_PKGS_ZONE2-Num_CSNF_WP.Num_CSNF_WP[Bin_1]),0)	IIE only

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
CSNF_WP_BIN2_2	Number of Zone 2 CSNF Bin 2 WPs in the mean infiltration scenario	if((CSNF_WP_BIN3 == Num_CSNF_WP.Num_CSNF_WP[Bin_3]) , (if (((NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4 - CSNF_WP_BIN3) > Num_CSNF_WP.Num_CSNF_WP[Bin_2]) , Num_CSNF_WP.Num_CSNF_WP[Bin_2] , (NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4 – CSNF_WP_BIN3))), 0)	IIE only
CSNF_WP_BIN2_3	Number of Zone 2 CSNF Bin 2 WPs in the high infiltration scenario	if((CSNF_WP_BIN3 == Num_CSNF_WP.Num_CSNF_WP[Bin_3]), (NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4 - CSNF_WP_BIN5 – CSNF_WP_BIN3), 0)	IIE only
CSNF_WP_BIN1_1	Number of Zone 2 CSNF Bin 1 WPs in the low infiltration scenario	if((NUM_CSNF_PKGS_ZONE2-Num_CSNF_WP.Num_CSNF_WP[Bin_1]) > 0,(Num_CSNF_WP.Num_CSNF_WP[Bin_1]) ,NUM_CSNF_PKGS_ZONE2)	IIE only
CSNF_WP_BIN1_2	Number of Zone 2 CSNF Bin 1 WPs in the mean infiltration scenario	if((CSNF_WP_BIN5 == Num_CSNF_WP.Num_CSNF_WP[Bin_5]), (NUM_CSNF_PKGS_ZONE2 – CSNF_WP_BIN4 - CSNF_WP_BIN3 – CSNF_WP_BIN2 – CSNF_WP_BIN5),0)	IIE only
CDSP_WP_BIN5_2	Number of Zone 2 CDSP Bin 5 WPs in the mean infiltration scenario	if(((NUM_CDSP_PKGS_ZONE2 – Num_CDSP_WP.Num_CDSP_WP[Bin_4] – Num_CDSP_WP.Num_CDSP_WP[Bin_3]) > Num_CDSP_WP.Num_CDSP_WP[Bin_2]), (NUM_CDSP_PKGS_ZONE2 – Num_CDSP_WP.Num_CDSP_WP[Bin_4] – Num_CDSP_WP.Num_CDSP_WP[Bin_3] – Num_CDSP_WP.Num_CDSP_WP[Bin_2]), 0)	IIE only
CDSP_WP_BIN5_3	Number of Zone 2 CDSP Bin 5 WPs in the high infiltration scenario	if((CDSP_WP_BIN4 == Num_CDSP_WP.Num_CDSP_WP[Bin_4]) , (if (((NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4) > Num_CDSP_WP.Num_CDSP_WP[Bin_5]) , Num_CDSP_WP.Num_CDSP_WP[Bin_5] , (NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4))) , 0)	IIE only
CDSP_WP_BIN4_2	Number of Zone 2 CDSP Bin 4 WPs in the mean infiltration scenario	if((NUM_CDSP_PKGS_ZONE2-Num_CDSP_WP.Num_CDSP_WP[Bin_4]) > 0,(Num_CDSP_WP.Num_CDSP_WP[Bin_4]),NUM_CDSP_PKGS_ZONE2)	IIE only
CDSP_WP_BIN4_3	Number of Zone 2 CDSP Bin 4 WPs in the high infiltration scenario	if((NUM_CDSP_PKGS_ZONE2-Num_CDSP_WP.Num_CDSP_WP[Bin_4]) > 0,(Num_CDSP_WP.Num_CDSP_WP[Bin_4]),NUM_CDSP_PKGS_ZONE2)	IIE only

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
CDSP_WP_BIN3_2	Number of Zone 2 CDSP Bin 3 WPs in the mean infiltration scenario	if((CDSP_WP_BIN4 == Num_CDSP_WP.Num_CDSP_WP[Bin_4]) , (if (((NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4) > Num_CDSP_WP.Num_CDSP_WP[Bin_3]) , Num_CDSP_WP.Num_CDSP_WP[Bin_3] , (NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4))), 0)	IIE only
CDSP_WP_BIN3_3	Number of Zone 2 CDSP Bin 3 WPs in the high infiltration scenario	if((CDSP_WP_BIN5 == Num_CDSP_WP.Num_CDSP_WP[Bin_5]) , (if (((NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4 - CDSP_WP_BIN5) > Num_CDSP_WP.Num_CDSP_WP[Bin_3]) , Num_CDSP_WP.Num_CDSP_WP[Bin_3] , (NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4 – CDSP_WP_BIN5))), 0)	IIE only
CDSP_WP_BIN2_1	Number of Zone 2 CDSP Bin 2 WPs in the low infiltration scenario	if((NUM_CDSP_PKGS_ZONE2-Num_CDSP_WP.Num_CDSP_WP[Bin_1]) > 0,(NUM_CDSP_PKGS_ZONE2-Num_CDSP_WP.Num_CDSP_WP[Bin_1]),0)	IIE only
CDSP_WP_BIN2_2	Number of Zone 2 CDSP Bin 2 WPs in the mean infiltration scenario	if((CDSP_WP_BIN3 == Num_CDSP_WP.Num_CDSP_WP[Bin_3]) , (if (((NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4 - CDSP_WP_BIN3) > Num_CDSP_WP.Num_CDSP_WP[Bin_2]) , Num_CDSP_WP.Num_CDSP_WP[Bin_2] , (NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4 – CDSP_WP_BIN3))), 0)	IIE only
CDSP_WP_BIN2_3	Number of Zone 2 CDSP Bin 2 WPs in the high infiltration scenario	if((CDSP_WP_BIN3 == Num_CDSP_WP.Num_CDSP_WP[Bin_3]), (NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4 - CDSP_WP_BIN5 – CDSP_WP_BIN3), 0)	IIE only
CDSP_WP_BIN1_1	Number of Zone 2 CDSP Bin 1 WPs in the low infiltration scenario	if((NUM_CDSP_PKGS_ZONE2-Num_CDSP_WP.Num_CDSP_WP[Bin_1]) > 0,(Num_CDSP_WP.Num_CDSP_WP[Bin_1]),NUM_CDSP_PKGS_ZONE2)	IIE only
CDSP_WP_BIN1_2	Number of Zone 2 CDSP Bin 1 WPs in the mean infiltration scenario	if((CDSP_WP_BIN5 == Num_CDSP_WP.Num_CDSP_WP[Bin_5]),(NUM_CDSP_PKGS_ZONE2 – CDSP_WP_BIN4 - CDSP_WP_BIN3 – CDSP_WP_BIN2 – CDSP_WP_BIN5),0)	IIE only
CSNF_WP_BIN5	Number of Zone 2 CSNF Bin 5 WPs in the current realization	If Infiltration_Scenario == 1 then 0 elseif Infiltration_Scenario == 2 then CSNF_WP_BIN5_2 else CSNF_WP_BIN5_3	IIE only

Table 6-136. Parameter Details for Supplemental TSPA-SR Parameters (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs	Applicability
BIN_Probability_Mean	Discrete distribution used to randomly assign the IIE released mass to a specific bin, mean infiltration scenario	See Table 6-135	IIE Only
CSNF_WP_BIN4	Number of Zone 2 CSNF Bin 4 WPs in the current realization	If Infiltration_Scenario == 1 then 0 elseif Infiltration_Scenario == 2 then CSNF_WP_BIN4_2 else CSNF_WP_BIN4_3	IIE only
CSNF_WP_BIN3	Number of Zone 2 CSNF Bin 3 WPs in the current realization	If Infiltration_Scenario == 1 then 0 elseif Infiltration_Scenario == 2 then CSNF_WP_BIN3_2 else CSNF_WP_BIN3_3	IIE only
CSNF_WP_BIN2	Number of Zone 2 CSNF Bin 2 WPs in the current realization	If Infiltration_Scenario == 1 then CSNF_WP_BIN2_1 elseif Infiltration_Scenario == 2 then CSNF_WP_BIN2_2 else CSNF_WP_BIN2_3	IIE only
CSNF_WP_BIN1	Number of Zone 2 CSNF Bin 1 WPs in the current realization	If Infiltration_Scenario == 1 then CSNF_WP_BIN2_1 elseif Infiltration_Scenario == 2 then CSNF_WP_BIN2_2 else 0	IIE only
CDSP_WP_BIN5	Number of Zone 2 CDSP Bin 5 WPs in the current realization	If Infiltration_Scenario == 1 then 0 elseif Infiltration_Scenario == 2 then CDSP_WP_BIN5_2 else CDSP_WP_BIN5_3	IIE only
CDSP_WP_BIN4	Number of Zone 2 CDSP Bin 4 WPs in the current realization	If Infiltration_Scenario == 1 then 0 elseif Infiltration_Scenario == 2 then CDSP_WP_BIN4_2 else CDSP_WP_BIN4_3	IIE only
CDSP_WP_BIN3	Number of Zone 2 CDSP Bin 3 WPs in the current realization	If Infiltration_Scenario == 1 then 0 elseif Infiltration_Scenario == 2 then CDSP_WP_BIN3_2 else CDSP_WP_BIN3_3	IIE only
CDSP_WP_BIN2	Number of Zone 2 CDSP Bin 2 WPs in the current realization.	If Infiltration_Scenario == 1 then CDSP_WP_BIN2_1 elseif Infiltration_Scenario == 2 then CDSP_WP_BIN2_2 else CDSP_WP_BIN2_3	IIE only
CDSP_WP_BIN1	Number of Zone 2 CDSP Bin 1 WPs in the current realization	If Infiltration_Scenario == 1 then CDSP_WP_BIN2_1 elseif Infiltration_Scenario == 2 then CDSP_WP_BIN2_2 else 0	IIE only

Results and Verification

The discussion below presents the results of the simulation and the verification of the model implementation. The results of every sub-component of the IIE model are not presented here. The model logic is nearly identical to the nominal case logic and the results of the model calculations are very similar. The model logic in many instances was copied from the nominal case model and modified to use bin selected parameters. The model logic for all calculated output is the same as the nominal case scenario logic and thus the verification of each model sub-component, done in the previous sections of this document, are not repreated in this section.

For the median value simulation, the mean infiltration scenario prevails. Accordingly, in container \TSPA_Model\Disruptive_Events\Indirect_Release_Zone1\Bin_Selector_Parameters\ the value of *BIN_Selector* should equal the value assigned to *BIN_Probability_Mean*. The value assigned to *BIN_Probability_Mean* for the median value simulation is 4. The value of *BIN_Selector* was extracted from the simulation results. Its value was also 4. This confirms that the model logic implemented for *BIN_Selector* works as intended and furthermore it verifies that the model passes values correctly from one component of the TSPA_SR model to another.

For the median value simulation the IIE should occur at 50 years. A plot of the number of failed Zone 1 CSFN and CDSP packages confirms that the event occurred between the 31.25 year time step and the 62.5 year time step. Figure 6-212 shows these results.

The time step interval shown in Figure 6-212 is 31.25 years for the first 250 years. The time step after the indirect intrusive indirect event is scheduled to occur is 62.5 years. An analysis of the plotted data shows that the number of CSNF waste packages failed jumps from 0 at 31.25 years to 130 at 62.5 years. Furthermore the number of failed CDSP waste packages jumps from 0 at 31.25 years to 65 at 62.5 years. It is concluded that the intrusive indirect event occurs at the scheduled time.

Continuing with the analysis, the median value for *Num_Pkgs_Intr_Zone1* is 195. The 62.5 year time step indicates that a total of 195 packages have been failed. Furthermore, the percentage of CSNF waste packages in the repository is 67 percent (7,860 out of 11,770) and 67 percent of the failed packages (128 out of 192) are CSNF waste packages. Similarly, the median value for *Num_Pkgs_Intr_Zone2* (in container \TSPA_Model\Disruptive_Events\ Indirect_Release_ Zone2\) is 1680.84. Subtracting out the 195 Zone 1 packages leaves 1485.84 Zone 2 packages (parameter *NUM_PKGS_INTR_ZONE2*), which partition out to 992 CSNF packages (parameter *NUM_CSNF_PKGS_ZONE2*) and 493.84 CDSP packages (parameter *NUM_CDSP_PKGS_ ZONE2*).

In the nominal case there would be 4009 CSNF packages in Bin 4, therefore all 992 Zone 2 CSNF packages are placed in Bin 4 (parameter *CSNF_WP_BIN4_2*). Likewise, in the nominal case there would be 2066 CSNF packages in Bin 4, therefore all 493.84 Zone 2 CDSP packages are placed in Bin 4. A check of the Bin 4 source terms finds for CSNF that there are 24 always drip packages, 96 intermittant drip packages, and 872 no drip packages, the total of which matches the 992 Zone 2 CSNF packages. A check of the Bin 4 source terms finds for CDSP that there are 12 always drip packages, 46 intermittant drip packages, and 435.84 no drip packages (note this value is rounded up to 436 by the source term element), the total of which matches the 493.84 Zone 2 CDSP packages.

Having verified that the timing of the indirect intrusive indirect event and the number of packages it impacts are implemented correctly, the next result of significance is to demonstrate that the mass released during an intrusive indirect event supplements the correct nominal case bin.

For CSNF Zone 1 waste packages, the annual mass flux of each radionuclide released from the EBS after an intrusive indirect event is given by *CSNF_Intr_RelRate*. An evaluation of the five parameters, *CSNF_Bin1_Input, CSNF_Bin1_Input, CSNF_Bin1_Input, CSNF_Bin1_Input, and CSNF_Bin1_Input* show that bins 1, 2, 3, and 5 release no mass from the intrusive indirect event and bin 4 releases mass equivalent to the cummulative mass released from the CSNF intrusive indirect event waste packages. The same holds true for CDSP Zone 1 waste packge release. It is concluded that the bin selector logic for assigning the mass release works properly. Furthermore, since other bin selector parameters, *Q_Evap_Num, Q_Flux_Denom, RH_Invert, Invert_Saturation, Invert_Temperature, Temp_WP,* and *QFlux* use the same logic principles, it is concluded that they function properly as well.

Figure 6-213 shows the annual flux of each radionuclide released from CSNF waste packages impacted by and intrusive indirect event. The radionuclides released at the first time step after failure, 62.5 years, is the fraction of the CSNF inventory which is not bound in a matrix, the gap inventory. The entire gap inventory mass is available for transport from the EBS once the packages have failed and is only limited by the initial inventory and solubility constraints. The release of radionuclides from the waste packages continues to occur after the initial impact as the CSNF matrix degrades and frees the bound inventory. In the IIE component this release of radionuclides from the source term to the waste form and on to the invert cell is governed by the parameter, *CSNF_Deg_Rate* and the solubility of each radionuclide species. An analysis was performed to determine the significance of each of these parameters.

Figure 6-214 shows the time history result of *CSNF_Deg_Rate*. It should be noted that the high temperatures early on in the repository life cause the CSNF matrix degradation rate to be very high. For instance at 62.5 years the CSNF waste degradation rate reaches it peak and is 0.1247 yr^{-1}. In comparison the degradation rate at 20,000 years, when the temperature has returned to near ambient conditions, is 0.01099 yr^{-1}. Using a sustained degradation rate equivalent to the 62.5 year value, over 99.9 percent of the CSNF waste form would be degraded in 62.5 years (two time steps after the waste packages have failed) as opposed to 49.4 percent at the 20,000 year rate. Both of these rates are suggest that the CSNF waste form will not contribute significantly to reduce the mass released from the source term. It is concluded that the transport of radionuclides from the source term is not limited by waste form dissolution and is limited only by solubility constraints and the initial inventory.

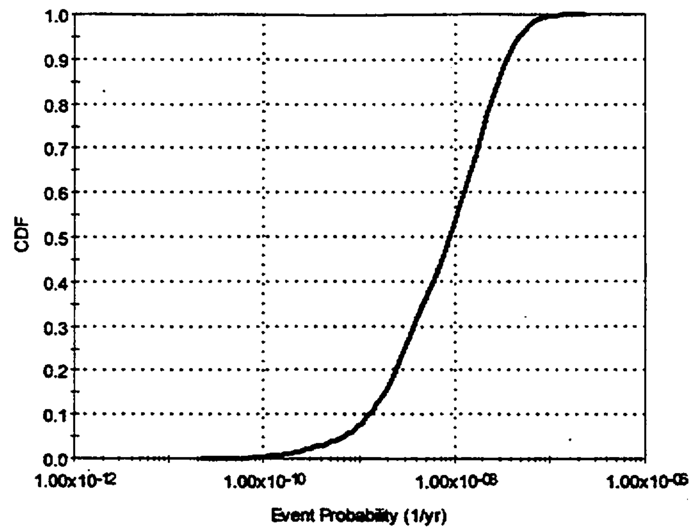


Figure 6-194. Cumulative Distribution Function for Intrusive Indirect Events

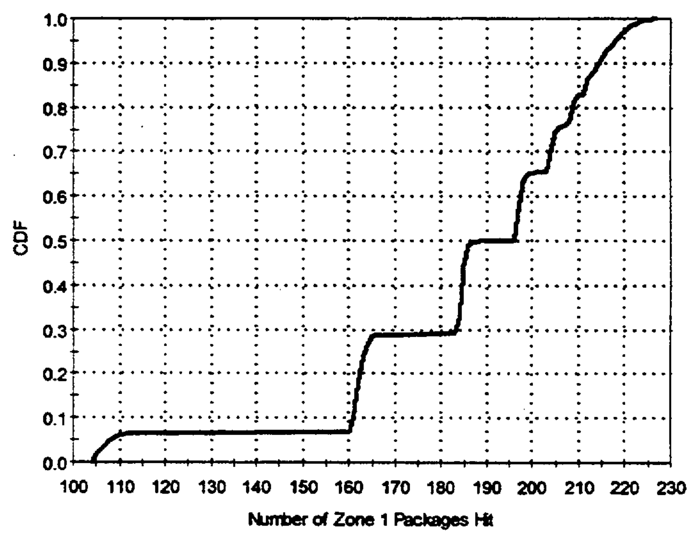


Figure 6-195. Cumulative Distribution Function for the Number of Zone 1 Waste Packages Hit by an Intrusive Indirect Event

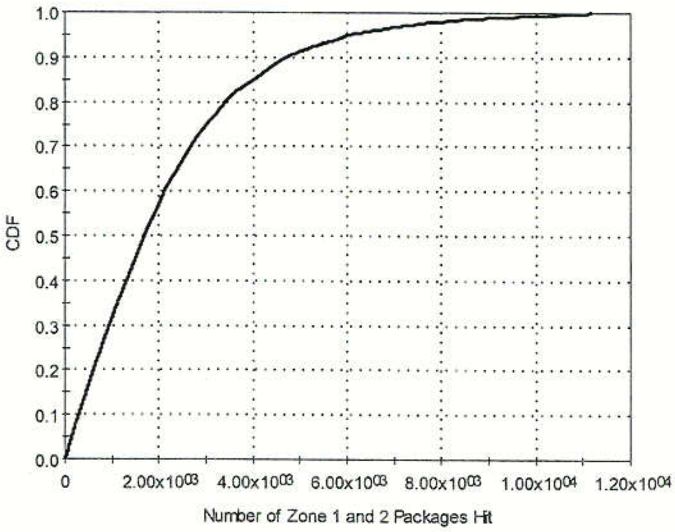


Figure 6-196. Cumulative Distribution Function for the Number of Zone 1 and 2 Waste Packages Hit by an Intrusive Indirect Event

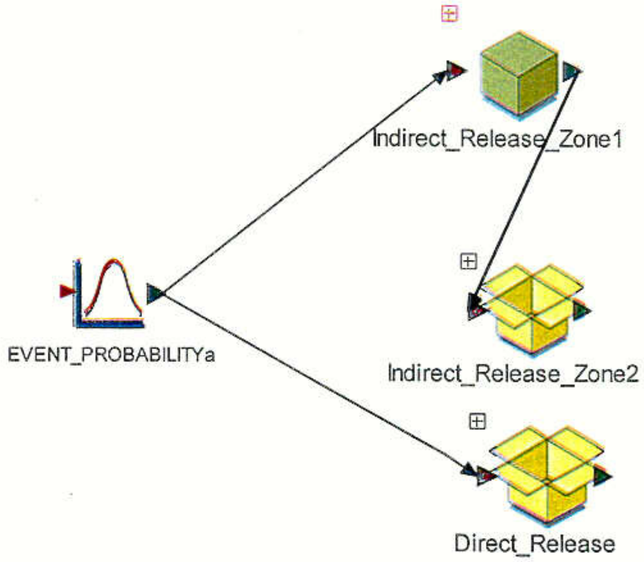


Figure 6-197. The Disruptive Event Component in the TSPA-SR Model

C30

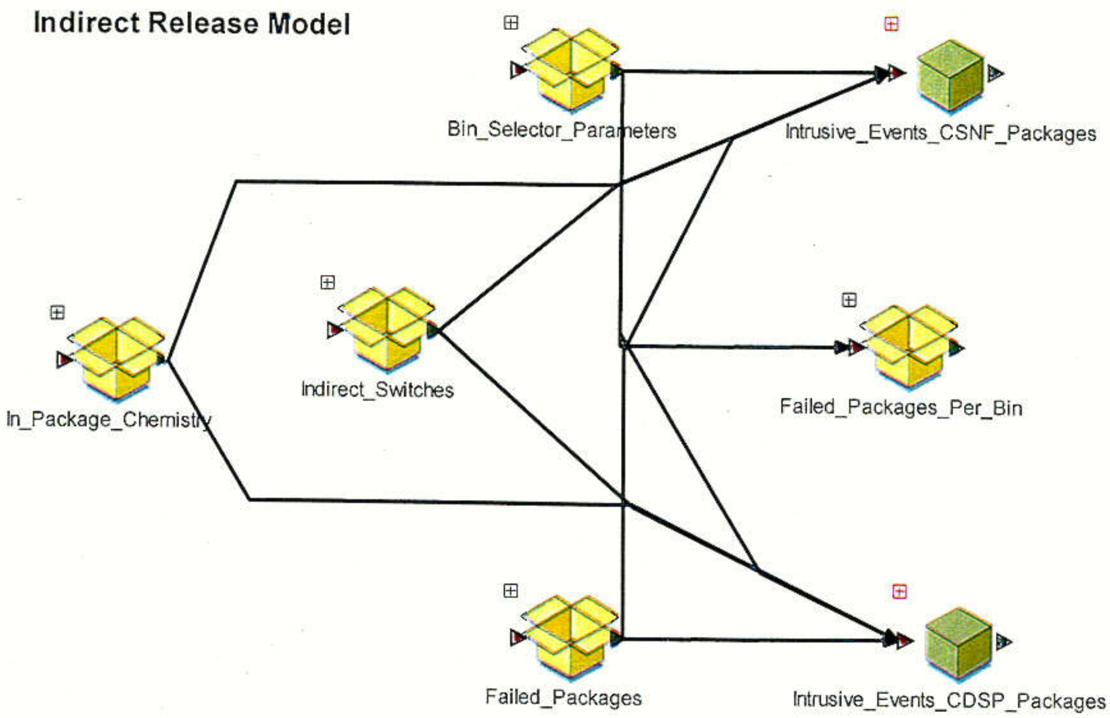


Figure 6-198. Zone 1 Intrusive Indirect Release Component in the TSPA-SR Model

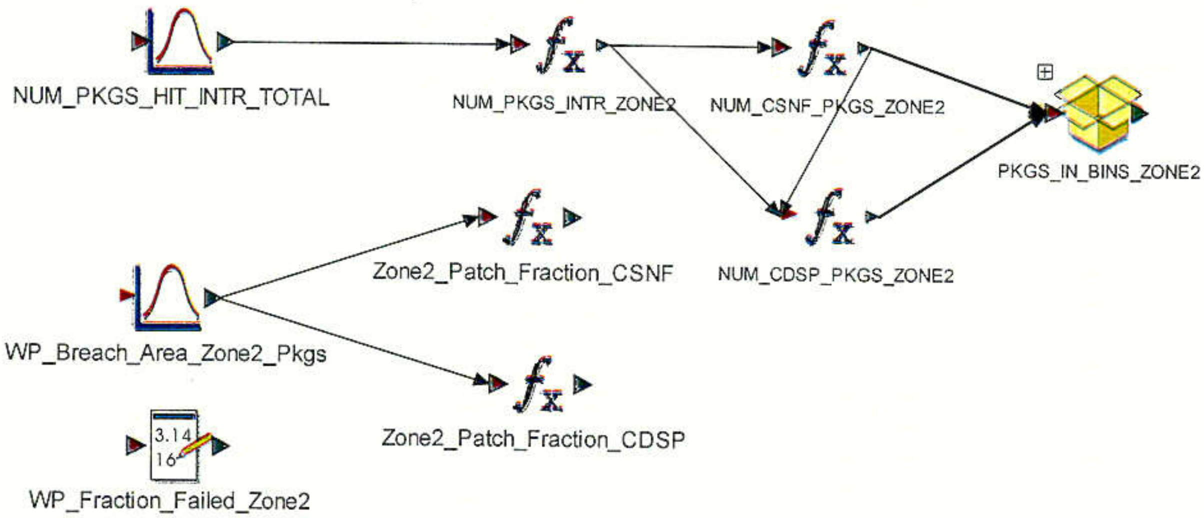


Figure 6-199. Zone 2 Intrusive Indirect Release Component in the TSPA-SR Model

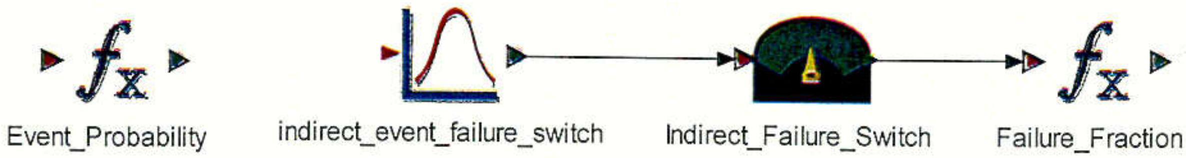
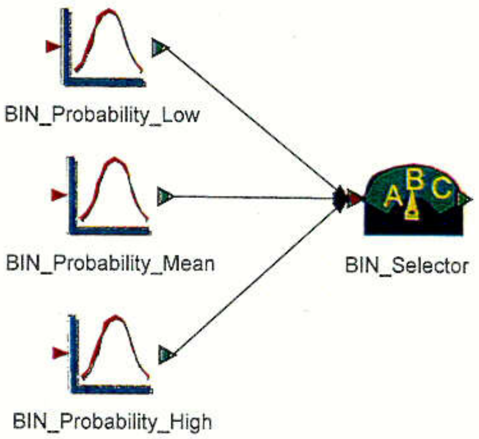


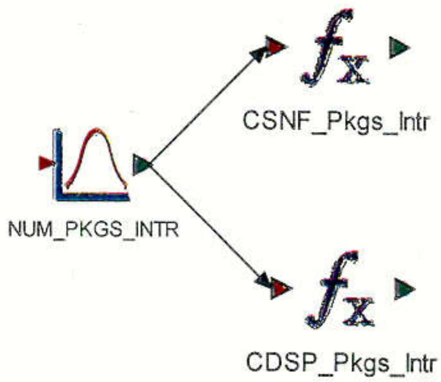
Figure 6-200. Intrusive indirect event Initiation Switch

C31



\TSPA_Model\Disruptive_Events\Indirect_Release\Bin_Selector_Parameters\

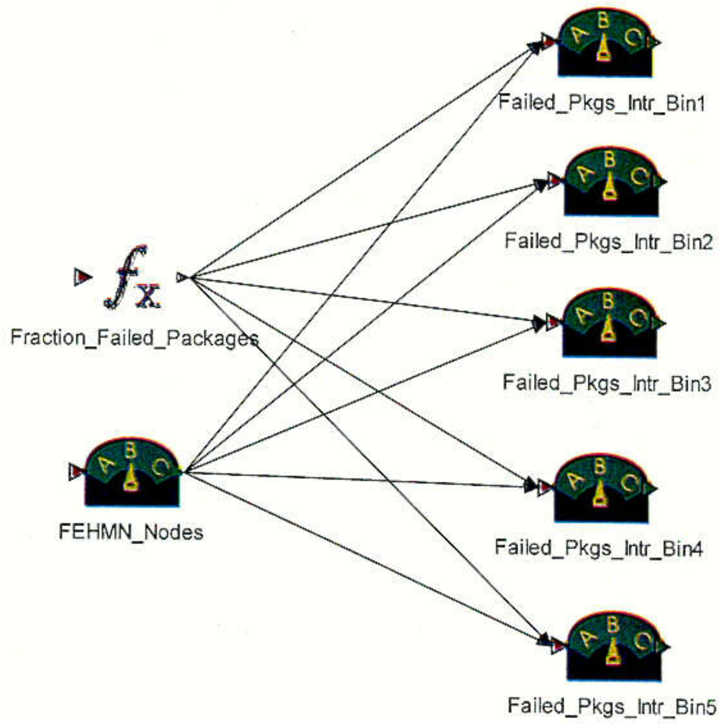
Figure 6-201. The Bin Selector for Intrusive indirect event Releases



\TSPA_Model\Disruptive_Events\Indirect_Release\Failed_Packages\

Figure 6-202. The Calculation of Impacted Packages

C32



\TSPA_Model\Disruptive_Events\Indirect_Release\Failed_Packages_Per_Bin\

Figure 6-203. The FEHMN Nodes Calculation Container for an Intrusive Indirect Release

C33

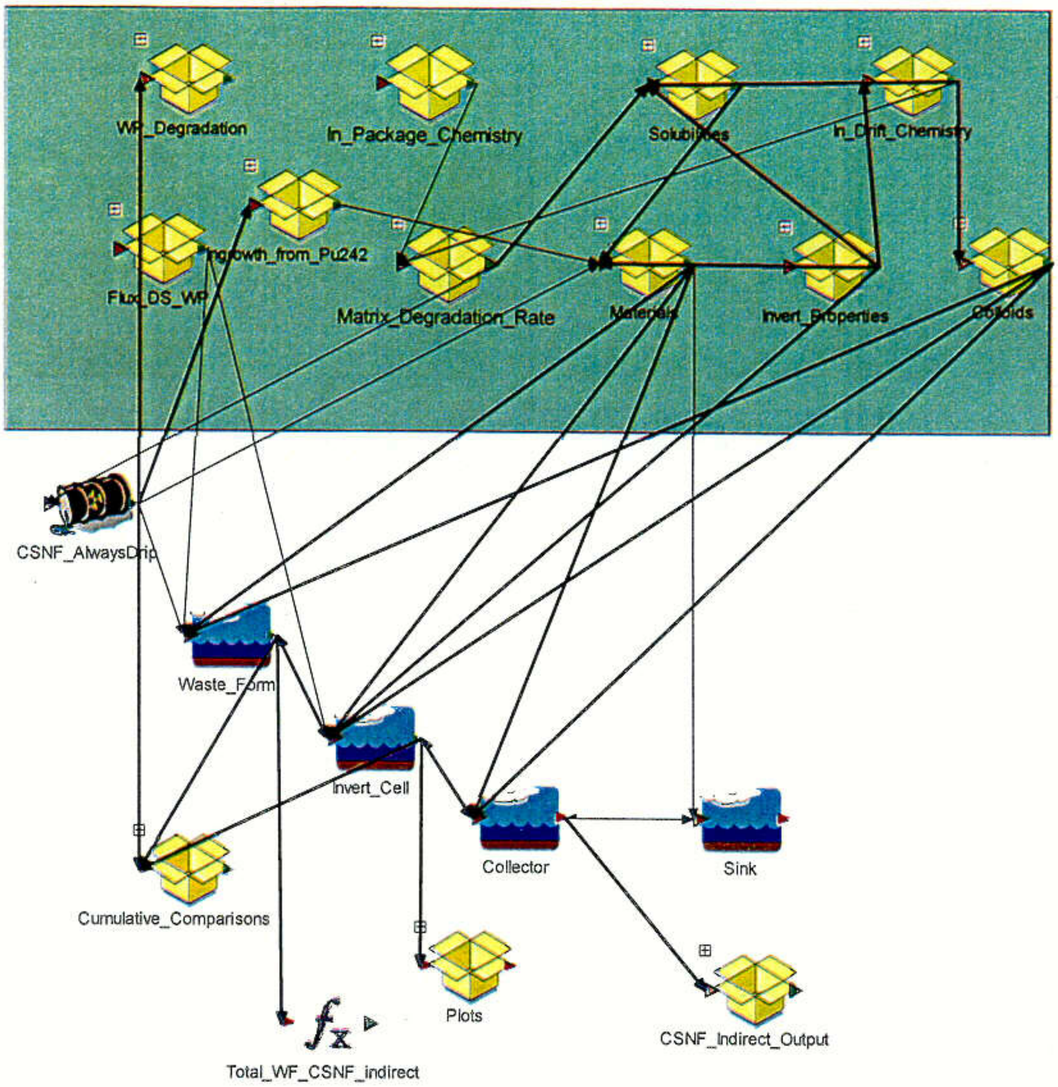


Figure 6-204. An Illustration of the Intrusive Indirect Event Impacting CSNF Waste Package Release from the EBS

C34

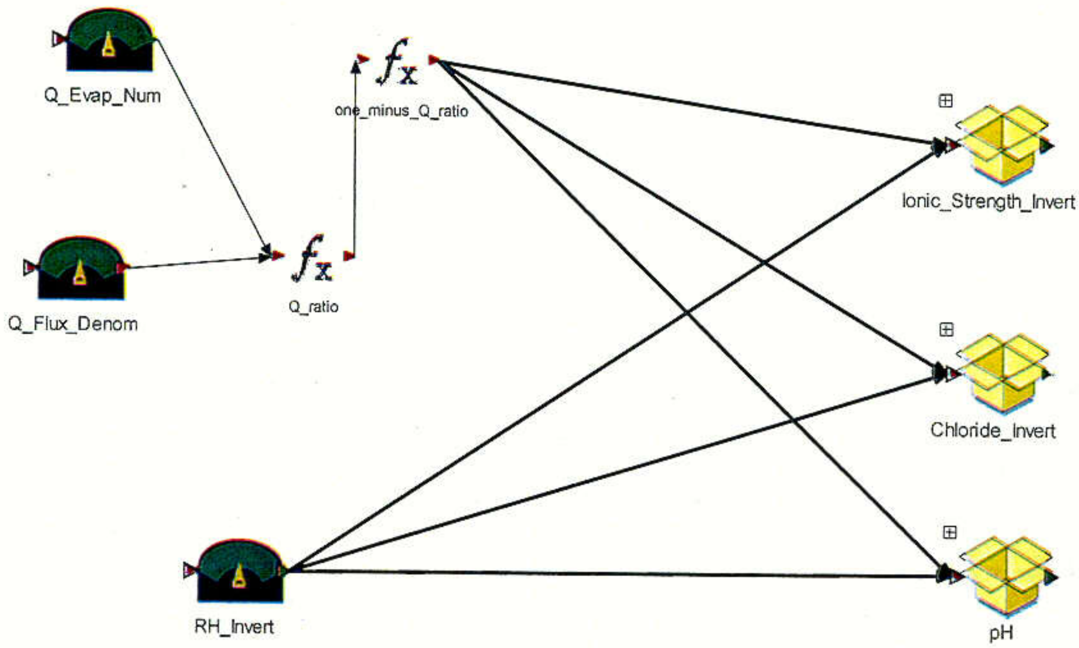


Figure 6-205. The In-Drift Chemistry Component within the Intrusive Indirect Event Component

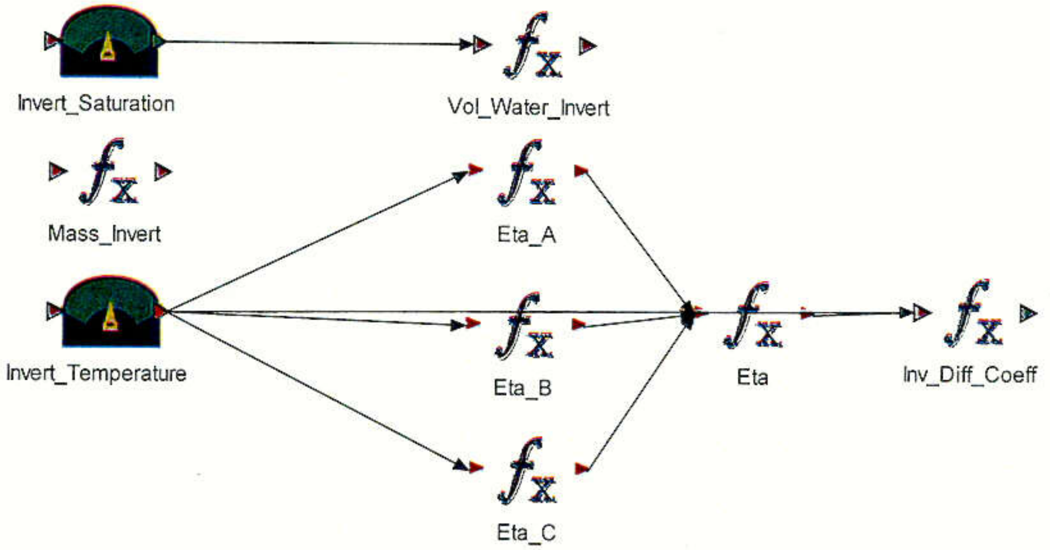


Figure 6-206. Graphical Representation of the Invert Properties Container within the Intrusive Indirect Event Component

C35

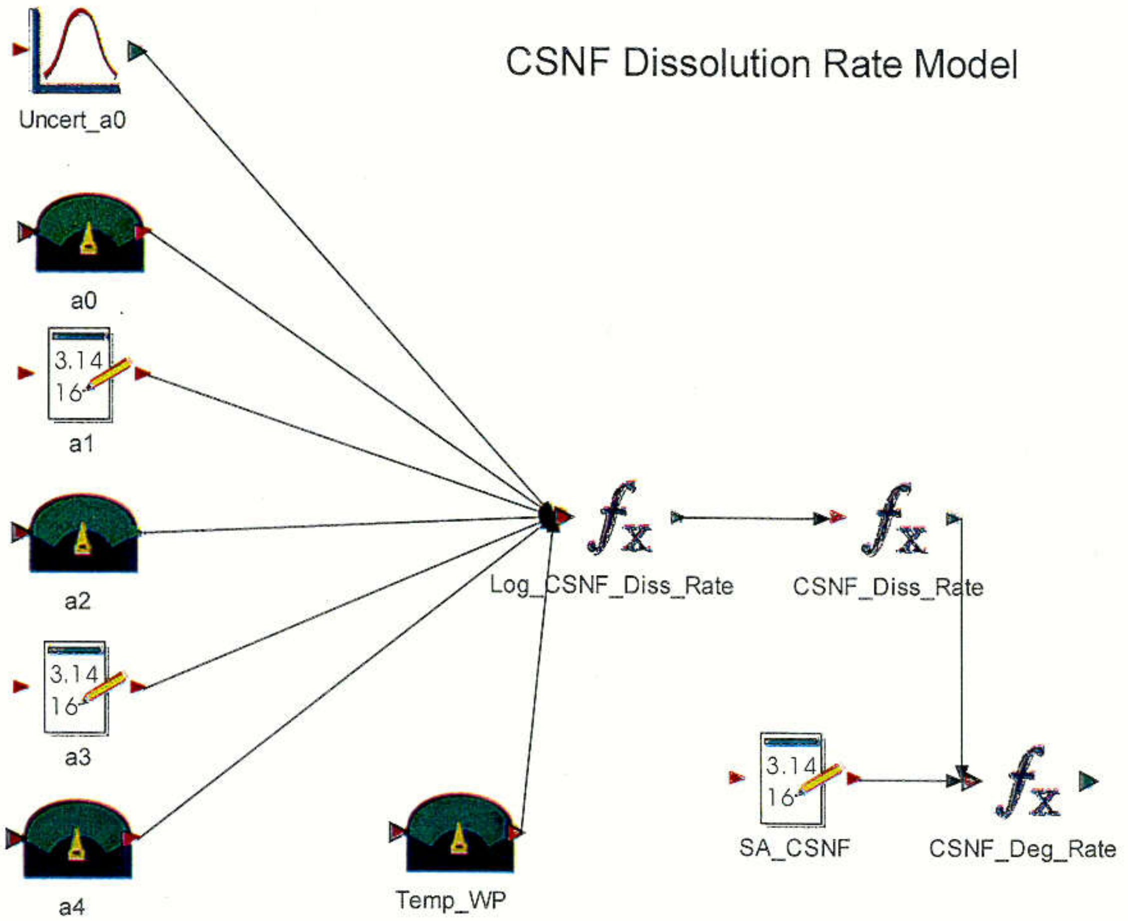
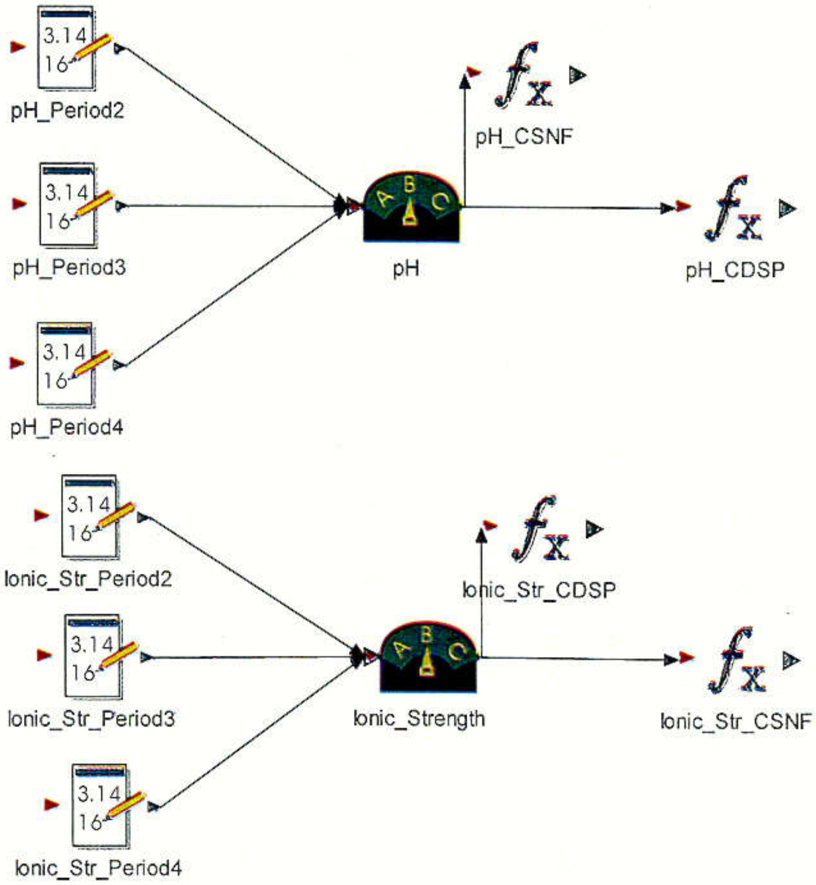


Figure 6-207. The Matrix Degradation Rate Container within the Intrusive Indirect Event Component

\TSPA_Model\Disruptive_Events\Indirect_Release\Intrusive_Events_CSNF_Packages\Flux_DS_WP\

Figure 6-208. Seepage Flux Through the Drift in the Intrusive Indirect Event Component of the TSPA-SR Model

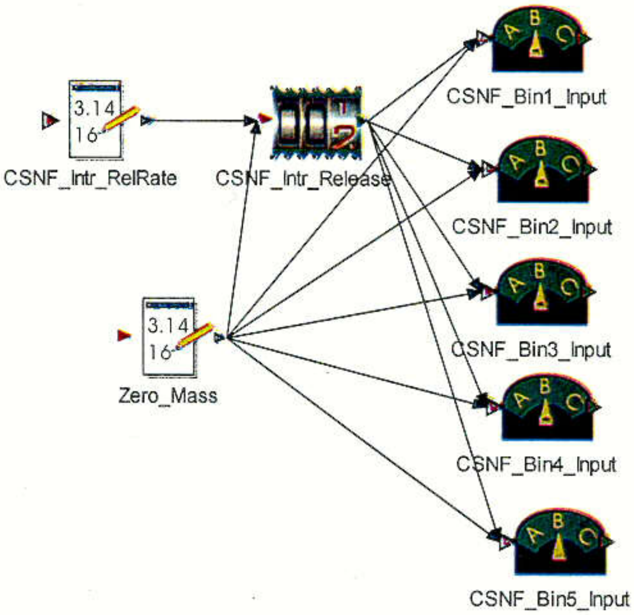
C36



\TSPA_Model\Disruptive_Events\Indirect_Release\In_Package_Chemistry\

Figure 6-209. In-Package Chemistry Component Implementation for Intrusive Indirect Releases

C37



\TSPA_Model\Disruptive_Events\Indirect_Release\Intrusive_Events_CSNF_Packages\CSNF_Indirect_Output\

Figure 6-210. Illustration of the Calculation of Mass Released from the EBS to the UZ for an Intrusive Indirect Event

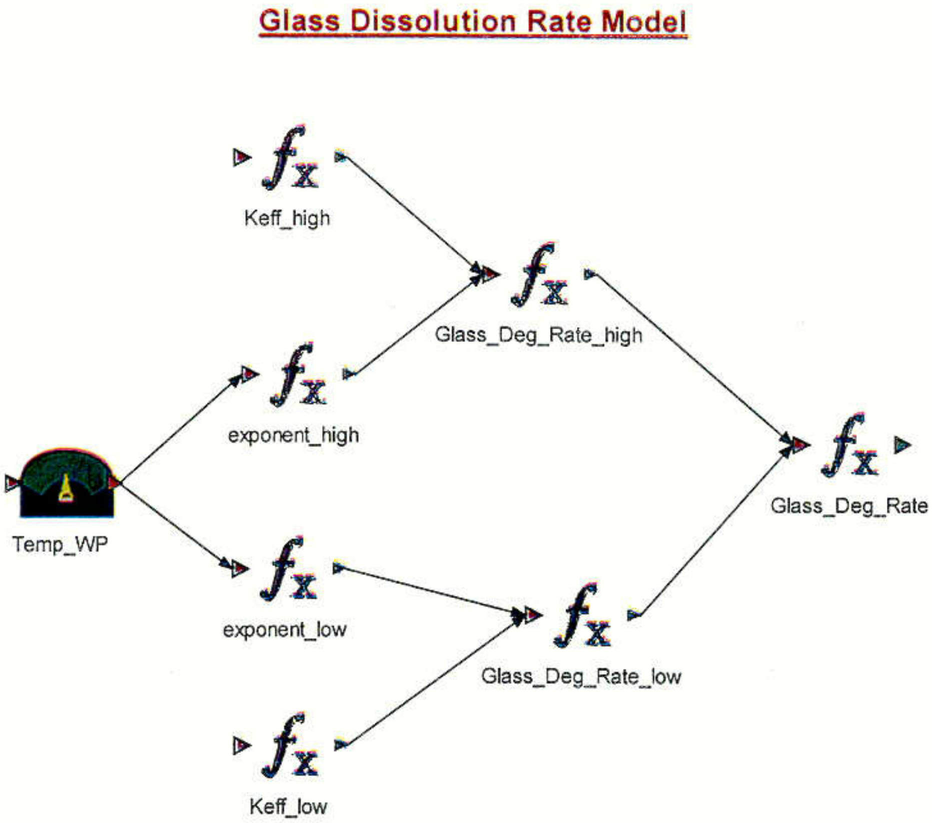


Figure 6-211. Illustration of the Calculation of CDSP Waste Form Dissolution Rates for an Intrusive Indirect Event

C38

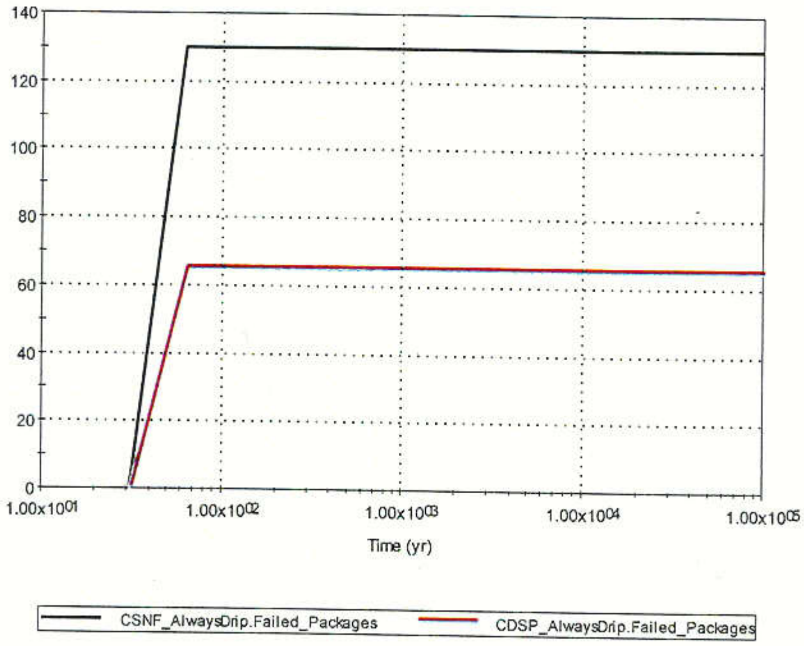


Figure 6-212. Time History Result of the Number of Failed Waste Packages Impacted by an Indirect Intrusive Event: Median Value Simulation

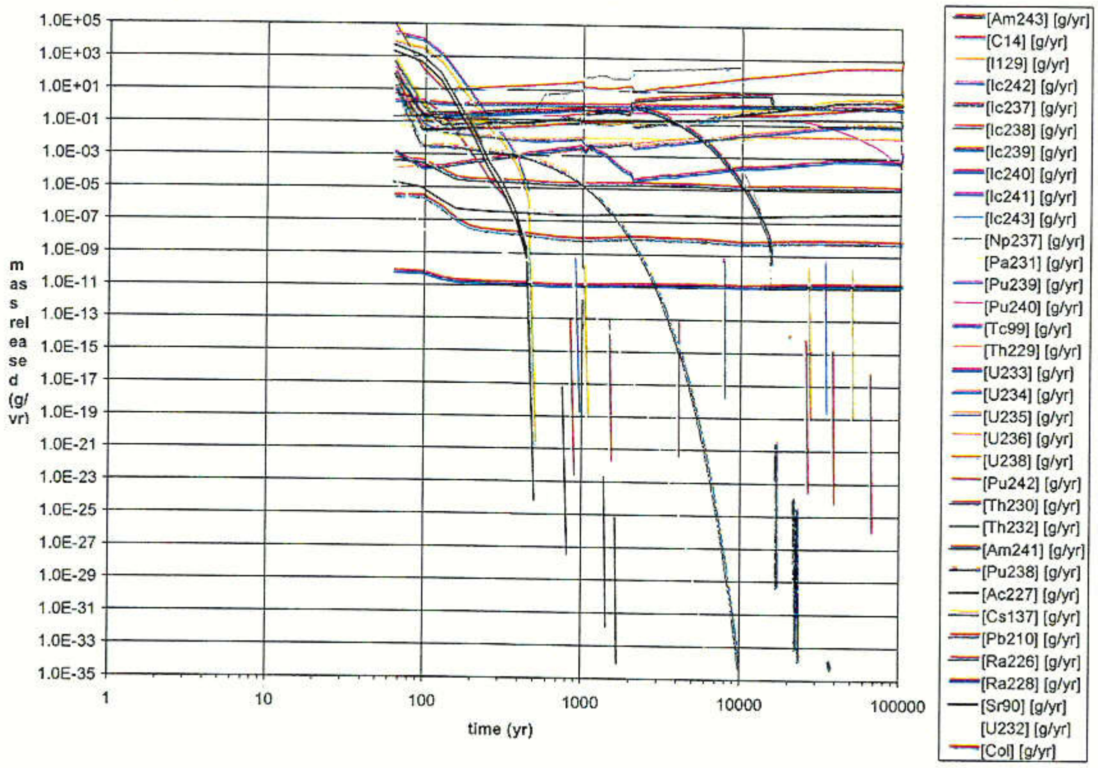


Figure 6-213. Time History Result of the Mass Released from Zone 1 CSNF Packages by an Indirect Intrusive Event: Median Value Simulation

C39

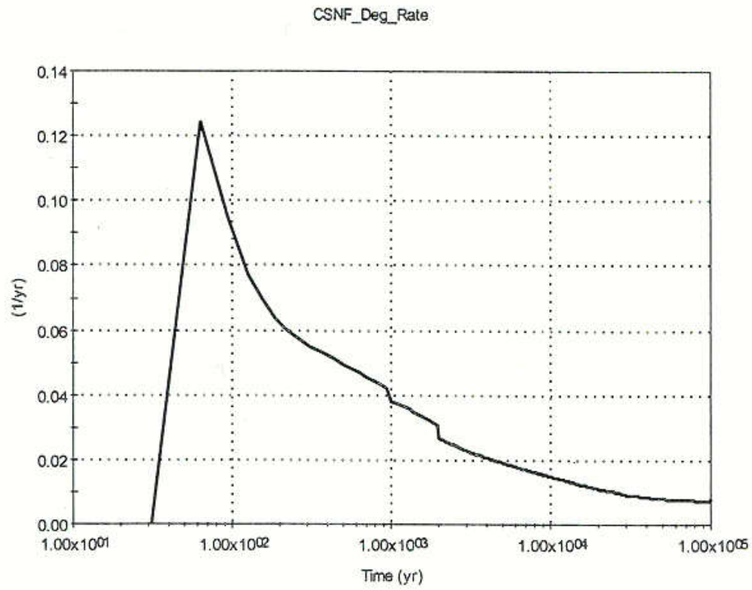


Figure 6-214. Time History Result of the CSNF Matrix Degradation Rate in the Indirect Intrusive Indirect Event Component: Median Value Simulation

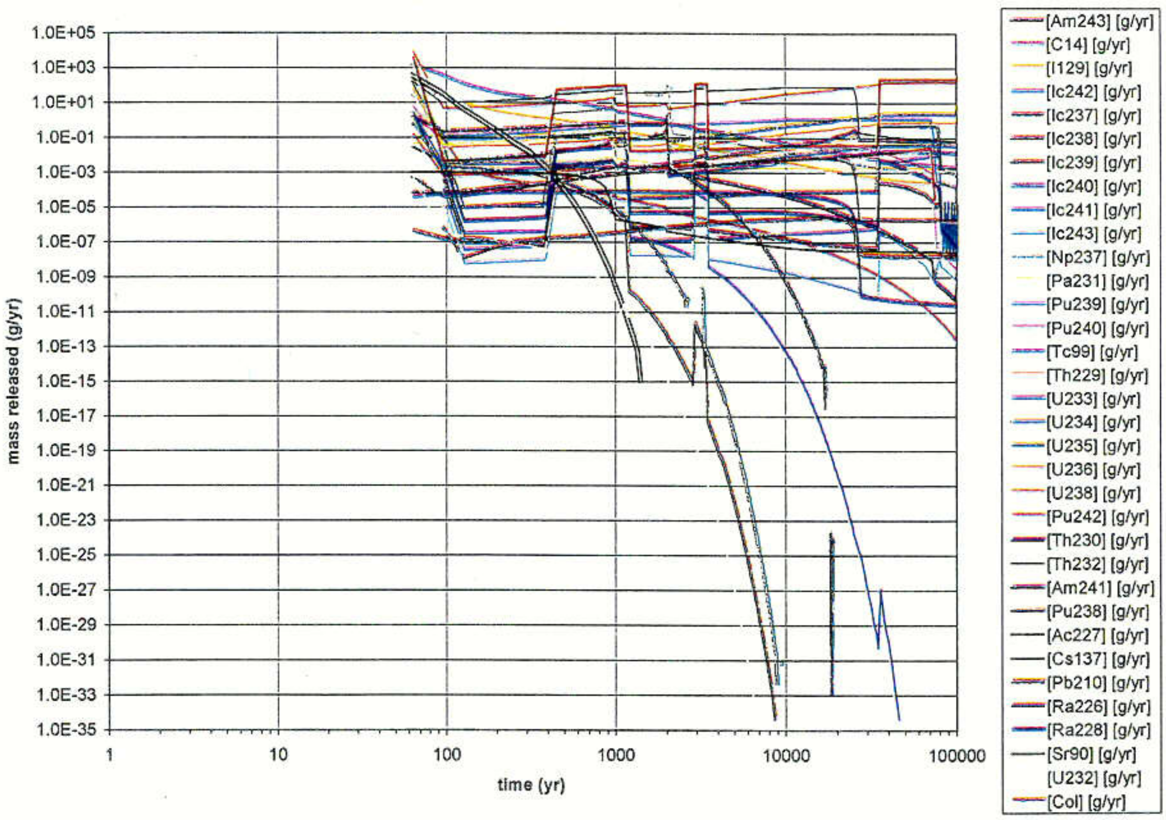


Figure 6-215. Time History Result of the Mass Released from CDSP EBS by an Indirect Intrusive Indirect Event: Median Value Simulation

C40

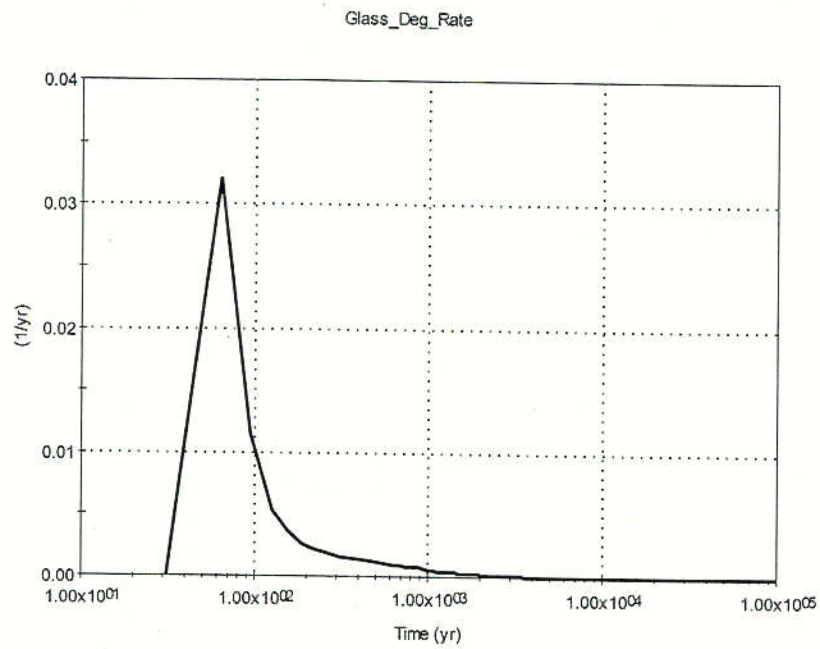


Figure 6-216. Time History Result of the CDSP HLW Glass Degradation Rate in the Indirect Intrusive Indirect Event Component: Median Value Simulation

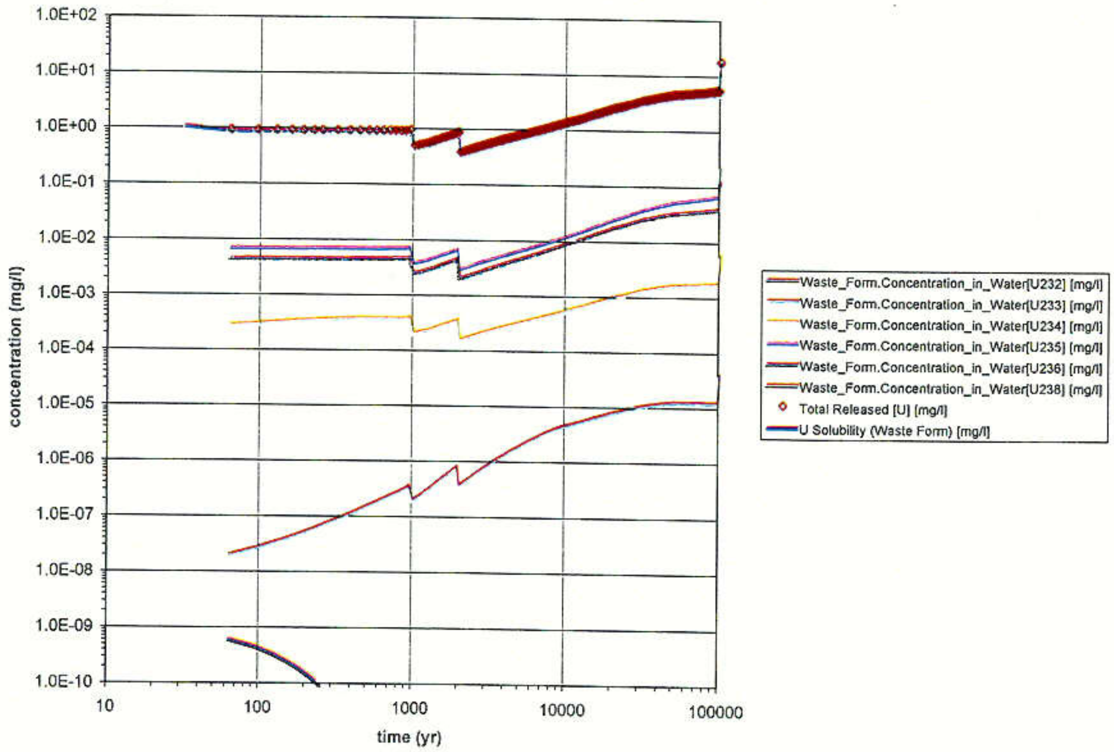


Figure 6-217. Time History Result of the Igneous Intrusive Event CSNF Waste Form Uranium Concentration and Solubility

C41

For CDSP waste packages, the mass of each radionuclide released from the EBS after an intrusive indirect event is shown in Figure 6-215. As the figure indicates, the initial mass release occurs at the 62.5 year time step and is orders of magnitude larger than the mass released at later times. The radionuclides released at the first time step are the fraction of the CDSP inventory in each waste package which is not bound in a matrix (i.e., the free DSNF inventory). The release of radionuclides from the waste packages continues to occur after the initial failure as the CDSP matrix degrades and frees the remaining bound inventory. In the IIE component this release is governed by the parameter, *Glass_Deg_Rate, DSNF_Deg_Rate,* and the solubility of each radionuclide species. An analysis was performed to assess the contribution of each parameter on the release of radionuclides to the unsaturated zone.

Figure 6-216 shows the time history result of *Glass_Deg_Rate*. It should be noted that the high temperatures early on in the repository life cause the CDSP matrix degradation rate to be very high. Accordingly 86 percent of the HLW matrix degrades in the first two time steps (62.5 years) after the waste packages first fail. Furthermore, the DSNF waste form degradation rate is conservatively set to completely degrade in one timestep (as discussed in Section 6.3.4.4 Dissolution Rate Model). It is concluded, for similar reasons discussed previously for the CSNF releases, that the transport of radionuclides from the source term is limited only by solubility constraints and initial inventory limitiations.

To further assess the limitations of the mass released by the IIE, a solubility analysis was performed. The solubility of Uranium, *Solubility_U,* for Zone 1 CSNF packages is plotted with the concentration of each Uranium isotope, ^{232}U, ^{233}U, ^{234}U, ^{235}U, ^{236}U, and ^{238}U, in Figure 6-217. As discussed previously in Section 6.3.4.5 Dissolved Concentration Limits, the solubility of Uranium is a function of pH, temperature, and the carbon dioxide fugacity. Temporal changes in pH and temperature will cause the solubility of Uranium to change over time as shown in the plotted results. The sum total of the Uranium isotope concentrations leaving the waste form and entering the invert should not exceed the waste form solubility limit of Uranium. The orange diamonds in Figure 6-217 plot the sum total of the Uranium isotopes in solution. These markers track the solubility limit directly. Thus the concentration of Uranium isotopes in solution is equal to and does not exceed the solubility limit. This confirms that radionuclide release is solubility limited in the IIE component.

6.3.9.3 Human Intrusion

Overview

The human intrusion scenario is based on a stylized scenario specified in the proposed NRC regulations at 10 CFR Part 63 [64 FR 8640 [101680]], with consideration given to the proposed EPA radiation protection standards at 40 CFR Part 197 [64 FR 46976 [105065]] where they differ from the proposed NRC regulations. Table 6-137 summarizes the human intrusion scenario assumptions outlined in the proposed NRC regulations and identifies areas where the proposed EPA regulations give additional and/or conflicting assumptions. Table 6-137 also summarizes the TSPA-SR model resolution of these differences.

Table 6-137. Human Intrusion Scenario Regulatory Assumptions

NRC Base Assumptions (from proposed 10 CFR 63 and accompanying supplemental information)	EPA Additional and/or Conflicting Assumptions (from proposed 40 CFR 197 and accompanying supplemental information)	Conceptualization for TSPA-SR
Assumed intrusion is a drilling event	Assumed intrusion is acute and inadvertent	Inadvertent drilling event
Drilling result is a single, nearly vertical borehole that penetrates a waste package and extends down to the SZ	Borehole penetrates a degraded waste package	Single vertical borehole from surface through a single waste package to SZ
Intrusion occurs 100 years after closure	Intrusion time should take into account the earliest time after disposal that a waste package could degrade sufficiently that current drilling techniques could lead to waste package penetration without recognition by the drillers	Intrusion occurs at 100 years. Later intrusion times examined in sensitivity simulations.
Borehole properties (diameter, drilling fluids) are based on current practices for resource exploration	Borehole results from exploratory drilling for ground water	Borehole diameter consistent with a water well
Borehole is not adequately sealed to prevent infiltrating water	Natural degradation processes gradually modify the borehole, the result is no more severe than the creation of a ground water flow path from the crest of Yucca Mountain through the repository and to the water table	Infiltration and transport through the borehole assumes a degraded, uncased borehole, with properties similar to a fault pathway
Hazards to the drillers or to the public from material brought to the surface by the assumed intrusion should not be considered	Only consider releases through the borehole to the SZ. Consider releases which occur gradually through air and water pathways, not suddenly as with direct removal	Groundwater is only pathway considered
A separate consequence analysis is required, identical to the performance assessment, except for the occurrence of the specified human intrusion scenario	Unlikely natural processes and events are not included, but analysis could include disturbances by other processes or events which are likely to occur	Intrusion borehole is applied to nominal case. Effects of volcanism are not included.
Peak dose not to exceed 25 mrem/year in the first 10,000 years	Peak dose not to exceed 15 mrem/year in the first 10,000 years	Does not affect simulation.

The human intrusion scenario is run entirely within GoldSim. All human intrusion input parameters are specified within GoldSim. Human intrusion output is dose to the critical group from the human intrusion event.

Inputs to the TSPA Model

Based on the regulatory guidance summarized in Table 6-137, the TSPA-SR human intrusion scenario was conceptualized to include five key components:

- Infiltration of water down the borehole and into the penetrated waste package
- Mobilization and release of radionuclides from the penetrated waste package
- Transport of radionuclides down the borehole to the water table

- Transport of radionuclides through the saturated zone
- Biosphere exposure pathways and dose calculation at the receptor location.

General human intrusion input parameters are given in Table 6-138. Input parameters for each of the five key components of human intrusion are given in Table 6-139 through Table 6-143. The human intrusion scenario is identical to the nominal scenario, except for the intrusion borehole. Therefore, only input parameters that are new or changed from the nominal scenario are listed in Table 6-138 through Table 6-143.

Table 6-138. General Human Intrusion Input Parameters

Parameter Name	Description	Parameter Value	Reference
HumanIntr_Failure_Time	Time at which waste package is penetrated by the Human Intrusion event	100 years	Assumption [a]
HumanIntr_Dist	Parameter that selects which type of WP is penetrated	Uniform Distribution 0-1	Assumption [a]
CSNF_Pkg_fraction	Fraction of CSNF packages in the repository	0.667799	Assumption [a]
Bin_Number	Waste package infiltration bin in which the penetrated WP resides	4	Assumption [a]
Drip_Number	Dripping environment in which the penetrated WP resides	2 (sometimes drips)	Assumption [a]

NOTE: [a] Refer to Section 5.2 for the assumptions pertaining to Section 6.3.9.3.

Table 6-139. Human Intrusion Input Parameters for Infiltration of Water Down the Borehole

Parameter Name	Description	Parameter Value	Reference
Drill_Patch	Cross-sectional area of the borehole. ross-sectional area of the intrusion in the WP	0.0324 m^2	Assumption [a]
Borehole_Low_ Infiltration	Borehole infiltration rate in the low infiltration scenario	CDF [-] (mm/year) 0.000 0.000 0.050 0.000 0.100 0.000 0.150 0.000 0.200 0.000 0.250 0.000 0.300 0.000 0.350 0.000 0.400 0.000 0.450 0.000 0.500 0.000 0.550 0.048 0.600 0.128 0.650 0.219 0.700 0.343 0.750 0.569	Assumption [a] (Derived from Glacialu.dat, DTN GS000308311221.005 [147613])

Table 6-139. Human Intrusion Input Parameters for Infiltration of Water Down the Borehole (Continued)

Parameter Name	Description	Parameter Value	Reference
Borehole_Low_ Infiltration (Continued)	Borehole infiltration rate in the low infiltration scenario	0.800 1.102 0.850 2.320 0.900 4.905 0.950 11.456 0.960 13.837 0.970 17.158 0.980 22.301 0.990 34.774 0.995 57.013 0.999 125.730 1.000 370.330	Assumption [a] (Derived from Glacialu.dat, DTN GS000308311221.005 [147613])
Borehole_Mean_ Infiltration	Borehole infiltration rate in the mean infiltration scenario	CDF [-] (mm/year) 0.000 0.000 0.050 0.000 0.100 0.000 0.150 0.000 0.200 0.000 0.250 0.000 0.300 0.000 0.350 0.000 0.400 0.252 0.450 0.405 0.500 0.582 0.550 0.805 0.600 1.111 0.650 1.600 0.700 2.814 0.750 6.195 0.800 12.329 0.850 20.681 0.995 300.200 0.999 871.430 1.000 3902.500	Assumption (Derived from Glacialu.dat, DTN GS000308311221.005 [147613])
Borehole_Mean_ Infiltration (Continued)	Borehole infiltration rate in the mean infiltration scenario (Continued)	0.900 32.428 0.950 54.620 0.960 63.891 0.970 84.034 0.980 119.210 0.990 203.160	
Borehole_High_ Infiltration	Borehole infiltration rate in the high infiltration scenario	CDF [-] (mm/yr) 0.000 0.000 0.050 0.000 0.100 0.000 0.150 0.000 0.200 0.000 0.250 0.000 0.300 0.000 0.350 0.000 0.400 0.500 0.450 0.770 0.500 1.053 0.550 1.416 0.600 1.917 0.650 2.653	Assumption [a] (Derived from Glacialu.dat, DTN GS000308311221.005 [147613])

Table 6-139. Human Intrusion Input Parameters for Infiltration of Water Down the Borehole (Continued)

Parameter Name	Description	Parameter Value	Reference
Borehole_High_ Infiltration (Continued)	Borehole infiltration rate in the high infiltration scenario	0.700 4.895 0.750 11.001 0.800 19.688 0.850 34.482 0.900 60.793 0.950 102.100 0.960 119.450 0.970 153.070 0.980 216.840 0.990 370.910 0.995 540.130 0.999 1711.900 1.000 7489.300	Assumption [a] (Derived from Glacialu.dat, DTN GS000308311221.005 [147613])

NOTE: [a] Refer to Section 5.2 for the assumptions pertaining to Section 6.3.9.3.

Table 6-140. Human Intrusion Input Parameters for Mobilization and Release of Radionuclides

Parameter Name	Description	Parameter Value	Reference
Clad_Fraction_Perforated	Fraction of CSNF cladding perforated in penetrated WP	1.0	Assumption [a]

NOTE: [a] Refer to Section 5.2 for the assumptions pertaining to Section 6.3.9.3.

Table 6-141. Human Intrusion Input Parameters for Transport of Radionuclides Down the Borehole

Parameter Name	Description	Parameter Value	Reference	DTN
Am_Devit	Kd for Americium in the borehole rubble (devitrified units of UZ)	Uniform Distribution Min = 100 ml/g, Max = 2000 ml/g	Per the nominal TSPA case	
Cs_Devit	Kd for Cesium in the borehole rubble (devitrified units of UZ)	Uniform Distribution Min = 10 ml/g, Max = 700 ml/g	CRWMS M&O 2000 [152773] Table 2a	b
Np_Devit	Kd for Neptunium in the borehole rubble (devitrified units of UZ)	Beta Distribution Mean = 0.3 ml/g, Std Dev = 0.09 ml/g Min = 0 ml/g Max = 1 ml/g	Per the nominal TSPA case	
Pa_Devit	Kd for Protactinium in the borehole rubble (devitrified units of UZ)	Uniform Distribution Min = 0 ml/g, Max = 100 ml/g	Per the nominal TSPA case	
Pu_Devit	Kd for Plutonium in the borehole rubble (devitrified units of UZ)	Uniform Distribution Min = 5 ml/g, Max = 70 ml/g	Per the nominal TSPA case	

Table 6-141. Human Intrusion Input Parameters for Transport of Radionuclides Down the Borehole (Continued)

Parameter Name	Description	Parameter Value	Reference	DTN
Sr_Devit	Kd for Strontium in the borehole rubble (devitrified units of UZ)	Uniform Distribution Min = 5 ml/g, Max = 30 ml/g	CRWMS M&O 2000 [152773] Table 2a	b
Th_Devit	Kd for Thorium in the borehole rubble (devitrified units of UZ)	Uniform Distribution Min = 100 ml/g, Max = 2000 ml/g	Per the nominal TSPA case	
U_Devit	Kd for Uranium in the borehole rubble (devitrified units of UZ)	Beta Distribution Mean = 0.5 ml/g, Std Dev = 0.15 ml/g Min = 0 ml/g Max = 2 ml/g	Per the nominal TSPA case	
UZ_Fault_Pathway: Porosity	Porosity in the borehole rubble (matrix porosity of a UZ fault)	0.19	Assumption [a]	
Borehole_to_SZ: Area	Cross-sectional area of borehole from the penetrated WP to the SZ	0.0324 m^2	Assumption [a]	
Borehole_to_SZ: Dispersivity	Dispersivity in the borehole from the penetrated WP to the SZ	20 m	Assumption [a]	
Borehole_to_SZ: Fluid Saturation	Fluid saturation in the borehole from the penetrated WP to the SZ	0.2	Assumption [a]	
Borehole_to_SZ: Source Zone Length	Length of source zone in the borehole	0.0 m	Assumption [a]	

NOTE: [a] Refer to Section 5.2 for the assumptions pertaining to Section 6.3.9.3.
[b] DTN: LA0003AM831341.001 [148751]

Table 6-142. Human Intrusion Input Parameters for Transport of Radionuclides Through the SZ

Parameter Name	Description	Parameter Value	Reference
SZ_Input_Pointer	Stochastic parameter that selects which SZ source region (1 or 3) the borehole intercepts	Discrete Distribution 0.5, 1 0.5, 3	Assumption [a]

NOTE: [a] Refer to Section 5.2 for the assumptions pertaining to Section 6.3.9.3.

Table 6-143. Human Intrusion Input Parameters for the Biosphere

Parameter Name	Description	Parameter Value	Reference
BDCF_Cs137	Biosphere dose conversion factor for Cs-137	Log-Normal Distribution Geometric Mean = 18.413 (mREM/year)/(pCi/l) Geometric Standard Deviation = 1.163	CRWMS M&O 2000 [144054], Table 7
BDCF_Sr90a	Biosphere dose conversion factor for Sr-90	Log-Normal Distribution Geometric Mean = 1.121 (mREM/yr)/(pCi/l) Geometric Standard Deviation = 2.736	CRWMS M&O 2000 [144054], Table 7
BDCF_Sr90b	Shifted Value for Sr-90 Log Normal Distribution	1.525 (mREM/yr)/(pCi/l)	CRWMS M&O 2000 [144054], Table 7

DTN: MO0003SPAABS07.006 [148872]

Implementation

The human intrusion scenario is identical to the nominal scenario, except for the intrusion borehole. The human intrusion case GoldSim file is derived from the nominal case GoldSim file (see Figure 6-218), but was run separately.

Parameters in the *Human_Intrusion_Parameters* container control waste package failure, borehole seepage, and selection of the NFE and SZ region associated with the waste package (see Figure 6-219).

The time at which the waste package is failed is set by the parameter *HumanIntr_Failure_Time*. The parameter *HI_Package_Failure_Time* determines how long the waste package has been failed. This value is passed on to the parameter *Avg_Pkg_FailTime*. The parameter *HumanIntr_WP_Failure* is equal to 0 if the waste package is not failed, and is equal to 1 when the waste package is failed. It is used in as the outer barrier fraction failed in the CSNF and CDSP source terms. It is also used in parameter *QFlux_WP* to switch on borehole seepage through the waste package upon package failure.

The stochastic HumanIntr_Dist generates a uniform random number between 0 and 1. Its value is compared to fraction of CSNF packages (*CSNF_Pkg_fraction*). If it is less than *CSNF_Pkg_fraction* switches *HumanIntr_CSNF_Switch* and *HumanIntr_CDSP_Switch* are set equal to 1 and 0, respectively. Conversely, if it is greater than *CSNF_Pkg_fraction* switches *HumanIntr_CSNF_Switch* and *HumanIntr_CDSP_Switch* are set equal to 0 and 1, respectively. The switches are used in the probability setting of the source term outer barrier failure mode to select which package (e.g., CSNF or CDSP). It is also used in parameter *QFlux_WP* to switch on borehole seepage through the waste package upon package failure.

Infiltration down the borehole to the penetrated waste package was not explicitly modeled. Instead, a volumetric flux (parameter *Borehole_Seepage*) was calculated and applied directly into the penetrated package. The infiltration rate was sampled from distributions of infiltration rate (*Borehole_High_Infiltration, Borehole_Mean_Infiltration*, and *Borehole_Low_Infiltration*). Switch *Borehole_Infiltration* selected the value of one of the infiltration rate distributions based on the value of *Infiltration_Scenario*. The volumetric flux was calculated as the product of the infiltration rate and the cross-sectional area of the borehole (parameter *Drill_Patch*).

The parameters *Bin_Number* and *Drip_Number* are used to set which infiltration bin (bin 1 through bin 5) and which seepage condition (always seeps, sometimes seeps, never seeps) the failed waste package is in.

Figure 6-220 shows a graphical illustration of Human Intrusion case in the *CSNF_Packages* container in the *Engineered_Barrier_System* container. The Human Intrusion case in the *CDSP_Packages* container is the same (see Figure 6-221), except that there is not a *Cladding* container (CDSP has no cladding) nor a *Bad_Infiltration_Bin_3* or *Bad_Infiltration_Bin_5* container. Figure 6-222 and Figure 6-223 show the CSNF and CDSP EBS for the Human Intrusion scenario, respectively.

Mobilization and release of radionuclides from the penetrated waste package followed similar waste form degradation processes as for the nominal scenario. For CSNF, radionuclides were exposed through unzipping of cladding and then mobilized by dissolution in the infiltrating water, as in the nominal case. The difference from the nominal case was that all of the cladding was initially assumed to be perforated as a result of damage from drill-bit penetration (parameter *Clad_Fraction_Perforated*) and therefore could not limit unzipping. For CDSP, radionuclides were exposed through glass or DSNF degradation and then mobilized by dissolution in the infiltrating water, as in the nominal case.

The NFE parameters (*WP_Temp, Q_Flux, Q_Evap, invert_RH, T_invert,* and *Liquid_Sat*) for the failed waste package are selected based on the *bin_number* (see Figure 6-224). The seepage flux is selected based on the *bin_number* and *drip_number* (see Figure 6-225). Note that *seepage_flux* is not used in the Human Intrusion scenario (*borehole_seepage* is used instead) the logic to allow its proper selection is incorporated.

For both fuel types in-drift boundary condition chemistry was used in place of in-package chemistry in the calculation of in-package solubilities, FeOx colloids and groundwater colloids. It is also used for calculating CDSP waste form colloids and waste form dissolution rate, and CSNF waste form dissolution rate. This was done by deleting the in-package chemistry calculations at the environment level, which causes all instances of *pH_CSNF, pH_CDSP, Ionic_Str_CSNF,* and *Ionic_Str_CSNF* in the functions for solubilities, FeOx colloids, and groundwater colloids to default the values prescribed in the *In_Package_Chemistry* container (see Figure 6-226) (this container is located in the *Waste_Form* container in the *Engineered_Barrier_System* container).

Regardless of which type of waste package was sampled, the released mass was accumulated in the *EBS_HumanIntr_Release* cell for transport down the borehole to the SZ (see Figure 6-227).

Transport of radionuclides down the borehole to the water table was modeled using a GoldSim pipe *Borehole_to_SZ* (see Figure 6-228). The pipe was assumed to have flow properties consistent with a UZ fault (see Figure 6-229 and Table 6-141). The volumetric flux through the borehole was assumed equivalent to the modeled infiltration flux (parameter *Borehole_Seepage*).

The Kd values were consistent with the Kd values for devitrified units of the UZ. The mass release from the borehole was accumulated in the *SourceTerm_to_SZ* cell for transfer to the SZ.

Transport of radionuclides through the saturated zone was identical to the nominal case, except for the release point from the UZ. For the human intrusion scenario, the location where the borehole entered the SZ was randomly sampled between SZ source regions 1 and 3, which underlie the repository footprint (see parameter *SZ_Input_Pointer* in Table 6-142).

The biosphere exposure pathways and dose calculation at the receptor location were also identical to the nominal case, except for the addition of BDCFs for Cs-137 and Sr-90 (see Figure 6-230). These two radionuclides were not included in the nominal case because of short half-lives.

Results and Verification

Since the human intrusion case is based on the nominal case, only those parts that are specific to the human intrusion case are verified here. Realization 1 from the 100-realization human intrusion case is used for verification.

In the *Human_Intrusion_Parameters* container the stochastic *HumanIntr_Dist* is equal to 0.01540. When compared against the *CSNF_Pkg_fraction* (0.667799) the *HumanIntr_CSNF_Switch* and *HumanIntr_CDSP_Switch* take on values of 1 and 0, respectively. The parameter *HumanIntr_WP_Failure* changes from 0 to 1 at the time of waste package failure (100 years) (see Figure 6-231). *HI_Package_Failure_Time*, the duration that the waste package has been failed is equal to 0 years up to 100 years; there after it is equal to the simulation time (*ETime*) minus the time of waste package failure (see Figure 6-232).

The *Infiltration_Scenario* has a value of 2 for realization 1. Hence the switch *Borehole_Infiltration* takes on the value of the stochastic *Borehole_Mean_Infiltration* (1.2372e-011 mm/yr). Multiplying it in the parameter Borehole_Seepage by the Drill_Patch value of 0.324 m^2 yields a value of 4.00852e-016 m^3/yr.

Examination of the CSNF source term parameter (*CSNF_HI_Pkg*) finds that it has one package fail at 100 years. Conversely, the CDSP source term parameter (*CDSP_HI_Pkg*) shows no waste package failures.

The parameter *Clad_Fraction_Perforated* has a value of 0 for *ETime* less than 100 years, and is equal to 1 for greater than or equal 100 years.

Since the parameter *Bin_Number* is equal to 4, Bin 4 near field environment values are used in *Peak_Temp, WP_Temp, Q_Flux, Q_Evap, invert_RH, T_invert,* and *Liquid_Sat* switches.

The outflow from *EBS_HumanIntr_CSNF_OUT* was compared to that from *EBS_HumanIntr_Release* to verify that the source term is correctly passed from the EBS on to the UZ. In the *Unsaturated_Zone_Transport* container *SZin_Cell_3* was checked to verify that the source term was correctly passed from the *Borehole_to_SZ* to the SZ parameters *SZ_External* and *Pipe_5km*.

In the Biosphere container the connections between the added BDCFs for Cs-137 and Sr-90 (*BDCF_Cs137* and *BDCF_Sr90*) and the parameter *BDCF_Nominal* were verified.

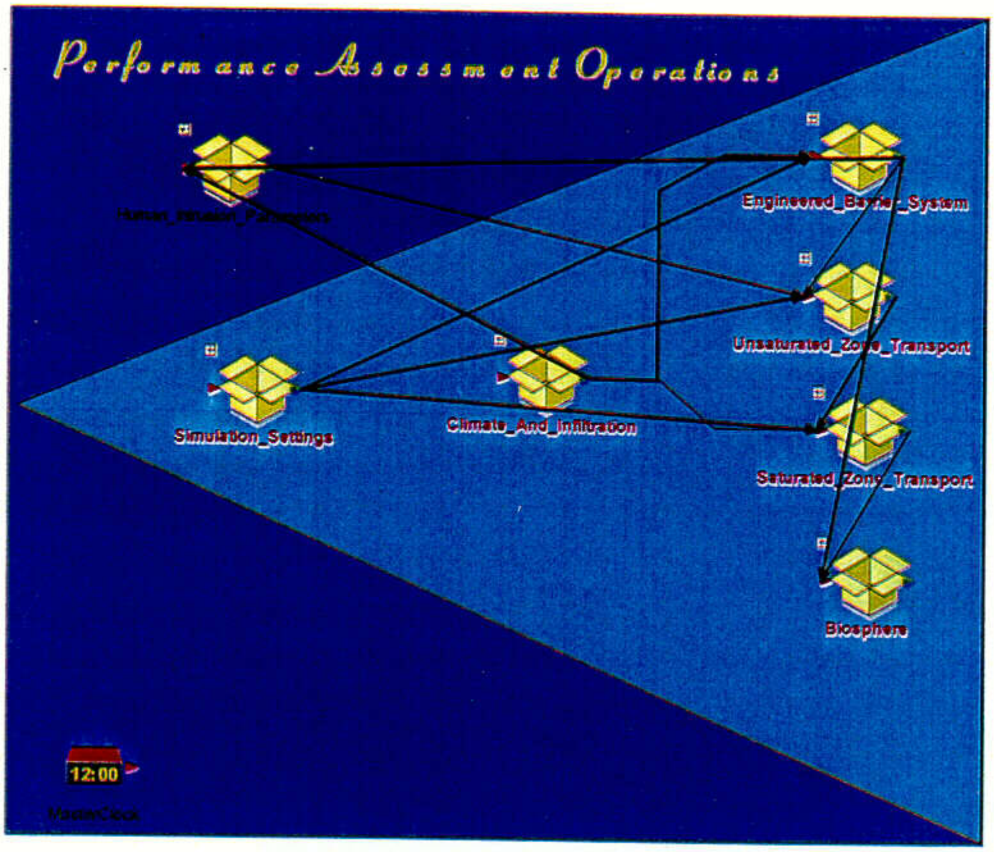


Figure 6-218. Overview of Human Intrusion Case GoldSim File

C42

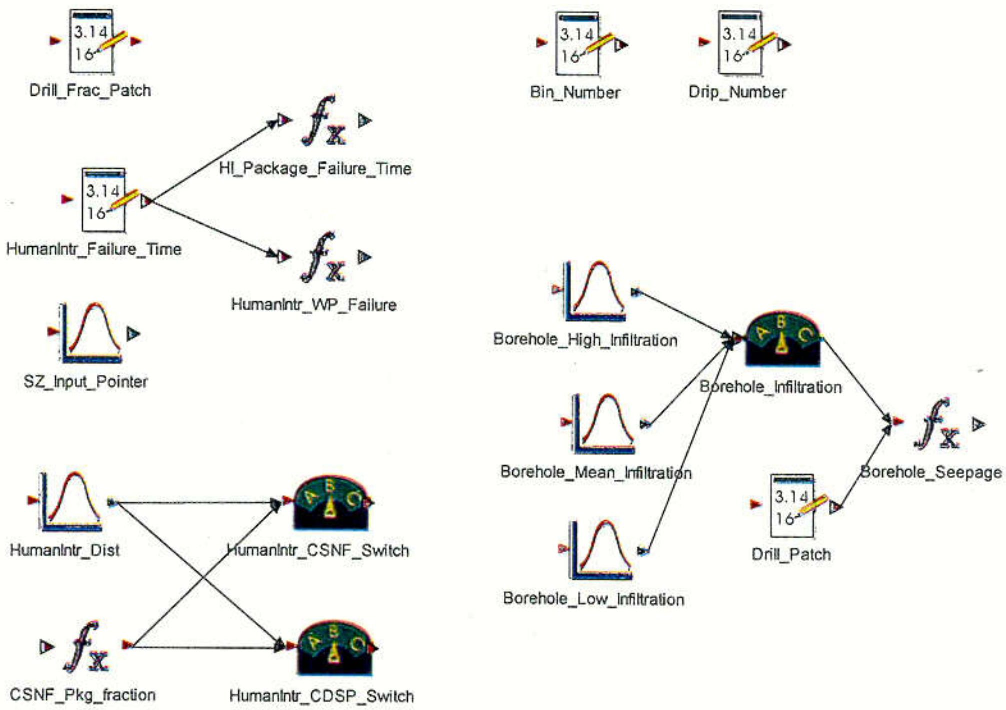


Figure 6-219. Human Intrusion Scenario Parameters in the Human_Intrusion_Parameters Container

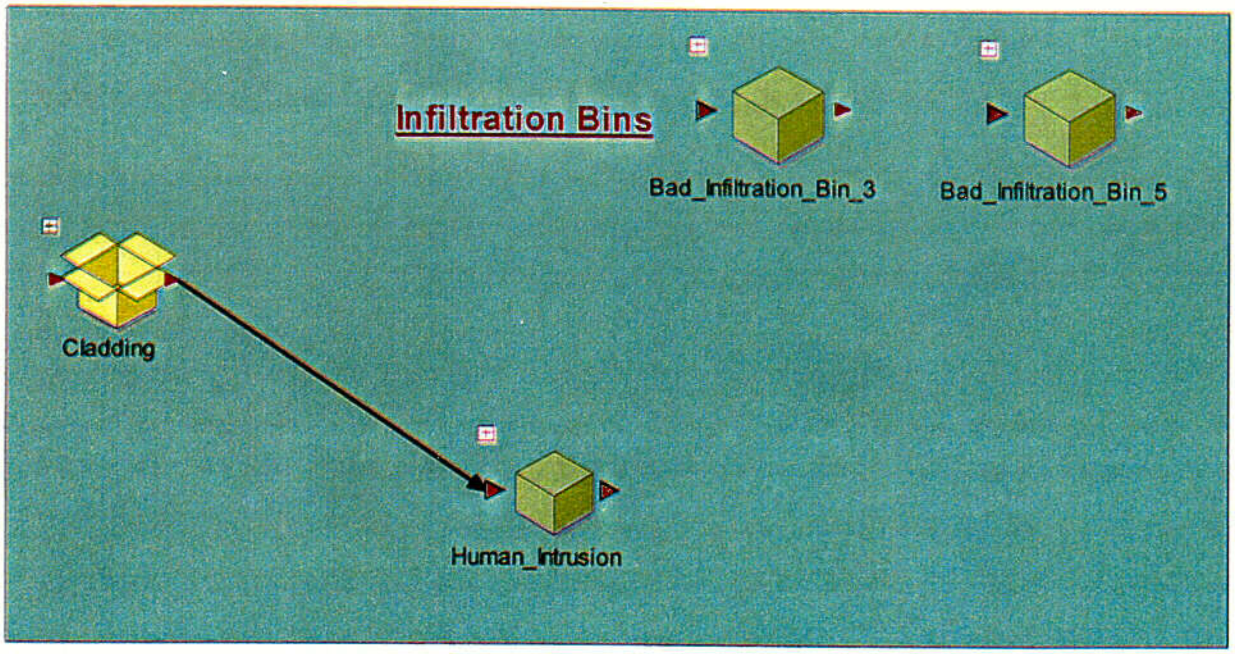


Figure 6-220. An Illustration of the Human Intrusion Case for CSNF at the EBS Level

C43

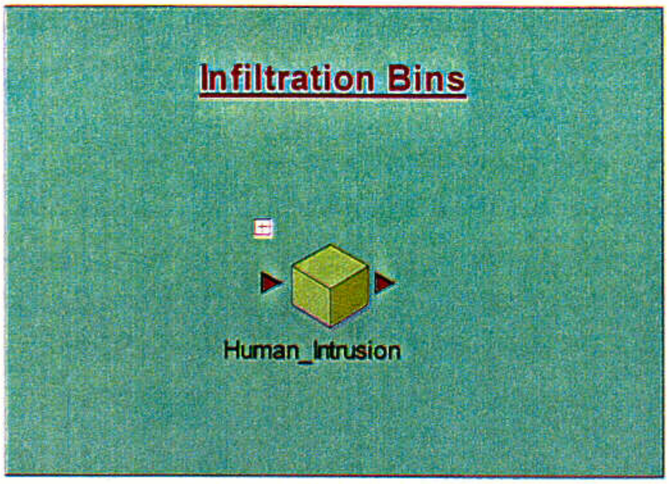


Figure 6-221. An Illustration of the Human Intrusion Case for CDSP at the EBS Level

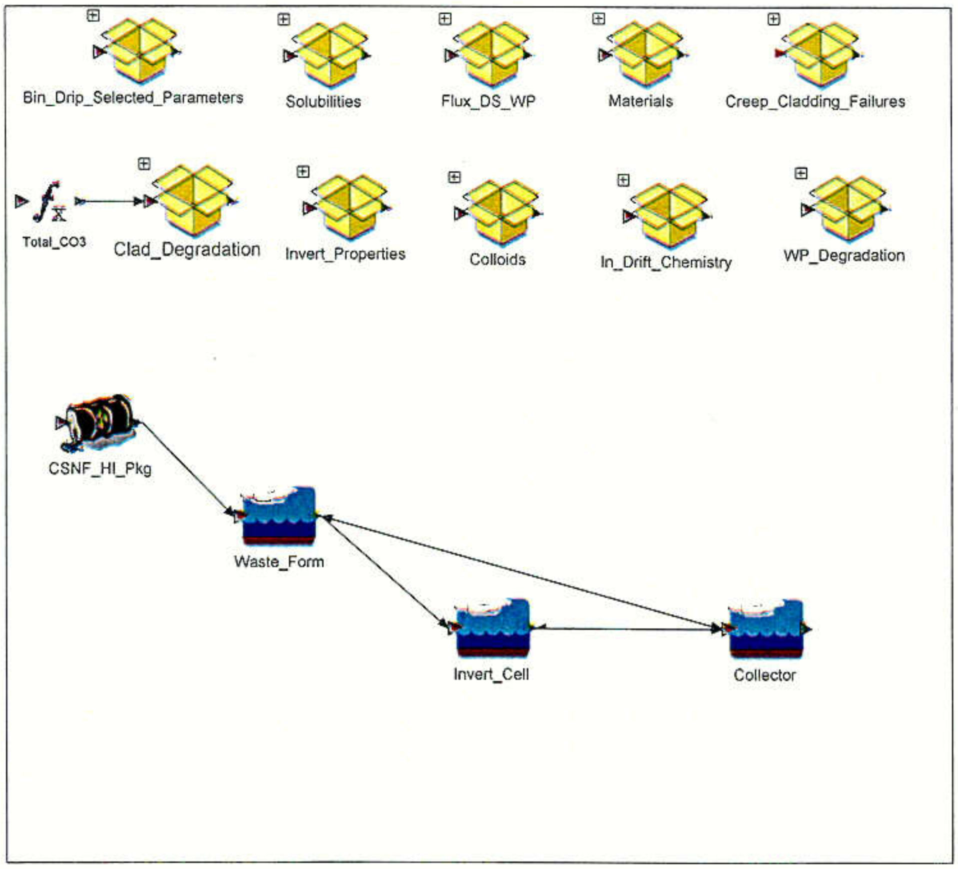


Figure 6-222. The Human Intrusion Case for CSNF at the Environment Level

C44

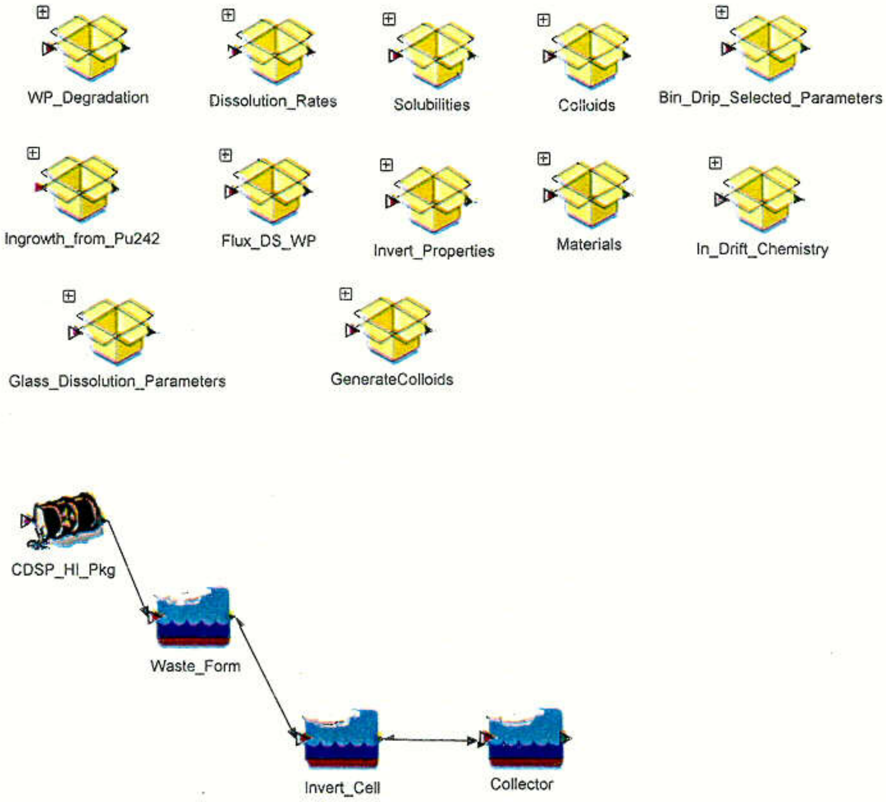


Figure 6-223. The Human Intrusion Case for CDSP at the Environment Level

C45

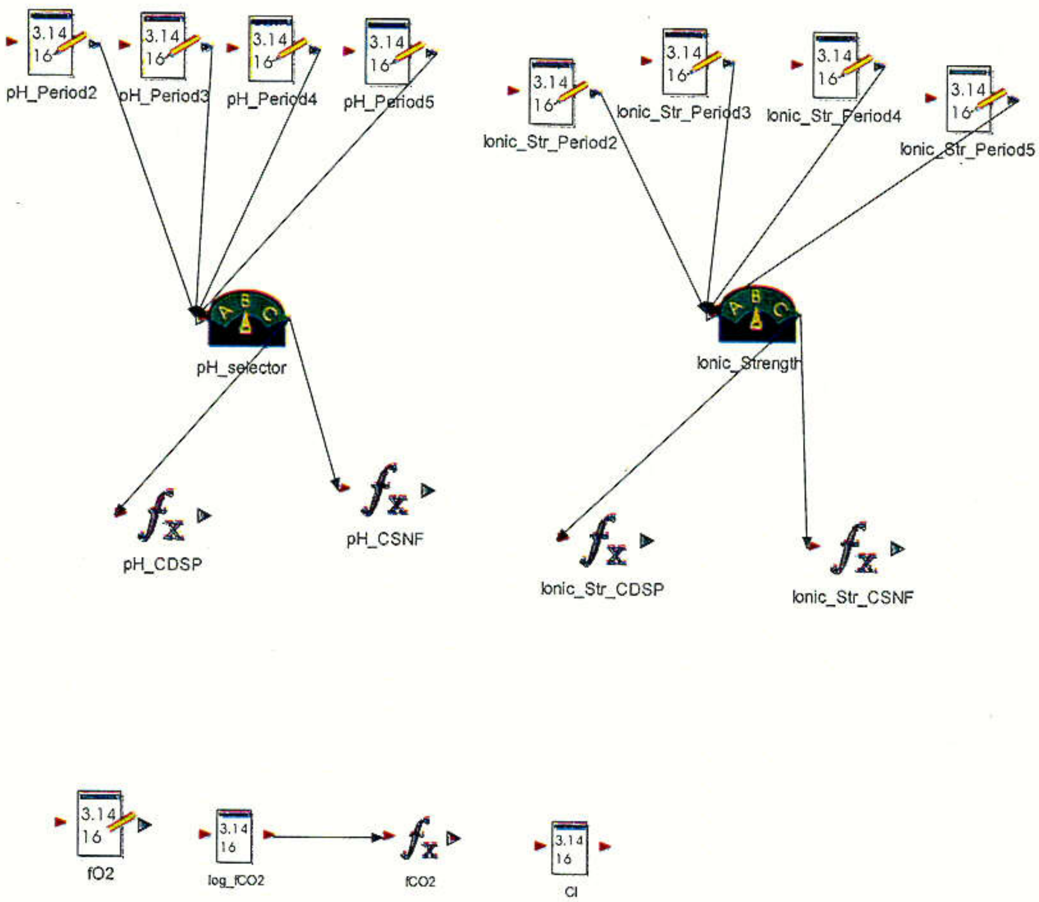


Figure 6-224. An Illustration of In-Package Chemistry for the Human Intrusion Scenario

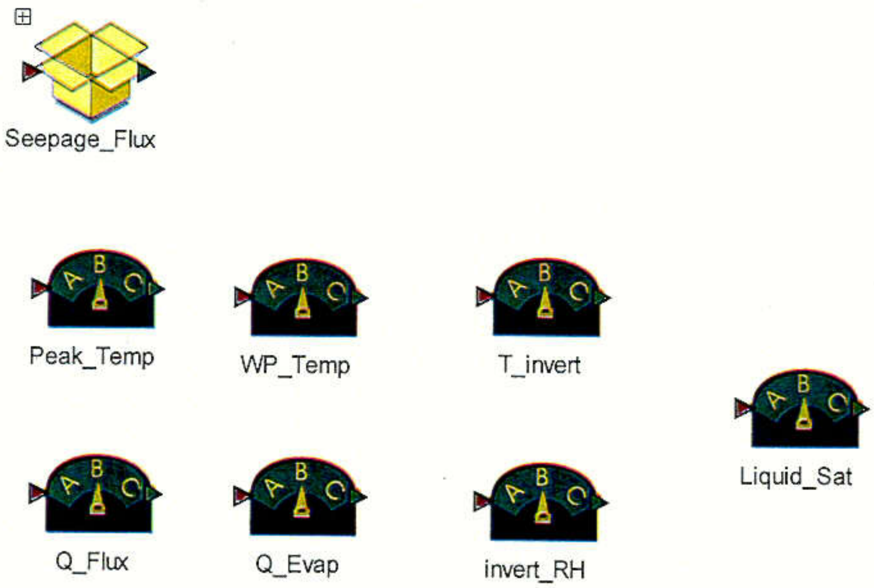


Figure 6-225. Selection of NFE Parameters for the Human Intrusion Scenario

C46

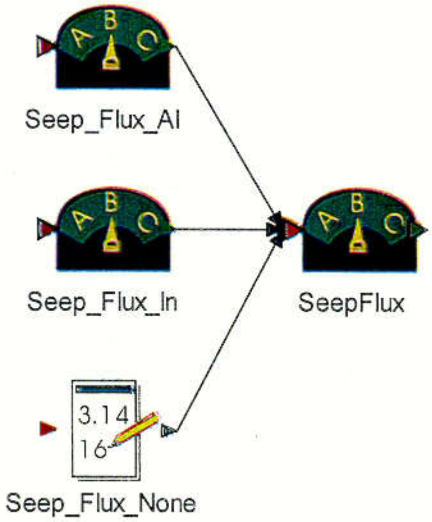


Figure 6-226. Selection of Seepage Flux for the Human Intrusion Scenario

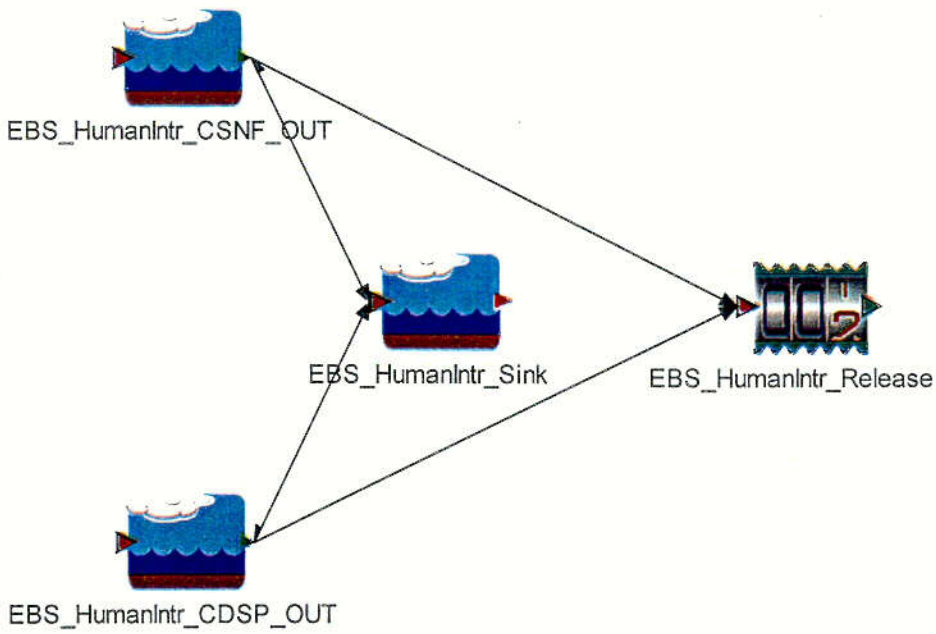


Figure 6-227. An Illustration of Source Term Transport from the EBS to the Borehole for the Human Intrusion Case

C47

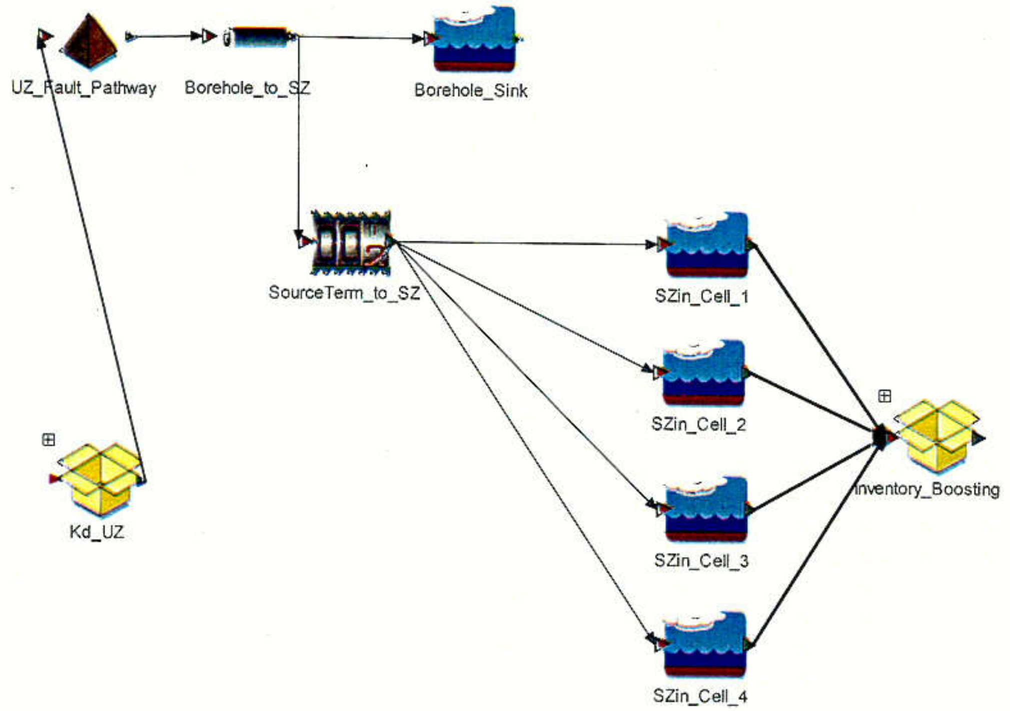


Figure 6-228. An Illustration of the Borehole between the EBS and the SZ in the Human Intrusion Case

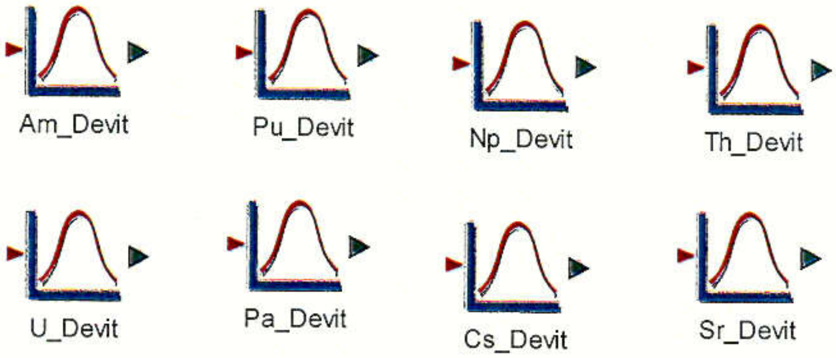


Figure 6-229. The Human Intrusion Case Borehole Kds

C48

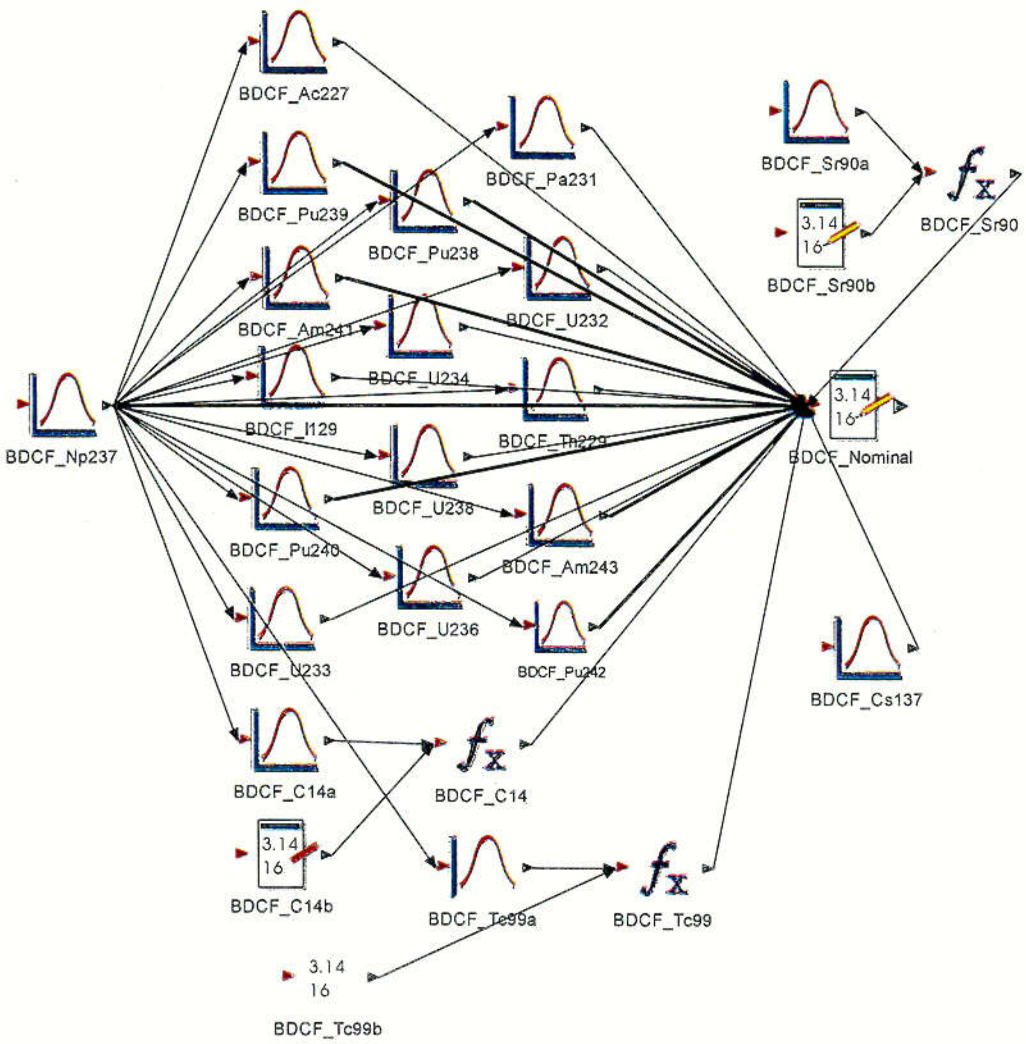


Figure 6-230. The Human Intrusion Case BDCFs

C49

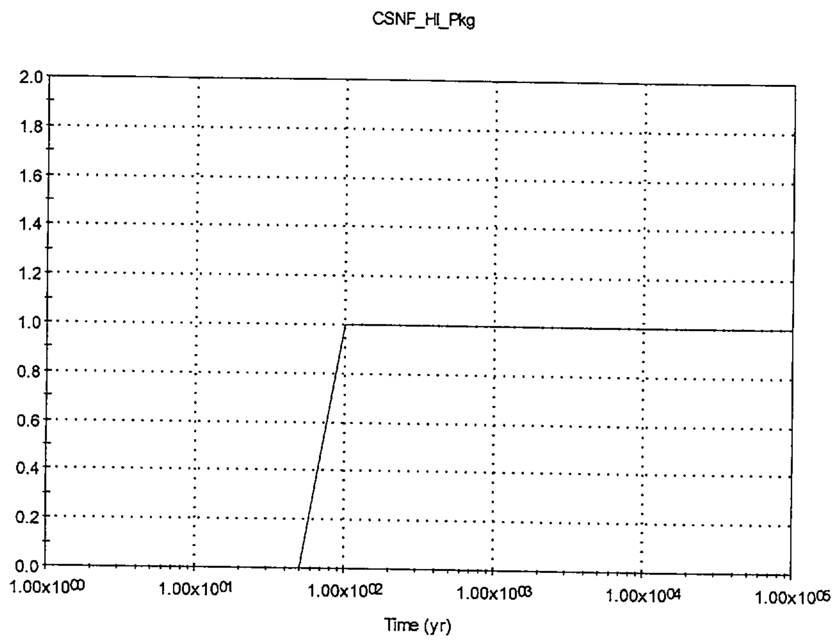


Figure 6-231. Plot of Parameter *HumanIntr_WP_Failure* vs. Simulation Time

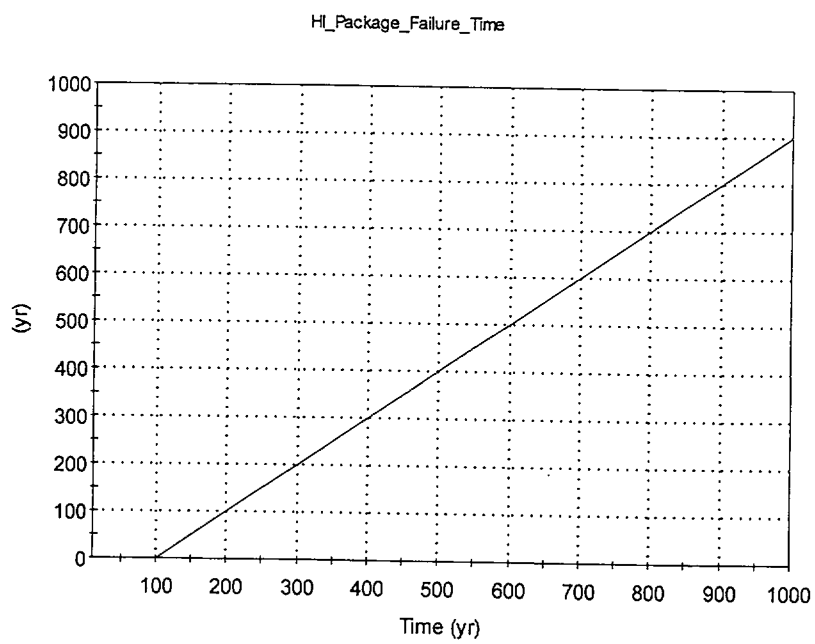


Figure 6-232. Plot of Human Intrusion Waste Package Failure time vs. Simulation Time

6.3.9.4 Seismic Cladding Event

Overview

A seismic analysis has shown that a severe earthquake (a once in a million years event) would fail most of the fuel rods, but a more moderate frequency event would fail no rods. Within the TSPA-SR model, a seismic event is included as part of the cladding model, and when this event occurs, all the cladding is assumed to fail. The event frequency is 1.10×10^{-6} events per year (CRWMS M&O 2000 [147210], Section 6.4.1).

Inputs to the TSPA Model

The data inputs to the TSPA model for seismic cladding model are listed in Table 6-144. A detailed discussion on the TSPA cladding degradation model can be found in *Clad Degradation – Summary and Abstraction* (CRWMS M&O 2000 [147210]).

Table 6-144. Data Inputs to the TSPA Model for Cladding Degradation

Parameter Name	Description	Parameter Value	Reference AMR
Seismic_Clad_Event	Frequency of a very severe seismic event that would fail all cladding	1.10×10^{-6}	CRWMS M&O 2000 [147210], Section 6.4.1; MOL.20000602.0055 [147210]

DTN: MO0004SPACLD07.043 [151368]

Implementation

Commercial spent nuclear fuel cladding is modeled as degrading in two distinct steps: (1) perforation of the cladding through the formation of small cracks or holes, and (2) unzipping (splitting) of the cladding. The mechanisms for cladding perforation that are modeled in the TSPA are the following:

- Initial failures due to reactor operation, dry storage handling, and transportation
- Creep and stress corrosion cracking
- Localized corrosion
- Stainless steel cladding failures
- Failures due to a very severe seismic event.

Cladding failure is a based upon the Cladding AMR (CRWMS M&O 2000 [147210]), with the exception of the seismic event, all of the above processes are address by model abstraction implemented as discussed in Section 6.3.4.3 of this AMR (see Figure 6-233). The nature of the seismic event requires that it be modeled as a disruptive event. The clad degradation abstraction indicates that the seismic analysis shows that most fuel rods would fail from a very severe earthquake (once in a million years event), but no rods would fail for more moderate frequency events (CRWMS M&O 2000 [147210], Section 6.4.1). Thus, the seismic failures have been implemented into the TSPA-SR model as a disruptive event. An event generator, *Seismic_Clad_Event*, has been defined to represent a very severe earthquake. The frequency of occurrence is 1.10×10^{-6} yr^{-1} (CRWMS M&O 2000 [147210], Section 6.4.1). The event generator is set with the option for random occurrence that cannot reoccur (e.g., only one seismic clad event per simulation or realization for a probabilistic simulation). The event will cause all the CSNF cladding to fail and therefore be immediately available for unzipping when the waste package fails. This is implemented within each source term group for CSNF fuels (e.g., all environments for CSNF BIN1 through BIN5, always drip, intermittent drip, and no drip). Within each source term for inner barrier failure the following logic has been implemented:

```
if( Switch_Zone2 == 1, 1, if(Seismic_Clad_Event.Num_Events==1,1,Clad_Fraction_Perforated))
```

When the seismic event hits, the source term will have a fraction equal to 1 for inner barrier failure. Thus, all the remaining rods will be perforated and begin to unzip.

Results and Verification

Table 6-145 shows TSPA model results for the first 1,000,000 years for the seismic cladding event parameter (*Seismic_Clad_Event*) that has a median value of 328,000 years for the seismic event. CSNF BIN4 for the nodrip environment for can be considered for verification of the seismic model. No drip environments only have an initial fraction perforated, as there is no localized corrosion due to the lack of a flux through the packages (see Section 6.3.4.3). No drip environments will have no perforations beyond the initial value until seismic clad event occurs. For BIN4 no drip the initial perforation is equal to 0.07. The unzipping rate is equal to 3.5005 yr^{-1} at 328,000 years when the seismic clad event occurs (approximately 29,000 years to unzip).

Table 6-145. GoldSim Results for the Seismic Clad Event (number of events) Versus Time for a Median Value Simulation

Time (yr)	Seismic Clad Event (num.events)
0 - 324,00	0
328,000-1,000,000	1

Figure 6-234 shows a time history plot of the cumulative release of Np237 and Tc99 from the waste package in Bin 4 (no drip) for a median value simulation. The seismic clad event is plotted on the same graph for comparison. It can be seen that given an unzipping time of approximately 29,000 years and a 328,000 year failure of all fuel rods, the cumulative release of these species increases starting at 360,000 years when all the rods are unzipped.

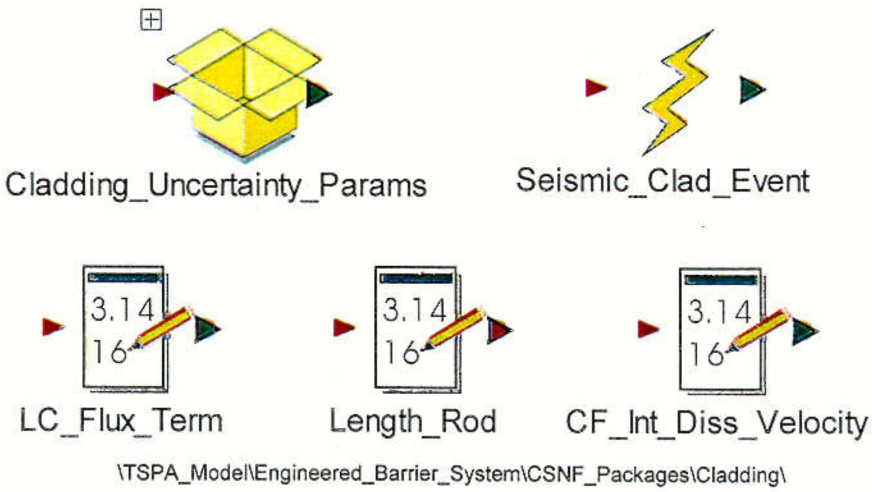
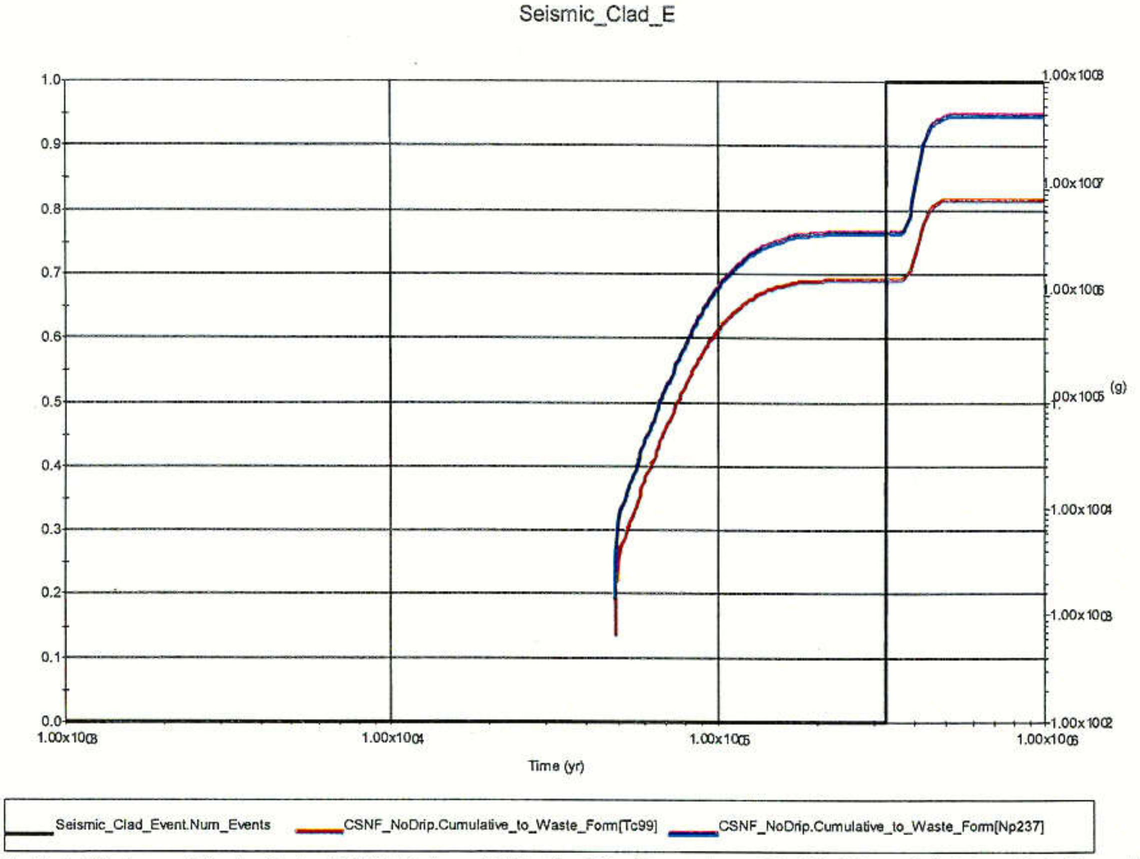


Figure 6-233. Cladding Model Parameters that are Defined Globally in the TSPA-SR Model



\TSPA_Model\Engineered_Barrier_System\CSNF_Packages\Infiltration_Bin_4\Intermittent_Drip\Clad_Degradation\Avg_Clad_Exposed\

Figure 6-234. Time History of the Average Cladding Exposed for Bin 4, Intermittent Drip (Median Value Simulation)

C50

6.4 SIMULATION SETTINGS

Overview

The Total System Performance Assessment (TSPA) Site Recommendation (SR) model was designed to run under a variety of simulation scenarios (see Sections 6, 6.2, and 6.3.9).

However, the base case model was only run in two basic modes: (a) as a single realization with all the uncertain component model parameters set at their median value and (b) as a probabilistic case where a Monte Carlo sampling method was adopted, by running numerous realizations of the repository system with sampled values from the probability distributions of the uncertain model parameters. For TSPA SR, the simulations were preformed for two major time intervals: (a) from 0 to 100,000 years; and (b) from 0 to 1,000,000 years. The simulation time frames and timesteps were set to optimize the computational efficiency of the model run, as well as to provide sufficient detail in the results to capture the behavior of the dominate processes that occur over the duration of each simulation time interval. External codes (e.g., DLLs) were called and in some cases rely on supporting input files that must be available during run time. Individual input files for some external codes were modified or replaced for a particular simulation mode (see implementation section below).

Inputs to the TSPA Model

Inputs to the model file regarding settings for a given simulation are located in the simulation settings pull-down menu within GoldSim and in the simulation settings container in the model file. In addition, external files that support the DLLs coupled to the model may also contain input parameters related to the simulation settings.

Implementation

The basic simulation settings are related to the type of run (e.g., nominal or igneous, backfill or no backfill), the duration of the run, and whether the model is to be run for a single realization with median values for uncertain parameters or run for multiple realizations with sampled values for uncertain parameters. Figure 6-235 shows the data elements (*Median_Value_Run, Backfill_Case,* and *Case_Selector*) located in the *Simulation_Settings* container that are used to set the simulation settings within the model file. The values that these data elements can take on are shown in Table 6-146. The two switches (*Realz_Number_SZ* and *Realz_Number_UZ*) in the *Simulation_Settings* container determines the realization number that is passed to fehmn_sr.dll (the UZ flow and transport code) and szconv_sr.dll (the SZ flow and transport code) (see Table 6-146).

Table 6-146. Parameters in *Simulation_Settings* Container

TSPA Parameter	Description	Parameter Value/Other Inputs
Median_Value_Run	Determines whether the run is to be made as a single realization with median values used for uncertain parameters or as multiple realizations with sampled values used for uncertain parameters	1 (median value, single realization run) 2 (probabilistic, multiple realization run)
Backfill_Case	Determines whether the run is to be made with or without backfill	0 (without backfill) 1 (with backfill)
Case_Selector	Determines what type of case is to be run.	1 (nominal, 1E5 year run) 2 (igneous, 1E5 year run) 3 (nominal, 1E6 year run)
Realz_Number_SZ	Determines what realization number is passed to szconv_sr.dll	If Median_Value_Run = 1 then 1, else if Master.Clock.Realization <= 100 then Master.Clock.Realization else if Master.Clock.Realization <= 200 then Master.Clock.Realization-100 else Master.Clock.Realization-200
Realz_Number_UZ	Determines what realization number is passed to fehmn_sr.dll	If Median_Value_Run = 0 then Master.Clock.Realization else 1

The *Simulation_Settings* container also has two containers. The Switches container contains elements whose values are set based on the values of the *Case_Selector* and *Backfill_Case* elements (see Figure 6-236 and Table 6-147). The logic in these parameters ensures that the appropriate conditions are propagated throughout the model file. The file elements (*Fehmn_input_file*, *SZ_Input_File*, and *Fehmn_gold*) in the *External_Files* container and the *Multiple_Realization_Files* container cause external files to be written to the "Networked" folders associated with the slave processors used when the simulation run over the network (see Figure 6-237, Figure 6-238, and Table 6-148). These external files contain input data for DLLs coupled to the GoldSim model file and contain values that define simulation settings for the DLLs.

Table 6-147. Parameters in Switches Container

TSPA Parameter	Description	Parameter Value/Other Inputs
CSNF_Probability	If this parameter takes on a value equal to 0, CSNF WPs do not fail. If this parameter takes on a value equal to 1, CSNF WPs fail per the WAPDEG-generated WP failure curve.	if(Case_Selector =2 and Backfill_Case = 1, 0, 1)
CDSP_Probability	If this parameter takes on a value equal to 0, causes CDSP WPs do not fail. If this parameter takes on a value equal to 1, CDSP WPs fail per the WAPDEG-generated WP failure curve.	if(Case_Selector = 2 and Backfill_Case = 1, 0, 1)
Volcano_Period	Sets the volcano period to the smallest time step in the simulation for the igneous case.	if(Case_Selector = 2,31.25{yr}, 250{yr})
Closure_Time	Sets the repository closure time in the simulation for the igneous case.	if(Case_Selector = 2, 62.5{yr}, 1e4 {yr})

Table 6-147. Parameters in Switches Container (Continued)

TSPA Parameter	Description	Parameter Value/Other Inputs
Switch_Wapdeg	This parameter was used in draft versions of the model. It is not currently used anywhere in the model file.	if(Case_Selector = 2,0,1)
Switch_Direct_Release	Activates the Direct_Release model if the simulation is for the igneous case.	if(Case_Selector = 2,1,0)
Switch_Indirect_Releas e	Activates the Indirect_Release model if the simulation is for the igneous case.	if(Case_Selector =2,1,0)
Switch_Zone2	Activates the Zone 2 indirect model for the igneous case	if((Switch_Indirect_Release = 1 and Backfill_Case = 0),1, 0)
Run_Time	Sets the model run time based on Case_Selector	If Case_Selector = 1 then Run_Time = 1E5{yr} else if Case_Selector = 2 then Run_Time = 1E5{yr} else Run_Time = 1E6{yr}
Waste_Packages	Sets the number of waste packages used by WAPDEG based on Case_Selector.	If Case_Selector = 1 then Waste_Packages = 400 else if Case_Selector = 2 then Waste_Packages = 10 else Waste_Packages = 400
TimeStep_Length	Forces a non-zero time-step length at t=0{yr}. Equal to the Master.Clock.Timestep_Length for t>0{yr}	If MasterClock.Timestep_Length = 0{yr} then TimeStep_Length = 1{yr} else TimeStep_Length = MasterClock.Timestep_Length
Indirect_Source_CSNF	Passes the indirect CSNF source on to the Source_Release_CSNF_Total parameter. A zero value is passed in the nominal case.	If Switch_Indirect_Release = 1 then Indirect_Release_Zone1.Waste_Form_CSNF_Indir ect else Zero_Mass/TimeStep_Length
Indirect_Source_CDSP	Passes the indirect CDSP source to the Source_Release_CDSP_Total parameter. A zero value is passed in the nominal case.	If Switch_Indirect_Release = 1 then Indirect_Release_Zone1.Waste_Form_CDSP_Indir ect else Zero_Mass/TimeStep_Length
Indirect_Unexposed_Ma ss_CSNF	Passes the indirect CSNF unexposed mass to the Total_CSNF_unexpos_Source_Term parameter. A zero value is passed in the nominal case.	If Switch_Indirect_Release = 1 then Indirect_Release_Zone1.Unexposed_Mass else Zero_Mass
Indirect_Unexposed_Ma ss_CDSP	Passes the indirect CDSP unexposed mass to the Total_CDSP_unexpos_Source_Term parameter. A zero value is passed in the nominal case.	If Switch_Indirect_Release = 1 then Indirect_Release_Zone1.Unexposed_Mass_2 else Zero_Mass
Indirect_WP_Release_ CSNF	Passes the indirect CSNF release from the waste package to the WF_Release_CSNF_Total parameter. A zero value is passed in the nominal case.	If Switch_Indirect_Release = 1 then Indirect_Release_Zone1.Total_WF_CSNF_indirect else Zero_Mass/TimeStep_Length
Indirect_WP_Release_ CDSP	Passes the indirect CDSP release from the waste package to the WF_Release_CDSP_Total parameter. A zero value is passed in the nominal case.	If Switch_Indirect_Release = 1 then Indirect_Release_Zone1.Total_WF_CDSP_indirect else Zero_Mass/TimeStep_Length

Table 6-148. Parameters in *External_Files* Container

TSPA Parameter	Description	Parameter Value /Other Inputs
Fehmn_input_file	FEHMN input file. Enter the total simulation time in days [first two entries in the line below 'time']	fm_pchm1.dat
SZ_Input_File	SZ_Convolute input file. Enter the size of the 1st time-step in years [line #10]	sz_convolute2.dat
Fehmn_gold	FEHMN input file. Contains simulation settings and batch file commands. Median Value Run: 10 [number of DLL inputs] ts0 [DLL will execute fehmn_ts0.bat file using the following commands] del fm_pchm1.zone2 copy fm_pchm1.zone2.0200 fm_pchm1.zone2 [deletes fm_pchm1.zone2 file, then creates a new fm_pchm1.zone2 file from fm_pchm1.zone2.0200] Multiple Realization Run: 10 [number of DLL inputs] ts0 [DLL will execute fehmn_ts0.bat file using the following commands] del ptrk.multrlz copy ptrk.multrlz.%1 ptrk.multrlz [Deletes the ptrk.multrlz file, then creates a new ptrk.multrlz file from either the ptrk.multrlz.0100, ptrk.multrlz.0200, or ptrk.multrlz.0300 file. The file used depends on the infiltration state (%1) passed by the DLL to the batch file.] del fm_pchm1.zone2 copy fm_pchm1.zone2.%1 fm_pchm1.zone2 [Deletes fm_pchm1.zone2 file, then creates a new fm_pchm1.zone2 file from either the fm_pchm1.0100, fm_pchm1.0200, or fm_pchm1.0300 file. The file used depends on the infiltration state (%1) passed by the DLL to the batch file.]	fehmn.gold
particle_tracking_file_multrlz	FEHMN input file. Particle tracking file used by FEHMN when simulation is run.	ptrk.multrlz
particle_tracking_file_100	FEHMN input file. Particle tracking file for low infiltration	ptrk.multrlz.0100
particle_tracking_file_200	FEHMN input file. Particle tracking file for mean infiltration	ptrk.multrlz.0200
particle_tracking_file_300	FEHMN input file. Particle tracking file for high infiltration	ptrk.multrlz.0300
UZ_Params_Multi_File	FEHMN input file. Contains the values for the stochastic UZ parameters used in a multiple realization run.	UZ_Params_Multi.sr

Along with the simulation settings parameters, the simulation settings pull-down menu is also used to input run-specific values to the simulation (see Figure 6-239). Both Time Options and Monte Carlo options are set in this menu.

The length of the run is specified in the "duration" entry. The number of timesteps is specified in the "# Timesteps" entry. The timestep length is calculated by GoldSim as the length of the run divided by the number of timesteps. Smaller timestep sizes can be used in the simulation via the Customized Timesteps menu (see Figure 6-240 and Table 6-149). A timestep subdivision factor is also input in the Customized Timesteps menu. This factor is used by GoldSim to determine the size of its internal timestep (Golder Associates 2000 [143556], Section 6). Table 6-153 summarizes the time option settings used for the nominal case runs (single median value run and multiple realization run) and for the igneous case run.

Table 6-149. Time Option Settings

Parameter	Nominal (multiple realization)		Nominal (median)		Igneous (multiple realization)	
# Timesteps	100		100		50	
Duration (years)	100,000		1,000,000		100,000	
Customized Timesteps (years)	0 – 10,000	1000	0 – 10,000	10,000	0 – 250	31.25
	10,000–100,000	500	10,000 – 100,000	500	250 – 4000	62.50
			100,000 – 150,000	1000	4000 – 10,000	125.00
			150,000 – 200,000	2000		
			200,000 – 1,000,000	4000		
Timestep subdivision factor	12		12		16	

The simulation time frames, timestep sizes, and timestep subdivision factors were chosen to optimize the computational efficiency of the model run, as well as to ensure sufficient detail in the results to capture the behavior of the dominate processes that occur over the duration of each simulation time interval.

In the Monte Carlo options a median value simulation is run by selecting "Run median value simulation"; a multiple realization simulation is run by selecting "# Realizations" and then setting it and "# Histories to save" equal to the number of realizations to be run (see Figure 6-239). The Latin-Hypercube Sampling option is selected, as it better models the "tails" of input probability distributions when the number of realizations is relatively small (DOE 1998 [100500]). In order to reproduce probabilistic sampling results the "Repeat Monte Carlo sampling sequences" option is selected. This causes the same random seed to be used each time the simulation is run. A value of 194446649 was used for the random seed.

Results and Verification

The links between the elements within the *Simulation_Settings* container been verified to be correct, as have the links from those elements to other parts of the model. The values of *Realz_Number_UZ* and *Realz_Number_SZ*, as well as the elements in the *Switches* container (see Table 6-147) were examined under both nominal and igneous settings and were found to be correct. The files associated with the external file elements in the *External_Files* container for the nominal and igneous cases were examined and found to contain the correct values per

Table 6-148. DTN: MO0009MWDMED01.020 [152838] contains the GoldSim model files and external files that were examined.

Verification of the simulation settings requires that the selections for a dynamic simulation for both the median value run and the Monte Carlo analysis be considered as appropriate to address the desired resolution of the run, as well as the uncertainty in the distributions for the stochastic parameters. The basis for the selections defined above rely on previous TSPA analyses (DOE 1998 [100550], CRWMS M&O 1995 [100198]), as well as conceptual and mechanical considerations of the dynamics of the TSPA-SR model.

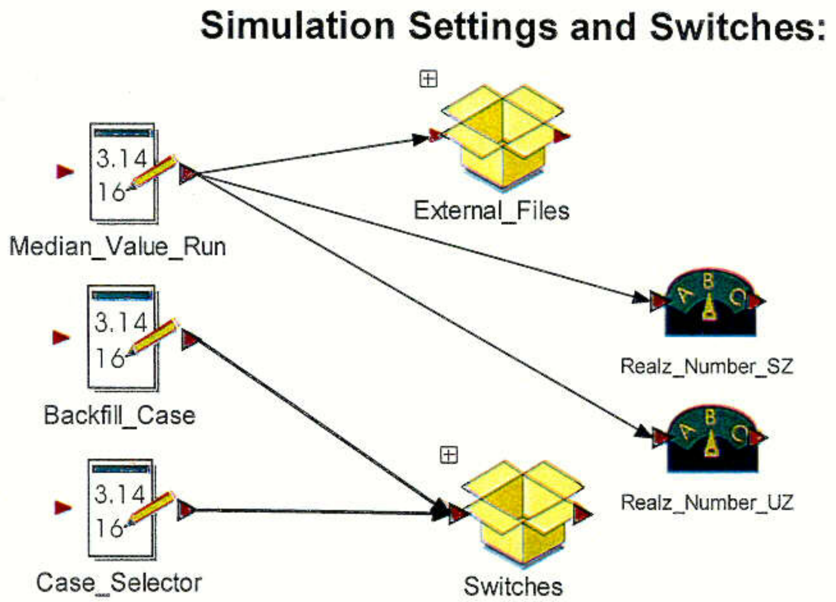


Figure 6-235. Simulation Settings and Switch in the *Simulation_Settings* Container for the TSPA-SR Model

C51

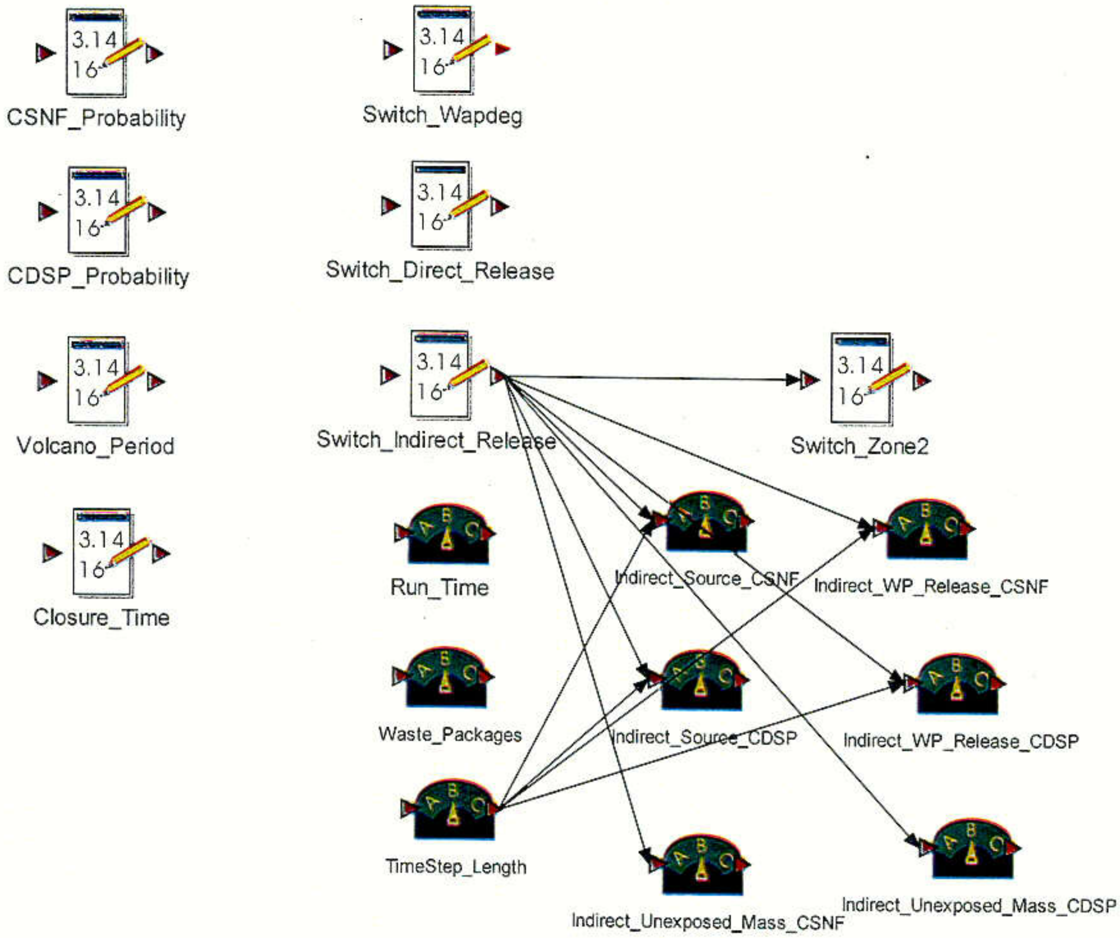


Figure 6-236. Simulation Settings and Switch for the TSPA-SR Model

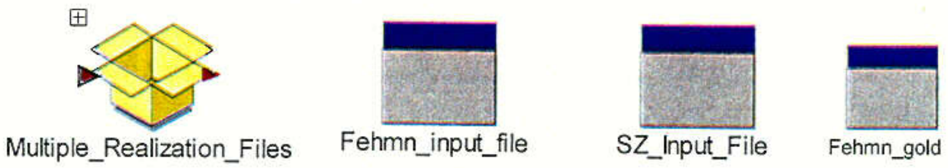


Figure 6-237. File Elements in the *External_Files* Container

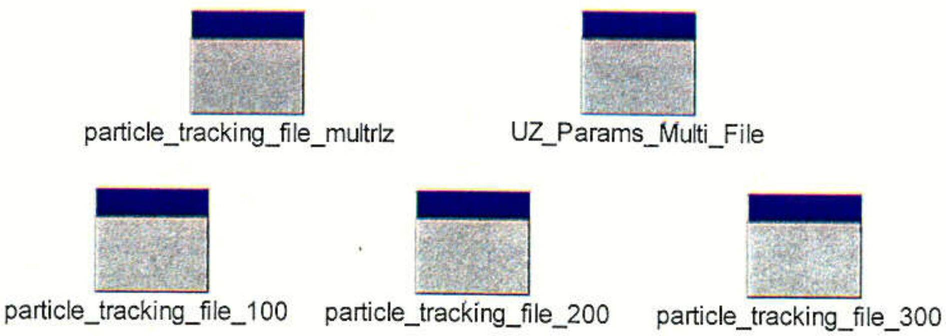


Figure 6-238. File Elements in the *Multiple_Realization_Files* Container

C52

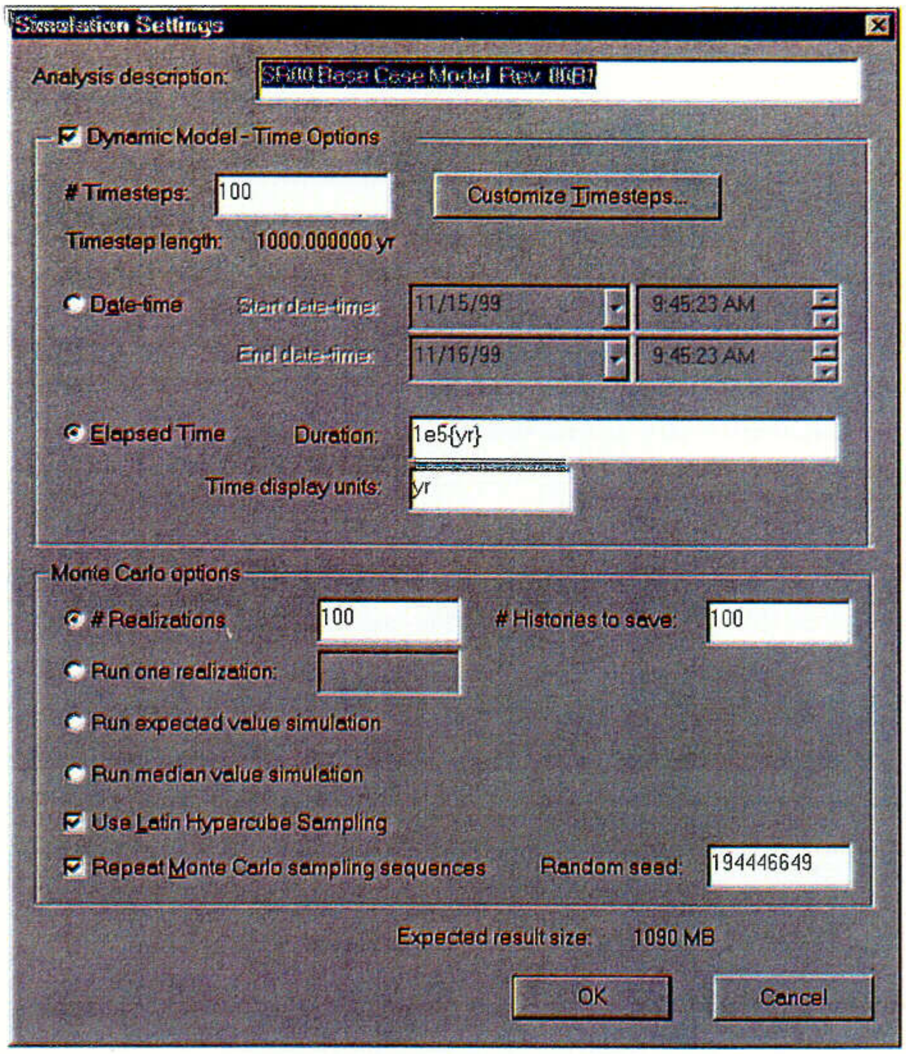


Figure 6-239. Simulation Settings Pulldown Menu for the Nominal Case, 1E5 years, 100 Realizations

C53

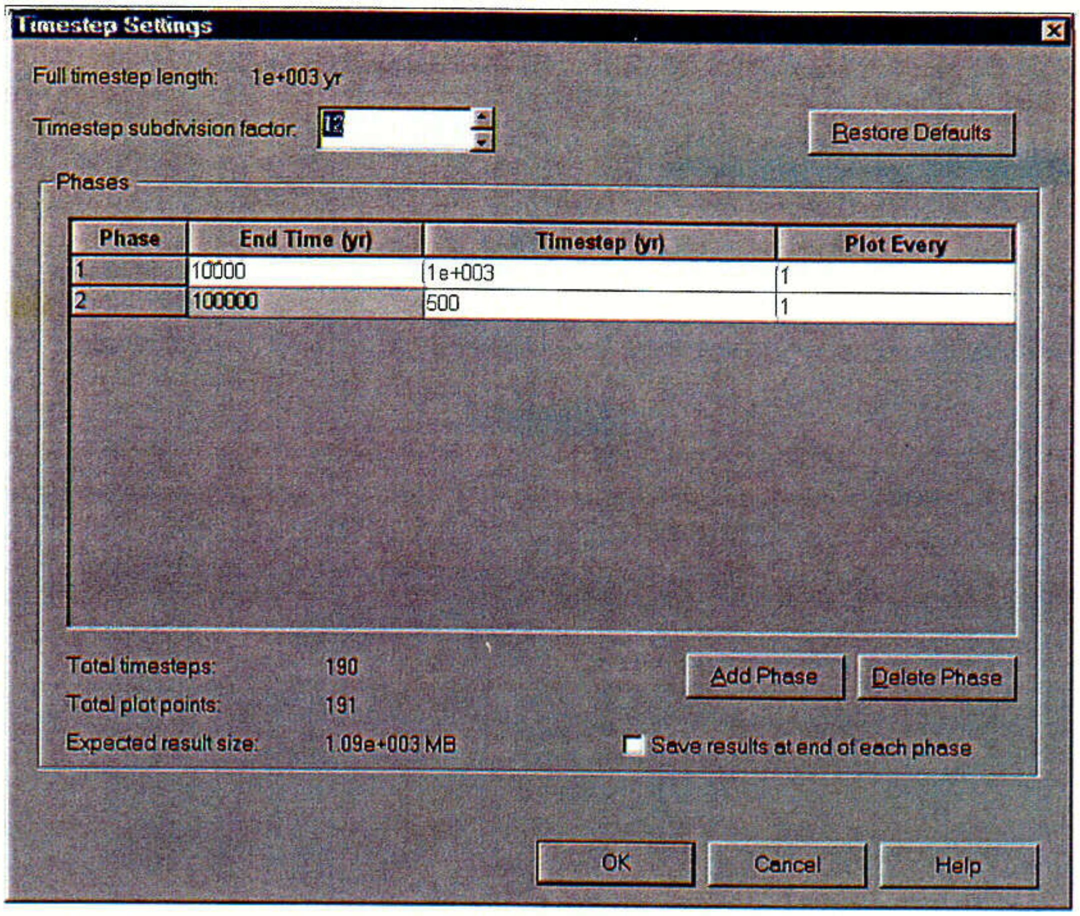


Figure 6-240. Timestep Settings for Nominal Case, 1E5, 100 Realizations

6.5 MODEL VALIDATION

Development of the TSPA-SR model was based on supporting abstraction and process-level models that represent different aspects of the repository. These abstraction and process-level models were specifically developed for use in the TSPA-SR model for Yucca Mountain, and are therefore, the output from these models are appropriate for use as inputs to this model.

The hierarchical aspect of total system performance assessment modeling is based on a sequence of modeling activities that starts with the development of process level models that are intended to capture the key aspects of the natural and engineered system for which they were developed. In turn these process models are frequently simplified into what are termed abstraction models. These simplified models are compared to the process models on which they are based to build confidence and to insure that the key aspects of the system are being captured. Once confidence in these processes and abstractions is demonstrated, they become key components in determining the validity of the total system model, where the total system model is probabilistic and stochastic in nature and is intended to capture the behavior of the entire system. These modeling activities are intended to build upon each other sequentially so that when the total system model is finalized, one is confident that the total system is adequately represented.

C54

Currently, model validation is defined as "a process to determine and document the adequacy of the scientific basis (i.e., confidence) for a model and to demonstrate that the model is appropriate and adequate for its intended use" (AP-3.10Q [152363]). Thus, model validation of the total system model depends upon the confidence building activities that are conducted for the key underlying process and abstraction models. The scientific process established on the Yucca Mountain project to accomplish model validation includes comparing analyses or modeling results to data acquired from the laboratory, field experiments, natural and man-made analog studies or other relevant observations such as classical case histories from the literature. In addition to these technical confidence-building activities, the documentation process insures the traceability, transparency and quality assurance of key modeling inputs such as data, assumptions, and computer software. Given that the component models of the TSPA-SR model undergo verification and validation independently within the source AMRs (see Section 4, Table 4-1 for list of AMR inputs) it remains to be demonstrated that the integrated model is validated, with emphasis on integrated data or results and the flow of data from each sub-component to the next. Criteria that can be used to demonstrate this integrated model validation consist of (1) the evaluation of the final results (in this case dose) in comparison to intermediate sub-system results, (2) the mechanical aspects of implementing the AMR abstractions within the TSPA-SR model, including appropriate use of associated DLLs, and (3) ensuring proper data is passed between each DLL and the GoldSim code. Only when both the verification of the subsystem models and the review of the integrated model has been completed can confidence in the model be demonstrated. This process has been done in compliance with alternative model validation approached defined within AP-3.10Q [152363] Section 5.3 Part C, item 3 (Sections 6.5.1 through 6.5.4 of this document). Additionally, further basis for the model validation is provided in Section 6.5.5 Peer Review, in accordance with AP-3.10Q [152363], Section 5.3 Part C, item 1.

For each of the process-level or abstraction analyses or models used as direct inputs to or component models of the TSPA-SR model a "Results and Verification" subsection is included in Sections 6.3.1 through 6.3.9 of this AMR. These subsections show the results from a median-value simulation (i.e., median values for all input parameters), and show that the process-level or abstraction models from the supporting AMRs have been implemented appropriately into the TSPA-SR model. The following section discusses an "integrated testing" approach to the validation of the TSPA-SR model.

6.5.1 Integrated Model Testing

Validation testing of the integrated TSPA-SR model has been conducted in two phases. This section briefly reviews the strategy, and discusses the details of implementing the validation testing scheme for the TSPA-SR model.

In phase-1, the computer model refers to a digital rendering of the conceptual model of the true physical system, namely, the YMP site. The validation, thus, relates to verifying that all aspects of the conceptual model are correctly implemented in the construction of the input for the simulation code "GoldSim". The adequacy of the chosen conceptual model to represent the complexity of the YMP site can best be addressed by a peer-review-process and is beyond the scope of this verification exercise, but *is* discussed in Section 6.5.5.

Specific criteria used to demonstrate the phase-1 verification include:

6.5.1.1 Verify that data input values for each parameter are consistent with AMR values.

6.5.1.2 Verify that all subsystem equations are input/implemented correctly per AMR recommendations.

6.5.1.3 Verify that each subsystem's input/output is as expected and as required.

Verification in phase-2 ensures that the simulation code GoldSim provides correct output for a given input (model). This verification was undertaken with a focus on the complexity in the different simulated processes related to the natural and engineered barrier systems, and also with a focus on the architecture or structure of the code. The verification of the code input and verification of the code output for a given subsystem (or AMR) input should provide assurance that the TSPA-SR results are correct as modeled.

Specific criteria used to demonstrate the phase-2 verification include:

6.5.1.4 Verify that external and internal component models transfer data appropriately.

6.5.1.5 Verify that each component model (as identified in Section 6.3 and described in Sections 6.3.1 through 6.3.9) within the integrated model calculates as expected.

6.5.1.6 Verify that total system results are consistent with subsystem results.

6.5.1.7 Verify that each downstream component model is responding as expected to upstream feeds.

Figure 6-241 illustrates the two phases of this verification scheme.

6.5.2 Phase-1: Verification of the TSPA-SR Model

Verification of the TSPA-SR model, specifically the inputs into the GoldSim code, consisted of ensuring that the input construction is in complete accord with the conceptual models of the different processes as developed in a series of relevant and applicable AMRs. The conceptual models provided in various AMRs have been converted into corresponding segments of model input, which are then integrated to become the TSPA-SR Model.

Translation of the conceptual model to input for the TSPA-SR model was accomplished and all components of the conceptual model were reviewed to ensure that they were incorporated. This review process also involved the author of each AMR, who was required to verify the input construction. Figure 6-242 illustrates this procedure.

6.5.2.1 Accuracy of the Input Data Fields (Criteria 6.5.1.1, 6.5.1.2, 6.5.1.3)

The correctness of all alphanumeric entries in the input fields has been checked thoroughly (Criteria 6.5.1.1). The verification includes checking primarily the data elements, the algebraic and/or logical expressions, and the set-up of the selector switches (Criteria 6.5.1.2). Refer to each component model section in Section 6.3.1 through 6.3.9 (Criteria 6.5.1.1, 6.5.1.2, 6.5.1.3).

Data and Function Elements

Data elements contain either a scalar numerical value (e.g., glass dissolution rate) or a vector of values (e.g., radionuclide inventory with numeric fields defining the inventory and the alphanumeric fields defining the radionuclide identifiers). Data elements may also contain vectors of entries that point to a scalar data element defined elsewhere in the model. Sometimes data elements may include an "if-then" statement or simple expressions. For each component model all data element entries have been thoroughly scrutinized.

Similarly function elements have been checked for the correctness of the formula (algebraic and/or logical) contained in the element. Each section in 6.3 defines the data inputs and verifies the correctness of each.

Selector Switches

Selector switches are used in the model to set-up "if-then" situations. The simplest switch selects the type of climate depending upon the elapsed simulation time (a deterministic trigger, see Section 6.3.1.1). The switches may be triggered by a random variable drawn from a prescribed statistical distribution to simulate chance occurrences (e.g., infiltration scenario). More complex switches are used to represent nested "if-then" situations. For example, modeling of the in-drift thermal-hydrologic (T-H) environment is based on looking up a table of multi-dimensional responses generated from an extensive process model study (the abstraction process, see Section 6.3.2.1), which includes spatial heterogeneity of T-H response over the repository area. If one location is considered in a given submodel, such as the CSNF temperature in the 3-10 mm/year infiltration bin, the required table is selected from the full suite of tables that cover all repository locations. In this case the selector switch is a function of the waste package type (CDSP or CSNF), the infiltration bin (ranging from 1 through 5) and the infiltration scenario (low, mean, or high). The setup of all selector switches has been checked for correctness.

Stochastic Variables

The distributions of the various stochastic variables have been checked for agreement with the distributions prescribed in the relevant AMR.

Tables

Many of the data elements contain vectors and matrices that are simply tables. As a part of the verification of the data elements, they have been verified. In addition some tables are generated by abstractions of the underlying process models. All of these latter tables are generated by computer programs that have been qualified. For example, the tables of thermal-hydrology variables in the drift at different locations are generated by the NUFT computer program (see Section 6.3.2.1) and are used to populate the tables in GoldSim. They are retrieved from the Technical Data Management System (TDMS) and are verified before being included in the TSPA-SR model. Like all data, table values are verified to be entered into the TSPA-SR model correctly during the technical review associated with the subsection of the model that contains the data table.

6.5.3 Phase-2: Verification of GoldSim TSPA-SR Model

Phase-1 verification provides assurance that the TSPA-SR Model is in full conformity with the conceptual model of the YMP site. Phase-2 verification seeks to ensure that TSPA-SR Model provides the correct output for a given set of inputs, based on the full-scale complexity of the YMP site.

It must be stated at the outset, that the GoldSim code has passed through a series of rigorous tests by its developers to ensure the correctness of its output (Golder Associates 2000 [151202]). Nevertheless, considering the complexity of the processes simulated in the natural and engineered barrier systems at the YMP site, many of which are handled via external DLLs (e.g., WAPDEG, SZ_Convolute, FEHM, etc.) and considering the fact that some of these codes derive their input from the output of another upstream code(s), the need to validate the code performance under the full-scale complexity of a realistic YMP model is warranted. Phase-2 verification addresses the validation of the TSPA model from this perspective.

The Phase-2 verification consists of three stages. Figure 6-243 explains stages 1 and 2 and Figure 6-244 explains stage 3.

6.5.3.1 Stage-1: Function Evaluations and Selector Switches (Criteria 6.5.1.4)

The GoldSim code can compute some model outputs/equations via data elements that employ user-prescribed functions. These may depend upon the intermediate output of another segment of this model. For example, in the in-package chemistry model, the pH in the package is a function of three factors: (1) the water flux through the package, (2) the waste package degradation rate, and (3) the degree of clad coverage for CSNF packages or the glass dissolution rate for CDSP packages. The pH values are modeled by linear regression equations which form a response surface based upon: (1) upper bound and lower bound dissolution rates of the waste form, (2) early (i.e., less than 1000 years after waste package breach) and late (greater than 1000 years after waste package breach) times; and (3) CSNF and CDSP waste packages. For each of these combinations (eight total), pH is computed based on the three factors cited above, which are time-dependent. The first factor is derived from the EBS transport model, the second one from the waste package degradation model, and the third from the cladding degradation model. Thus, correct evaluation of the pH in the model would firstly imply the correct selection of the appropriate regression equation, depending upon the waste package type (CSNF/CDSP), waste form dissolution rate (high/low), and the time (early/late), and implies correct functioning of the selector elements. Secondly, it would imply the selection of the correct input variables at the selected time from the relevant submodel. Thirdly, it would imply the correct evaluation of the chosen regression equation. Section 6.3.4.2 explains the details of the in-package chemistry model and its verification. The model computations of pH are verified implying simultaneously the correctness of the relevant selector switches, correct data transfers, and the correct function evaluations.

Similar verification of the TSPA-SR Model components is given in the relevant results and verification discussions in Section 6.3 (e.g., near-field environment in Section 6.3.2.1, in-drift geochemical environment in Section 6.3.2.2, EBS transport in Section 6.3.5, etc.). Note that these verifications are undertaken while using the full-scale TSPA Model. These model subsections involve the computations performed by GoldSim code itself and not by an external DLL. Verification of the computations performed by external DLLs under the command of the GoldSim code are presented below.

6.5.3.2 Stage-2: Dynamically Linked Library Routines (Criteria 6.5.1.5)

In addition to the direct computations undertaken within the GoldSim code, whose verification has been demonstrated above in Section 6.5.2.1, major process simulations are performed via external codes, such as WAPDEG, FEHM, etc. Those routines were initially built and validated as independent stand-alone codes (see Section 3.0). They were then incorporated into the TSPA model as DLLs. Some input data are transmitted from the GoldSim code through an argument list and other input data are read from data files. Most of these data are output from another component model (or a DLL). As an example, SEEPAGE DLL reads data files of time-dependent infiltration at several locations in the repository, which were generated by the near field environment thermal-hydrology process model abstractions. The correctness of each type of input to a DLL is verified and can be referenced by DTN.

It remains to be demonstrated that the outputs of a DLL are correct when the full scale TSPA-SR Model is implemented. It has been mentioned in Section 6.5.2.2 that the DLLs have been validated under the GoldSim code command or under an equivalently mimicked computational environment. In view of such extensive validation of the DLLs, their performance in the full scale TSPA-SR Model is assured to be correct, if there are no errors in data transfers to the DLLs. The final validation of the integrated model is, therefore, directed at checking the data transfers to and from a DLL.

The inputs to a DLL from the TSPA model were written to an output file, taking care to identify the DLL from which the data was printed (see Attachment IX for examples of output files for select DLLs). Those data were compared to their correct values that are known outputs from another upstream DLL or model component. In some cases it is possible to infer that such data transfers are error-free automatically. For example, when the seepage DLL reads the data of time-dependent infiltration at different locations from a file developed by the thermal-hydrology model abstraction and if the data transfer has already been validated per AP-SI.1Q [153201], it follows that the data transfer would be correct in the integrated model too. Even so, to avoid any possible pit-falls, the data transfers to and from SEEPAGE DLL have been verified in the integrated model (see Section 6.3.1.2).

Several examples can be shown to verify that the input to the DLL is correct by reviewing the various log files for each associated DLL. Attachment IX contains several excerpts from log files of associated DLLs that echo the information passed to the DLL from the GoldSim code. Verification that the data is correct adds to the information necessary to conclude model validation. For example, in Attachment IX, the WAPDEG.DLL output file WAP4DLL.ina echoes the input array posed from the GoldSim code. Spot checking this output file for consistency with the input array in the GoldSim code (*WAPDEG_Inputs*) verifies the data was passed correctly. For instance line 11 in the input array (WAPDEG_Inputs) is 1.0e-4, while in line number 18 of the WAPDEG.DLL output file WAP4DLL.ina (see Attachment IX), the identical value of 1.0e-4 is echoed. Again, checking line 1090 in *WAPDEG_Inputs* yields a value of 10 which is also echoed in the WAPDEG DLL output file, line 92.

Another example relates to the input for the FEHM DLL for UZ transport. Under the integrated model, the input to the FEHM DLL at selected times from the EBS have been compared to the expected output. Attachment IX has an excerpt from the FEHM DLL output file fehm.out. This file contains both input values passed to the FEHM DLL from the GoldSim code and associated input files, as well as output from the DLL. Since the DLL itself is already qualified, we only need to demonstrate that the data transfer between the GoldSim code and the DLL is functioning correctly. Line 225 in Attachment IX lists two values passed from the GoldSim code to FEHM DLL during the median value base case run. The first is the *Flowfield_Index* the next is the *Realz_Number_UZ*. For the median value case the *Flowfield_Index* is 200 during the first climate state and the value for *Realz_Number_UZ* is 1. Both values are echoed correctly within the FEHM DLL output file. The values read into the FEHM DLL from the associated input files have also been verified using this method.

Additional examples of external DLL output files are show in Attachment IX for both the SZ_Convolute DLL and the ASHPLUME DLL. Since each DLL is validated individually under AP-SI.1Q, the verification of error-free data transfers between the different components in GoldSim, when the integrated model is implemented, provides unequivocal assurance that the output from the TSPA-SR model is correct, even for an input model which encompasses the full complexity of the conceptual model of the YMP site.

6.5.3.3 Stage-3: Integrated Model Output Testing (Criteria 6.5.1.6 and 6.5.1.7)

Integrated model output testing can be accomplished by careful evaluation of the model results, in this case dose, in response to the upstream feeds. For the TSPA model the general measure of performance is dose. The dose is calculated from the concentration of radionuclide species in the groundwater 20 km downgradient from the potential repository (see Section 6.3.8). To validate the model it is expected that a particle or given mass can be followed through the entire system, from the waste form to the EBS, to the UZ, to the SZ, and out to the accessible environment. Since each subsystem component is integrated within the total system model, each is affected by the logical order of models/processes that precede it, i.e., by upstream models. The total system integrated model is the sum of the subsystem models coupled together using common input data and propagating changes in a logical order through the system, during a simulation. It can be demonstrated through a series of plots (Figure 6-245 through Figure 6-250) that the integrated total system model is performing as expected.

The logical order of influence during a TSPA model simulation is a function of the scenario class and the scenario. The scenario is defined at runtime by a combination of certain model parameters. For instance, *case_selector* and *infiltration_scenario* would be selected first to define a nominal or igneous scenario with low, medium, or high infiltration (see Climate and Simulation Settings, Sections 6.3.1.1 and 6.4 respectively). The next logical step in the model simulation are calls to external DLLs. For a nominal scenario case this would include waste package degradation (WAPDEG DLL, Section 6.3.3) and seepage (Seepage DLL, Section 6.3.1.2), and for an igneous scenario, it would include those two DLLs plus the ASHPLUME DLL. After the DLLs are completed various internal model components, like the number of waste packages per bin, are calculated (see Section 6.3.1.2).

To validate the coupled model it must be demonstrated that the coupled processes are working as expected. Figure 6-245 is the first of several figures that demonstrate the correct coupling in the integrated model. It shows that the calculated number of failed packages in a source term group (in this case CSNF BIN4 Intermittent Drip) is correctly computed as the total number of packages in the source term times the fraction of failed packages calculated by the WAPDEG DLL (see Section 6.3.3). Figure 6-245 shows the number of packages failed, equal to 1 at 41,000 years when the fraction failed is equal to 0.0025. The number of failed packages continues to grow as the fraction failed calculated by the WAPDEG DLL grows, reaching a maximum of 388 packages in the CSNF BIN4 Intermittent Drip environment when the WAPDEG failure fraction reaches 1.

As can be seen in Figures 6-246 to 6-250, the coupled models begin to re-calculate only after waste package failure. For example, Figure 6-246 shows how pH is dependent upon the average waste package failure time, since it can be seen that the pH begins to fluctuate immediately after the first failure at 41,000 years. Figure 6-247 then shows how the clad unzipping rate is a function of pH and therefore also a function of the waste package failure. It clearly indicates how the clad unzipping rate is a mirror image of pH. Thus, Figure 6-246 and Figure 6-247 build confidence in the correct working of the total system model, i.e., the coupled processes are behaving as expected.

A simpler coupled process, yet equally important, is shown in Figure 6-248. The drip shield patch fraction is calculated as the ratio of the number of patches in the drip shield to the total number possible (see Section 6.3.3). The flux through the drip shield (*Qflux_DS*), (i.e., the flux that reaches the waste packages in dripping and sometimes dripping environments), is calculated as the patch fraction (i.e., fraction of degraded drip shield surface area) times the seepage into the drift (see Section 6.3.5). As can be seen in Figure 6-248, the *Qflux_DS* parameter is clearly scaled to the patch openings in the drip shield because as the fraction of drip shield patches approaches 1, the flux through the drip shield (*Qflux_DS*) approaches its maximum value.

Figure 6-249 and Figure 6-250 trace the mass release through the system for two radionuclides, Tc-99 and Np-237. Shown are cumulative mass exiting the waste packages (Section 6.3.3), cumulative mass exiting the engineered barrier system (Section 6.3.5), cumulative mass exiting the unsaturated zone (Section 6.3.6), and cumulative mass exiting the saturated zone (Section 6.3.7). As seen in Figure 6-249, the mass of Tc-99 released from the waste packages through the EBS begins only after waste packages have failed (as demonstrated by comparison to the waste package failure curve produced by the WAPDEG DLL). Prior to waste package failure, the mass retained in the waste packages is only decreased by natural decay of the radioactive species. Following the Tc-99 mass through the system model, the mass released from the UZ overlays the mass released from the EBS. Thus, the mass released from the UZ is clearly controlled by the mass input from the EBS—in this case they overlay since there is virtually no retardation of Tc-99 in the UZ. Continuing on and following the mass through the SZ, Figure 6-249 shows a very minimal retardation through the SZ for Tc-99, with the SZ cumulative release bounded by the release from the UZ and EBS. For Tc-99, it can be concluded that the total system model behaves as expected for Tc-99.

Figure 6-250 traces a weakly sorbing radionuclide, Np-237, through the total system model. As with Tc-99, release of Np-237 is initially controlled by the waste package failure, then dissolution or unzipping of the fuel matrix. Also, in contrast to Tc-99, Np-237 has a fairly low solubility, which further constrains its release. Additionally, Np-237 can be retarded in both the UZ and SZ. As seen in Figure 6-250, the Np-237 released from the EBS begins as the waste package failures start, similar to the behavior of Tc-99. In this case however, there is a delay in the release from the UZ as compared to the EBS cumulative breakthrough due to sorption of Np in the matrix of the UZ. The cumulative mass released from the SZ is even further delayed because the SZ alluvium retards the release of the Np species (Section 6.3.7). Again Figure 6-250 demonstrates, this time for a sorbing species, that the cumulative release of mass through the total system model behaves as expected.

This section has validated and tested the integrated model against criteria 6.5.1.6 and 6.5.1.7, viz., that total system results are consistent with subsystem results and that each downstream model is responding as expected to upstream feeds. For example, the EBS release is controlled by the waste package failures, pH, dissolution of the matrix (unzipping rate), and the water flux (Figure 6-245 to Figure 6-248); the UZ release mimics the EBS behavior with appropriate modifications due to retardation in the UZ; and the SZ mimics the behavior of the EBS and UZ with appropriate changes due to retardation in the SZ (Figure 6-249 and Figure 6-250).

The processes and models analyzed in this section are not the only coupled processes in the integrated model. Other factors such as T-H effects, solubility, colloid concentration, etc. simultaneously influence several of the submodels. However, as required for model validation (AP-3.10Q [152363]), confidence in the integrated TSPA Model has been demonstrated by following the key subsystem model outputs in a logical order of influence through the total system.

6.5.4 Summary of Integrated Model Testing

The TSPA-SR model has been carefully scrutinized to establish its agreement with the conceptual models developed in the relevant AMRs. This included verifying all of the data fields in the TSPA-SR model. This verification ensures that the input digital model is in accord with the conceptual model (Criteria 6.5.1.1, 6.5.1.2, and 6.5.1.3).

The internal computations performed within the GoldSim code have been verified to be correct when the integrated model is implemented (Criteria 6.5.1.4).

All the external dynamically linked library routines (DLLs) have been verified under the GoldSim code command. The data transfers to and from the DLLs in the GoldSim code have been verified when the integrated model is implemented (Attachment IX). More details of the verifications can be found in the results and verification subsections under Section 6.3 (Criteria 6.5.1.5).

The integrated total system model behaves as expected, and results from each subsystem model component are consistent with the entire total system model (Criteria 6.5.1.6 and 6.5.1.7).

This sequence of validation tests provides confidence that the TSPA model output does represent the performance of the underlying conceptual models and that the integrated model is behaving as it should.

6.5.5 Peer Review

As stated in AP-3.10Q [152363], an alternative approach to model validation is the Peer Review Process or "review by international collaborations." The TSPA-SR integrated model is very similar conceptually to the preceding model used for performance assessment of Yucca Mountain, the TSPA-VA model. TSPA-VA model underwent extensive peer review (Budnitz et al. 1999, MOL.19990317.0328 [102726]) and the TSPA-VA Peer Review Panel concluded, in part:

> "The Panel believes that the basic framework or architecture of the TSPA-VA is sound, as is the use of abstractions of component models for purposes of computational efficiency. Where the Panel has concerns, it is more often due to the specific methods applied and the details of the component models, rather than with how the models were linked."

Because the TSPA-SR model is quite similar in architecture to the TSPA-VA model, this conclusion adds confidence to the validity of the integrated TSPA-SR model. If there are concerns, it would appear they would rest in the validation of the underlying component models rather than in the integrated system model. Validation of the component models is an exercise for the supporting AMRs listed in Table 4-1.

With respect to the "review by international collaborations" validation criterion, the GoldSim risk-based methodology and software has been used by nuclear waste management programs in other countries. For example, the URL **http://www.goldsim.com/Software/modules2.asp** provides the following documentation:

"The GoldSim Contaminant Transport Module has been used to address complex contaminant transport problems in North and South America, Europe, Asia, and Australia. A few of the more high-profile applications of the software (specifically, in the area of radioactive waste management) are listed below:

> <u>Spanish Radioactive Waste Disposal Research.</u> ENRESA, the Spanish radioactive waste management agency, has been using GoldSim (and RIP) since 1992 to evaluate potential host rocks as part of a program to select a disposal site for the nation's spent nuclear fuel.
>
> <u>Evaluation of Waste Disposal Sites, Los Alamos, New Mexico.</u> Los Alamos National Laboratory is using GoldSim to aid in characterizing risks and to help identify monitoring requirements for low-level radioactive waste disposal areas.
>
> <u>Remediation and Closure of Mine Workings and Facilities.</u> GoldSim has been used in Germany to evaluate alternative remediation and closure options for abandoned mine workings and tailings facilities associated with former uranium mining operations."

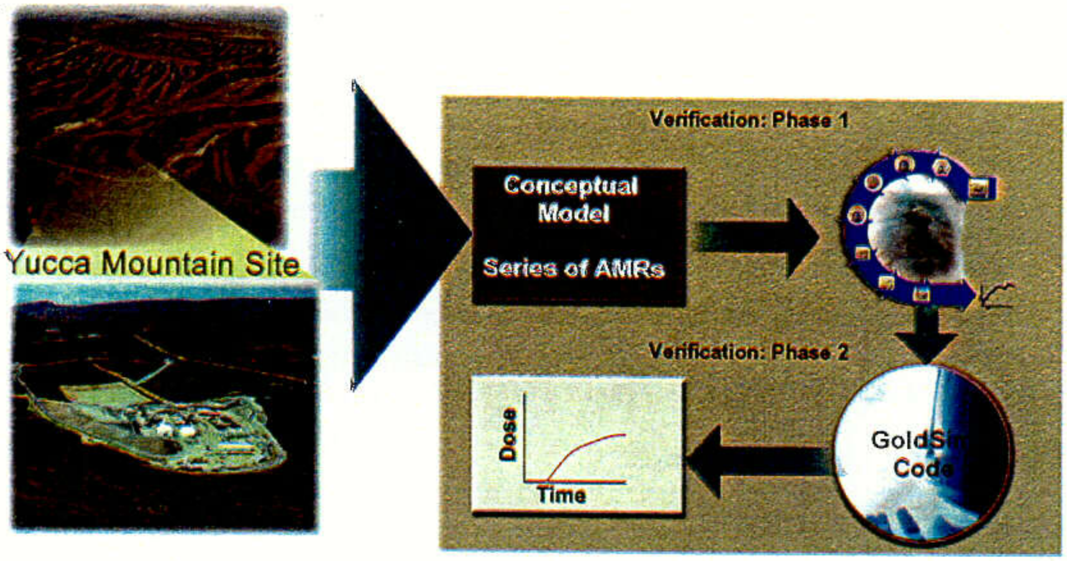


Figure 6-241. Validation of the Integrated Model: Two Phases

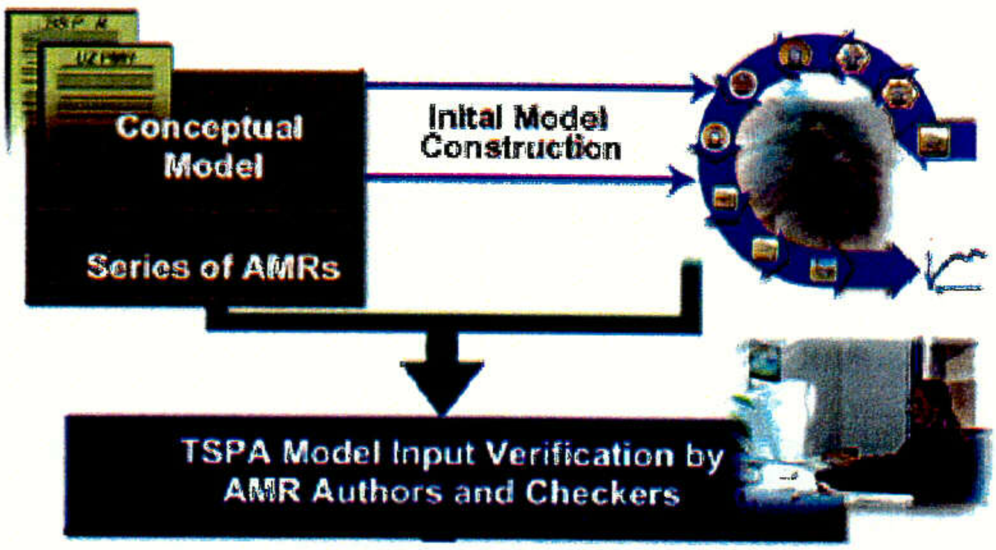


Figure 6-242. Validation of the Integrated Model: Phase 1

C55

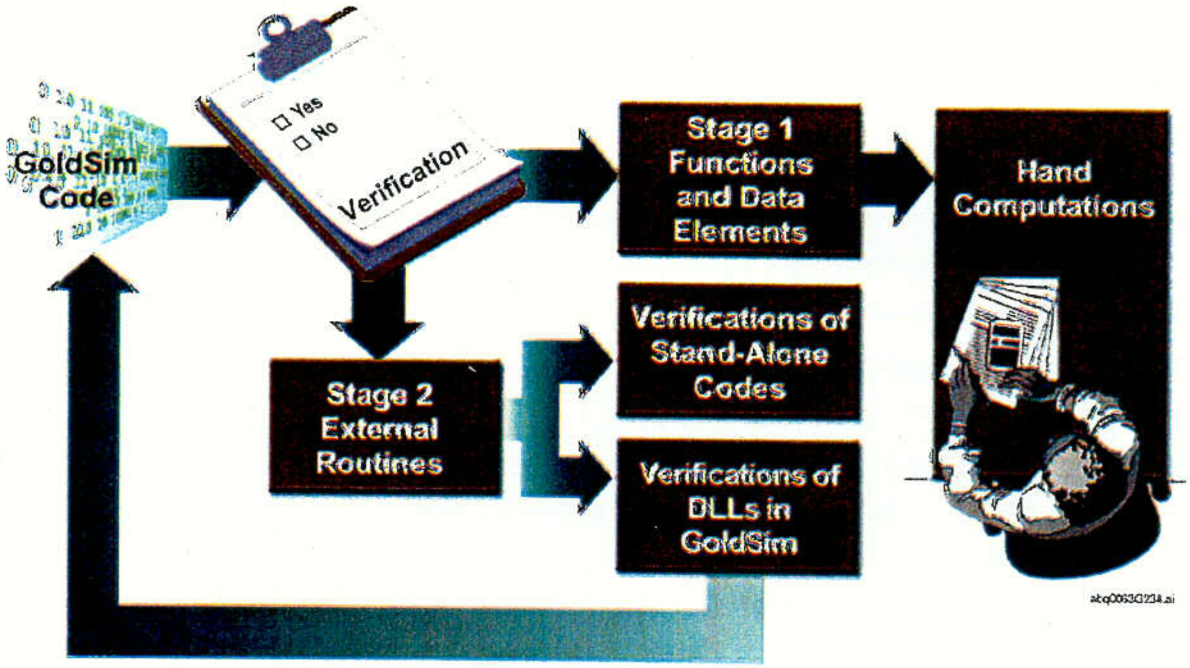


Figure 6-243. Validation of TSPA Model Codes: Stages 1 and 2

C56

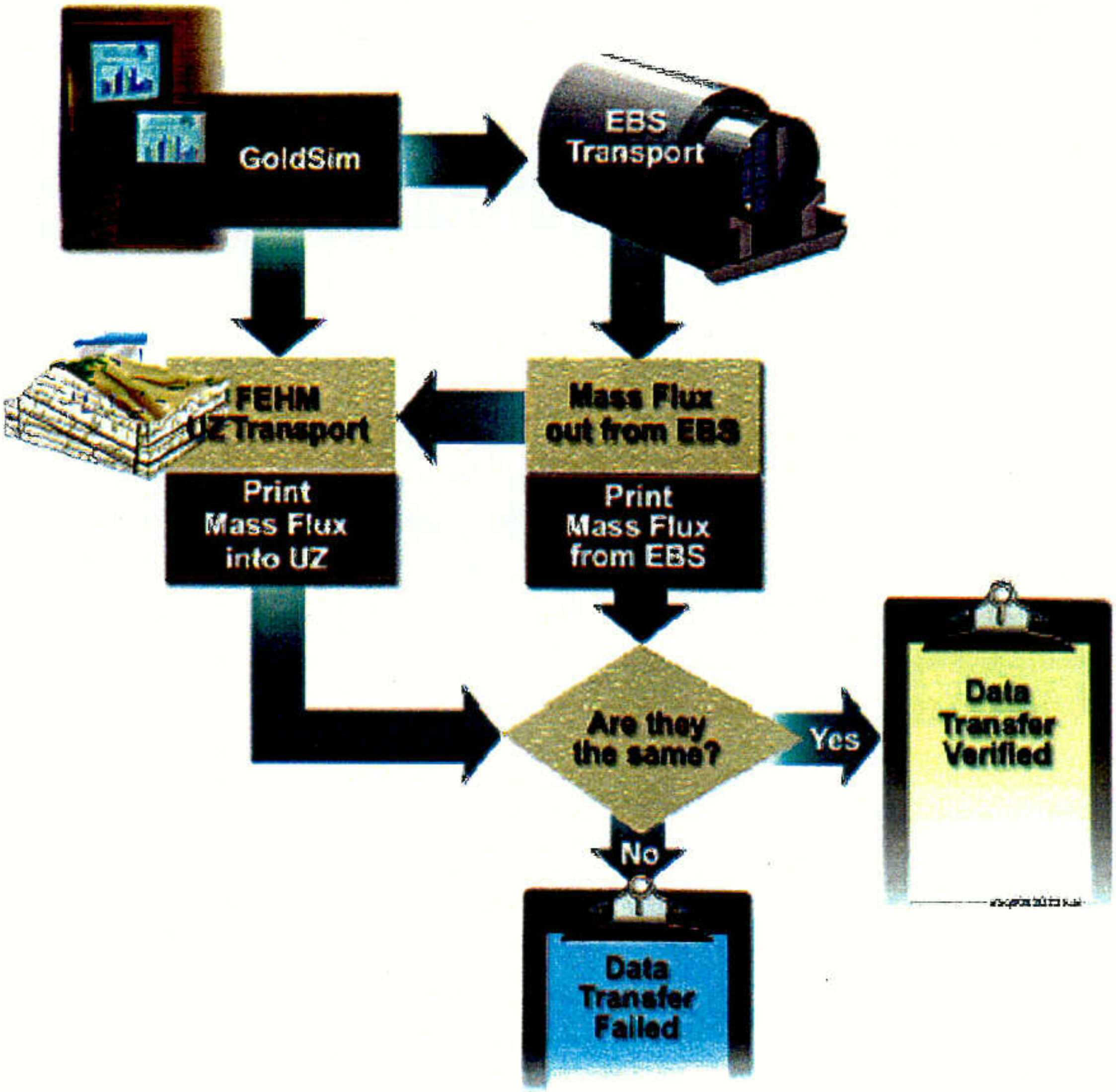


Figure 6-244. Verification of Integrated TSPA Model: Stage 3

C57

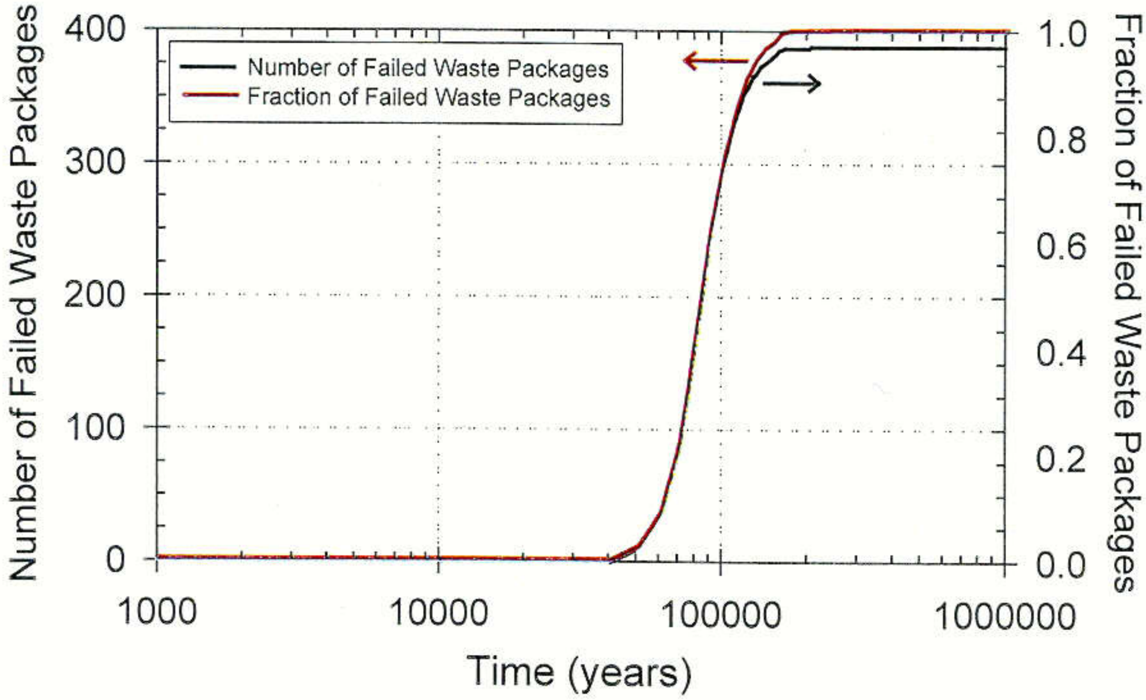


Figure 6-245. Validation of the TSPA-SR Model: Stage 3. Fraction of Failed Packages (from WAPDEG DLL) Versus Number of Failed Packages in a GoldSim Source Term Group (CSNF BIN 4 Intermittent Drip)

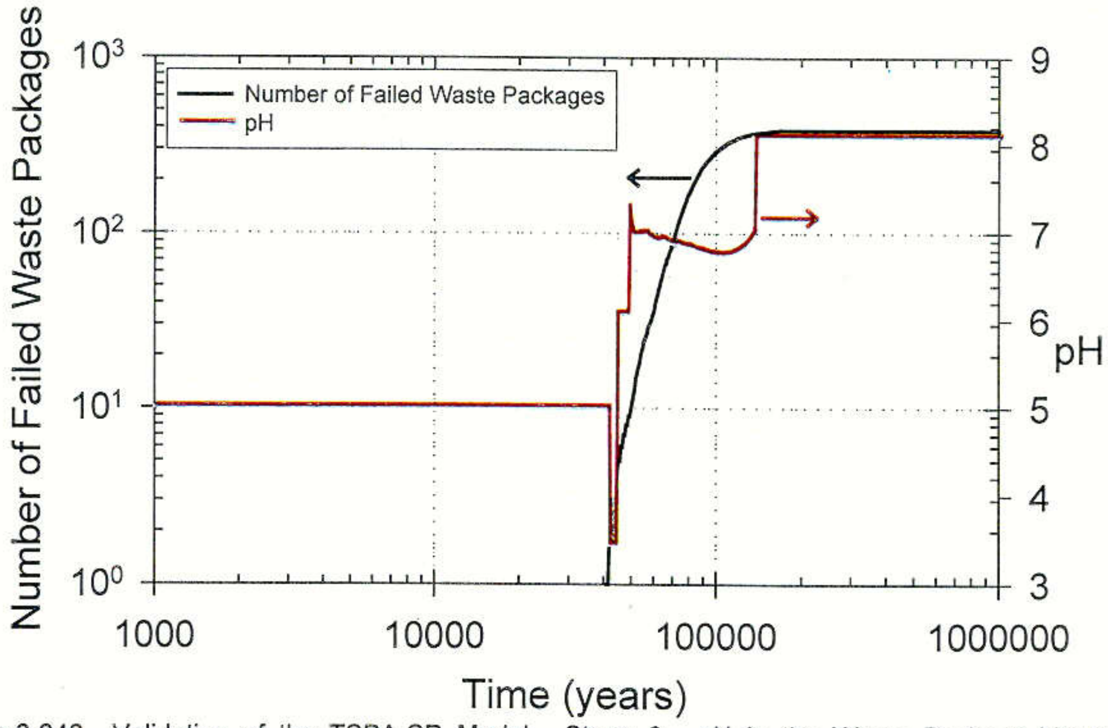


Figure 6-246. Validation of the TSPA-SR Model: Stage 3. pH in the Waste Package Versus the Number of Failed Packages (CSNF BIN 4 Intermittent Drip)

C58

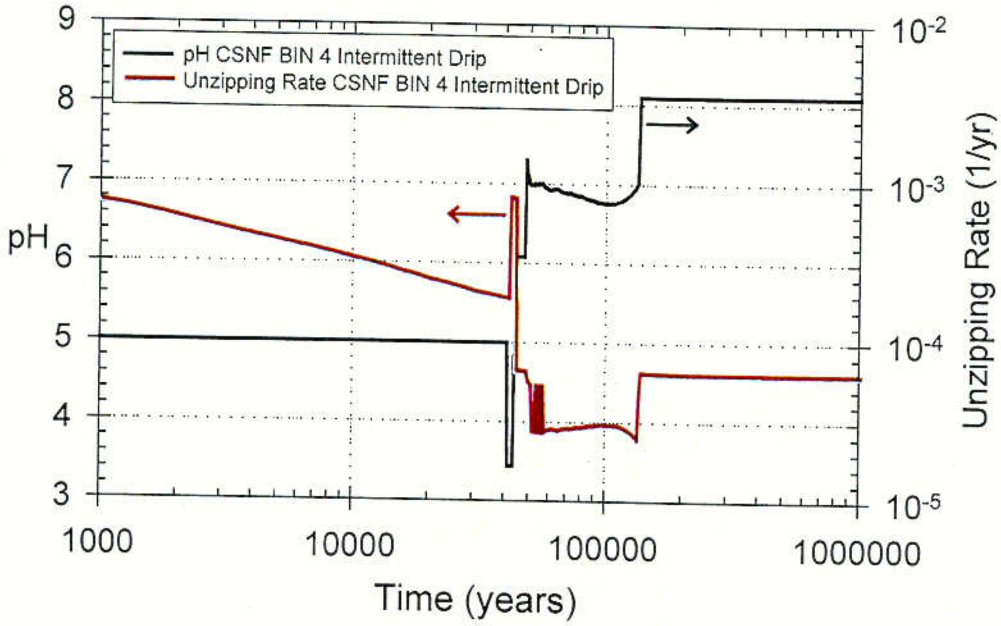


Figure 6-247. Validation of the TSPA-SR Model: Stage 3. Clad Unzipping Rate Versus pH in the Waste Package (CSNF BIN 4 Intermittent Drip)

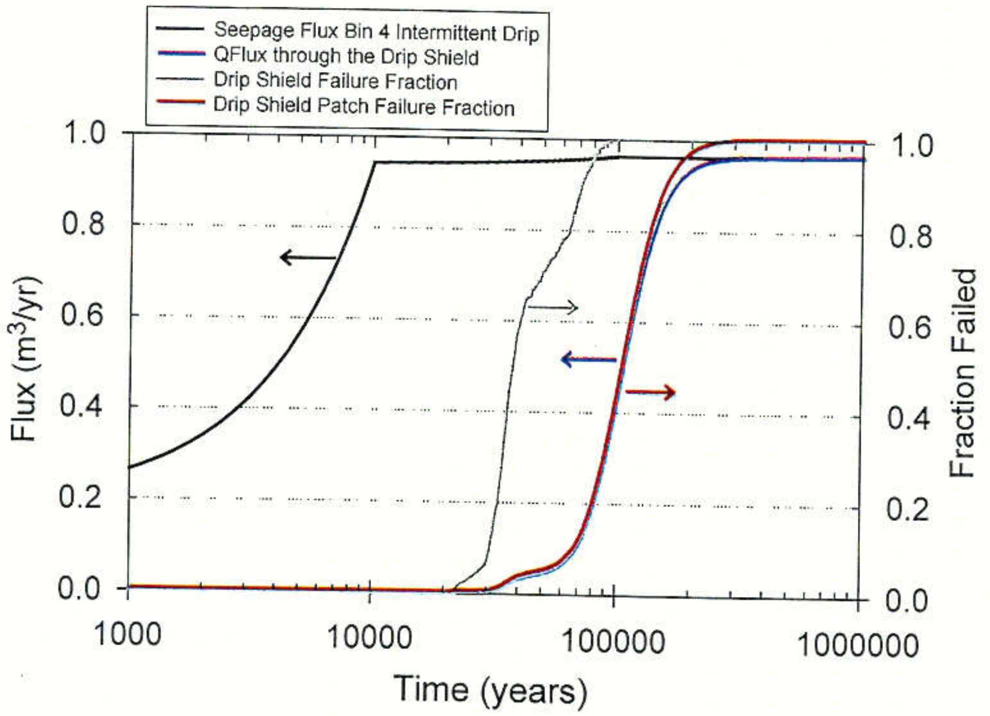


Figure 6-248. Validation of the TSPA-SR Model: Stage 3. Flux Through the Drip Shield (*Qflux_DS*) Versus Drip Shield Patch Failure Fraction. Also, Shown are Seepage into the Drift and Fraction of Failed Dripshields (CSNF BIN 4 Intermittent Drip)

C59

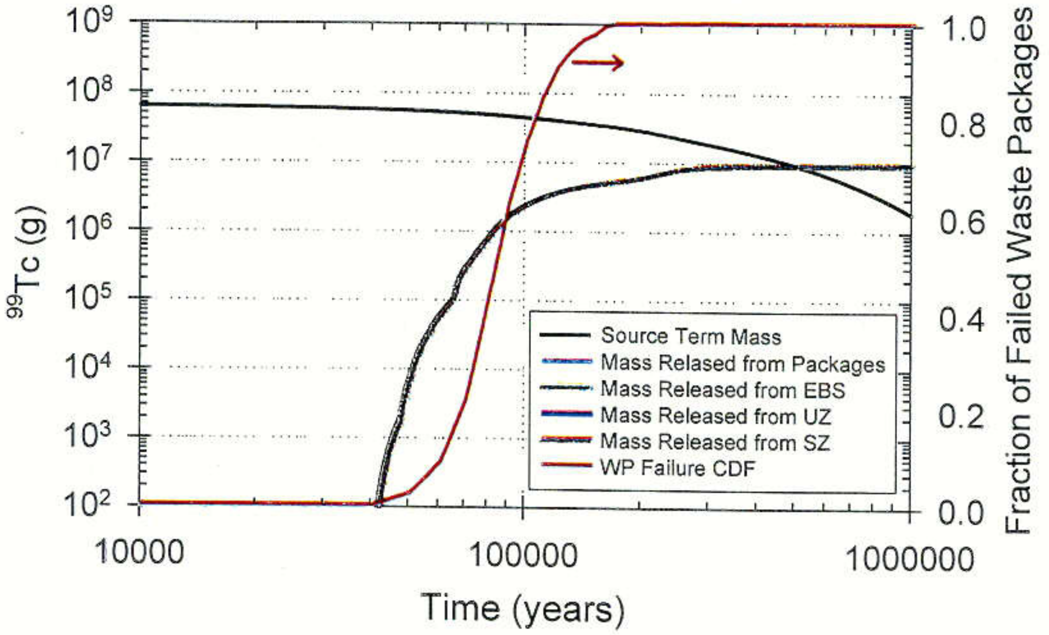


Figure 6-249. Validation of the TSPA-SR Model: Stage 3. Cumulative ^{99}Tc Mass Released from Various Submodels of the Total Integrated System Model; Compared to the CDF of Waste Package Failures

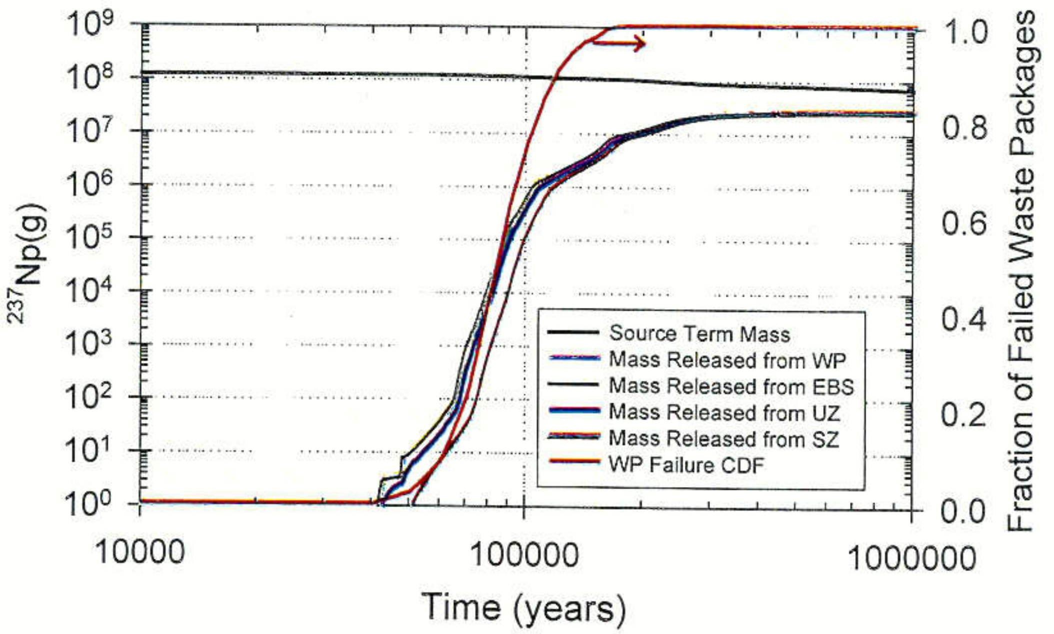


Figure 6-250. Validation of the TSPA-SR Model: Stage 3. Cumulative ^{237}Np Mass Released from Various Submodels of the Total Integrated System Model; Compared to the CDF of Waste Package Failures

C60

7. CONCLUSIONS

This AMR is used to document and validate the integration of information from multiple process models (which are documented in the supporting AMRs listed in Table 4-1) into a comprehensive total system model. The resulting integrated total system model (the "TSPA-SR model") is used to analyze the performance of the potential repository system at Yucca Mountain as part of the Site Recommendation process.

The TSPA-SR model requires numerous data feeds, as well as process and abstraction submodels. These data and submodels have been documented by component and as an integrated whole in this document. Each model component presented in Section 6 of this AMR contains a result and verification section. These subsections show that each model component obtained from supporting AMRs has been implemented correctly into the TSPA-SR model. Validation of the integrated TSPA-SR model, i.e., correct coupling of the individual components, has been demonstrated in Section 6.5.

The results of this model will allow quantitative assessment of repository performance that meets the intent of the technical requirements proposed by the NRC at proposed 10 CFR Part 63 (64 FR 8640 [101680]), and radiation protection standards proposed by the EPA at proposed 40 CFR Part 197 (64 FR 46976 [105065]).

Model Uncertainties

The model is based on inputs from multiple sources, many of which define an uncertainty range for the specific information feed. In order to demonstrate compliance with 10 CFR Part 63 (64 FR 8640 [101680]) and 40 CFR Part 197 (64 FR 46976 [105065]), the model was run in a stochastic mode to evaluate the overall uncertainty in the performance (annual dose) that results from the uncertainty in many of the input parameters (see *Total System Performance Assessment for the Site Recommendation*, [143665]). The key uncertain parameters are identified in the discussion of the model inputs for each applicable model component described in Section 6 of this AMR.

Model Inputs and Developed Data

This document may be affected by technical product input information that requires confirmation. Any changes to the document that may occur as a result of completing the confirmation activities will be reflected in subsequent revisions. The status of the technical product input information quality may be confirmed by review of the DIRS database.

The AMR's feeding the TSPA-SR model contain, in some cases, data that is in the process of being qualified. The outcome of that effort will lead to qualification of data, and may also cause some of the data to be modified. This information will be reviewed and treated similarly to the TBV's in terms of evaluating the impact and then revising the model when appropriate.

Developed data associated with the validation of the integrated model, as documented in the AMR, is contained in the following DTNs:

- The TSPA-SR cases "SR00_037ne6" and "SR00_038ne6" in DTN: MO0009MWDMED01.020 [152838] contain the TSPA-SR model and results for a nominal-scenario median-value simulation, with and without the implementation of seismic cladding failure, respectively.

- The case "SR00_001ie5" in DTN: MO0009MWDMED01.020 [152838] is the median value simulation for the igneous scenario that includes both eruptive and groundwater release doses.

- Realization #1 in the case SR00_005hm5 in DTN: MO0008MWDHUMAN.000 [152186] was used to verify the human intrusion model.

In addition, the numerous simulations (and associated developed data) necessary to demonstrated compliance with NRC and EPA regulations are documented in the DTNs listed in Appendix G of *Total System Performance Assessment for the Site Recommendation* [143665].

Restrictions on Subsequent Use

The assumptions imbedded in the model are clearly stated in the assumptions section of this document (Section 5). These assumptions are consistent with the assumptions generated in the supporting AMRs (Table 4-1). The integrated TSPA-SR model has not been utilized such that the assumptions are overridden, or such that the key components and subsystem models are used out of the range of their intended use. Future use of the model is restricted to the bounds of these assumptions. However, if future testing and process modeling warrant changes to the underlying assumptions, then the integrated TSPA-SR model will be modified accordingly and documented in future ICNs of this AMR.

8. REFERENCES

The following is a list of the references cited in this document. Column 1 represents the unique six digit DIRS number, which is placed in the text following the reference callout (e.g., CRWMS M&O 2000 [144054]). The purpose of these numbers is to assist the reader in locating a specific reference. Within the reference list, multiple sources by the same author (e.g., CRWMS M&O 2000) are ordered numerically by the DIRS number.

8.1 DOCUMENTS CITED

153283 Andrews, R.W. 2000. "AP-2.21Q Activity Evaluation for TSPA-SR" Interoffice Correspondence from R.W. Andrews (CRWMS M&O) to Distribution (CRWMS M&O), October 31, 2000, LV.PA.RWA.10/00-089, with attachment. ACC: MOL.20001129.0179.

100309 Barnard, R.W.; Wilson, M.L.; Dockery, H.A.; Gauthier, J.H.; Kaplan, P.G.; Eaton, R.R.; Bingham, F.W.; and Robey, T.H. 1992. *TSPA 1991: An Initial Total-System Performance Assessment for Yucca Mountain.* SAND91-2795. Albuquerque, New Mexico: Sandia National Laboratories. ACC: NNA.19920630.0033.

102726 Budnitz, B.; Ewing, R.C.; Moeller, D.W.; Payer, J.; Whipple, C.; and Witherspoon, P.A. 1999. *Peer Review of the Total System Performance Assessment-Viability Assessment Final Report.* Las Vegas, Nevada: Total System Performance Assessment Peer Review Panel. ACC: MOL.19990317.0328.

100111 CRWMS M&O 1994. *Total System Performance Assessment - 1993: An Evaluation of the Potential Yucca Mountain Repository.* B00000000-01717-2200-00099 REV 01. Las Vegas, Nevada: CRWMS M&O. ACC: NNA.19940406.0158.

100198 CRWMS M&O 1995. *Total System Performance Assessment - 1995: An Evaluation of the Potential Yucca Mountain Repository.* B00000000-01717-2200-00136 REV 01. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19960724.0188.

101112 CRWMS M&O 1998. *Software Routine Report for SZ_CONVOLUTE V1.0.* CSCI: 30038-2999 v 1.0. DI: 30038-2999, Rev. 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19981103.0084.

142321 CRWMS M&O 1999. *Characterize Framework for Seismicity and Structural Deformation at Yucca Mountain, Nevada.* ANL-CRW-GS-000003 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000510.0175.

150744 CRWMS M&O 1999. *ASHPLUME Version 1.4LV User's Manual.* 10022-UM-1.4LV-00, Rev. 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000622.0082.

111880 CRWMS M&O 2000. *Summary of In-Package Chemistry for Waste Forms.* ANL-EBS-MD-000050 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000217.0217.

122797 CRWMS M&O 2000. *UZ Flow Models and Submodels.* MDL-NBS-HS-000006 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990721.0527.

122799 CRWMS M&O 2000. *UZ Colloid Transport Model.* ANL-NBS-HS-000028 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000822.0005.

122894 CRWMS M&O 2000. *Seepage Model for PA Including Drift Collapse.* MDL-NBS-HS-000002 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990721.0525.

123913 CRWMS M&O 2000. *Abstraction of Flow Fields for RIP (ID:U0125).* ANL-NBS-HS-000023 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000127.0089.

125156 CRWMS M&O 2000. *Waste Form Colloid-Associated Concentrations Limits: Abstraction and Summary.* ANL-WIS-MD-000012 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000525.0397.

127818 CRWMS M&O 2000. *In-Drift Precipitates/Salts Analysis.* ANL-EBS-MD-000045 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000512.0062.

129284 CRWMS M&O 2000. *EBS Radionuclide Transport Abstraction.* ANL-WIS-PA-000001 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000526.0329.

129287 CRWMS M&O 2000. *In-Package Chemistry Abstraction.* ANL-EBS-MD-000037 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000418.0818.

134732 CRWMS M&O 2000. *Analysis of Base-Case Particle Tracking Results of the Base-Case Flow Fields (ID: U0160).* ANL-NBS-HS-000024 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000207.0690.

135773 CRWMS M&O 2000. *Abstraction of Models of Stress Corrosion Cracking of Drip Shield and Waste Package Outer Barrier and Hydrogen Induced Corrosion of Drip Shield.* ANL-EBS-PA-000004 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000526.0326.

136045 CRWMS M&O 2000. *Initial Cladding Condition.* ANL-EBS-MD-000048 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000523.0150.

136060 CRWMS M&O 2000. *CSNF Waste Form Degradation: Summary Abstraction.* ANL-EBS-MD-000015 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000121.0161.

136281 CRWMS M&O 2000. *Evaluate Soil/Radionuclide Removal by Erosion and Leaching.* ANL-NBS-MD-000009 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000310.0057.

136383 CRWMS M&O 2000. *Inventory Abstraction.* ANL-WIS-MD-000006 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000414.0643.

139440 CRWMS M&O 2000. *Input and Results of the Base Case Saturated Zone Flow and Transport Model for TSPA.* ANL-NBS-HS-000030 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000526.0330.

139563 CRWMS M&O 2000. *Igneous Consequence Modeling for TSPA-SR.* ANL-WIS-MD-000017 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000501.0225.

139610 CRWMS M&O 2000. *Multiscale Thermohydrologic Model.* ANL-EBS-MD-000049 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000609.0267.

141044 CRWMS M&O 2000. *Characterize Framework for Igneous Activity at Yucca Mountain, Nevada (T0015).* ANL-MGR-GS-000001 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000720.0541.

141418 CRWMS M&O 2000. *Particle Tracking Model and Abstraction of Transport Processes.* ANL-NBS-HS-000026 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000502.0237.

142004 CRWMS M&O 2000. *Abstraction of Drift Seepage.* ANL-NBS-MD-000005 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000322.0671.

142022 CRWMS M&O 2000. *Drift-Scale Coupled Processes (DST and THC Seepage) Models.* MDL-NBS-HS-000001 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990721.0523.

143244 CRWMS M&O 2000. *Analysis of Infiltration Uncertainty.* ANL-NBS-HS-000027 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000525.0377.

143378 CRWMS M&O 2000. *Disruptive Event Biosphere Dose Conversion Factor Analysis.* ANL-MGR-MD-000003 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000303.0216.

143420 CRWMS M&O 2000. *Defense High Level Waste Glass Degradation.* ANL-EBS-MD-000016 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000329.1183.

143569 CRWMS M&O 2000. *Summary of Dissolved Concentration Limits.* ANL-WIS-MD-000010 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000525.0372.

143665 CRWMS M&O 2000. *Total System Performance Assessment for the Site Recommendation.* TDR-WIS-PA-000001 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20001005.0282.

144054 CRWMS M&O 2000. *Abstraction of BDCF Distributions for Irrigation Periods.* ANL-NBS-MD-000007 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000517.0257.

144055 CRWMS M&O 2000. *Distribution Fitting to the Stochastic BDCF Data.* ANL-NBS-MD-000008 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000517.0258; MOL.20000601.0753.

144056 CRWMS M&O 2000. *Groundwater Usage by the Proposed Farming Community.* ANL-NBS-MD-000006 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000407.0785.

144128 CRWMS M&O 2000. *Design Analysis for UCF Waste Packages.* ANL-UDC-MD-000001 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000526.0336.

144164 CRWMS M&O 2000. *DSNF and Other Waste Form Degradation Abstraction.* ANL-WIS-MD-000004 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000223.0502.

144167 CRWMS M&O 2000. *In-Package Source Term Abstraction.* ANL-WIS-MD-000018 REV 00. Las Vegas, Nevada: CRWMS M&O. Submit to RPC URN-0278.

144229 CRWMS M&O 2000. *General Corrosion and Localized Corrosion of Waste Package Outer Barrier.* ANL-EBS-MD-000003 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000202.0172.

144551 CRWMS M&O 2000. *Calculation of Probability and Size of Defect Flaws in Waste Package Closure Welds to Support WAPDEG Analysis.* CAL-EBS-PA-000003 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000424.0676.

144692 CRWMS M&O 2000. *Non-Disruptive Event Biosphere Dose Conversion Factor Sensitivity Analysis.* ANL-MGR-MD-000010 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000420.0074.

144700 CRWMS M&O 2000. *Biosphere Dose Conversion Factors for Reasonably Maximally Exposed Individual and Average Member of Critical Group.* CAL-MGR-MD-000002 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000306.0251.

145774 CRWMS M&O 2000. *Unsaturated Zone Flow and Transport Model Process Model Report.* TDR-NBS-HS-000002 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000320.0400.

146460 CRWMS M&O 2000. *Environment on the Surfaces of the Drip Shield and Waste Package Outer Barrier.* ANL-EBS-MD-000001 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000328.0590.

147210 CRWMS M&O 2000. *Clad Degradation - Summary and Abstraction.* ANL-WIS-MD-000007 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000602.0055.

147505 CRWMS M&O 2000. *Colloid-Associated Radionuclide Concentration Limits: ANL.* ANL-EBS-MD-000020 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000329.1187.

147639 CRWMS M&O 2000. *Aging and Phase Stability of Waste Package Outer Barrier.* ANL-EBS-MD-000002 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000410.0407.

147641 CRWMS M&O 2000. *Calculation of General Corrosion Rate of Drip Shield and Waste Package Outer Barrier to Support WAPDEG Analysis.* CAL-EBS-PA-000002 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000319.0047.

147648 CRWMS M&O 2000. *Abstraction of Models for Pitting and Crevice Corrosion of Drip Shield and Waste Package Outer Barrier.* ANL-EBS-PA-000003 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000526.0327.

147972 CRWMS M&O2000. *Uncertainty Distribution for Stochastic Parameters.* ANL-NBS-MD-000011 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000526.0328.

148713 CRWMS M&O 2000. *Repository Safety Strategy: Plan to Prepare the Safety Case to Support Yucca Mountain Site Recommendation and Licensing Considerations.* TDR-WIS-RL-000001 REV 04. Three volumes. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20001003.0112.

149860 CRWMS M&O 2000. *Abstraction of NFE Drift Thermodynamic Environment and Percolation Flux.* ANL-EBS-HS-000003 REV 00 ICN 01. Las Vegas, Nevada: CRWMS M&O. Submit to RPC URN-0644.

149862 CRWMS M&O 2000. *Multiscale Thermohydrologic Model.* ANL-EBS-MD-000049 REV 00 ICN 01. Las Vegas, Nevada: CRWMS M&O. Submit to RPC URN-0574.

150559 CRWMS M&O 2000. *Revised Grams Per Package for Use in TSPA-SR.* Input Transmittal 00306.T. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000628.0043.

150806 CRWMS M&O 2000. *The Development of Information Catalogued in REV00 of the YMP FEP Database.* TDR-WIS-MD-000003 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000705.0098.

150824 CRWMS M&O 2000. *Unsaturated Zone Flow and Transport Model Process Model Report.* TDR-NBS-HS-000002 REV 00 ICN 1. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000629.0915.

151566 CRWMS M&O 2000. *WAPDEG Analysis of Waste Package and Drip Shield Degradation.* ANL-EBS-PA-000001 REV 00 ICN 01. Las Vegas, Nevada: CRWMS M&O. Submit to RPC URN-0628.

151708 CRWMS M&O 2000. *Precipitates/Salts Model Results for THC Abstraction.* CAL-EBS-PA-000008 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000801.0001.

151853 CRWMS M&O 2000. *Monitored Geologic Repository Project Description Document.* TDR-MGR-SE-000004 REV 02. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20001031.0062.

152496 CRWMS M&O 2000. *GVP Software Routine Report.* SDN: 10341-SRR-1.02-00. Las Vegas, Nevada: CRWMS M&O.

152497 CRWMS M&O 2000. *MFD Software Routine Report.* SDN: 10342-SRR-1.01-00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000908.0033.

152499 CRWMS M&O 2000. *SCCD Software Routine Report.* SDN: 10343-SRR-2.01-00. Las Vegas, Nevada: CRWMS M&O.

152773 CRWMS M&O 2000. *Unsaturated Zone and Saturated Zone Transport Properties* (U0100). ANL-NBS-HS-000019 REV 00. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000829.0006.

153016 CRWMS M&O 2000. *SZ_CONVOLUTE V2.0 User's Manual.* 10207-UM-2.0-00, LV-2000-041. Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000908.0050.

153034 CRWMS M&O 2000. *Biosphere Dose Conversion Factors for Radionuclides Identified as being Important Contributors to Dose afterTen Thousand Years*. CAL-NBS-PA-000001 REV 00. Las Vegas, Nevada: CRWMS M&O. Submit to RPC URN-0647.

153289 CRWMS M&O 2000. *Technical Work Plan For Total System Performance Assessment.* TWP-MGR-PA-00001 REV 00. Las Vegas, Nevada: CRWMS M&O. URN-0722.

100550 DOE (U.S. Department of Energy) 1998. *Total System Performance Assessment.* Volume 3 of *Viability Assessment of a Repository at Yucca Mountain.* DOE/RW-0508. Washington, D.C.: U.S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.19981007.0030.

107790 DOE (U.S. Department of Energy) 1999. *DOE Spent Nuclear Fuel Information in Support of TSPA-SR.* DOE/SNF/REP-0047, Rev. 0. [Washington, D.C.]: U.S. Department of Energy, Office of Environmental Management. TIC: 245482.

149540 DOE (U.S. Department of Energy) 2000. *Quality Assurance Requirements and Description.* DOE/RW-0333P, Rev. 10. Washington, D.C.: U.S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.20000427.0422.

105655 Dyer, J.R. 1999. "Revised Interim Guidance Pending Issuance of New U.S. Nuclear Regulatory Commission (NRC) Regulations (Revision 01, July 22, 1999), for Yucca Mountain, Nevada." Letter from J.R. Dyer (DOE/YMSCO) to D.R. Wilkins (CRWMS M&O), September 3, 1999, OL&RC:SB-1714, with enclosure, "Interim Guidance Pending Issuance of New NRC Regulations for Yucca Mountain (Revision 01)." ACC: MOL.19990910.0079.

153112 Edward E. Johnson, Inc. 1966. *Ground Water and Wells: A Reference Book for the Water-Well Industry.* 1st Edition. Saint Paul, Minnesota: Edward E. Johnson. TIC: 208449.

100393 Finsterle, S.; Pruess, K.; and Fraser, P. 1996. *ITOUGH2 Software Qualification.* LBNL-39489. Berkeley, California: Lawrence Berkeley National Laboratory. ACC: MOL.19970619.0040.

143556 Golder Associates 2000. *User's Guide GoldSim Graphical Simulation Environment.* Version 6.02. Draft #3. Redmond, Washington: Golder Associates. TIC: 247324.

151202 Golder Associates 2000. *Software Code: GoldSim.* 6.04.007. 10344-6.04.007.

100987 Jarzemba, M.S.; LaPlante, P.A.; and Poor, K.J. 1997. *ASHPLUME Version 1.0 - A Code for Contaminated Ash Dispersal and Deposition, Technical Description and User's Guide.* CNWRA 97-004, Rev. 1. San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. TIC: 239303.

101701 Knauss, K.G.; Bourcier, W.L.; McKeegan, K.D.; Merzbacher, C.I.; Nguyen, S.N.; Ryerson, F.J.; Smith, D.K.; Weed, H.C.; and Newton, L. 1990. "Dissolution Kinetics of a Simple Analogue Nuclear Waste Glass as a Function of pH, Time and Temperature." *Scientific Basis for Nuclear Waste Management XIII, Symposium held November 27-30, 1989, Boston, Massachusetts*. Oversby, V.M. and Brown, P.W., eds. *176,* 371-381. Pittsburgh, Pennsylvania: Materials Research Society. TIC: 203658.

101079 LaPlante; P.A. and Poor, K. 1997. *Information and Analyses to Support Selection of Critical Groups and Reference Biospheres for Yucca Mountain Exposure Scenarios.* CNWRA 97-009. San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. TIC: 236454.

100464 Leigh, C.D.; Thompson, B.M.; Campbell, J.E.; Longsine, D.E.; Kennedy, R.A.; and Napier, B.A. 1993. *User's Guide for GENII-S: A Code for Statistical and Deterministic Simulations of Radiation Doses to Humans from Radionuclides in the Environment.* SAND91-0561. Albuquerque, New Mexico: Sandia National Laboratories. TIC: 231133.

131202 Lide, D.R., ed. 1991. *CRC Handbook of Chemistry and Physics.* 72nd Edition. Boca Raton, Florida: CRC Press. TIC: 3595.

105729 Liu, H.H.; Doughty, C.; and Bodvarsson, G.S. 1998. "An Active Fracture Model for Unsaturated Flow and Transport in Fractured Rocks." *Water Resources Research, 34,* (10), 2633-2646. Washington, D.C.: American Geophysical Union. TIC: 243012.

101464 Neuman, S.P. 1990. "Universal Scaling of Hydraulic Conductivities and Dispersivities in Geologic Media." *Water Resources Research, 26,* (8), 1749-1758. Washington, D.C.: American Geophysical Union. TIC: 237977.

100474 Nitao, J.J. 1998. *Reference Manual for the NUFT Flow and Transport Code, Version 2.0.* UCRL-MA-130651. Livermore, California: Lawrence Livermore National Laboratory. TIC: 238072.

100413 Pruess, K. 1991. *TOUGH2-A General-Purpose Numerical Simulator for Multiphase Fluid and Heat Flow.* LBL-29400. Berkeley, California: Lawrence Berkeley Laboratory. ACC: NNA.19940202.0088.

153328 Steward, S.A. 1999. Estimate of the Porosity of the Alteration Phases that Grow on CSNF During Long-Term Contact with Dripping Groundwater. Letter from S.A. Steward (LLNL) to R. Rechard (SNL), November 29, 1999, LLYMP9911113. ACC: MOL.20000425.0925.

142632 Stroupe, E.P. 2000. "Approach to Implementing the Site Recommendation Design Baseline." Interoffice correspondence from E.P. Stroupe (CRWMS M&O) to D.R. Wilkins, January 26, 2000, LV.RSO.EPS.1/00-004, with attachment. ACC: MOL.20000214.0480.

136368 USGS (U.S. Geological Survey) 2000. *Future Climate Analysis.* ANL-NBS-GS-000008 REV 00. Denver, Colorado: U.S. Geological Survey. ACC: MOL.20000629.0907.

102865 Weast, R.C., ed. 1979. *CRC Handbook of Chemistry and Physics.* 60th Edition, 1979 - 1980. Boca Raton, Florida: CRC Press. TIC: 245312.

100191 Wilson, M.L.; Gauthier, J.H.; Barnard, R.W.; Barr, G.E.; Dockery, H.A.; Dunn, E.; Eaton, R.R.; Guerin, D.C.; Lu, N.; Martinez, M.J.; Nilson, R.; Rautman, C.A.; Robey, T.H.; Ross, B.; Ryder, E.E.; Schenker, A.R.; Shannon, S.A.; Skinner, L.H.; Halsey, W.G.; Gansemer, J.D.; Lewis, L.C.; Lamont, A.D.; Triay, I.R.; Meijer, A.; and Morris, D.E. 1994. *Total-System Performance Assessment for Yucca Mountain – SNL Second Iteration (TSPA-1993).* SAND93-2675. Executive Summary and two volumes. Albuquerque, New Mexico: Sandia National Laboratories. ACC: NNA.19940112.0123.

100836 Wolery, T.J. 1992. *EQ3NR, A Computer Program for Geochemical Aqueous Speciation-Solubility Calculations. Theoretical Manual, User's Guide, and Related Documentation (Version 7.0).* UCRL-MA-110662 PT III. Livermore, California: Lawrence Livermore National Laboratory. TIC: 205154.

100097 Wolery, T.J. and Daveler, S.A. 1992. *EQ6, A Computer Program for Reaction Path Modeling of Aqueous Geochemical Systems: Theoretical Manual, User's Guide, and Related Documentation (Version 7.0).* UCRL-MA-110662 PT IV. Livermore, California: Lawrence Livermore National Laboratory. TIC: 205002.

100528 Zyvoloski, G.A.; Robinson, B.A.; Dash, Z.A.; and Trease, L.L. 1995. *Models and Methods Summary for the FEHM Application.* LA-UR-94-3787, Rev. 1. Los Alamos, New Mexico: Los Alamos National Laboratory. TIC: 222337.

100615 Zyvoloski, G.A.; Robinson, B.A.; Dash, Z.V.; and Trease, L.L. 1997. *User's Manual for the FEHM Application–A Finite-Element Heat- and Mass-Transfer Code.* LA-13306-M. Los Alamos, New Mexico: Los Alamos National Laboratory. TIC: 235999.

8.2 CODES, STANDARDS, AND REGULATIONS

105065 64 FR 46976. Environmental Radiation Protection Standards for Yucca Mountain, Nevada. Readily available.

101680 64 FR 8640. Disposal of High-Level Radioactive Wastes in a Proposed Geologic Repository at Yucca Mountain, Nevada. Readily available.

151627 AP-2.14Q, Rev. 0, ICN 1. Review of Technical Products. [Washington, D.C.]: U.S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.20000405.0477.

152174 AP-2.21Q, Rev. 0, ICN 0. *Quality Determinations and Planning for Scientific, Engineering, and Regulatory Compliance Activities.* [Washington, D.C.]: U.S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.20000802.0003.

152363 AP-3.10Q, Rev. 2, ICN 3. *Analyses and Models.* Washington, D.C.: U. S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.20000918.0282.

153184 AP-3.15Q, Rev. 2, ICN 0. *Managing Technical Product Inputs.* Washington, D.C: U.S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.20001109.0051.

153201 AP-SI.1Q, Rev. 2, ICN 4, ECN 1. *Software Management.* Washington, D.C.: U.S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.20000223.0508.

153202 AP-SV.1Q, Rev. 0, ICN 2. *Control of the Electronic Management of Information.* Washington, D.C.: U.S. Department of Energy, Office of Civilian Radioactive Waste Management. ACC: MOL.20000831.0065.

143799 EPA (U.S. Environmental Protection Agency) 1999. *Background Information Document for* 40 CFR 197, *Environmental Radiation Protection Standards for Yucca Mountain, Nevada.* Washington, D.C.: U.S. Environmental Protection Agency, Office of Radiation and Indoor Air. TIC: 246926.

146971 LANL (Los Alamos National Laboratory) 1999. *Software Code: FEHM V2.00.* V2.00. SUN Ultra Sparc. 10031-2.00-00.

150986 NLP-2-0, Rev. 5, ICN 1. *Determination of Importance Evaluations.* Las Vegas, Nevada: CRWMS M&O. ACC: MOL.20000713.0360.

107770 NRC (U.S. Nuclear Regulatory Commission) 1998. "Proposed Rule 10 CFR Part 63, Disposal of High-Level Radioactive Wastes in a Proposed Geologic Repository at Yucca Mountain, Nevada. SECY-98-225." Washington, D.C.: U.S. Nuclear Regulatory Commission. Accessed 10/20/98. TIC: 240520. Http://www.nrc.gov/NRC/COMMISSION/SECYS/1998-225scy.html

149372 NRC (U.S. Nuclear Regulatory Commission) 2000. *Issue Resolution Status Report Key Technical Issue: Total System Performance Assessment and Integration.* Rev. 2. Washington, D.C.: U.S. Nuclear Regulatory Commission. TIC: 247614.

100477 Nuclear Energy Agency 1991. *Disposal of Radioactive Waste: Can Long-Term Safety Be Evaluated?.* Paris, France: Nuclear Energy Agency, Organization for Economic Co-operation and Development. TIC: 226870.

100478 OECD (Organisation for Economic Co-operation and Development) 1991. *Disposal of Radioactive Waste: Review of Safety Assessment Methods.* Paris, France: Organisation for Economic Co-operation and Development, Nuclear Energy Agency. TIC: 226871.

104262 QAP-2-3, Rev. 10. *Classification of Permanent Items.* Las Vegas, Nevada: CRWMS M&O. ACC: MOL.19990316.0006.

8.3 DATA, LISTED BY DATA TRACKING NUMBER

8.3.1 Source Data

147613 GS000308311221.005. Preliminary Net Infiltration Modeling Results for 3 Climate Scenarios for FY99. Submittal date: 03/01/2000.

151139 GS000308315121.003. Meteorological Stations Selected to Represent Future Climate States at Yucca Mountain, Nevada. Submittal date: 03/14/2000.

148751 LA0003AM831341.001. Preliminary Revision of Probability Distributions for Sorption Coefficients (K_Ds). Submittal date: 3/29/2000.

149557 LA0003JC831362.001. Preliminary Matrix Diffusion Coefficients for Yucca Mountain Tuffs. Submittal date: 4/10/2000.

147285 LA0003MCG12213.002. Cumulative Probabilities for Colloid Transport Between One Matrix and Another Calculated from Interpolation of Pore Volume Data from Yucca Mountain Hydrologic (Stratigraphic) Samples. Submittal date: 03/10/2000.

153251 LA0007MCG12213.001. Colloid Size Distribution. Submittal date: 07/31/2000.

106787 LB990501233129.001. Fracture Properties for the UZ Model Grids and Uncalibrated Fracture and Matrix Properties for the UZ Model Layers for AMR U0090, "Analysis of Hydrologic Properties Data". Submittal date: 08/25/1999.

106785 LB990701233129.001. 3-D UZ Model Grids for Calculation of Flow Fields for PA for AMR U0000, "Development of Numerical Grids for UZ Flow and Transport Modeling". Submittal date: 09/24/1999.

118710 LB990801233129.007. TSPA Grid Flow Simulations for AMR U0050, "UZ Flow Models and Submodels" (Flow Field #7). Submittal date: 11/29/1999.

118717 LB990801233129.009. TSPA Grid Flow Simulations for AMR U0050, "UZ Flow Models and Submodels" (Flow Field #9). Submittal date: 11/29/1999.

118722 LB990801233129.011. TSPA Grid Flow Simulations for AMR U0050, "UZ Flow Models and Submodels" (Flow Field #11). Submittal date: 11/29/1999.

126111 LB991091233129.004. Calibrated Fault Properties for the UZ Flow and Transport Model for AMR U0035, "Calibrated Properties Model". Submittal date: 10/22/1999.

104055 LB997141233129.001. Calibrated Basecase Infiltration 1-D Parameter Set for the UZ Flow and Transport Model, FY99. Submittal date: 07/21/1999.

119933 LB997141233129.002. Calibrated Upper-Bound Infiltration 1-D Parameter Set for the UZ Flow and Transport Model, FY99. Submittal date: 07/21/1999.

119940 LB997141233129.003. Calibrated Lower-Bound Infiltration 1-D Parameter Set for the UZ Flow and Transport Model, FY99. Submittal date: 07/21/1999.

142973 LL000122051021.116. Summary of Analyses of Glass Dissolution Filtrates. Submittal date: 01/27/2000.

145940 LL000207751021.119. CSNF Alteration Phase Porosity Estimates. Submittal Date: 2/18/2000.

145943 LL000210651021.121. Analysis of Knauss Data. Submittal date: 02/29/00.

142902 LL991109851021.095. Colloid Size and Concentration Investigations in Scientific Notebook SN 1381. Submittal date: 01/10/2000.

150040 MO0002SPACRI02.002. Critical Group BDCF (1 Million Year Radionuclides). Submittal date: 02/17/2000. Submit to RPC URN-0591.

149168 MO0002SPALOO46.010. Lookup Tables for PH, CL, and Ionic Strength Predicted by Precipitates/Salts Model for THC Abstraction. Submittal date: 02/07/2000.

148338 MO0002SPASDC00.002. Self-Diffusion Coefficient of Water. Submittal date: 02/24/2000.

146857 MO0003MWDTAB45.013. EQ3/6 Input/Output Files for In-Drift Precipitates/Salts Lookup Tables. Submittal date: 03/06/2000.

148872 MO0003SPAABS07.006. Abstracted BDCF Distributions with Soil Erosion for Use in TSPA-SR. Submittal date: 03/23/2000. Submit to RPC URN-0560.

148453 MO0003SPAABS08.004. Abstracted BDCF Distributions for Use in TSPA-SR. Submittal date: 03/21/2000. Submit to RPC URN-0561.

147949 MO0003SPAHIG12.002. Highest and Lowest Observed or Expected Masses of Iron-(hydr)Oxide Colloids Per Unit Volume or Mass of Water. Submittal date: 03/02/2000.

147952 MO0003SPAHLO12.004. Highest And Lowest Observed or Expected Groundwater Colloid Masses Per Unit Volume or Mass of Water; Values of Ionic Strength Above Which Groundwater Colloid Dispersions Are Unstable and Below Which Groundwater Colloid Dispersions Are Stable (Within Defined pH Range). Submittal date: 03/16/2000.

147951 MO0003SPAION02.003. Values Of Ionic Strength That Define The Stability Limits Of Iron-(Hydr)Oxide Colloids. Submittal date: 03/03/2000.

147953 MO0003SPALOW12.001. Lowest Observed or Expected Concentration of Radionuclide Element Rn Associated with Waste-Form Colloids. Submittal date: 03/02/2000.

151075 MO0003SPASGU01.003. Stochastic Groundwater Usage in Amargosa Valley for TSPA-SR. Submittal date: 03/21/2000. Submit to RPC URN-0562.

151368 MO0004SPACLD07.043. Clad Degradation - Summary and Abstraction. Submittal date: 04/04/2000. Submit to RPC URN-0563.

151063 MO0004SPADEC00.002. Decay Rate. Submittal date: 04/18/2000. Submit to RPC URN-724

151062 MO0004SPAFRE00.003. Free Fraction. Submittal date: 04/19/2000.

148810 MO0004SPAKDS42.005. Kds for Pu and Am on Waste Form, Iron (hydr)Oxide, and Groundwater Colloids. Submittal date: 04/10/2000.

151713 MO0004SPASOL10.002. Radionuclide Solubility Limits. Submittal date: 04/24/2000.

152837 MO0006SPABDC01.007. BDCFS for Radionuclides That May Be Important to TSPA Dose Calculations After 10,000 Years. Submittal date: 06/26/2000. Submit to RPC URN-0725

153029 MO0006SPASTR01.003. Area of Stress Corrosion Crack for the EBS Transport Abstraction. Submittal date: 06/09/2000. Submit to RPC URN-0726

151712 MO0007RIB00091.000. Defense High Level Waste Glass Degradation. Submittal date: 07/26/2000.

152186 MO0008MWDHUMAN.000. Human Intrusion Cases TSPA SR, REV 00B Human Intrusion Scenario, No Backfill. Submittal date: 08/31/2000. Submit to RPC URN-0576.

151714 MO0008MWDNM501.005. TSPA SR, REV 00B, Case SR00_047NM5.---Base Case; Nominal Scenario; No Backfill; 100 Realizations; 100,000 Years. Submittal date: 08/15/2000. Submit to RPC URN-0569.

151547 MO0008MWDVEB03.003. Volcanic Eruption Biosphere Dose Conversion Factors. Submittal date: 08/02/2000.

152838 MO0009MWDMED01.020. Median Value Realization AMR Base Cases for the TSPA_SR Model. TSPA_SR, Rev. 00B, Cases SR00_037NE6, SR00_038NE6 AND SR00_001IE5. Base Case; No Backfill; Median Value Realization. Submittal date: 09/26/2000. Submit to RPC URN-0727

152839 MO0009MWDNM601.018. MILLION-YEAR SENSITIVITY CASES FOR THE NOMINAL SCENARIO. TSPA_SR, REV. 00B, CASE SR00_023NM6 AND CASE SR00_024NM6. Submittal date: 09/20/2000.

153127 MO0010MWDWAP01.009. WAPDEG models for tspa-sr. ---*.gsm files are goldsim 6.04.007/ WAPDEG 4.0 inputs and outputs---*.jnb files are sigmaplot 4.0 graphs---files in the runfiles directory are WAPDEG 4.0 input files and dlls---files in the prewap_for_no_backfill directory are for the prewap routine. Submittal date: 10/24/2000. URN-0723

144565 MO9910SPAFWPWF.001. Weld Flaws of Waste Packages. Submittal date: 10/22/1999.

139569 MO9911SPACDP37.001. In-Package Chemistry Abstraction for Co-Disposal Packages. Submittal date: 11/24/1999.

148596 MO9912SPAPAI29.002. PA Initial Abstraction of THC Model Chemical Boundary Conditions. Submittal date: 01/11/2000.

147198 SN0001T0872799.006. In-Drift Thermodynamic Environment and Percolation Flux. Submittal date: 01/27/2000.

149556 SN0003T0503100.001. Weighting Factors for Low, Middle and High Climate Infiltration Rate Maps. Submittal date: 03/20/2000.

151021 SN0003T0810599.010. Revised Average Radionuclide Activities for Commercial Spent Nuclear Fuel (CSNF) and Co-Disposal Waste Packages for Total System Performance Assessment-Site Recommendation (TSPA-SR) and Final Environmental Impact Statement (TSPA-FEIS). Submittal date: 03/15/2000.

149288 SN0004T0501600.004. Updated Results of the Base Case Saturated Zone (SZ) Flow and Transport Model. Submittal date: 04/10/2000.

151515 SN0004T0501600.005. Updated Input Files to the Base Case Saturated Zone (SZ) Flow and Transport Model for TSPA Abstractions. Submittal date: 04/10/2000.

149254 SN0004T0571599.004. Uncertainty Distributions for Stochastic Parameters Revision to Include New U Sorption Coefficients in the Alluvium and Supporting Electronic Files. Submittal date: 04/10/2000.

151514 SN0005T0581699.005. Geometric Means and Standard Deviations for Fracture Aperture Distributions for Unsaturated Zone (UZ) Transport in TSPA-SR. Submittal date: 05/24/2000.

152110 SN0005T0810599.012. Updated Waste Package Radionuclide Inventory Approximations for Total System Performance Assessment-Site Recommendation (TSPA-SR). Submittal date: 05/25/2000.

150856 SN0006T0502900.002. Updated Igneous Consequence Data for Total System Performance Assessment-Site Recommendation (TSPA-SR). Submittal date: 06/15/2000.

152545 SN0007T0872799.014. Abstraction of Thermal Hydrologic (TH) Data for TSPA-SR for the No Backfill Repository Design. Submittal date: 07/05/2000.

111485 SN9907T0872799.001. Heat Decay Data and Repository Footprint for Thermal-Hydrologic and Conduction-Only Models for TSPA-SR (Total System Performance Assessment-Site Recommendation). Submittal date: 07/27/1999.

108437 SN9908T0872799.004. Tabulated In-Drift Geometric and Thermal Properties Used in Drift-Scale Models for TSPA-SR (Total System Performance Assessment-Site Recommendation). Submittal date: 08/30/1999.

126110 SN9910T0581699.002. Post-Processed Flow Fields for RIP: Developed Data from AMR U0125 (Abstract Flow Fields for RIP). Submittal date: 10/15/1999.

146902 SN9912T0511599.002. Revised Seepage Abstraction Results for TSPA-SR (Total System Performance Assessment-Site Recommendation). Submittal date: 12/15/1999.

136370 SN9912T0512299.002. Annual Surface Soil Removal Estimates for Amargosa Valley Soils. Submittal date: 12/09/1999.

146903 SN9912T0581699.003. Files to Support Base-Case Particle-Tracking Analyses (AMR U0160) for TSPA-SR. Submittal date: 12/13/1999.

8.3.2 DTNs That Superceded Source Data

153267 MO0010SPAABS08.007. ABSTRACTED BDCF DISTRIBUTIONS FOR USE IN TSPA-SR. Submittal date: 10/27/2000. URN-728

153268 MO0011SPAABS07.009. ABSTRACTED BDCF DISTRIBUTIONS WITH SOIL EROSION FOR USE IN TSPA-SR. Submittal date: 11/15/2000. URN-729

152980 SN0009T0810599.014. Updated Average Radionuclide Activities for CSNF and Codisposal Waste Packages for TSPA-SR and TSPA-FEIS. Submittal date: 09/20/2000. URN-0643

152993 SN0011T0810599.023. Final Waste Package Radionuclide Inventory Approximations for Total System Performance Assessment-Site Recommendation (TSPA-SR). Submittal date: 11/01/2000. Submit to RPC URN-0642

INTENTIONALLY LEFT BLANK

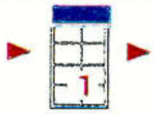
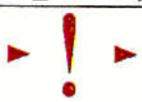
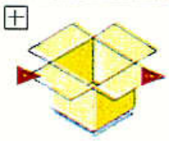
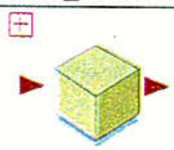
ATTACHMENT I
GOLDSIM GRAPHICAL ELEMENTS

ATTACHMENT I

GOLDSIM GRAPHICAL ELEMENTS

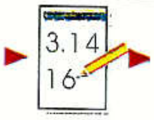
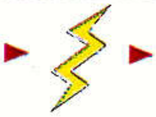
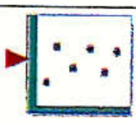
The table that follows shows all the GoldSim elements that are used in the TSPA-SR model, along with a short description of each element. For more in-depth information on any of these GoldSim elements, refer to the GoldSim user's manual (Golder Associates 2000 [143556]).

Table I.1. GoldSim Elements

Element Appearance	Title and Definition
 One_Dimensional_Table	1-D Table: A one-dimensional look-up table.
 Two_Dimensional_Table	2-D Table: A two-dimensional look-up table.
 Accumulator	Accumulator: Integrates values over time.
 Cell_Pathway	Cell Pathway: Equivalent to a mixing cell. Cell pathways can represent partitioning, solubility, and mass transport. Networks of cells behave mathematically as a coupled system of differential equations.
 Consequence_Generator	Consequence Generator: This element generates a specific consequence when triggered by an event.
 Global_Container	Container (Global): A way to organize model elements. The hierarchy is analogous to a folder scheme in the Windows operating system.
 Localized_Container	Container (Localized): Localization is a special feature of the container. In this form, nothing outside the container can "see" what is inside the container. Elements within the container look for necessary information inside the container first, then look outside the container.

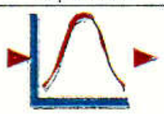
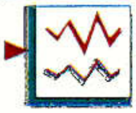
C61

Table I.1. GoldSim Elements (Continued)

 Data	Data: This element is used to define scalar, vector, or matrix data for use by the model.
 Event_Generator	Event Generator: Generates a discreet event in a manner defined by the user. An example of a generated event is the formation of a volcanic cone.
 Expression	Expression: The expression element is used to define mathematical equations.
 External	External: The external element links external functions to the GoldSim generated model via a DLL.
 External_Pathway	External Pathway: This element is a specialized form of the external element. It links external transport pathways via a DLL at run time, and the linkage is defined by input and output mass flux links.
 File	File: This element dynamically copies a file to a specified directory for use by an external function.
 Fluid	Fluid: A liquid that may dissolve in a reference fluid or may partition species between itself and a reference fluid.
 MasterClock	MasterClock: A special element used to define the simulation settings before running a model. Timesteps, total time simulated, and a variety of Monte Carlo options are set from the MasterClock element.
 Multi_Variate_Result	Multi-Variate Result: A result element in which two or more results are graphically displayed or compared.

C62

Table I.1. GoldSim Elements (Continued)

 Pipe_Pathway	Pipe Pathway: Act as a conduit for a single fluid and model advection and dispersion. A Laplace transform approach provides analytical solutions.
 Receptor	Receptor: A specialized expression used to make the vector calculations resulting in contaminant impact. Receptors can produce Risk, Dose, or Hazard Index results with user defined impact equations.
 Water	Reference Fluid: All partitioning coefficients must be defined in relation to a reference fluid. Each model must contain at least one reference fluid, known as the "Master Reference Fluid." The Master Reference Fluid is named "Water" by default, but this name may be changed.
 Selector	Selector: A specialized form of expression used to create complex nested if-then statements.
 Solid	Solid: Used to represent media such as soil or rock. Solids are defined by bulk density, porosity, and tortuosity. In addition, species can be partitioned between a solid and a reference fluid.
 Source	Source: A way to introduce mass into the model when a simple equation is insufficient. Source outputs are essentially release rates for species into one or several transport pathways.
 Species	Species: This element, which cannot be deleted or renamed, is where all species that undergo transport are defined. For the TSPA-SR model, radionuclides and colloid-associated radionuclides make up the species list.
 Stochastic	Stochastic: A probability distribution. Many different types of distributions are available, including uniform, discrete, cumulative, Poisson, and Weibull distributions.
 Time_History_Result	Time History Results: Shows a particular output history as a function of time in a stand-alone element. This allows the user to collect and group time history results in a container, simplifying the presentation of results.

C63

INTENTIONALLY LEFT BLANK

ATTACHMENT II

SEEPDLL AVERAGE SEEPAGE FLUX AND SEEPAGE FRACTION

ATTACHMENT II

SEEPDLL AVERAGE SEEPAGE FLUX AND SEEPAGE FRACTION

II.1 SOFTWARE ROUTINE IDENTIFICATION

II.1.1 SOFTWARE NAME AND VERSION NUMBER

seepdll Ver 1.0

II.1.2 NAME AND VERSION OF INDUSTRY STANDARD SOFTWARE UNDER WHICH ROUTINE WAS DEVELOPED

This routine was developed using MS Visual Fortran Professional Edition 6.0.A.

II.1.3 SRR DOCUMENT IDENTIFICATION NUMBER:

N/A

II.1.4 SRR MEDIA NUMBER (IF APPLICABLE):

N/A

II.2 DESCRIPTION AND TESTING

II.2.1 OVERVIEW

The **seepdll** routine calculates seepage fractions (the fraction of waste packages experiencing seepage) and seepage flows (the amount of seepage onto a given waste package). The routine is called (in the form of direct linked library) by the TSPA model. The seepage fractions and seepage flows calculated by the routine are passed back to the TSPA model, which uses this information in its calculation of waste package and waste form degradation.

The **seepdll** routine uses data developed in Abstraction of Drift Seepage (CRWMS M&O 2000 [142004]) as inputs. These data were developed from a model that postulates seepage is a function of percolation flux, . The percolation flux is a function of water infiltration into the unsaturated zone, which is in turn a function of climate during the life of the repository. A detailed description of the model including assumptions and the influence of perturbing physical processes can be found in CRWMS M&O 2000 [142004].

Because repository life is assumed to be one million years and the repository extends through several geo-hydrologic units covering many different locations, parameters utilized in the analysis cannot be adequately characterized by single values. These parameters may vary due to inherent uncertainty as well as varying both temporally and spatially. Therefore a probabilistic, rather than deterministic, approach is utilized. Using a Monte Carlo simulation, in which multiple realizations of the model are generated and repository performance computed for each one accomplish this.

II.2.2 INPUTS

The data used by the seepdll routine consists of percolation flux projections for 623 locations in the repository over the life of the repository, as well as data distributions representing the uncertainty associated with seepage fraction, mean seepage flux and seepage flux standard deviation. Also included are flow-focusing factor distributions that account for the effect of flow channeling in fractures.

II.2.2.1 Percolation Flux

The **ReadMasterFile** subroutine in the **seepdll** routine reads in the names of the T-H input files. The **master.in** file contains fifteen file names (3 infiltration scenarios, 2 fuel types, and 5 infiltration bins). If an input file does not exist for a given infiltration scenario "dummy" is inserted as a place holder and the code skips executing the calculation subroutine for a "dummy" file.

Separate percolation flux verses time data are provided for High Level Waste (HLW) packages and Commercial Spent Nuclear Fuel (CSNF) packages. This distinction is made to account for the differences in size between the two types of packages.

The percolation flux verses time data for each waste package type described above are divided into five bins based on water infiltration flux. The range of infiltration fluxes for each bin is

- Bin 1 0 – 3 mm/yr
- Bin 2 3 – 10 mm/yr
- Bin 3 10 – 20 mm/yr
- Bin 4 20 – 60 mm/yr
- Bin 5 > 60 mm/yr

Each bin contains percolation flux time histories (covering 0 to 1 million years) at multiple locations. A sample listing of data for mean infiltration from bin 5 for CNSF is presented in Figure II-1. The only information used from the header rows are the 'number of rows' and the 'fraction of this history'. The first column lists the time steps from zero to one-million years. The next column lists percolation flux values (mm/yr) associated with each time step at five meters above the drift. The third column lists percolation flux values (mm/yr) associated with each time step at three meters above the drift. The **seepdll** routine only utilizes the three-meter values in calculating seepage fraction and seepage flow rate.

Each infiltration scenario (e.g., Low , Base, High) has a separate set of percolation flux data. Low infiltration has data only in Bin 1 and Bin 2. Mean infiltration has data in all five bins. High infiltration only has data in Bins 2 through 5.

```
Infiltration Bin:
 qinf >   60.0 mm/yr
 RIP_csnf_qperc_d0010500_bin-60_mean
  Time (yr),  Percolation Flux at 5 m (mm/yr), Percolation Flux at 3 m (mm/yr)
The number of Rows = 352
The fraction of this history=0.000576
 Coordinate Location:
The easting coordinate = 170208.78 m
The northing coordinate = 234316.70 m
 Infiltration rate:
qinf =  61.00266 mm/yr
      0.00   0.153137E+02   0.153137E+02
      1.00   0.143936E+02   0.145005E+02
      2.00   0.144035E+02   0.145514E+02
      5.00   0.145207E+02   0.147774E+02
        .          .              .
        .          .              .            (data skipped)
        .          .              .
 800000.00   0.610016E+02   0.609857E+02
1000000.00   0.610027E+02   0.609868E+02
The number of Rows = 352
The fraction of this history=0.000960
 Coordinate Location:
The easting coordinate = 170228.75 m
The northing coordinate = 234315.60 m
 Infiltration rate:
qinf =  60.79187 mm/yr
      0.00   0.152617E+02   0.152617E+02
      1.00   0.143430E+02   0.144497E+02
      2.00   0.143530E+02   0.145008E+02
      5.00   0.144701E+02   0.147270E+02
        .          .              .
        .          .              .            (data skipped)
        .          .              .
 800000.00   0.607917E+02   0.607758E+02
1000000.00   0.607919E+02   0.607760E+02
The number of Rows = 352
The fraction of this history=0.001153
 Coordinate Location:
The easting coordinate = 170256.20 m
The northing coordinate = 234314.20 m
 Infiltration rate:
qinf =  60.37322 mm/yr
      0.00   0.151589E+02   0.151589E+02
      1.00   0.142427E+02   0.143489E+02
      2.00   0.142530E+02   0.144006E+02
      5.00   0.143699E+02   0.146269E+02
        .          .              .
        .          .              .            (data skipped)
        .          .              .
 800000.00   0.603734E+02   0.603575E+02
1000000.00   0.603732E+02   0.603574E+02
```

Figure II-1. Partial Listing of Percolation Flux Data in Bin 5 (Mean Infiltration)

II.2.2.2 Seepage Fraction, Mean Seepage Flow, and Seepage Flow Standard Deviation Distributions

Seepage fraction, mean seepage flow, and seepage flow standard deviation (as a function of percolation flux) are developed in CRWMS M&O 2000 [142004], Section 6.4 Table 11, in the form of triangle distributions. The distributions are implemented as a set of 1-D look-up tables (see Tables II-1, II-2, and II-3). Linear interpolation is used to select the value of the dependent parameter for percolation flux values within the range of the tables. Linear extrapolation is used to select the dependent parameter for percolation fluxes above the range of tables.

Table II-1. Triangular Distribution for Seepage Fraction $f_s(q)$

q [mm/yr]	min [-]	peak [-]	max [-]
0.0	0.0	0.0	0.0
3.4	0.0	0.0	0.0
5.0	0.0	0.0	1.97E-03
9.9	0.0	0.0	3.00E-02
14.6	0.0	2.45E-03	5.75E-02
73.2	0.0	0.250	0.744
97.9	0.0	0.292	0.779
213.0	4.91E-03	0.487	0.944
500.0	6.01E-02	0.925	0.999
549.2	6.96E-02	1.0	1.0
5383.4	1.0	1.0	1.0

Table II-2. Triangular Distribution for Mean Seepage Flow Rate, $\mu_{Qs}(q)$

q [mm/yr]	min [m^3/yr]	peak [m^3/yr]	max [m^3/yr]
0.0	0.0	0.0	0.0
3.4	0.0	0.0	0.0
5.0	0.0	0.0	3.21E-03
9.9	0.0	0.01	30E-02
14.6	0.0	7.95E-03	2.26E-02
73.2	0.0	1.06E-01	4.04E-01
97.9	0.0	3.54E-01	9.17E-01
213.0	2.84E-01	1.51E+00	3.31E+00
500.0	9.92E-01	5.50E+00	1.30E+01
549.2	1.11E+00	6.19E+00	1.46E+01
5383.4	1.30E+01	7.34E+01	1.77E+02

Table II-3. Triangular Distribution for Seepage Flow Rate Standard. Deviation, $\sigma_{QS}(q)$

q [mm/yr]	min [m^3/yr]	peak [m^3/yr]	max [m^3/yr]
0.0	0.0	0.0	0.0
3.4	0.0	0.0	0.0
5.0	0.0	0.0	3.16E-03
9.9	0.0	0.0	1.39E-02
14.6	0.0	7.09E-03	2.45E-02
73.2	0.0	1.98E-01	4.09E-01
97.9	0.0	3.66E-01	7.33E-01
213.0	1.88E-01	1.15E+00	2.24E+00
500.0	1.05E+00	4.48E+00	5.74E+00
549.2	1.20E+00	5.05E+00	6.33E+00
5383.4	1.57E+01	6.11E+01	6.52E+01

II.2.2.3 Seepage Flow Rate Distribution

The seep flow rate is derived from an inverse Beta distribution. The parameters controlling range and shape of the distribution are determined using the mean seep flow rate and flow rate standard deviation calculated from their respective triangular distributions. The minimum value of the Beta distribution is set to zero. The maximum value of the distribution is set equal to the mean plus ten standard deviations. The shaping factors for the distribution are determined from the mean seep flow rate and flow rate standard deviation. The seepage flow for a given realization is then randomly sampled from the Beta distribution using a random number (*r*) generated by the **seepdll** routine.

II.2.2.4 Flow Focusing Factor (*F*)

A separate flow focusing factor (*F*) is provided for the Low, Base and High infiltration scenarios. This factor accounts for the effects of flow channeling in fractures above the drifts. The range of the flow focusing factor, was calculated using information based on the mean log weep spacing (CRWMS M&O 2000 [142004]Sections 6.3.3.1 & 6.3.3.2,). The flow focusing factor is a stochastic multiplicative factor for which a log-uniform distribution is appropriate (CRWMS M&O 2000 [142004]Section 6.3.3.2). The maximum value of its distribution is dependent upon the infiltration scenario (Low, Medium, or High) (Table II-4).

Table II-4. Log-Uniform Distributions for Flow Focusing Factor (*F*)

Low Infiltration	Medium Infiltration	High Infiltration
Min = 1	Min = 1	Min = 1
Max = 47.3	Max = 22.4	Max = 9.7

The flow focusing factor used by the **seepdll** routine is generated by GoldSim and passed to the routine during execution.

II.2.2.5 Random Number (*R*)

A uniform random number, (*R*), on the interval (0;1) generated by GoldSim is used by the **seepdll** routine to evaluate the triangular distributions for seepage fraction, mean seepage flow and seepage flow standard deviation for each realization.

II.2.2.6 Random Number Seeds

Two random number seeds are generated by GoldSim. These seeds are passed to the **seepdll** routine for use by the International Mathematical and Statistical Libraries (IMSL) **random_number** subroutine.

II.2.3 DESCRIPTION OF SOFTWARE ROUTINE INCLUDING THE EXECUTION ENVIRONMENT

II.2.3.1 Development and Execution Environment

The **seepdll** routine is a FORTRAN program directly linked to and run with GoldSim. The code was developed and tested in the Windows NT 4.0 operating system. It was compiled with Digital FORTRAN Professional 6.0.A as a DLL (dynamically-linked library) for use within the performance assessment simulator Golder 2000 [143556]. The routine operates in a Windows 95/98 or Windows NT environment

II.2.3.2 Main Program

The basic structure of **seepdll** is comprised of an **if-then-else** construct where the statement blocks within it are executed depending on the value of the method variable passed by GoldSim to the dll. The **if-then-else** construct is structured (in pseudo code; with comments in *italics*) as follows:

```
if method=0 then
  initialize the dll

elseif method=2 then
  report the version of the dll

elseif method=3 then
  report the number of input and output arguments

elseif method=1 then
  perform the dll's calculations
  (more detailed pseudo code for this statement block is given below)

elseif method=99
  terminate the dll
```

Upon **method**=1 the dll begins executing the statement block that perform the seepage fraction and seepage flow calculations. The pseudo code for this statement block is as follows:

```
F=in(1)
get the flow focus factor (F) from GoldSim
```

```
R=in(2)
get the random number (R) from GoldSIM

inf_master=in(3)
get the infiltration state (inf_master) from GoldSIM

seed(1)=in(4)
seed(2)=in(5)
call random_number(put=seed(1:2))
set random number generator with seeds from GoldSim

call ReadMasterFile
read in the names of infiltration bin input files

call ReadDistributionData
read in the seepage fraction, mean seepage flow, and seepage flow standard
deviation distributions

nStart=0
initialize out() vector counter

do m=1,2
  execute calculation for CSNF (m=1) and HLW (m=2)

  do n=1,5
    potentially execute calculation for all five infiltration rate bins

    if FileNames(m,inf_master,n) □ 'dummy' then

      call CountDataSets
      counts the number of spatial locations in the infiltration bin

      call ReadPercData
      reads in the percolation flux histories in the infiltration bin

      call AllocateArrays
      allocates dynamic arrays

      call DoCalculations
      calculate the seepage flows and seepage fractions

      call GenerateOutput
      generate the output vector that is passed to GoldSIM

      call DeallocateArrays
      dallocate the dynamic arrays

    else

      call NoDataOutput
      infiltration bin has no data, generate 'no-data' output for GoldSIM

   endif

  end do
  end of loop through infiltration bins 1 to 5
```

```
end do
end of loop through CSNF and HLW
```

Descriptions of the implicit subroutines called by the main program block of the dll are given in Sections 2.3.4 through 2.3.11. Explicit subroutines used by the dll are described in Sections 2.3.12 through 2.3.14

II.2.3.3 Subroutine ReadMasterFile

The **ReadMasterFile** subroutine reads in the names of the T-H input files from the **master.in** file. This file contains fifteen file names (3 infiltration scenarios, 2 fuel types, and 5 infiltration bins). If an infiltration flux bin does not contain data for a given infiltration scenario (e.g., high infiltration has no data in bin 1), "dummy" is used in the **master.in** file as a placeholder.

II.2.3.4 Subroutine ReadDistributionData

The **ReadDistributionData** subroutine reads the seepage fraction, seepage flow, and seepage flow standard deviation triangular distributions from the files **SeepFrac.dat**, **SeepFlowMean.dat**, and **SeepFlowSD.dat**, respectively.

II.2.3.5 Subroutine CountDataSets

If the infiltration bin contains data then the **CountDataSets** subroutine is called. This subroutine reads through the selected data file counting the number of spatial locations in that bin.

II.2.3.6 Subroutine ReadPercData

This subroutine reads the percolation flux histories associated with each location in the infiltration bin into the PercHis array

II.2.3.7 Subroutine AllocateArrays

The AllocateArays subroutine allocates the dynamic arrays to sizes appropriate for the data to be stored in them.

II.2.3.8 Subroutine DoCalculations

This subroutine loops through each location in the percolation flux bin calculating the always seeps and sometimes seeps averages for selected percolation flux bin. The subroutine is structured (in pseudo code; with comments in *italics*) as follows:

```
do k=1,nLocations
loop through space
 call random_number(rn)
generate uniform random number used to evaluate beta distribution function
for seepage flow


 do j=1,nTimes
```

```
 loop through time

  perc=percHis(j,k)
  grab percolation flux from percHis array

  perc=perc*F
  scale the percolation flux by flow focus factor

  SeepFlow=0.0
  initialize SeepFlow to zero

  call SeepageFraction
  calculate seepage fraction

  if (SeepFrac .gt. 0.0) then
   call SeepageFlow
  end if
  if seepage fraction is not equal to 0.0, then calculate the seepage flow


  SeepFracOut(j,k)=SeepFrac
  SeepFlowOut(j,k)=SeepFlow
  store the seepage fraction and seepage flow values

 end do
end do

call NeverSometimesAlwaysSeeps
determine what fraction of locations always, sometimes, and never see seeps

call SeepageAveraging
calculate the average seepage for always and sometimes seeps locations
```

II.2.3.9 Subroutine SeepageFraction

First, the **SeepageFraction** subroutine checks to see if the percolation flux **(perc)** is outside the bounds of the seepage fraction response surface. If so, the seepage fraction is set equal to 0.0 (below the lower bound) or 1.0 (above the upper bound) as is appropriate. If the percolation flux is within the bounds of the response surface the **Interp** subroutine is called to find the minimum, peak, and maximum values of the seepage fraction triangle distribution. The subroutine **TriDist** is then called to generate the seepage fraction from the triangle distribution based on the cumulative probability value **(R)**.

The seepage fraction is then divided by the flow focus factor **(F)**, and then compared to **rn**. If the seepage fraction is less than **rn**, the seepage fraction and the seepage flow are set equal to 0.0. If the seepage fraction is greater than or equal to **rn**, **rn** is normalized by the seepage fraction **(rnp)**.

II.2.3.10 Subroutine SeepageFlow

The **Interp** subroutine is called to find the minimum, peak, and maximum values of the mean seepage flow and seepage standard deviation triangle distributions. The subroutine **TriDist** is then called to generate the mean seepage flow and seepage standard deviation from their respective distributions, based on the cumulative probability value (**R**).

If both the mean seepage flow and seepage standard deviation are equal to zero, the seepage flow is set equal to zero. If either has a non-zero value, the values of two parameters are passed along with **rnp** to the **BetaDist** subroutine, which returns the value for seepage flow.

II.2.3.11 Subroutine NeverSometimesAlwaysSeeps

The **NeverSometimesAlwaysSeeps** subroutine is called by the **DoCalculations** subroutine. When called it checks all of the time steps at each location to determine if the location sometimes seeps, always seeps, or never seeps. This is accomplished by nested do loops that loop through each time step at each location. Two Flags, **SeepFracYes,** and **SeepFracNo,** are used to denote whether or not seepage occurs at any time step associated with a given location. These flags are initialized to a value of –1. An IF statement is used to check whether or not the **SeepFracOut** value is greater than zero for each time step. If the value is greater than zero the **SeepFracYes** flag for that location is set to 1. If the value is equal to zero the **SeepFracNo** flag for that location is set to 1.

After each time step at each location has been checked and the seepage fraction flags, **SeepFracYes** and **SeepFracNo**, have been set for each location the flags are checked to determine if the location never leaks, sometimes leaks, or always leaks. A DO statement is used to loop through each location. At each location nested IF statements are used set the value of the variable **FracFlag(k)** in accordance with the following logic (Table II-5).

Table II-5. Logic Controlling the Value of the FracFlag Variable

Value of SeepFracYes(k)	Value of SeepFracNo(k)	Value FracFlag(k) Assumes	Condition Represented
1	-1	1	Always Seeps
-1	1	3	Never Seeps
1	1	2	Sometimes Seeps

When this step is completed the program returns to the **DoCalculations** subroutine.

II.2.3.11 Subroutine SeepageAveraging

The **SeepageAveraging** Subroutine is called by the **DoCalculations** subroutine to calculate the weighted average of the always seeps locations and the weighted average of the sometimes seeps locations for each time step.

A DO statement is used to loop through each time step. Another DO statement nested inside the first loops through each location for that time step. Nested IF statements inside the DO loop are used to determine if the location always seeps or sometimes seeps. If the **FracFlag** value equals 1 the always seeps counter is incremented by one and the **WeightAlways** variable is incremented by the **LocationWeight(k)** variable. The **LocationWeight(k)** variable is the fraction of this history value read from the Percolation Flux data files. Next the **SeepAlwaysAvg(j)** variable is incremented by the value of the **SeepFlowOut(j,k)** variable.

If the **FracFlag** value equals 2 the sometimes seeps counter is incremented by one and the **WeightSometimes** variable is incremented by the **LocationWeight(k)** variable. The **LocationWeight(k)** variable is the fraction of this history value read from the Percolation Flux data files. Next the **SeepSometimesAvg(j)** variable is incremented by the value of the **SeepFlowOut(j,k)** variable. If the **FracFlag** value equals 3 the summation steps are bypassed and the next location is evaluated.

When all of the locations for the time step have been evaluated the sum of the always seeps locations and the sometimes seeps locations are averaged. The value of the always seeps location counter is checked with an IF statement to determine whether or not it is equal to zero. If so, the always seeps average is set to zero. If not, the sum of the always seeps locations is divided by the value of the **WeightAlways** variable to obtain the weighted average of the always seeps locations.

Next the value of the sometimes seeps counter is checked with an IF statement to determine whether or not it is equal to zero. If so, the sometimes seeps average is set to zero. If not, the sum of the sometimes seeps locations is divided by the value of the **WeightSometimes** variable to obtain the weighted average of the sometimes seeps locations.

When this step is completed the program returns to the **DoCalculations** subroutine.

II.2.3.12 Subroutine GenerateOutput

Output from **seepdll** is passed to GoldSim via the **out( )** vector. The GoldSim TSPA model expects the following output for each infiltration bin:

- A one-dimensional table containing the time steps and the attendant average seepage flow rates of the "always seeps" locations.
- The fraction of "always seeps" locations.
- A one-dimensional table containing the time steps and the attendant average seepage flow rates of the "sometimes seeps" locations.
- The fraction of "sometimes seeps" locations.

A one-dimensional tables is passed in the **out( )** vector in the following format:

```
out(n) = 1              ["1" denotes a 1-d table]
out(n+1) = j            [# of rows in the table]
out(n+1+1) = 1st independent variable value
out(n+1+2) = 2nd independent variable value
out(n+1+3) = 3rd independent variable value
    .
    .
    .
```

```
out(n+1+j)   = j^th independent variable value
out(n+1+j+1) = 1^st dependent variable value
out(n+1+j+2) = 2^nd dependent variable value
out(n+1+j+3) = 3^rd dependent variable value
    .
    .
    .
out(n+1+j+j) = j^th dependent variable value
```

Hence a one-dimensional table takes up 2(j+1) elements in the **out( )** vector, where "j" is equal to the number of rows in the table. The total number of elements required for both 1-d tables and the "always seeps" and "sometimes seeps" fractions is 4(j+1)+2.

A counter (**nStart**) is used to keep track of the total number of elements used. It is intialized to 0 at the start of the dll. When output is stored in the **out( )** vector, it is stored in elements **nStart**+1 to **nStart**+4(j+1)+2. The counter is then incremented at the end of the **GenerateOutput** subroutine by 4(j+1)+2 so that the output for the next bin processed is written contiguous to the previous output.

II.2.3.13 Subroutine NoDataOutput

If a given bin has no data the **NoDataOutput** subroutine is executed. This subroutine stores two 1-d tables containing 0 m^3/yr values for seepage in the **out( )** vector, as well as zero values for the "always seeps" and "sometimes seeps" seepage fraction. The specific values and their locations in the out( ) vector are given in Table II-6.

Table II-6. Data Stored in the Out() Vector when an Infiltration Bin Contains no Data

Out vector	Value	Comment
out(nStart+1)	1	1st 1-d table flag
out(nStart+2)	2	# of rows in 1st table
out(nStart+3)	0	years
out(nStart+4)	1000000	years
out(nStart+5)	0	m^3/yr
out(nStart+6)	0	m^3/yr
out(nStart+7)	0	fraction of locations that "always seep"
out(nStart+8)	1	2nd 1-d table flag
out(nStart+9)	2	# of rows in 2nd table

Table II-6. Data Stored in the Out() Vector when an Infiltration Bin Contains no Data (Continued)

Out vector	Value	Comment
out(nStart+10)	0	years
out(nStart+11)	1000000	years
out(nStart+12)	0	m^3/yr
out(nStart+13)	0	m^3/yr
out(nStart+14)	0	fraction of locations that "sometimes seep"

Once the above data are stored, the nStart counter is incremented by 14 so that the output from the calculations on the next bin will be stored contiguous to the output from the current bin.

II.2.3.13 Subroutine DeallocateArrays

This subroutine deallocates all of the allocated dynamic arrays.

II.2.4 OUTPUT

The output for each loop through the main program consists of:

- A one-dimensional table containing the time steps and the always seeps average for each time step.
- The fraction of always seeps locations
- A second one-dimensional table listing the time steps and the sometimes seeps averages for each time step and
- The fraction of sometimes seeps locations.

II.2.5 DESCRIPTION OF TEST CASES

The logic and calculations of the **seepdll** routine have been verified by running a series of test cases. First, test cases that check individual subroutines were performed. These tests were then followed by an overall test that verified proper linking and integration of the subroutines with the main body of the program.

II.2.5.1 Subroutine Interp Testing

The **Interp** subroutine interpolates minimum, peak, and maximum values from the triangular distributions for seepage fraction, mean seepage flow, and mean seepage flow standard deviation. These interpolated values are then used by the **TriDist** subroutine to calculate mean and standard deviation values that are used to determine seepage fraction and seepage flow.

II.2.5.1.1 Subroutine Description

The **Interp** subroutine first checks to see if the independent variable is below the range of the response surface. If so, the minimum, peak, and maximum values are set equal to the minimum, peak, and maximum values associated with the minimum value of the independent variable.

If the independent variable is greater than the minimum value of the independent variable, an **if-statement** is used to determine if the random number is greater than the maximum value of the independent variable in the triangular distribution. If so, the subroutine linearly extrapolates values for the minimum, peak, and maximum.

If the independent variable is between the minimum and maximum values of the independent variable the values for minimum, peak, and maximum are linearly interpolated. This is accomplished by looping through the independent variable value using an **if-statement** to logically check that the random number is greater than the i^{th} value of the independent variable and less than the $i^{th}+1$ value of the independent variable.

II.2.5.1.2 Testing

This subroutine was tested by replicating the subroutine in an executable that reads an input test file( ) and writes the results an output test file ( ). The output was verified to agree with the results from an EXCEL spreadsheet that replicates the subroutines logic and calculations.

II.2.5.1.2.1 Test File Inputs

There are two test input files: **PercHis.dat** and **SeepFlowMean.dat**. These files contain data generated to test the subroutine's logic and calculations over the range of values expected for both the triangular distributions and the random number for which the min, peak, and max values will be evaluated.

PercHis.dat contains the independent variable values. These values cover the range of values the independent variable is expected to assume. The contents of the **PercHis.dat** file are shown in Figure II-2. The number in the 1st row denotes the number of independent variable values to follow. The remaining rows contain the independent variable values.

```
5
-1.0
5.0
80.0
5383.4
5450.0
```

Figure II-2. Values Used to Simulate the Independent Variable in Testing the Interp Subroutine

The triangular distribution used to test the subroutine is in the **SeepFlowMean.dat** file. The contents of this file are shown in Figure II-3. The first value in the file denotes the number of lines of data contained in the file. The remaining rows of data contain the independent variable value, minimum, peak, and maximum distribution values, respectively.

11			
0.0	0.0	0.0	0.0
3.4	0.0	0.0	0.0
5.0	0.0	0.0	3.21E-03
9.9	0.0	0.0	1.30E-02
14.6	0.0	7.95E-03	2.26E-02
73.2	0.0	1.06E-01	4.04E-01
97.9	0.0	3.54E-01	9.17E-01
213.0	2.84E-01	1.51E+00	3.31E+00
500.0	9.92E-01	5.50E+00	1.30E+01
549.2	1.11E+00	6.19E+00	1.46E+01
5383.4	1.30E+01	7.34E+01	1.77E+02

q min peak max
mean seepage flow rate

Figure II-3. Seepage Fraction Triangular Distribution Used to Test the Interp Subroutine

II.2.5.1.2.2 Test File Executable

The source code used to test the **Interp** subroutine is presented below. The subroutine was visually verified to contain the same logic and calculations as the subroutine in **seepdll**.

```
program SeepInterp

integer(4) nDistRows, nSize

real(8) Min, Peak, Max

real(8), allocatable :: PercValue(:), PercHis(:)
real(8), allocatable :: MeanSeepFlowMin(:)
real(8), allocatable :: MeanSeepFlowPeak(:)
real(8), allocatable :: MeanSeepFlowMax(:)
call ReadData

open(unit=13, file='output.dat')

do i=1,nSize

 call Interp(PercHis(i), PercValue, MeanSeepFlowMin, MeanSeepFlowPeak,   &
             MeanSeepFlowMax, nDistRows, Min, Peak, Max)

 write(13,1010) PercHis(i), Min, Peak, Max
 1010 format(F8.2, " ", E9.3, " ", E9.3, " ", E9.3)

end do

close(13)

contains

!***********************************************************************
!***********************************************************************
! read in data from files
subroutine ReadData

!!write(unit1,*) "entering subroutine ReadDistributionData"
```

```
open(unit=11, file='SeepFlowMean.dat')

! read in the number of data sets (rows) in the file
read(11,*) nDistRows

! set the size of the seepage flow and
! seepage fraction parameter vectors
allocate(PercValue(1:nDistRows))

allocate(MeanSeepFlowMin(1:nDistRows))
allocate(MeanSeepFlowPeak(1:nDistRows))
allocate(MeanSeepFlowMax(1:nDistRows))

! read in mean seepage flow data (triangle distribution)
do i=1,nDistRows
 read(11,*) PercValue(i), MeanSeepFlowMin(i), &
            MeanSeepFlowPeak(i), MeanSeepFlowMax(i)
end do

close(11)

open(unit=12, file='PercHis.dat')

read(12,*) nSize

allocate(PercHis(1:nSize))

do i=1,nSize
 read(12,*) PercHis(i)
end do

close(12)


!!write(unit1,*) "exiting subroutine ReadDistributionData"
!!write(unit1,*) " "

end subroutine ReadData
!************************************************************************
!************************************************************************
end program SeepInterp

!************************************************************************
!************************************************************************
! subroutine that interpolates between points on a given
! response surface
subroutine Interp(ind, IndData, DepMin, DepPeak, DepMax, &
                  nRows, Min, Peak, Max)

! variable listing
!
! ind                - independent variable
! IndData()          - range of data for independent variable
! DepMin()           - range of dependent variable minimum values
! DepPeak()          - range of dependent variable peak values
! DepMax()           - range of dependent variable maximum values
! nRows                   - number of rows in response surface data set
! Min                - interpolated minimum value
! Peak               - interpolated peak value
! Max                - interpolated maximum value

integer(4) nRows
```

```
real(8) IndData(nRows), DepMin(nRows)
real(8) DepPeak(nRows), DepMax(nRows)
real(8) ind, Min, Peak, Max

!write(666,*) "entering Interp subroutine"

! for independent variable values below the range of the response surface,
! set the dependent variables equal to the "floor" values
if (ind .le. IndData(1)) then
 Min=DepMin(1)
 Peak=DepPeak(1)
 Max=DepMax(1)

! for independent variable values above the range of the response surface,
! linearly extrapolate the values of the  dependent variables
elseif (ind .ge. IndData(nRows)) then

 Min=DepMin(nRows)   &
          + (ind-IndData(nRows))/(IndData(nRows)-IndData(nRows-1))   &
                  *(DepMin(nRows)-DepMin(nRows-1))
 Peak=DepPeak(nRows)  &
          + (ind-IndData(nRows))/(IndData(nRows)-IndData(nRows-1))   &
                  *(DepPeak(nRows)-DepPeak(nRows-1))
 Max=DepMax(nRows)   &
          + (ind-IndData(nRows))/(IndData(nRows)-IndData(nRows-1))   &
                  *(DepMax(nRows)-DepMax(nRows-1))

else

 do i=1,nRows-1     ! loop through the range of the independent variable

! if the independent variable is between the i-th and i-th plus 1 values
! in the independent variable range, interpolate the Min, Peak and Max
! values
  if ((ind .ge. IndData(i)) .and. (ind .lt. IndData(i+1))) then

   Min=DepMin(i)   &
          + (ind-IndData(i))/(IndData(i+1)-IndData(i))   &
                  *(DepMin(i+1)-DepMin(i))
   Peak=DepPeak(i)   &
          + (ind-IndData(i))/(IndData(i+1)-IndData(i))   &
                  *(DepPeak(i+1)-DepPeak(i))
   Max=DepMax(i)   &
          + (ind-IndData(i))/(IndData(i+1)-IndData(i))   &
                  *(DepMax(i+1)-DepMax(i))

  end if
 end do

end if

!write(666,*) " Min, Peak, Max"
!write(666,*) Min, Peak, Max
!write(666,*) "exiting Interp subroutine"
!write(666,*) " "

end subroutine Interp
!************************************************************************
!************************************************************************
```

II.2.5.1.2.3 Test File Output

The output generated for this test case is presented in Figure II-4.

```
  -1.00 0.000E+00 0.000E+00 0.000E+00
   5.00 0.000E+00 0.000E+00 0.321E-02
  80.00 0.000E+00 0.174E+00 0.545E+00
5383.40 0.130E+02 0.734E+02 0.177E+03
5450.00 0.132E+02 0.743E+02 0.179E+03
```

Figure II-4. Output File From Triangle Distribution Subroutine Test Case

II.2.5.1.2.4 Independent Verification

Operation of the Interp subroutine was verified by replicating the subroutines calculations in an Excel spreadsheet. This spreadsheet is shown in Figure II-5. Visual comparison of the solutions generated by the SeepInterp program with those generated by the EXCEL spreadsheet show they agree, thus verifying the performance of the Interp subroutine.

Input Data From SeepMeanFlow.dat			
11			
0	0	0	0
3.4	0	0	0
5	0	0	3.21E-03
9.9	0	0	1.30E-02
14.6	0	7.95E-03	2.26E-02
73.2	0	1.06E-01	4.04E-01
97.9	0	3.54E-01	9.17E-01
213	2.84E-01	1.51E+00	3.31E+00
500	9.92E-01	5.50E+00	1.30E+01
549.2	1.11E+00	6.19E+00	1.46E+01
5383.4	1.30E+01	7.34E+01	1.77E+02
q	min	peak	max
	mean seepage flow rate		
Test Case Values	**Test Case Solutions**		
	Min	**Peak**	**Max**
-1	0.00E+00	0.00E+00	0.00E+00
5	0.00E+00	0.00E+00	3.21E-03
80	0.00E+00	1.74E-01	5.45E-01
5383.4	1.30E+01	7.34E+01	1.77E+02
5450	1.32E+01	7.43E+01	1.79E+02

Figure II-5. EXCEL Spreadsheet Solutions To Interp Test Case

II.2.5.1.4 Conclusions

Visual inspection of the test case outputs given the same inputs verify the Interp subroutine in seep-dll correctly interpolates values from the triangular distributions.

II.2.5.2 Subroutine TriDist Testing

Seepage fraction, mean seepage flow, and seepage flow standard deviation are calculated by the subroutine **TriDist**. The subroutine is passed the minimum, peak, and maximum values of the triangle distribution for the parameter of interest, as well as a cumulative probability value (**R**). The subroutine returns the value of the independent variable (e.g. seepage fraction, mean seepage flow, or seepage flow) that yields the passed cumulative probability value from the triangle distribution.

II.2.5.2.1 Discussion of Triangle Distribution CDF

The general form of the triangle distribution CDF is given by:

$$F(x)=0 \qquad x<a \qquad \text{(Eq. II-1a)}$$

$$F(x)=\frac{(x-a)^2}{(b-a)(c-a)} \qquad a\le x\le b \qquad \text{(Eq. II-1b)}$$

$$F(x)=1-\frac{(c-x)^2}{(c-b)(c-a)} \qquad b<x<c \qquad \text{(Eq. II-1c)}$$

$$F(x)=1 \qquad x\ge c \qquad \text{(Eq. II-1d)}$$

where

$F(x)$ - cumulative probability
a - minimum value
b - peak value
c - maximum value
x - independent variable

Equations (II-1b) and (1c) can be rearranged as quadratic equations terms of x

$$x^2+[-2a]x+[F(x)(b-a)(c-a)]=0 \qquad \text{for (1b)} \qquad \text{(Eq. II-2a)}$$
$$Ax^2+Bx+C=0$$

where

$$A=1$$
$$B=-2a$$
$$C=[F(x)(b-a)(c-a)]$$

$$x^2+[-2c]x+[(F(x)-1)(c-b)(c-a)]=0$$
$$Ax^2+Bx+C=0$$

for (1c) (Eq. II-2b)

where

$$A=1$$
$$B=-2c$$
$$C=[(F(x)-1)(c-b)(c-a)]$$

which have a general form solution of

$$x=\frac{-B\pm\sqrt{[B^2-4AC]}}{2A} \qquad \text{(Eq. II-3)}$$

The four possible combinations of minimum, peak, and maximum values are shown in Table II-7, along with the equations that applicable for each case.

Table II-7. Combinations of Triangle Distribution Minimum, Peak, and Maximum Values

Case	Conditions	Governing Equations
Case 1	max = min peak = min	none, trivial solution (x = 0.0)
Case 2	max > peak peak = min	degenerate case eq. 2a is applicable
Case 3	max > peak peak > min	Both eqns. 2a and 2b are applicable
Case 4	peak = max min < peak	degenerate case eq. 2b is applicable

II.2.5.2.2 Subroutine Description

An if-then-else construct is used to determine which case from Table II-1 is applicable for the set of minimum, peak, and maximum values that are passed to the subroutine. The appropriate quadratic equation coefficients are then generated and used to calculate the distribution value associated with the cumulative probability value (R). The value calculated is then passed back to the calling statement.

II.2.5.2.3 Testing

This subroutine was tested by replicating the subroutine in an executable that calls an input test file(TriTestData.dat) and writes the results an output test file (TriTestOut.ou). The output was verified to agree with the results from an EXCEL spreadsheet that replicated the subroutines logic and calculations.

II.2.5.2.3.1 Test File Inputs

The input test file contains data generated to test each of the subroutines' logic and calculations over the range of values expected for both the triangular distributions and the random number used to evaluate them. The input file is presented in Figure II-6.

The first line of data simulates the random number generated to evaluate the triangular distribution. The second through seventh lines are the minimum, peak, and maximum values for six representative triangular distributions that will test each of the logic functions and calculations over the full range of expected values.

```
0.00     0.10   0.25   0.50   0.75   0.90   1.00
0.00E+000.00E+00      0.00E+00
0.00E+000.00E+00      1.97E-03
0.00E+002.45E-03      5.75E-02
4.91E-034.87E-01      9.44E-01
6.96E-021.00E+00      1.00E+00
1.00E+00       1.00E+00      1.00E+00
```

Figure II-6. Input File Used to Test the Triangle Distribution Subroutine

II.2.5.2.3.2 Test File Executable

The source code for the file used to test the TriDist subroutine is presented below. The subroutine was visually verified to contain the same logic and calculations as the subroutine in seepdll.

```
program TriDistTest

real(8) Mn(6), Peak(6), Mx(6)
real(8) rand(7)
real(8) OutMatrix(6,7)

! open input and output files
open(unit=20, file='TriTestData.dat')
open(unit=21, file='TriTestOut.ou')

! read inputs
read(20,*) (rand(j), j=1,7)
do i=1,6
 read(20,*) Mn(i), Peak(i), Mx(i)
end do

! loop through the six triangle distributions
do i=1,6
 ! loop through the seven 'random' numbers
 do j=1,7
  call TriDist(Mn(i), Peak(i), Mx(i), rand(j), OutMatrix(i,j))
 end do
 ! write results to output file
 write(21,1000) (OutMatrix(i,j), j=1,7)
 1000 format(7(E10.4, "  "))
end do

! close input and output files
close(20)
close(21)

end program TriDistTest
```

```
!*********************************************************************
!*********************************************************************
subroutine TriDist(Min, Peak, Max, R, solution)

! variable listing
!
! Min               - minimum value of triangle distribution
! Peak              - peak value of triangle distribution
! Max               - maximum value of triangle distribution
! R                      - random number at which distribution is evaluated
! solution          - result returned by the subroutine
! sol1, sol2 - intermidiate results
! A, B, C           - coefficients of quadratic equation
! PeakNorm          - normalized peak value of triangle distribution

real(8) Min, Peak, Max, R, solution, sol1, sol2
real(8) A, B, C


! case 1
if ( (Peak .eq. Min) .and.  &
     (Max .eq. Min)  .and.  &
        (Min .eq. 0.0) )then
 solution = 0.0

! case 2
elseif ( (Peak .eq. Min) .and. (Peak .le. Max) ) then
 write(22,*) "case 2"
 A=1.0
 B=-2*Max
 C=(Max**2)+((R-1)*(Max-Peak)*(Max-Min))
 solution=(-B-(B**2 - 4*A*C)**0.5)/(2*A)

! case 4
elseif ( (Min .lt. Peak) .and. (Peak .eq. Max) ) then
 A=1.0                                                ! the distribution
 B=-2*Min
 C=(Min**2)-(R*(Peak-Min)*(Max-Min))
 solution=(-B+(B**2 - 4*A*C)**0.5)/(2*A)

! case 3
elseif ( (Peak .gt. Min) .and. (Peak .lt. Max) ) then

 A=1.0
 B=-2*Min
 C=(Min**2)-(R*(Peak-Min)*(Max-Min))
 sol1 = (-B+(B**2 - 4*A*C)**0.5)/(2*A)

 A=1.0
 B=-2*Max
 C=(Max**2)+((R-1)*(Max-Peak)*(Max-Min))
 sol2=(-B-(B**2 - 4*A*C)**0.5)/(2*A)

 if (sol1 .le. Peak) then
  solution=sol1
 else
  solution=sol2
 end if

end if
```

```
end subroutine TriDist
!***********************************************************************
!***********************************************************************
```

II.2.5.2.3.3 Test File Output

The output generated for this test case is presented in Figure II-7.

```
0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
0.0000E+00  0.1011E-03  0.2639E-03  0.5770E-03  0.9850E-03  0.1347E-02  0.1970E-02
0.0000E+00  0.4125E-02  0.8776E-02  0.1772E-01  0.2937E-01  0.3971E-01  0.5750E-01
0.4910E-02  0.2177E+00  0.3413E+00  0.4807E+00  0.6164E+00  0.7368E+00  0.9440E+00
0.6960E-01  0.3638E+00  0.5348E+00  0.7275E+00  0.8754E+00  0.9523E+00  0.1000E+01
0.1000E+01  0.1000E+01  0.1000E+01  0.1000E+01  0.1000E+01  0.1000E+01  0.1000E+01
```

Figure II-7. Output File From Triangle Distribution Subroutine Test Case

II.2.5.2.4 Independent Verification

Operation of the TriDist subroutine was verified by replicating the subroutines logic and calculations in an Excel spreadsheet. This spreadsheet is replicated in Figures II-8 – II-14.

Random Number		Triangle Distribution		
0		Min	Peak	Max
0.1		0.00E+00	0.00E+00	0.00E+00
0.25		0.00E+00	0.00E+00	1.97E-03
0.5		0.00E+00	2.45E-03	5.75E-02
0.75		4.91E-03	4.87E-01	9.44E-01
0.9		6.96E-02	1.00E+00	1.00E+00
1		1.00E+00	1.00E+00	1.00E+00

Figure II-8. Subroutine Test Data As Entered In The Spreadsheet

The first row of data contains the values used to represent random numbers. The three rows on the right contain the min, peak, and max values of the triangular distributions used to test the subroutine.

The spreadsheet was set up to display intermediate results and the value of the random number used for each realization of each data set. The logic used to select the appropriate case is displayed in the cell immediately below the cell identifying the case.

The remaining cells display the results from the calculations of the coefficients A, B, and C for each data set along with the solution realized for that data set. If the data set was not applicable to a given case, the cells below that case display N/A.

The results for each data set are presented in Figures II-9 through II-14. Visual comparison of the solutions from the spreadsheet demonstrates agreement between the results obtained from the test case and those obtained from the spreadsheet.

			1st Data Set				
			Case 1	Case 2	Case 4	Case 3	
			Peak = Min & Max = Min & Min = 0	Peak = Min & Peak <= Max	Min < Peak & Peak = Max	Peak > Min & Peak < Max	
		A	0	N/A	N/A	N/A	N/A
		B	0	N/A	N/A	N/A	N/A
		C	0	N/A	N/A	N/A	N/A
Random #	0					N/A	N/A
		Solution	0	N/A	N/A	N/A	
		A	0	N/A	N/A	N/A	N/A
		B	0	N/A	N/A	N/A	N/A
		C	0	N/A	N/A	N/A	N/A
Random #	0.1					N/A	N/A
		Solution	0	N/A	N/A	N/A	
		A	0	N/A	N/A	N/A	N/A
		B	0	N/A	N/A	N/A	N/A
		C	0	N/A	N/A	N/A	N/A
Random #	0.25					N/A	N/A
		Solution	0	N/A	N/A	N/A	
		A	0	N/A	N/A	N/A	N/A
		B	0	N/A	N/A	N/A	N/A
		C	0	N/A	N/A	N/A	N/A
Random #	0.5					N/A	N/A
		Solution	0	N/A	N/A	N/A	
		A	0	N/A	N/A	N/A	N/A
		B	0	N/A	N/A	N/A	N/A
		C	0	N/A	N/A	N/A	N/A
Random #	0.75					N/A	N/A
		Solution	0	N/A	N/A	N/A	
		A	0	N/A	N/A	N/A	N/A
		B	0	N/A	N/A	N/A	N/A
		C	0	N/A	N/A	N/A	N/A
Random #	0.9					N/A	N/A
		Solution	0	N/A	N/A	N/A	
		A	0	N/A	N/A	N/A	N/A
		B	0	N/A	N/A	N/A	N/A
		C	0	N/A	N/A	N/A	N/A
Random #	1					N/A	N/A
		Solution	0	N/A	N/A	N/A	

Figure II-9. Spreadsheet Solutions For First Data Set

2nd Data Set					
	Case 1	Case 2	Case 4	Case 3	
	Peak = Min & Max = Min & Min = 0	Peak = Min & Peak <= Max	Min < Peak & Peak = Max	Peak > Min & Peak < Max	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-3.940E-03	N/A	N/A	N/A
C	NA	0.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	0.000E+00	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-3.940E-03	N/A	N/A	N/A
C	NA	3.881E-07	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.011E-04	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-3.940E-03	N/A	N/A	N/A
C	NA	9.702E-07	N/A	N/A	N/A
				N/A	N/A
Solution	NA	2.639E-04	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-3.940E-03	N/A	N/A	N/A
C	NA	1.940E-06	N/A	N/A	N/A
				N/A	N/A
Solution	NA	5.770E-04	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-3.940E-03	N/A	N/A	N/A
C	NA	2.911E-06	N/A	N/A	N/A
				N/A	N/A
Solution	NA	9.850E-04	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-3.940E-03	N/A	N/A	N/A
C	NA	3.493E-06	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.347E-03	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-3.940E-03	N/A	N/A	N/A
C	NA	3.881E-06	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.970E-03	N/A	N/A	

Figure II-10. Spreadsheet Solutions For Second Data Set

3rd Data Set					
	Case 1	Case 2	Case 4	Case 3	
	Peak = Min & Max = Min & Min = 0	Peak = Min & Peak <= Max	Min < Peak & Peak = Max	Peak > Min & Peak < Max	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	0.000E+00	-1.150E-01
C	NA	N/A	N/A	0.000E+00	1.409E-04
				0.000E+00	1.238E-03
Solution	NA	N/A	N/A	0.000E+00	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	0.000E+00	-1.150E-01
C	NA	N/A	N/A	-1.409E-05	4.574E-04
				3.753E-03	4.125E-03
Solution	NA	N/A	N/A	4.125E-03	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	0.000E+00	-1.150E-01
C	NA	N/A	N/A	-3.522E-05	9.322E-04
				5.935E-03	8.776E-03
Solution	NA	N/A	N/A	8.776E-03	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	0.000E+00	-1.150E-01
C	NA	N/A	N/A	-7.044E-05	1.724E-03
				8.393E-03	1.772E-02
Solution	NA	N/A	N/A	1.772E-02	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	0.000E+00	-1.150E-01
C	NA	N/A	N/A	-1.057E-04	2.515E-03
				1.028E-02	2.937E-02
Solution	NA	N/A	N/A	2.937E-02	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	0.000E+00	-1.150E-01
C	NA	N/A	N/A	-1.268E-04	2.990E-03
				1.126E-02	3.971E-02
Solution	NA	N/A	N/A	3.971E-02	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	0.000E+00	-1.150E-01
C	NA	N/A	N/A	-1.409E-04	3.306E-03
				1.187E-02	5.750E-02
Solution	NA	N/A	N/A	5.750E-02	

Figure II-11. Spreadsheet Solutions for Third Data Set

4th Data Set					
	Case 1	Case 2	Case 4	Case 3	
	Peak = Min & Max = Min & Min = 0	Peak = Min & Peak <= Max	Min < Peak & Peak = Max	Peak > Min & Peak < Max	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	-9.820E-03	-1.888E+00
C	NA	N/A	N/A	2.411E-05	4.620E-01
				4.910E-03	2.889E-01
Solution	NA	N/A	N/A	4.910E-03	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	-9.820E-03	-1.888E+00
C	NA	N/A	N/A	-4.525E-02	5.049E-01
				2.177E-01	3.225E-01
Solution	NA	N/A	N/A	2.177E-01	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	-9.820E-03	-1.888E+00
C	NA	N/A	N/A	-1.132E-01	5.693E-01
				3.413E-01	3.767E-01
Solution	NA	N/A	N/A	3.413E-01	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	-9.820E-03	-1.888E+00
C	NA	N/A	N/A	-2.263E-01	6.766E-01
				4.807E-01	4.808E-01
Solution	NA	N/A	N/A	4.807E-01	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	-9.820E-03	-1.888E+00
C	NA	N/A	N/A	-3.395E-01	7.838E-01
				5.876E-01	6.164E-01
Solution	NA	N/A	N/A	6.164E-01	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	-9.820E-03	-1.888E+00
C	NA	N/A	N/A	-4.074E-01	8.482E-01
				6.432E-01	7.368E-01
Solution	NA	N/A	N/A	7.368E-01	
A	N/A	N/A	N/A	1.000E+00	1.000E+00
B	NA	N/A	N/A	-9.820E-03	-1.888E+00
C	NA	N/A	N/A	-4.527E-01	8.911E-01
				6.778E-01	9.440E-01
Solution	NA	N/A	N/A	9.440E-01	

Figure II-12. Spreadsheet Solutions For Fourth Data Set

5th Data Set					
	Case 1	Case 2	Case 4	Case 3	
	Peak = Min & Max = Min & Min = 0	Peak = Min & Peak <= Max	Min < Peak & Peak = Max	Peak > Min & Peak < Max	
A	N/A	N/A	1.000E+00	N/A	N/A
B	NA	N/A	-1.392E-01	N/A	N/A
C	NA	N/A	4.844E-03	N/A	N/A
				N/A	N/A
Solution	NA	N/A	6.960E-02	N/A	
A	N/A	N/A	1.000E+00	N/A	N/A
B	NA	N/A	-1.392E-01	N/A	N/A
C	NA	N/A	-8.172E-02	N/A	N/A
				N/A	N/A
Solution	NA	N/A	3.638E-01	N/A	
A	N/A	N/A	1.000E+00	N/A	N/A
B	NA	N/A	-1.392E-01	N/A	N/A
C	NA	N/A	-2.116E-01	N/A	N/A
				N/A	N/A
Solution	NA	N/A	5.348E-01	N/A	
A	N/A	N/A	1.000E+00	N/A	N/A
B	NA	N/A	-1.392E-01	N/A	N/A
C	NA	N/A	-4.280E-01	N/A	N/A
				N/A	N/A
Solution	NA	N/A	7.275E-01	N/A	
A	N/A	N/A	1.000E+00	N/A	N/A
B	NA	N/A	-1.392E-01	N/A	N/A
C	NA	N/A	-6.444E-01	N/A	N/A
				N/A	N/A
Solution	NA	N/A	8.754E-01	N/A	
A	N/A	N/A	1.000E+00	N/A	N/A
B	NA	N/A	-1.392E-01	N/A	N/A
C	NA	N/A	-7.742E-01	N/A	N/A
				N/A	N/A
Solution	NA	N/A	9.523E-01	N/A	
A	N/A	N/A	1.000E+00	N/A	N/A
B	NA	N/A	-1.392E-01	N/A	N/A
C	NA	N/A	-8.608E-01	N/A	N/A
				N/A	N/A
Solution	NA	N/A	1.000E+00	N/A	

Figure II-13. Spreadsheet Solutions for Fifth Data Set

6th Data Set					
	Case 1	Case 2	Case 4	Case 3	
	Peak = Min & Max = Min & Min = 0	Peak = Min & Peak <= Max	Min < Peak & Peak = Max	Peak > Min & Peak < Max	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-2.000E+00	N/A	N/A	N/A
C	NA	1.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.000E+00	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-2.000E+00	N/A	N/A	N/A
C	NA	1.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.000E+00	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-2.000E+00	N/A	N/A	N/A
C	NA	1.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.000E+00	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-2.000E+00	N/A	N/A	N/A
C	NA	1.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.000E+00	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-2.000E+00	N/A	N/A	N/A
C	NA	1.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.000E+00	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-2.000E+00	N/A	N/A	N/A
C	NA	1.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.000E+00	N/A	N/A	
A	N/A	1.000E+00	N/A	N/A	N/A
B	NA	-2.000E+00	N/A	N/A	N/A
C	NA	1.000E+00	N/A	N/A	N/A
				N/A	N/A
Solution	NA	1.000E+00	N/A	N/A	

Figure II-14. Spreadsheet Solutions For Sixth Data Set

II.2.5.2.5 Conclusions

These tests demonstrate that the TriDist subroutine provides correct solutions for triangular distributions over the full range of values to be encountered for the TSPA Site Recommendation.

II.2.5.3 Subroutine BetaDist Testing

The BetaDist subroutine is called by the SeepageFlow subroutine to select the seepage flow rate from its beta distribution based on the random number rnp (generated by the DLL). The distribution is specified by its minimum, maximum, mean, and standard deviation values. The ISML inverse beta distribution function dbetin is used to generate the seepage flow rate from the specified beta distribution.

II.2.5.3.1 Discussion of the Beta Distribution

The canonical form of the normalized beta distribution is given in terms of shape parameters p and q, which are related to its minimum, maximum, mean, and standard deviation by (Iman, R. L. and Shortencarier, M. J. 1984 [100905])

$$X = \frac{Y-a}{b-a} \qquad \text{normalized Y} \qquad \text{(Eq. II-1)}$$

$$\mu = \frac{aq+bp}{p+q} \qquad \text{mean of X} \qquad \text{(Eq. II-2)}$$

$$\sigma^2 = \frac{(b-a)^2 pq}{(p+q)^2 (p+q+1)} \qquad \text{std. dev. of X} \qquad \text{(Eq. II-3)}$$

where:

- Y - variable with a beta distribution on the interval (a,b)
- X - variable with a beta distribution on the interval (0,1)
- a - minimum value
- b - maximum value
- μ - mean of X
- σ - standard deviation of X
- p - p shape factor
- q - q shape factor

For a minimum value of zero, equation (II-2) can be rearranged to solve for q

$$q = p\left(\frac{\mu}{b} - 1\right) \qquad \text{(Eq. II-4)}$$

Shape parameter p can be solved for by substituting equation (II-4) into equation (II-3)

$$p = \frac{\left(\frac{b}{\mu}\right)^2 \left(\frac{b}{\mu} - 1\right)}{\sigma^2 \left(\frac{b}{\mu}\right)^3} \qquad \text{(Eq. II-5)}$$

II.2.5.3.2 Subroutine Description

The subroutine is passed the mean seepage flow, seepage flow standard deviation, and the normalized random number rnp.

The subroutine begins by calculating an upper bound value for the seepage flow distribution. The upper bound of the distribution is defined to be 10 standard deviations above the mean seepage flow and the lower bound is defined to be equal to zero (ref. Seepage AMR). The shape factors p and q are then calculated per equations (II-3) and (II-4).

The beta distribution is then evaluated for the random number rnp using the IMSL function dbetin. Since the dbetin function returns a result in terms of the normalized beta distribution, it is multiplied by the upper bound – per equation (II-1) – to obtain the result over the interval of the unnormalized distribution.

The dbetin function will return the best approximate value and an error flag if it does not converge in 100 iterations (possible for very small values of rnp). To account for this the nlrty function is used to receive the error code generated by the dbetin function. The variable itype is set equal to the error code received by the nlrty function. An if-statement then checks the value of itype and sets solution equal to zero (an appropriate value for very small values of rnp) when the dbetin function returns an error code.

II.2.5.3.3 Testing

The BetaDist subroutine was verified by writing an independent executable program, BetaDistTest, with the BetaDist subroutine imbedded in it. The output from this program was then compared with the output from an EXCEL spreadsheet that was created to provide solutions for the inverse beta distribution over the same range of values provided to the BetaDistTest program. These files are presented in Section 3.3 COMPUTER LISTING OF TEST DATA INPUT AND OUTPUT. Visual inspections of these files show that the outputs from both methods agree thus validating the operation of the BetaDist subroutine.

II.2.5.3.3.1 Test File Inputs

The input files used in testing the BetaDist subroutine are listed below. Each input contains two values representing the Mean and Standard Deviation of the distribution being evaluated. The first set contains values that will not cause the dbetin function to generate an error code. The second set contains values that will cause the dbetin function to generate an error code.

Input data that will not cause the dbetin function to generate an error code

0.6 0.1

Input data that will cause the dbetin function to generate an error code.

0.102416815820969 0.139750587454366

II.2.5.3.3.2 Test File Executable

The source code used to test the BetaDist subroutine is presented below. The subroutine was visually verified to contain the same logic and calculations as the subroutine in seep-dll. The test code uses a do loop to exercise the BetaDist subroutine over the full range of values (10^{-10} to 1) that the random number rnp is likely to assume.

```
program BetaDistTest

real(8) Mean, SD, rnp, solution

! force IMSL error messages to be written to the debug.dat file (unit=50)
call umach(-3,50)

! causes non-covergence error in DBETIN function to print error message
! but not stop execution
call erset(0,1,0)


open(unit=10, file='output.dat')
open(unit=20, file='input.dat')
open(unit=50, file='debug.dat')
open(unit=888, file='debug1.dat')

read(20,*) Mean, SD

rnp=1.0E-10
call BetaDist(Mean, SD, rnp, solution)
write(10,*) solution, rnp

rnp=0.05
do i=1,19
 call BetaDist(Mean, SD, rnp, solution)
  write(10,*) solution, rnp
 rnp=rnp+0.05
end do

rnp=0.999
call BetaDist(Mean, SD, rnp, solution)
write(10,*) solution, rnp


close(10)
close(20)
close(50)

end program BetaDistTest
!**********************************************************************
!**********************************************************************
subroutine BetaDist(Mean, SD, rnp, solution)
```

```
use numerical_libraries          ! accesses IMSL routines

! variable listing

! Mean              - beta distribution mean
! SD                - beta distribution std. dev.
! rnp               - random number at which beta distribution is evaluated
! solution          - result from evaluation of beta distribution

! UB                - beta distribution upper bound
! p                        - beta distribution shape factor
! q                        - beta distribution shape factor

real(8) Mean, SD, rnp, UB, p, q, solution

!write(666,*) "  entering BetaDist subroutine"

! upper-bound is [mean + 10*SD]
UB=Mean + 10*SD

! shape parameter "p" for the beta distribution
p=((UB**2)*(UB/Mean - 1)/((SD**2)*(UB/Mean)**3)) - Mean/UB

! shape parameter "q" for the beta distribution
q=p*(UB/Mean - 1)

!write(666,*) "  ", Mean, SD, rnd, UB

! calculate the seepage flux from the beta distribution
! and the random number "rnp"

solution=dbetin(rnp, p, q)*UB

!icode=iercd()
itype=n1rty(1)
if (itype .eq. 4) then
 write(888,*) "error level 4 -- fatal"
 write(888,*) rnp,  " rnp"
 write(888,*) Mean, " Mean"
 write(888,*) SD,   " SD"
 write(888,*) UB,   " UB"
 write(888,*) solution, " solution"
end if

if (itype .eq. 4) then
 solution=0.0
end if

!write(666,*) solution, " BetaDist solution"
!write(666,*) " "

!write(666,*) "  exiting BetaDist subroutine"
!write(666,*) " "

end subroutine BetaDist
!***********************************************************************
```

II.2.5.3.3 Test File Output

The test file output is presented in Figure II-15

0.123928027502555	1.000000013351432E-010
0.439308672056746	5.000000074505806E-002
0.472695875596229	0.100000001490116
0.495790596104236	0.150000002235174
0.514448768824864	0.200000002980232
0.530656612442197	0.250000003725290
0.545360677820781	0.300000004470348
0.559104915059639	0.350000005215406
0.572246515787419	0.400000005960464
0.585048217494191	0.450000006705523
0.597725513902499	0.500000007450581
0.610475758533888	0.550000008195639
0.623500949949910	0.600000008940697
0.637031530740469	0.650000009685755
0.651359028439432	0.700000010430813
0.666891275047425	0.750000011175871
0.684262723234823	0.800000011920929
0.704595604197020	0.850000012665987
0.730278457188909	0.900000013411045
0.768467497309891	0.950000014156103
0.919426861282465	0.999000012874603

Figure II-15. Inverse Beta Distribution Function Solutions From The BetaDistTest Program For The Data Set With Mean = .6 And Standard Deviation = .1

II.2.5.3.4 Independent Verification

Operation of the dbetin function within the BetaDist subroutine was verified by replicating the subroutines logic and calculations in an Excel spreadsheet. This spreadsheet is presented as Figure II-16.

Mean	**Std Dev**	**UB**	**Alpha**	**Beta**
0.6	0.1	1.6	22.125	36.875
0.10241	0.13975	1.49991	0.432065	5.89602
mp				
1.00E-10	0.100000	0.000000		
0.05	0.439309	0.0001967		
0.1	0.472696	0.000980166		
0.15	0.49579	0.002514062		
0.2	0.514449	0.004919457		
0.25	0.530657	0.008309119		
0.3	0.545361	0.01280079		
0.35	0.559105	0.018526962		
0.4	0.572247	0.025643866		
0.45	0.585048	0.034343447		
0.5	0.597725	0.044868072		
0.55	0.610476	0.057533329		
0.6	0.623501	0.072762176		
0.65	0.637032	0.091141179		
0.7	0.651359	0.113519652		
0.75	0.666891	0.141194454		
0.8	0.684263	0.176297815		
0.85	0.704596	0.222730862		
0.9	0.730278	0.288955993		
0.95	0.768468	0.400547789		
0.999	0.919434	0.899817834		

Figure II-16. EXCEL Spreadsheet Inverse Beta Distribution Function Solutions For The Data Set With Mean = .6 And Standard Deviation = .1

II.2.5.3.5 Conclusions

These tests demonstrate that the BetaDist subroutine provides correct solutions for distributions over the full range of values to be encountered for the TSPA Site Recommendation.

II.2.5.4 Integrated Testing

The integrated test for seep-dll was performed by inserting write statements, which write intermediate results to an output file, into the routine and associated subroutines. The intermediate results are then checked using hand calculations. This was necessary since the seepdll routine generates random numbers making it impossible to replicate these calculations independently unless the random numbers and intermediate values are known.

The write statements inserted for testing are commented out during GoldSim runs to conserve execution time. Prior to running, the routine is checked with a comparison program that identifies any differences between the files. The only differences are that the write statements have been commented out.

II.2.5.4.1 Integrated Test Output

The output for the test case was written to the Seep debug.dat file presented below.

```
method=1; begin calculation

flow focus factor    11.59972
big 'R' random number    0.697516240267493
inf_master         2
entering subroutine ReadMasterFile
dummy
dummy
```

These are the values passed from GoldSim for the test case

```
dummy
dummy
dummy

dummy
dummy
dummy
dummy
dummy


dummy
dummy
dummy
dummy
CSNF_mean_tc_bin5.dat

dummy
dummy
dummy
dummy
dummy


dummy
dummy
dummy
dummy
dummy

dummy
dummy
dummy
dummy
dummy
exiting subroutine ReadMasterFile

entering subroutine ReadDistributionData
mean seepage flow data (triangle distribution)
0.00 0.000E+00 0.000E+00 0.000E+00
3.40 0.000E+00 0.000E+00 0.000E+00
5.00 0.000E+00 0.000E+00 0.321E-02
9.90 0.000E+00 0.000E+00 0.130E-01
14.60 0.000E+00 0.795E-02 0.226E-01
73.20 0.000E+00 0.106E+00 0.404E+00
97.90 0.000E+00 0.354E+00 0.917E+00
213.00 0.284E+00 0.151E+01 0.331E+01
500.00 0.992E+00 0.550E+01 0.130E+02
549.20 0.111E+01 0.619E+01 0.146E+02
5383.40 0.130E+02 0.734E+02 0.177E+03

std. dev. seepage flow data (triangle distribution)
```

This section of the output shows that all of the input files except the one for CSNF bin 5 were set to "dummy" and will not be processed for the test case.

This section shows the triangular distributions used in the test case

0.00 0.000E+00 0.000E+00 0.000E+00
3.40 0.000E+00 0.000E+00 0.000E+00
5.00 0.000E+00 0.000E+00 0.316E-02
9.90 0.000E+00 0.000E+00 0.139E-01
14.60 0.000E+00 0.709E-02 0.245E-01
73.20 0.000E+00 0.198E+00 0.409E+00
97.90 0.000E+00 0.366E+00 0.733E+00
213.00 0.188E+00 0.115E+01 0.224E+01
500.00 0.105E+01 0.448E+01 0.574E+01
549.20 0.120E+01 0.505E+01 0.633E+01
5383.40 0.157E+02 0.611E+02 0.652E+02

seepage fraction data (triangle distribution)
0.00 0.000E+00 0.000E+00 0.000E+00
3.40 0.000E+00 0.000E+00 0.000E+00
5.00 0.000E+00 0.000E+00 0.197E-02
9.90 0.000E+00 0.000E+00 0.300E-01
14.60 0.000E+00 0.245E-02 0.575E-01
73.20 0.000E+00 0.250E+00 0.744E+00
97.90 0.000E+00 0.292E+00 0.779E+00
213.00 0.491E-02 0.487E+00 0.944E+00
500.00 0.601E-01 0.925E+00 0.999E+00
549.20 0.696E-01 0.100E+01 0.100E+01
5383.40 0.100E+01 0.100E+01 0.100E+01

exiting subroutine ReadDistributionData

entering subroutine NoDataOutput
exiting subroutine NoDataOutput

entering subroutine NoDataOutput
exiting subroutine NoDataOutput

entering subroutine NoDataOutput
exiting subroutine NoDataOutput

entering subroutine NoDataOutput
exiting subroutine NoDataOutput

This section shows that the dummy files for CSNF were processed as "dummy" files and no data was processed for the first four infiltration bins.

```
 entering subroutine CountDataSets

# of spatial locations in data set 3

exiting subroutine CountDataSets

entering subroutine ReadPercData
exiting subroutine ReadPercData

entering subroutine AllocateArrays
exiting subroutine AllocateArrays
```

This shows that the CountDataSets subroutine correctly counted the number of data sets in CSNF infiltration bin 5

```
entering subroutine DoCalculations

little 'r' random number    3.920868194323862E-007

j-th time in the k-th location
j, k            1           1

perc      15.3137000000000
perc*F      177.634705204105

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
3.401367528689465E-003  0.427084861119032      0.893302574793027
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000      -6.802735057378930E-003 -0.262977479451479
A, B, C
0.516225946456557      sol1

1.00000000000000       -1.78660514958605      0.672492696879612
A, B, C
0.539047309126813      sol2

case 3b
0.539047309126813      solution

exiting subroutine TriDist

calculated SeepFrac   0.539047309126813
SeepFrac/F   4.647069821345081E-002

rn < SeepFrac   3.920868194323862E-007 4.647069821345081E-002
normalized rn (rn/SeepFrac)   8.437291336390934E-006
exiting subroutine SeepageFraction

4.647069821345081E-002 SeepFrac > 0.0
entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
0.196738977219513        1.15481076642872        2.57473370593766
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
0.130235660976297       0.909110415986261        1.77696351644298
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000      -0.393477954439026       -1.55043781584616
A, B, C
1.45735154299995      sol1

1.00000000000000       -5.14946741187531        5.60789629011472
A, B, C
1.56411143902646      sol2

case 3b
1.56411143902646      solution
```

This section shows the routine has generated a random number, r, has read the perc flux at the 1st time step for the 1st spatial location in the infiltration bin and has scaled the perc flux by the flow focus factor. The following sections show the intermediate results as the routine loops through the time steps and spatial locations in the infiltration bin.

```
exiting subroutine TriDist

entering subroutine TriDist

1.00000000000000      -0.260471321952594      -0.877669343900772
A, B, C
1.07608484096786      sol1

1.00000000000000       -3.55392703288596       2.72531439084597
A, B, C
1.11947971594031      sol2

case 3b
1.11947971594031      solution

exiting subroutine TriDist

entering BetaDist subroutine
1.56411143902646        1.11947971594031       0.0000000E+00
12.7589085984296
8.785619952793255E-004 BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

4.647069821345081E-002 SeepFrac
8.785619952793255E-004 SeepFlow

j-th time in the k-th location
j, k            2           1

perc     39.4073000000000
perc*F     457.113833912754

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
5.185300520433762E-002  0.859550032243158      0.990781396742862
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000      -0.103706010408675      -0.526286427115071
A, B, C
0.779159791412051      sol1

1.00000000000000       -1.98156279348572      0.944376678874699
A, B, C
0.797724158450592      sol2

case 3a
0.779159791412051      solution

exiting subroutine TriDist

calculated SeepFrac   0.779159791412051
SeepFrac/F   6.717054127478553E-002

rn < SeepFrac   3.920868194323862E-007  6.717054127478553E-002
normalized rn (rn/SeepFrac)   5.837184158281716E-006
```

```
exiting subroutine SeepageFraction

6.717054127478553E-002 SeepFrac > 0.0

entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
0.886204161708118        4.90377769098219        11.5520315352425
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
0.921192072588132        3.98240093006784        5.21699797454578
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000       -1.77240832341624       -29.1037332336376
A, B, C
6.35329580625428       sol1

1.00000000000000       -23.1040630704849        112.000573003349
A, B, C
6.92074018315859       sol2

case 3b
6.92074018315859       solution

exiting subroutine TriDist

entering subroutine TriDist

1.00000000000000       -1.84238414517626       -8.32399418701757
A, B, C
3.94982031349470       sol1

1.00000000000000       -10.4339959490916        25.6128182438853
A, B, C
3.95040821360128       sol2

case 3a
3.94982031349470       solution

exiting subroutine TriDist

entering BetaDist subroutine
6.92074018315859        3.94982031349470        0.0000000E+00
46.4189433181056
3.787721127930804E-002 BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

6.717054127478553E-002 SeepFrac
3.787721127930804E-002 SeepFlow

j-th time in the k-th location
j, k              3           1

perc     60.6116000000000
perc*F     703.077877844620
```

```
entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
9.921565047921771E-002   1.00000000000000        1.00000000000000
exiting Interp subroutine

entering subroutine TriDist

case 4
1.00000000000000      -0.198431300958435      -0.556129612183380
A, B, C
0.851527658547276      solution

exiting subroutine TriDist

calculated SeepFrac   0.851527658547276
SeepFrac/F   7.340929853607261E-002

rn < SeepFrac   3.920868194323862E-007  7.340929853607261E-002
normalized rn (rn/SeepFrac)   5.341105653525875E-006
exiting subroutine SeepageFraction

7.340929853607261E-002 SeepFrac > 0.0
entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
1.48847171560393        8.32936787264426        19.7693697741025
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
1.66155087268772        6.83413285614806        8.20389654311215
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000       -2.97694343120786       -85.0142463025088
A, B, C
10.8281609155003       sol1

1.00000000000000       -39.5387395482050        327.568491313068
A, B, C
11.8157863080081       sol2

case 3b
11.8157863080081       solution

exiting subroutine TriDist

entering subroutine TriDist

1.00000000000000       -3.32310174537544       -20.8437697741530
A, B, C
6.51999929545422       sol1

1.00000000000000       -16.4077930862243        64.5932200998258
A, B, C
6.55747667252574       sol2

case 3a
6.51999929545422       solution
```

```
exiting subroutine TriDist

entering BetaDist subroutine
11.8157863080081        6.51999929545422        0.0000000E+00
77.0157792625504
8.296809876617674E-002 BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

7.340929853607261E-002 SeepFrac
8.296809876617674E-002 SeepFlow

j-th time in the k-th location
j, k            4           1

perc     61.0027000000000
perc*F     707.614530202007

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
0.100088783852540        1.00000000000000        1.00000000000000
exiting Interp subroutine

entering subroutine TriDist

case 4
1.00000000000000      -0.200177567705079      -0.554858924739575
A, B, C
0.851671573299355      solution

exiting subroutine TriDist

calculated SeepFrac   0.851671573299355
SeepFrac/F   7.342170527458904E-002

rn < SeepFrac   3.920868194323862E-007  7.342170527458904E-002
normalized rn (rn/SeepFrac)   5.340203117947546E-006
exiting subroutine SeepageFraction

7.342170527458904E-002 SeepFrac > 0.0
entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
1.49962987962887        8.39244106054299        19.9217739656626
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
1.67515838979130        6.88673294812430        8.25914306255268
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000       -2.99925975925773       -86.3219740301543
A, B, C
10.9108393558880       sol1

1.00000000000000       -39.8435479313251        332.631030112911
A, B, C
```

```
11.9064107283514       sol2

case 3b
11.9064107283514       solution

exiting subroutine TriDist

entering subroutine TriDist

1.00000000000000       -3.35031677958260       -21.1276682118048
A, B, C
6.56737913733458       sol1

1.00000000000000       -16.5182861251054        65.4802229082491
A, B, C
6.60589740038901       sol2

case 3a
6.56737913733458       solution

exiting subroutine TriDist

entering BetaDist subroutine
11.9064107283514         6.56737913733458        0.0000000E+00
77.5802021016972
8.387285757181594E-002 BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

7.342170527458904E-002 SeepFrac
8.387285757181594E-002 SeepFlow

little 'r' random number    2.548044275764261E-002

j-th time in the k-th location
j, k             1           2

perc     15.2617000000000
perc*F     177.031519516087

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
3.375636497167551E-003  0.426062956608487       0.892437886361028
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000      -6.751272994335101E-003 -0.262111957610529
A, B, C
0.515355472557427      sol1

1.00000000000000       -1.78487577272206       0.671024620659100
A, B, C
0.538289950671098      sol2

case 3b
0.538289950671098      solution

exiting subroutine TriDist
```

```
calculated SeepFrac   0.538289950671098
SeepFrac/F   4.640540714226085E-002

rn < SeepFrac   2.548044275764261E-002  4.640540714226085E-002
normalized rn (rn/SeepFrac)   0.549083486748204
exiting subroutine SeepageFraction

4.640540714226085E-002 SeepFrac > 0.0
entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
0.195250665009284        1.14875270686878        2.56219310340569
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
0.129250440217413       0.905001835800277        1.76906602876405
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000     -0.390501330018568      -1.53609073268303
A, B, C
1.44992733844735     sol1

1.00000000000000      -5.12438620681138       5.55286438371634
A, B, C
1.55622634678629     sol2

case 3b
1.55622634678629     solution

exiting subroutine TriDist

entering subroutine TriDist

1.00000000000000     -0.258500880434827     -0.870597221614226
A, B, C
1.07121800755220     sol1

1.00000000000000      -3.53813205752811       2.70100358028747
A, B, C
1.11439738445646     sol2

case 3b
1.11439738445646     solution

exiting subroutine TriDist

entering BetaDist subroutine
1.55622634678629       1.11439738445646       0.0000000E+00
12.7002001913509
1.44771248540074     BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

4.640540714226085E-002 SeepFrac
1.44771248540074     SeepFlow
```

```
j-th time j, k               2            2

perc      39.2546000000000
perc*F      455.342555940437

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
5.151238906743114E-002  0.856846827532793       0.990441953229004
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000      -0.103024778134862      -0.524774992423929
A, B, C
0.777755129927099      sol1

1.00000000000000      -1.98088390645801       0.943032784860910
A, B, C
0.795653663398292      sol2

case 3a
0.777755129927099      solution

exiting subroutine TriDist

calculated SeepFrac   0.777755129927099
SeepFrac/F   6.704944689428492E-002

rn < SeepFrac    2.548044275764261E-002  6.704944689428492E-002
normalized rn (rn/SeepFrac)    0.380024652519758
exiting subroutine SeepageFraction

6.704944689428492E-002 SeepFrac > 0.0
entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
0.881834597929720       4.87915260697681        11.4922277598008
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
0.915872066970930       3.96184916822877        5.19539702366387
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000      -1.76366919585944       -28.8062047213775
A, B, C
6.32093718746445      sol1

1.00000000000000      -22.9844555196017       110.846821886712
A, B, C
6.88522472133598      sol2

case 3b
6.88522472133598      solution

exiting subroutine TriDist

entering subroutine TriDist
```

```
1.00000000000000      -1.83174413394186      -8.25353623235785
A, B, C
3.93122575690773      sol1

1.00000000000000      -10.3907940473277       25.3953388189841
A, B, C
3.93174698725693      sol2

case 3a
3.93122575690773      solution

exiting subroutine TriDist

entering BetaDist subroutine
6.88522472133598        3.93122575690773      0.0000000E+00
46.1974822904132
5.09972506780483      BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

6.704944689428492E-002 SeepFrac
5.09972506780483      SeepFlow

j-th time in the k-th location
j, k              3           2

perc      60.4060000000000
perc*F      700.692974431992

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
9.875664710014583E-002   1.00000000000000        1.00000000000000
exiting Interp subroutine

entering subroutine TriDist

case 4
1.00000000000000      -0.197513294200292      -0.556797423491058
A, B, C
0.851452003029436      solution

exiting subroutine TriDist

calculated SeepFrac   0.851452003029436
SeepFrac/F   7.340277635392234E-002

rn < SeepFrac   2.548044275764261E-002  7.340277635392234E-002
normalized rn (rn/SeepFrac)   0.347131866440377
exiting subroutine SeepageFraction

7.340277635392234E-002 SeepFrac > 0.0
entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
1.48260590500939        8.29621050258040       19.6892513854941
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
```

```
1.65439744513340        6.80648115860186        8.17485362724160
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000        -2.96521181001877       -84.3307805205829
A, B, C
10.7846971102251        sol1

1.00000000000000        -39.3785027709881       324.922699293827
A, B, C
11.7681453498856        sol2

case 3b
11.7681453498856        solution

exiting subroutine TriDist

entering subroutine TriDist

1.00000000000000        -3.30879489026680       -20.6952850979510
A, B, C
6.49509119752209        sol1

1.00000000000000        -16.3497072544832       64.1293468806174
A, B, C
6.53202528982843        sol2

case 3a
6.49509119752209        solution

exiting subroutine TriDist

entering BetaDist subroutine
11.7681453498856        6.49509119752209        0.0000000E+00
76.7190573251065
8.35458848705407        BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

7.340277635392234E-002 SeepFrac
8.35458848705407        SeepFlow

j-th time in the k-th location
j, k            4           2

perc      60.7919000000000
perc*F      705.169308220577

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
9.961817143858860E-002   1.00000000000000        1.00000000000000
exiting Interp subroutine

entering subroutine TriDist

case 4
1.00000000000000        -0.199236342877177      -0.555543873149570
A, B, C
```

```
0.851594004315110      solution

exiting subroutine TriDist

calculated SeepFrac   0.851594004315110
SeepFrac/F   7.341501813452445E-002

rn < SeepFrac   2.548044275764261E-002  7.341501813452445E-002
normalized rn (rn/SeepFrac)   0.347073983022829
exiting subroutine SeepageFraction

7.341501813452445E-002 SeepFrac > 0.0
entering subroutine SeepageFlow
entering Interp subroutine
Min, Peak, Max
1.49361571195703          8.35844507995222          19.8396292364862
exiting Interp subroutine

entering Interp subroutine
Min, Peak, Max
1.66782403897198          6.85838188857791          8.22936559822626
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000        -2.98723142391406         -85.6158785099687
A, B, C
10.8662762950025        sol1

1.00000000000000        -39.6792584729725          329.897536149541
A, B, C
11.8575648361135        sol2

case 3b
11.8575648361135        solution

exiting subroutine TriDist

entering subroutine TriDist

1.00000000000000        -3.33564807794397         -20.9744136665661
A, B, C
6.54184195969600        sol1

1.00000000000000        -16.4587311964525          65.0013848500803
A, B, C
6.57979798846012        sol2

case 3a
6.54184195969600        solution

exiting subroutine TriDist

entering BetaDist subroutine
11.8575648361135          6.54184195969600          0.0000000E+00
77.2759844330735
8.41891098620637        BetaDist solution

exiting BetaDist subroutine

exiting subroutine SeepageFlow

7.341501813452445E-002 SeepFrac
```

```
8.41891098620637        SeepFlow

little 'r' random number    0.352516161261067

j-th time in the k-th location
j, k              1            3

perc      15.1589000000000
perc*F      175.839067809772

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
3.324768227158843E-003  0.424042729999180       0.890728463845460
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000       -6.649536454317686E-003 -0.260404314343510
A, B, C
0.513633859295773       sol1

1.00000000000000        -1.78145692769092       0.668126981942338
A, B, C
0.536793138733435       sol2

case 3b
0.536793138733435       solution

exiting subroutine TriDist

calculated SeepFrac   0.536793138733435
SeepFrac/F   4.627636856872620E-002

rn >= SeepFrac    0.352516161261067       4.627636856872620E-002
SeepFrac and SeepFlow equal 0.0   0.000000000000000E+000
0.000000000000000E+000
exiting subroutine SeepageFraction

0.000000000000000E+000 SeepFrac
0.000000000000000E+000 SeepFlow

j-th time in the k-th location
j, k              2            3

perc      38.9021000000000
perc*F      451.253652959156

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
5.072609444883561E-002  0.850606620195506       0.989658365549664
exiting Interp subroutine

entering subroutine TriDist

1.00000000000000       -0.101452188897671      -0.521285023286587
A, B, C
0.774506558363563       sol1

1.00000000000000        -1.97931673109933       0.939931349087096
```

```
A, B, C
0.790931589761053      sol2

case 3a
0.774506558363563      solution

exiting subroutine TriDist

calculated SeepFrac   0.774506558363563
SeepFrac/F   6.676939097674697E-002

rn >= SeepFrac   0.352516161261067       6.676939097674697E-002
SeepFrac and SeepFlow equal 0.0   0.000000000000000E+000
0.000000000000000E+000
exiting subroutine SeepageFraction

0.000000000000000E+000 SeepFrac
0.000000000000000E+000 SeepFlow

j-th time in the k-th location
j, k            3           3

perc      59.9655000000000
perc*F      695.583295670986

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
9.777322789547090E-002   1.00000000000000        1.00000000000000
exiting Interp subroutine

entering subroutine TriDist

case 4
1.00000000000000       -0.195546455790942      -0.558227786638927
A, B, C
0.851289910346514      solution

exiting subroutine TriDist

calculated SeepFrac   0.851289910346514
SeepFrac/F   7.338880251522001E-002

rn >= SeepFrac   0.352516161261067       7.338880251522001E-002
SeepFrac and SeepFlow equal 0.0   0.000000000000000E+000
0.000000000000000E+000
exiting subroutine SeepageFraction

0.000000000000000E+000 SeepFrac
0.000000000000000E+000 SeepFlow

j-th time in the k-th location
j, k            4           3

perc      60.3732000000000
perc*F      700.312503459549

entering subroutine SeepageFraction

entering Interp subroutine
Min, Peak, Max
9.868342088013823E-002   1.00000000000000        1.00000000000000
```

```
exiting Interp subroutine

entering subroutine TriDist

case 4
1.00000000000000      -0.197366841760276      -0.556903949693028
A, B, C
0.851439933472115      solution

exiting subroutine TriDist

calculated SeepFrac   0.851439933472115
SeepFrac/F   7.340173585015518E-002

rn >= SeepFrac   0.352516161261067      7.340173585015518E-002
SeepFrac and SeepFlow equal 0.0   0.000000000000000E+000
0.000000000000000E+000
exiting subroutine SeepageFraction

0.000000000000000E+000 SeepFrac
0.000000000000000E+000 SeepFlow

1           1     k-th location, j-th time
4.647069821345081E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)

1           2     k-th location, j-th time
6.717054127478553E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)

1           3     k-th location, j-th time
7.340929853607261E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)

1           4     k-th location, j-th time
7.342170527458904E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)

2           1     k-th location, j-th time
4.640540714226085E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)

2           2     k-th location, j-th time
6.704944689428492E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)

2           3     k-th location, j-th time
7.340277635392234E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)
```

This section shows the results of setting the always seeps, sometimes seeps and never seeps flags for each time step at each location

```
2           4     k-th location, j-th time
7.341501813452445E-002  SeepFracOut(j,k)

location has at least one time with seepage
1.00000000000000      FracYes(k)

3           1     k-th location, j-th time
0.000000000000000E+000  SeepFracOut(j,k)

location has at least one time with no seepage
1.00000000000000      FracNo(k)

3           2     k-th location, j-th time
0.000000000000000E+000  SeepFracOut(j,k)

location has at least one time with no seepage
1.00000000000000      FracNo(k)

3           3     k-th location, j-th time
0.000000000000000E+000  SeepFracOut(j,k)

location has at least one time with no seepage
1.00000000000000      FracNo(k)

3           4     k-th location, j-th time
0.000000000000000E+000  SeepFracOut(j,k)

location has at least one time with no seepage
1.00000000000000      FracNo(k)

done calculating seepage fractions and seepage flows
now determine always, sometimes and never seepage flow fractions


1  location always seeps
-1.00000000000000        1.00000000000000      FracNo(k), FracYes(k)
1.00000000000000      FracFlag(k)=1

2  location always seeps
-1.00000000000000        1.00000000000000      FracNo(k), FracYes(k)
1.00000000000000      FracFlag(k)=1

3  location never seeps
1.00000000000000       -1.00000000000000      FracNo(k), FracYes(k)
3.00000000000000      FracFlag(k)=3
exiting subroutine DoCalculations

entering subroutine GenerateOutput
1st 1-d table written
0.666666666666667      fraction always seeps
exiting subroutine GenerateOutput
entering subroutine DeallocateArrays
exit subroutine DeallocateArrays

entering subroutine NoDataOutput
exiting subroutine NoDataOutput

entering subroutine NoDataOutput
exiting subroutine NoDataOutput

entering subroutine NoDataOutput
exiting subroutine NoDataOutput
```

```
entering subroutine NoDataOutput
exiting subroutine NoDataOutput

entering subroutine NoDataOutput
exiting subroutine NoDataOutput

finished with calculation do loops
```

2.5.4.2 Test Case Output Vector

The following 1d array presents the output of the test case. This vector is representative of the data passed to GoldSim for each infiltration bin.

```
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
```

0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
4.00000000000000
0.000000000000000E+000
1000.00000000000
10000.0000000000
1000000.00000000
0.724295523698010
2.56880113954207
4.21877829291012
4.25139192188909
0.666666666666667
1.00000000000000
4.00000000000000
0.000000000000000E+000
1000.00000000000
10000.0000000000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000

```
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
1.00000000000000
2.00000000000000
0.000000000000000E+000
1000000.00000000
0.000000000000000E+000
0.000000000000000E+000
0.000000000000000E+000
```

II.2.5.4.3 Conclusions

The above intermediate results were verified by hand calculations. The results of the hand calculations are consistent with the intermediate results presented above demonstrating that the routine correctly processes data providing outputs consistent with the input data.

II.2.6 DESCRIPTION OF TEST RESUTLTS

The testing described above demonstrates that the individual subroutines, **Interp, Tridist**, and **BetaDist**, within seep-dll produce outputs that can be independently verified. The integrated test demonstrates that all of the various subroutines are correctly linked, producing intermediate and final results that can be verified by inspection and hand calculations.

II.2.7 RANGE OF INPUT PARAMETER VALUES FOR WHICH RESULTS WERE VERIFIED

The seepdll routine was tested using the same data sets used during GoldSim runs. The only difference is that the test inputs are limited to the CSNF bin 5 data file to limit the results in the test case output to a number that can reasonably be verified.

II.2.8 LIMITATIONS ON SOFTWARE ROUTINE APPLICATIONS OR VALIDITY

This software routine was developed specifically for the TSPA model to evaluate seepage into emplacement drifts within the Yucca Mountain repository. It was written and compiled as a direct linked library (DLL) file. As such it can be called by any program written to utilize DLLs. For any output to be meaningful the routine would also have to have access to appropriate input data correctly describing percolation flux, seepage flows, seepage fractions, and flow focusing factors for the geological area being evaluated.

II.3. SUPPORTING INFORMATION:

II.3.1 DIRECTORY LISTING OF EXECUTABLE AND DATA FILES

Executable and data files must reside in the same directory.

II.3.2 COMPUTER LISTING OF SOURCE CODE

```
subroutine seepdll(method, state, in, out)

! attibute statements needed for the dll
!DEC$ ATTRIBUTES dllexport,c :: seepdll
!DEC$ ATTRIBUTES value :: method
!DEC$ ATTRIBUTES reference :: state
!DEC$ ATTRIBUTES reference :: in
!DEC$ ATTRIBUTES reference :: out

!*******************************************************************************
! variable listing
! (note that variables used in external subroutines
!  are defined in those subrou)
!*******************************************************************************

! stoichiocastic parameters

! F                    - perc. flux flow focus parameter passed from GoldSim [-]
! R                    - random number (passed from GoldSim) used to select values
!                         from triangle distributions [-]
! rn           - random number compared to seepage fraction [-]
! rnp          - random number normalized by the seepage flux; used to select
!                     seepage flow values from beta distribution [-]


! indices

! i                            - the i-th data element in triangle distribution tables
! inf_master - infiltration scenario [1=low, 2=mean, 3=high]
! j                            - the j-th time in a given infiltration bin
! k                            - the k-th spatial location in a given infiltration bin
```

```
! m                         - the m-th fuel type
! n                         - the n-th infiltration bin


! character variables

! dummy                                        - dummy variable used to read past
!                                                input file headers
! dummy2()                             - dummy variable used to read past
!                                                input file headers
! FileName(m,inf_master,n) - input file names
! infile                               - name of current input file


! counters or indicial limits

! nAlways            - number of locations that aways see seepage
! nDistRows          - number of rows in triangle distributions
! nLocations - number of spatial locations
! nOut               - number of values passed back to GoldSim (via the out vector)
! nStart             - starting index for tables passed back to GoldSim
! nTimes             - number of time steps
! nSometimes - number of locations that sometimes see seepage


! GoldSim input and output vectors

! in()        - vector used by GoldSim to pass information to the dll
! out()               - vector used by GoldSim to pass information from the dll


! percolation flux data

! percHis(j,k)              - percolation flux [ mm/yr]
! perc                      - percolation flux for the current location
!                                   and time step [ mm/yr]
! t(j)                      - time [ yr]


! seepage fraction and seepage flow variables

! SFwMean     - mean seepage flow
! SFwSD               - seepage flow standard deviation

! SeepFlow                          - seepage flow for the current location and time step
! SeepFlowOut(j,k)          - storage array for calculated seepage flows
! SeepFrac                          - seepage fraction for the current location and time
step
! SeepFracOut(j,k)          - storage array for calculated seepage fractions

! FracNo(k)                 - [ =1] no seep flow at at least one time at the
!                                    k-th location
!                                  [ =-1] seep flow at all times at the k-th location
! FracYes(k)          - [ =1] seep flow at at least one time at the k-th location
!                                  [ =-1] no seep flow at all times in the k-th
location
! FracFlag(k)         - [ =1] seepage at all times at the k-th location
!                                  [ =2] seepage sometimes at the k-th location
!                                  [ =3] no seepage at any time at the k-th location

! SeepAlwaysAvg(j)          - average of locations that always have seepage
!                                         at the j-th time [ m3/yr]
! SeepSometimesAvg(j)       - average of locations that sometimes have seepage
```

```
!                                              at the j-th time [ m3/yr]

! triangle distributions for seepage fraction, seepage flow mean, and
! seepage flow standard deviation as a function of percolation flux

! PercValue(i)                  - percolation flux [ mm/yr]

! SeepFracMax(i)          - seepage fraction maximum data set [ -]
! SeepFracMin(i)          - seepage fraction minimum data set [ -]
! SeepFracPeak(i)         - seepage fraction peak data set [ -]

! MeanSeepFlowMax(i)      - mean seepage flow maximum data set [ m3/yr]
! MeanSeepFlowMin(i)      - mean seepage flow minimum data set [ m3/yr]
! MeanSeepFlowPeak(i)     - mean seepage flow peak data set [ m3/yr]

! SdSeepFlowMax(i)        - std. dev. seepage flow maximum data set [ m3/yr]
! SdSeepFlowMin(i)        - std. dev. seepage flow minimum data set [ m3/yr]
! SdSeepFlowPeak(i)       - std. dev. seepage flow peak data set [ m3/yr]


! triangle distribution parameters for a given percolation flux

! SFcMax                  - seepage fraction maximum [ -]
! SFcMin                  - seepage fraction minimum [ -]
! SFcPeak                 - seepage fraction peak [ -]

! SFwMeanMax        - mean seepage flow maximum [ m3/yr]
! SFwMeanMin        - mean seepage flow minimum [ m3/yr]
! SFwMeanPeak       - mean seepage flow peak [ m3/yr]

! SFwSdMax                - std. dev. seepage flow maximum [ m3/yr]
! SFwSdMin                - std. dev. seepage flow minimum [ m3/yr]
! SFwSdPeak               - std. dev. seepage flow peak [ m3/yr]

!******************************************************************************

use numerical_libraries          ! accesses IMSL routines

! define static variables and arrays
real(8) F, R, rn, rnp, perc
real(8) SFcMin, SFcPeak, SFcMax
real(8) SFwMeanMin, SFwMeanPeak, SFwMeanMax
real(8) SFwSdMin, SFwSdPeak, SFwSdMax
real(8) SeepFrac, SeepFlow
real(8) SFwMean, SFwSD
real(8) nAlways, nSometimes
real(8) WeightAlways, WeightSometimes

integer(4) nLocations, nTimes, nDistRows, nOut, inf_master, nStart
integer(4) method, state, nunit, n, m
integer(4) DebugFlag

integer(4) seed(2)

integer(4) unit1, unit2, unit3

character*25 infile, dummy1, dummy2(10)
character*25 FileNames(2,3,5)

logical used

! define dynamic arrays
```

```
real(8), allocatable :: PercValue(:)

real(8), allocatable :: MeanSeepFlowMin(:)
real(8), allocatable :: MeanSeepFlowPeak(:)
real(8), allocatable :: MeanSeepFlowMax(:)

real(8), allocatable :: SdSeepFlowMin(:)
real(8), allocatable :: SdSeepFlowPeak(:)
real(8), allocatable :: SdSeepFlowMax(:)

real(8), allocatable :: SeepFracMin(:)
real(8), allocatable :: SeepFracPeak(:)
real(8), allocatable :: SeepFracMax(:)

real(8), allocatable :: percHis(:,:)
real(8), allocatable :: t(:), SeepAlwaysAvg(:), SeepSometimesAvg(:)
real(8), allocatable :: SeepFracOut(:,:), SeepFlowOut(:,:)
real(8), allocatable :: FracFlag(:), FracYes(:), FracNo(:)
real(8), allocatable :: LocationWeight(:)

real(8) in(*), out(*)             ! define in and out array for dll

! the output for one fuel type perc flux bin is comprised of:
!  - 1-d table flag (always seeps look-up table)
!  - # of rows in table
!  - 1 to n independent variable values
!  - 1 to n dependent variable values
!  - always seeps fraction
!  - 1-d table flag (sometimes seeps look-up table)
!  - # of rows in table
!  - 1 to n independent variable values
!  - 1 to n dependent variable values
!  - always seeps fraction
!
! for n=281, one fuel type bin has
!   [ 2* (2+( 2*281) +1)] =1130 elements
!
! hence for 10 fuel type bins there are 11300 elements in
! the out vector that is passed back to GoldSim

if (method.eq.0) then            ! initialize
 return

elseif (method.eq.2) then  ! report version
 out(1) = 1.0
 return

elseif (method.eq.3) then  ! report arguments
 out(1) = 5
 nOut=2* (2 + (2*457) + 1)
 out(2) = nOut*10
 return

elseif (method.eq.1) then  ! calculation

! open debug file
!used = .true.
unit1=666
!do while(used)
! inquire(unit=unit1, opened=used)
!  if(.not. used) then
!  used = .false.
```

```
  open(unit1, file='seep_debug.dat')
! else
!  used = .true.
!  unit1=unit1+1
! end if
!end do

! open output file
! contains the out() vector that is passed back to GoldSim
!used = .true.
unit2=777
!do while(used)
! inquire(unit=unit2, opened=used)
! if(.not. used) then
!  used = .false.
  open(unit2, file='seep_output.dat')
! else
!  used = .true.
!  unit2=unit2+1
! end if
!end do

! open test file
! contains minmum data set nessary for code V&V
!used = .true.
unit3=888
!do while(used)
! inquire(unit=unit3, opened=used)
! if(.not. used) then
!  used = .false.
  open(unit3, file='seep_test.dat')
! else
!  used = .true.
!  unit3=unit3+1
! end if
!end do

! force IMSL error messages to be written to the debug.dat file (unit=666)
call umach(-3,unit1)

! causes non-covergence error in DBETIN function to print error message
! but not stop execution
call erset(0,1,0)

 !!write(unit1,*) "method=1; begin calculation"
 !!write(unit1,*) " "

 ! random numbers passed to the DLL from GoldSim
 ! flow focus factor
 F=in(1)
 !!write(unit1,*) "flow focus factor  ", F

 ! random number for seepage fraction, seepage flow mean,
 ! and seepage flow standard deviation distributions
 R=in(2)
 !!write(unit1,*) "big 'R' random number  ", R

 ! infiltration state
 inf_master=in(3)
 !!write(unit1,*) "inf_master  ", inf_master
 !!write(unit1,*) " "

 ! seeds for random number generator
```

```
seed(1)=in(4)
seed(2)=in(5)

call random_seed(put=seed(1:2))

call ReadMasterFile

call ReadDistributionData ! read in percolation flux, seepage flow,
                                                   ! and seepage fraction data from files

nStart=0     ! initialize out() vector counter

do m=1,2     ! execute calculation for CSNF (m=1) and HLW (m=2)

 do n=1,5    ! exectute calculation for all five infiltration rate bins

  if (FileNames(m,inf_master,n) .ne. 'dummy') then
  write(unit1,*) 'm, inf_master, n ', m, inf_master, n
  write(unit1,*) FileNames(m,inf_master,n)
  write(unit1,*) " "
   call CountDataSets
   call ReadPercData
   call AllocateArrays
   call DoCalculations
   call GenerateOutput
   call DeallocateArrays
  else
   call NoDataOutput
  endif

 end do      ! end of loop through infiltration bins 1 to 5

end do              ! end of loop through CSNF and HLW

!!write(unit1,*) "finished with calculation do loops"

close(unit1)        ! close debug and output files
close(unit2)
close(unit3)

else if (method .eq. 99) then     ! ending dll

 !!write(unit1,*) "ending dll"

 close(unit1)        ! close debug and output files
 close(unit2)
 close(unit3)

else
 !!write(unit1,*) "bomb!!!"
 close(unit1)        ! close debug and output files
 close(unit2)
 close(unit3)
end if

! This is the end of the main program logic.
! Internal subroutines are beyond this point
contains
!**********************************************************************
!**********************************************************************
subroutine ReadMasterFile

!!write(unit1,*) "entering subroutine ReadMasterFile"
```

```
!!write(unit1,*) " "

open(unit=11, file='master.in')

do i=1,2      ! fuel type (CSNF, CDSP)
 !!write(unit1,*) " "
 do j=1,3     ! infiltration (low, mean, high)
 !!write(unit1,*) " "
  do k=1,5    ! infiltration bin (1,2,3,4,5)
   read(11,*) FileNames(i,j,k)
   !!write(unit1,*) FileNames(i,j,k)
  end do
 end do
end do

close(unit=11)

!!write(unit1,*) "exiting subroutine ReadMasterFile"
!!write(unit1,*) " "

end subroutine ReadMasterFile
!*********************************************************************
!*********************************************************************
! read in seepage flow and seepage fraction data from files
subroutine ReadDistributionData

!!write(unit1,*) "entering subroutine ReadDistributionData"

open(unit=11, file='SeepFlowMean.dat')

! read in the number of data sets (rows) in the file
read(11,*) nDistRows

! set the size of the seepage flow and
! seepage fraction parameter vectors
allocate(PercValue(1:nDistRows))

allocate(MeanSeepFlowMin(1:nDistRows))
allocate(MeanSeepFlowPeak(1:nDistRows))
allocate(MeanSeepFlowMax(1:nDistRows))

allocate(SdSeepFlowMin(1:nDistRows))
allocate(SdSeepFlowPeak(1:nDistRows))
allocate(SdSeepFlowMax(1:nDistRows))

allocate(SeepFracMin(1:nDistRows))
allocate(SeepFracPeak(1:nDistRows))
allocate(SeepFracMax(1:nDistRows))

! read in mean seepage flow data (triangle distribution)
!!write(unit1,*) "mean seepage flow data (triangle distribution)"
do i=1,nDistRows
 read(11,*) PercValue(i), MeanSeepFlowMin(i), &
           MeanSeepFlowPeak(i), MeanSeepFlowMax(i)
! write(unit1,1010) PercValue(i), MeanSeepFlowMin(i), &
!             MeanSeepFlowPeak(i), MeanSeepFlowMax(i)
 1010 format(F8.2, " ", E9.3, " ", E9.3, " ", E9.3)
end do
 !!write(unit1,*) " "
close(unit=11)

! read in std. dev. seepage flow data (triangle distribution)
open(unit=12, file='SeepFlowSD.dat')
```

```
! read in the number of data sets (rows) in the file
read(12,*) nDistRows
!!write(unit1,*) "std. dev. seepage flow data (triangle distribution)"
do i=1,nDistRows
 read(12,*) PercValue(i), SdSeepFlowMin(i), &
            SdSeepFlowPeak(i), SdSeepFlowMax(i)
! write(unit1,1010) PercValue(i), SdSeepFlowMin(i), &
!             SdSeepFlowPeak(i), SdSeepFlowMax(i)

end do
!!write(unit1,*) " "
close(unit=12)

! read in seepage fraction data (triangle distribution)
open(unit=13, file='SeepFrac.dat')

! read in the number of data sets (rows) in the file
read(13,*) nDistRows
!!write(unit1,*) "seepage fraction data (triangle distribution)"
do i=1,nDistRows
 read(13,*) PercValue(i), SeepFracMin(i), &
            SeepFracPeak(i), SeepFracMax(i)
! write(unit1,1010) PercValue(i), SeepFracMin(i), &
!             SeepFracPeak(i), SeepFracMax(i)
end do
!!write(unit1,*) " "
close(unit=13)

!!write(unit1,*) "exiting subroutine ReadDistributionData"
!!write(unit1,*) " "

end subroutine ReadDistributionData
!***********************************************************************
!***********************************************************************
subroutine CountDataSets

!!write(unit1,*) "entering subroutine CountDataSets"

infile=FileNames(m,inf_master,n)

! open the selected file
open(unit=60, file=infile)

nLocations=1

! read past 1st four rows of header information
do mm=1,4
 read(60,*) dummy1
end do

read(60,*) (dummy2(mm), mm=1,5), nTimes

! read past next six rows of header information
do mm=1,6
 read(60,*) dummy1
end do

! read past the 1st data set
do mm=1,nTimes
 read(60,*) dummy1
end do
```

```
! read through file until the end of the file is reached

do

! read the 1st row header information for the next data set
! if this read occurs at the end of the file, the 'eof' error
! causes the do loop to be exited
 read(60,*,end=100) (dummy2(mm), mm=1,5), nTimes

! if an 'eof' error did not occur, increment the locations counter
! and read through the given data set

 nLocations=nLocations+1

 do mm=1, (6+nTimes)
  read(60,*) dummy1
 end do

end do

100 continue ! line that the do-loop bails out to

! close the data file
close(unit=60)

!!write(unit1,*) " "
!!write(unit1,*) "# of spatial locations in data set ", nLocations
!!write(unit1,*) " "

!!write(unit1,*) "exiting subroutine CountDataSets"
!write(unit1,*) " "

end subroutine CountDataSets
!**********************************************************************
!**********************************************************************
subroutine ReadPercData

!write(unit1,*) "entering subroutine ReadPercData"

! open the selected file
open(unit=60, file=infile)

allocate(t(1:nTimes), percHis(1:nTimes,1:nLocations))
allocate(LocationWeight(1:nLocations))

do mm=1,5
 read(60,*) dummy1
end do

read(60,*) (dummy2(mm), mm=1,6), LocationWeight(1)

do mm=1,5
 read(60,*) dummy1
end do

do mm=1, nTimes
 read(60,*) t(mm), percHis(mm,1)
end do

do nn=2, nLocations

 read(60,*) dummy
 read(60,*) (dummy2(mm), mm=1,6), LocationWeight(nn)
```

```
 do mm=1,5
  read(60,*) dummy1
 end do

 do mm=1,nTimes
  read(60,*) t(mm), percHis(mm,nn)
 end do

end do

close(unit=60)

!write(unit1,*) "exiting subroutine ReadPercData"
!write(unit1,*) " "

end subroutine ReadPercData
!*********************************************************************
!*********************************************************************
subroutine AllocateArrays

!write(unit1,*) "entering subroutine AllocateArrays"

! set bounds on dynamic arrays
allocate(SeepAlwaysAvg(1:nTimes), SeepSometimesAvg(1:nTimes))
allocate(SeepFracOut(1:nTimes,1:nLocations))
allocate(SeepFlowOut(1:nTimes,1:nLocations))
allocate(FracYes(1:nLocations), FracNo(1:nLocations))
allocate(FracFlag(1:nLocations))

!write(unit1,*) "exiting subroutine AllocateArrays"
!write(unit1,*) " "

end subroutine AllocateArrays
!*********************************************************************
!*********************************************************************
subroutine DoCalculations

!write(unit1,*) "entering subroutine DoCalculations"
!write(unit1,*) " "

! calculate the seepage fraction and seepage flow
! for each percolation flux in the current fuel type perc flux bin

do k=1,nLocations                              ! loop through space

! generate random number used to evaluate beta distribution function
! for seepage flow

 call random_number(rn)
 !call random(rn)
 !write(unit1,*) "little 'r' random number  ", rn
 !write(unit1,*) " "

 do j=1,nTimes                                            ! loop through time

 !write(unit1,*) "j-th time in the k-th location"
 !write(unit1,*) "j, k  ", j, k
 !write(unit1,*) " "

  perc=percHis(j,k) ! grab percolation flux from percHis array
  !write(unit1,*) "perc  ", perc
```

```
! scale the percolation flux by flow focus factor
  perc=perc*F
  !write(unit1,*) "perc*F  ", perc
  !write(unit1,*)

  SeepFlow=0.0                    ! initialize SeepFlow to zero
  call SeepageFraction     ! calculate seepage fraction

! if seepage fraction is not equal to 0.0, then calculate
! the seepage flow
  if (SeepFrac .gt. 0.0) then
   !write(unit1,*) SeepFrac, " SeepFrac > 0.0"
   call SeepageFlow
  end if

  !write(unit1,*) SeepFrac, " SeepFrac"
  !write(unit1,*) SeepFlow, " SeepFlow"
  !write(unit1,*) " "

! store the seepage fraction and seepage flow values
  SeepFracOut(j,k)=SeepFrac
  SeepFlowOut(j,k)=SeepFlow

 end do
end do

call NeverSometimesAlwaysSeeps

call SeepageAveraging

!write(unit1,*) "exiting subroutine DoCalculations"
!write(unit1,*) " "

end subroutine DoCalculations
!*********************************************************************
!*********************************************************************
! subroutine that calculates the seepage fraction
subroutine SeepageFraction

!write(unit1,*) "entering subroutine SeepageFraction"
!write(unit1,*) " "

! set SeepFrac=0 if perc is <= 3.4 mm/yr
if (perc .le. PercValue(2)) then
 SeepFrac=0.0
 !write(unit1,*) perc, PercValue(2), " perc <= PercValue(2)"
 !write(unit1,*) SeepFrac, " SeepFrac = 0.0"

! set SeepFrac=1 if perc is >= xxxx.yy mm/yr
else if (perc .ge. PercValue(nDistRows)) then
 SeepFrac=1.0
 !write(unit1,*) perc, PercValue(nDistRows), " perc <= PercValue(nDistRows)"
 !write(unit1,*) SeepFrac, " SeepFrac = 1.0"

else

! if perc lies inside the bounds of the response surface then find the
! min, peak, and max triangle distribution values via interpolation
 call Interp(perc, PercValue, SeepFracMin, SeepFracPeak, &
             SeepFracMax, nDistRows, SFcMin, SFcPeak, SFcMax)

! caluclate the seepage fraction from the triangle distribution and the
! random parameter 'R' passed from GoldSim
```

```
 call TriDist(SFcMin, SFcPeak, SFcMax, R, SeepFrac)

end if

!write(unit1,*) "calculated SeepFrac ", SeepFrac

SeepFrac = SeepFrac/F      ! divide the seepage fraction
                                        ! by the flow focus factor

!write(unit1,*) "SeepFrac/F ", SeepFrac
!write(unit1,*)

!!!! if rn>=SeepFrac then set SeepFrac and SeepFlow equal to zero
if (rn .ge. SeepFrac) then
!!!if (rn .ge. 2.0) then
 !write(unit1,*) "rn >= SeepFrac ", rn, SeepFrac
 SeepFrac=0.0
 SeepFlow=0.0
 !write(unit1,*) "SeepFrac and SeepFlow equal 0.0 ", SeepFrac, SeepFlow
else
 !write(unit1,*) "rn < SeepFrac ", rn, SeepFrac
!!!rnp=rn
 rnp=rn/SeepFrac    ! normalize rn by the seepage fraction
  !write(unit1,*) "normalized rn (rn/SeepFrac) ", rnp
end if

!write(unit1,*) "exiting subroutine SeepageFraction"
!write(unit1,*) " "

end subroutine SeepageFraction
!*********************************************************************
!*********************************************************************
! subroutine that calculates the seepage flow
subroutine SeepageFlow

!write(unit1,*) "entering subroutine SeepageFlow"

! calculate the minimum, peak, and maximum mean seepage flow
! and seepage flow std. dev. from the triangle distributions

call Interp(perc, PercValue, MeanSeepFlowMin, MeanSeepFlowPeak,  &
             MeanSeepFlowMax, nDistRows, SFwMeanMin, SFwMeanPeak,  &
                     SFwMeanMax)
call Interp(perc, PercValue, SdSeepFlowMin, SdSeepFlowPeak,  &
             SdSeepFlowMax, nDistRows, SFwSdMin, SFwSdPeak,  &
                     SFwSdMax)

! evaluate the mean seepage flow and seepage flow std. dev. from
! the triangle distributions for those parameters
call TriDist(SFwMeanMin, SFwMeanPeak, SFwMeanMax, R, SFwMean)
call TriDist(SFwSdMin, SFwSdPeak, SFwSdMax, R, SFwSD)

if ( (SFwMean .eq. 0.0) .and. (SFwDS .eq. 0.0)) then
 SeepFlow = 0.0

else

! calculate the seepage flow from its beta distribution
 call BetaDist(SFwMean, SFwSD, rnp, SeepFlow)

end if

!write(unit1,*) "exiting subroutine SeepageFlow"
```

```
!write(unit1,*) " "

end subroutine SeepageFlow
!*************************************************************************
!*************************************************************************
subroutine NeverSometimesAlwaysSeeps

! initialize yes/no seepage fraction flags
do k=1,nLocations
 FracYes(k)=-1             ! no seepage
 FracNo(k)=-1       ! always seepage
end do

! determine if locations never seep, sometimes seep, or always seep
do k=1,nLocations                 ! loop through space
 do j=1,nTimes                                ! loop through time

 !write(unit1,*) " "
 !write(unit1,*) k, j, "     k-th location, j-th time"
 !write(unit1,*) SeepFracOut(j,k), "  SeepFracOut(j,k)"
 !write(unit1,*) " "

  if (SeepFracOut(j,k) .eq. 0.0) then
   FracNo(k)=1             ! location has at least one time with no seepage
   !write(unit1,*) "location has at least one time with no seepage"
   !write(unit1,*) FracNo(k), " FracNo(k)"

  else
   FracYes(k)=1     ! location has at least one time with seepage
   !write(unit1,*) "location has at least one time with seepage"
   !write(unit1,*) FracYes(k), " FracYes(k)"

  end if
 end do
end do

!write(unit1,*) " "
!write(unit1,*) "done calculating seepage fractions and seepage flows"
!write(unit1,*) "now determine always, sometimes and never seepage flow fractions"
!write(unit1,*) " "

do k=1,nLocations
 !write(unit1,*) " "

 if ((FracYes(k) .eq. 1) .and. (FracNo(k) .eq. -1)) then
  FracFlag(k)=1            ! always seeps
  !write(unit1,*) k, "  location always seeps"
  !write(unit1,*) FracNo(k), FracYes(k), " FracNo(k), FracYes(k)"
  !write(unit1,*) FracFlag(k), " FracFlag(k)=1"

 else if ((FracYes(k) .eq. -1) .and. (FracNo(k) .eq. 1)) then
  FracFlag(k)=3            ! never seeps
  !write(unit1,*) k, "  location never seeps"
  !write(unit1,*) FracNo(k), FracYes(k), " FracNo(k), FracYes(k)"
  !write(unit1,*) FracFlag(k), " FracFlag(k)=3"

 else if ((FracYes(k) .eq. 1) .and. (FracNo(k) .eq. 1)) then
  FracFlag(k)=2            ! sometimes seeps
  !write(unit1,*) k, "  location sometimes seeps"
  !write(unit1,*) FracNo(k), FracYes(k), " FracNo(k), FracYes(k)"
  !write(unit1,*) FracFlag(k), " FracFlag(k)=2"

 else
```

```
  FracFlag(k)=-1          ! error
  !write(unit1,*) " error!"
 end if
end do

end subroutine NeverSometimesAlwaysSeeps
!**********************************************************************
!**********************************************************************
subroutine SeepageAveraging

! calculate average (spatial) seepage flow at each time step
do j=1,nTimes
 nAlways=0                ! initialize "always seeps" counter
 nSometimes=0       ! initialize "sometimes seeps" counter

 WeightAlways=0.0
 WeightSometimes=0.0

 SeepAlwaysAvg(j)=0
 SeepSometimesAvg(j)=0

! sum up flows for "always" and "sometimes" seeps locations
 do k=1,nLocations
  if (FracFlag(k) .eq. 1) then
   nAlways=nAlways+1
   WeightAlways=WeightAlways+LocationWeight(k)
   SeepAlwaysAvg(j) = SeepAlwaysAvg(j)  &
                    + SeepFlowOut(j,k)*LocationWeight(k)
  else if (FracFlag(k) .eq. 2) then
   nSometimes=nSometimes+1
   WeightSometimes=WeightSometimes+LocationWeight(k)
   SeepSometimesAvg(j) = SeepSometimesAvg(j)  &
                       + SeepFlowOut(j,k)*LocationWeight(K)
  end if
 end do

! divide sum of "always" and "sometimes" flows by their respective
! number of locations
 if (nAlways .eq. 0) then
  SeepAlwaysAvg(j)=0
 else
 SeepAlwaysAvg(j)=SeepAlwaysAvg(j)/WeightAlways
 end if

 if (nSometimes .eq. 0) then
  SeepSometimesAvg(j)=0
 else
  SeepSometimesAvg(j)=SeepSometimesAvg(j)/WeightSometimes
 end if

end do

end subroutine SeepageAveraging
!**********************************************************************
!**********************************************************************
subroutine GenerateOutput

!write(unit1,*) "entering subroutine GenerateOutput"

! first 1-d table
out(nStart+1)=1                  ! 1-d table
write(unit2,*) out(nStart+1)
out(nStart+2)=nTimes                   ! # of rows
```

```
write(unit2,*) out(nStart+2)

! independent variable values (times)
do j=1,nTimes
 out(nStart+2+j)=t(j)
  write(unit2,*) out(nStart+2+j)
end do

! dependent variable values (always seeps flow rates)
do j=1,nTimes
 out(nStart+2+nTimes+j)=SeepAlwaysAvg(j)
 write(unit2,*) out(nStart+2+nTimes+j)
end do

!write(unit1,*) "1st 1-d table written"

! fraction of locations (always seeps)
out(nStart+2+nTimes+nTimes+1)=nAlways/nLocations
!write(unit1,*) out(nStart+2+nTimes+nTimes+1), " fraction always seeps"
write(unit2,*) out(nStart+2+nTimes+nTimes+1)

! second 1-d table
 out(nStart+2+nTimes+nTimes+2)=1              ! 1-d table
 write(unit2,*) out(nStart+2+nTimes+nTimes+2)
 out(nStart+2+nTimes+nTimes+3)=nTimes         ! # of rows
 write(unit2,*) out(nStart+2+nTimes+nTimes+3)

 do j=1,nTimes
  out(nStart+2+nTimes+nTimes+3+j)=t(j)
  write(unit2,*) out(nStart+2+nTimes+nTimes+3+j)
 end do

 do j=1,nTimes
  out(nStart+2+nTimes+nTimes+3+nTimes+j)=SeepSometimesAvg(j)
  write(unit2,*) out(nStart+2+nTimes+nTimes+3+nTimes+j)
 end do

 out(nStart+2+nTimes+nTimes+3+nTimes+nTimes+1)=nSometimes/nLocations
 write(unit2,*) out(nStart+2+nTimes+nTimes+3+nTimes+nTimes+1)

 nStart=nStart+2+nTimes+nTimes+3+nTimes+nTimes+1

!write(unit1,*) "exiting subroutine GenerateOutput"

end subroutine GenerateOutput
!**********************************************************************
!**********************************************************************
subroutine DeallocateArrays

!write(unit1,*) "entering subroutine DeallocateArrays"

! deallocate dynamic arrays
 deallocate(t, percHis)
 deallocate(SeepAlwaysAvg, SeepSometimesAvg)
 deallocate(SeepFracOut)
 deallocate(SeepFlowOut)
 deallocate(FracYes, FracNo)
 deallocate(FracFlag)
 deallocate(LocationWeight)

!write(unit1,*) "exit subroutine DeallocateArrays"
!write(unit1,*) " "
```

```
end subroutine DeallocateArrays
!*********************************************************************
!*********************************************************************
subroutine NoDataOutput

!write(unit1,*) "entering subroutine NoDataOutput"

! first 1-d table
out(nStart+1)=1                          ! 1-d table
write(unit2,*) out(nStart+1)
out(nStart+2)=2                          ! # of rows
write(unit2,*) out(nStart+2)

out(nStart+3)=0.0
write(unit2,*) out(nStart+3)
out(nStart+4)=1.0E+06
write(unit2,*) out(nStart+4)

out(nStart+5)=0.0
write(unit2,*) out(nStart+5)
out(nStart+6)=0.0
write(unit2,*) out(nStart+6)

! fraction of locations
out(nStart+7)=0.0
write(unit2,*) out(nStart+7)

! second 1-d table
out(nStart+8)=1                          ! 1-d table
write(unit2,*) out(nStart+8)
out(nStart+9)=2                          ! # of rows
write(unit2,*) out(nStart+9)

out(nStart+10)=0.0
write(unit2,*) out(nStart+10)
out(nStart+11)=1.0E+06
write(unit2,*) out(nStart+11)

out(nStart+12)=0.0
write(unit2,*) out(nStart+12)
out(nStart+13)=0.0
write(unit2,*) out(nStart+13)

out(nStart+14)=0
write(unit2,*) out(nStart+14)

nStart=nStart+14

!write(unit1,*) "exiting subroutine NoDataOutput"
!write(unit1,*) " "

end subroutine NoDataOutput
!*********************************************************************
!*********************************************************************
! This is the end of the seepdll subroutine. External subroutines are
! defined beyond this point
end subroutine seepdll

!*********************************************************************
!*********************************************************************
! subroutine that interpolates between points on a given
! response surface
subroutine Interp(ind, IndData, DepMin, DepPeak, DepMax, &
```

```
                    nRows, Min, Peak, Max)

! variable listing
!
! ind                 - independent variable
! IndData()           - range of data for independent variable
! DepMin()            - range of dependent variable minimum values
! DepPeak()           - range of dependent variable peak values
! DepMax()            - range of dependent variable maximum values
! nRows                      - number of rows in response surface data set
! Min                 - interpolated minimum value
! Peak                - interpolated peak value
! Max                 - interpolated maximum value

integer(4) nRows
real(8) IndData(nRows), DepMin(nRows)
real(8) DepPeak(nRows), DepMax(nRows)
real(8) ind, Min, Peak, Max

!write(666,*) "entering Interp subroutine"

! for independent variable values below the range of the response surface,
! set the dependent variables equal to the "floor" values
if (ind .le. IndData(1)) then
 Min=DepMin(1)
 Peak=DepPeak(1)
 Max=DepMax(1)

! for independent variable values above the range of the response surface,
! linearly extrapolate the values of the  dependent variables
elseif (ind .ge. IndData(nRows)) then

 Min=DepMin(nRows)  &
          + (ind-IndData(nRows))/(IndData(nRows)-IndData(nRows-1))   &
                 *(DepMin(nRows)-DepMin(nRows-1))
 Peak=DepPeak(nRows)  &
          + (ind-IndData(nRows))/(IndData(nRows)-IndData(nRows-1))   &
                 *(DepPeak(nRows)-DepPeak(nRows-1))
 Max=DepMax(nRows)  &
          + (ind-IndData(nRows))/(IndData(nRows)-IndData(nRows-1))   &
                 *(DepMax(nRows)-DepMax(nRows-1))

else

 do i=1,nRows-1      ! loop through the range of the independent variable

! if the independent variable is between the i-th and i-th plus 1 values
! in the independent variable range, interpolate the Min, Peak and Max
! values
  if ((ind .ge. IndData(i)) .and. (ind .lt. IndData(i+1))) then

   Min=DepMin(i)  &
           + (ind-IndData(i))/(IndData(i+1)-IndData(i))   &
                 *(DepMin(i+1)-DepMin(i))
   Peak=DepPeak(i)  &
           + (ind-IndData(i))/(IndData(i+1)-IndData(i))   &
                 *(DepPeak(i+1)-DepPeak(i))
   Max=DepMax(i)  &
           + (ind-IndData(i))/(IndData(i+1)-IndData(i))   &
                 *(DepMax(i+1)-DepMax(i))

  end if
 end do
```

```
end if

!write(666,*) " Min, Peak, Max"
!write(666,*) Min, Peak, Max
!write(666,*) "exiting Interp subroutine"
!write(666,*) " "

end subroutine Interp
!************************************************************************
!************************************************************************
subroutine TriDist(Min, Peak, Max, R, solution)

! variable listing

! Min                 - minimum value of triangle distribution
! Peak                - peak value of triangle distribution
! Max                 - maximum value of triangle distribution
! R                          - random number at which distribution is evaluated
! solution            - result returned by the subroutine
! sol1, sol2 - intermidiate results
! A, B, C             - quadratic equation coefficients

real(8) Min, Peak, Max, R, solution, sol1, sol2
real(8) A, B, C

!write(666,*) "entering subroutine TriDist"
!write(666,*) " "

! case 1
if ( (Peak .eq. Min) .and.  &
     (Max .eq. Min)  .and.  &
        (Min .eq. 0.0) )then

! write(666,*) "case 1"
 solution = 0.0

! case 2
elseif ( (Peak .eq. Min) .and. (Peak .le. Max) ) then
 A=1.0
 B=-2*Max
 C=(Max**2)+((R-1)*(Max-Peak)*(Max-Min))
 solution=(-B-(B**2 - 4*A*C)**0.5)/(2*A)
! write(666,*) "case 2"
! write(666,*) A, B, C, "  A, B, C"
! write(666,*) solution, " solution"


! case 4
elseif ( (Min .lt. Peak) .and. (Peak .eq. Max) ) then
 A=1.0                                                     ! the distribution
 B=-2*Min
 C=(Min**2)-(R*(Peak-Min)*(Max-Min))
 solution=(-B+(B**2 - 4*A*C)**0.5)/(2*A)
! write(666,*) "case 4"
! write(666,*) A, B, C, "  A, B, C"
! write(666,*) solution, " solution"

! case 3
elseif ( (Peak .gt. Min) .and. (Peak .lt. Max) ) then

 A=1.0
 B=-2*Min
```

```
 C=(Min**2)-(R*(Peak-Min)*(Max-Min))
 sol1 = (-B+(B**2 - 4*A*C)**0.5)/(2*A)
! write(666,*) A, B, C, "  A, B, C"
! write(666,*) sol1, " sol1"
! write(666,*) " "

 A=1.0
 B=-2*Max
 C=(Max**2)+((R-1)*(Max-Peak)*(Max-Min))
 sol2=(-B-(B**2 - 4*A*C)**0.5)/(2*A)
! write(666,*) A, B, C, "  A, B, C"
! write(666,*) sol2, " sol2"
! write(666,*) " "

 if (sol1 .le. Peak) then
  solution=sol1
!  write(666,*) "case 3a"

 else
  solution=sol2
!  write(666,*) "case 3b"

 end if

! write(666,*) solution, " solution"

end if

!write(666,*) " "
!write(666,*) "exiting subroutine TriDist"
!write(666,*) " "

end subroutine TriDist
!*********************************************************************
!*********************************************************************
subroutine BetaDist(Mean, SD, rnp, solution)

use numerical_libraries          ! accesses IMSL routines

! variable listing

! Mean                - beta distribution mean
! SD                  - beta distribution std. dev.
! rnp                 - random number at which beta distribution is evaluated
! solution            - result from evaluation of beta distribution

! UB                  - beta distribution upper bound
! p                          - beta distribution shape factor
! q                          - beta distribution shape factor

real(8) Mean, SD, rnp, UB, p, q, solution

!write(666,*) "  entering BetaDist subroutine"

! upper-bound is [mean + 10*SD]
UB=Mean + 10*SD

! shape parameter "p" for the beta distribution
p=((UB**2)*(UB/Mean - 1)/((SD**2)*(UB/Mean)**3)) - Mean/UB

! shape parameter "q" for the beta distribution
q=p*(UB/Mean - 1)
```

```
!write(666,*) "  ", Mean, SD, rnd, UB

! calculate the seepage flux from the beta distribution
! and the random number "rnp"

solution=dbetin(rnp, p, q)*UB

!icode=iercd()
itype=n1rty(1)
!if (itype .eq. 4) then
! write(888,*) "error level 4 -- fatal"
! write(888,*) rnp,  " rnp"
! write(888,*) Mean, " Mean"
! write(888,*) SD,   " SD"
! write(888,*) UB,   " UB"
! write(888,*) solution, " solution"
!end if

!write(666,*) solution, " BetaDist solution"
!write(666,*) " "

!write(666,*) "  exiting BetaDist subroutine"
!write(666,*) " "

end subroutine BetaDist
!*************************************************************************
```

II.3.3 COMPUTER LISTING OF TEST DATA INPUT AND OUTPUT

Test case inputs and outputs are listed for each test in Section 2 under the heading describing the specific test case.

ATTACHMENT III

SOILEXP SOIL ERROSION DLL

III.1 SOFTWARE ROUTINE IDENTIFICATION

III.1.1 SOFTWARE NAME AND VERSION NUMBER

soilexp V1.0

III.1.2 NAME AND VERSION OF INDUSTRY STANDARD SOFTWARE UNDER WHICH ROUTINE WAS DEVELOPED

This routine was developed using MS Visual Fortran Professional Edition 6.0.A.

III.1.3 SRR DOCUMENT IDENTIFICATION NUMBER:

N/A

III.1.4 SRR MEDIA NUMBER (IF APPLICABLE):

N/A

III.2 DESCRIPTION AND TESTING

III.2.1 OVERVIEW

The disruptive events analysis, discussed in section 6.3.9, considers two types of igneous events, intrusive and volcanic. Intrusive events are those where a magma dike is postulated to intersect portions of the repository releasing waste products which are transported by groundwater. Volcanic events are those where a rising magma dike forms a conduit to the surface resulting in a volcanic event with waste products entrained in the ash which is dispersed downwind and deposited on the ground.

For the analysis volcanic events are assumed to occur at regular intervals. The radionuclide concentration at deposition points increases following each volcanic event and is decreased by radionuclide decay and soil removal in the time period between volcanic events.

The soilexp routine calculates the cumulative soil removal factor used to calculate radionuclide concentration at deposition points over the life of the repository. The soilexp routine receives input from the TSPA software, calculates the cumulative soil removal factor for the time interval, and passes the result back to the TSPA software.

III.2.2 INPUTS

Inputs to **soilexp** for the TSPA software are listed in Table III.1.

Table III-1. Soilexp Inputs from the TSPA Software

Parameter	Variable Name
Time(yr)	etime
Period between Volcanic events (yr)	VolcanoPeriod
Time at which volcanic eruptions began (yr)	VolcanoStartTime
Soil removal rate (l/yr)	K

These values are stored in the TSPA software and passed to **soilexp** in an input vector, in().

III.2.3 DESCRIPTION OF SOFTWARE ROUTINE INCLUDING THE EXECUTION ENVIRONMENT

III.2.3.1 DEVELOPMENT AND EXECUTION ENVIRONMENT

This **soilexp** routine was developed using MS Visual Fortran Professional Edition 6.0.A. The routine is compiled as a DLL and may be called by any software capable of utilizing DLL's running in a Windows 95/NT operating environment.

III.2.3.2 MAIN PROGRAM

The basic structure of **soilexp** is comprised of an **if-then-else** construct where the statement blocks within it are executed depending on the value of the method variable passed by GoldSIM to the dll. The **if-then-else** construct is structured (in pseudo code; with comments in *italics*) as follows:

```
if method=0 then
  initialize the dll

elseif method=2 then
  report the version of the dll

elseif method=3 then
  report the number of input and output arguments

elseif method=1 then
  perform the dll's calculations
  (more detailed pseudo code for this statement block is given below)

elseif method=99
  terminate the dll
```

Upon **method**=1 the dll begins executing the statement block that perform the calculations. The pseudo code for this statement block is as follows with explanatory comments in italics:

get inputs from GoldSim

n=etime/VolcanoPeriod + 1
calculate number of periods (time steps)

nStart=VolcanoStartTime/VolcanPeriod + 1
set index at which volcanoes start

do I=1,n
loop through the number of time periods

timeVec(i)=(i-1)*VolcanoPeriod
calculate the timeVec values

If (etime .lt. volcanoStartTime) then
Determine whether or not the elapsed time is less than the time at which the first volcanic event occurs

sum =0
return a zero for the soil removal factor

else
calculate the soil removal factor for the given time step

do i=nStart,n
Loop through the time steps

sum=sum+exp(-k*(timeVec(n)-timeVec(i)))
Calculate soil removal factor

out(i)=sum
assign result to output vector

write the current number of time steps and the time vector to soilexp.ou'

```
end subroutine soilexp
```

III.2.6 DESCRIPTION OF TEST RESUTLTS

Soilexp was tested by calling the DLL from the TSPA software and comparing the output to results obtained using an EXCEL spreadsheet replicating the calculations performed by the routine. The input values and outputs obtained from soilexp and the EXCEL spreadsheet are presented below.

Visual comparison of the Total Soil Removal Factor obtained from both cases demonstrate they agree thus validating the performance of the soilexp routine.

Input values used to test the routine are listed in Table III.2

Table III-2 Test Case Input Values For Soilexp Validation

Parameter	Variable Name	Test Case Input Value
Time(yr)	etime	80
Period between Volcanic events (yr)	VolcanoPeriod	20
Time at which volcanic eruptions began (yr)	VolcanoStartTime	20
Soil removal rate (l/yr)	k	0.0046667

The test case output obtained from soilexp when called by the TSPA software is presented in Table III-3.

Table III-3 Test Case Output From The Soilexp Routine

	Time (yr)	Total Soil_Removal_Factor
0		0
20		1
40		1.9109
60		2.7406
80		3.4964

The excel spreadsheet used to validate the soilexp routine is presented in Figure III.1.

DtVolcano [yr]	k [-]					
20	0.0046667					
n-th time	Time [yr]					
1	0	0	0	0	0	0
2		20	20	20	20	20
3			40	40	40	40
4				60	60	60
5					80	80
6						100
n-th time	n-th soil removal factor					
1	0	0	0	0	0	0
2		1	0.9108899	0.8297205	0.755784	0.6884361
3			1	0.9108899	0.8297205	0.755784
4				1	0.9108899	0.8297205
5					1	0.9108899
6						1
Total soil removal factor	0	1	1.9108899	2.7406104	3.4963945	4.1848306

Figure III-1 Excel Spreadsheet Used To Validate The Soilexp Routine

III.2.7 RANGE OF INPUT PARAMETER VALUES FOR WHICH RESULTS WERE VERIFIED

The soilexp routine was validated over a range of 0 to 80 years. Since the routine calculates solutions to a simple algebraic expression modeling periodic buildup followed by periods of exponential removal it is considered valid for the range of real numbers. For results to be meaningful parametric inputs must be realistic and consistent with the real world situations being modeled.

III.2.8 LIMITATIONS ON SOFTWARE ROUTINE APPLICATIONS OR VALIDITY

This software routine was developed specifically for the TSPA model to evaluate seepage into emplacement drifts within the Yucca Mountain repository. It was written and compiled as a direct linked library (DLL) file. As such it can be called by any program written to utilize DLLs. For any output to be meaningful the routine would also have to have access to appropriate input data correctly describing volcanic event frequencies and soil removal rates for the geological area being evaluated.

III.3. SUPPORTING INFORMATION

III.3.1 DIRECTORY LISTING OF EXECUTABLE AND DATA FILES

Executable and data files must reside in the same directory.

III.3.2 COMPUTER LISTING OF SOURCE CODE

```
subroutine soilexp(method, state, in, out)

! attibute statements needed for the dll
!DEC$ ATTRIBUTES dllexport,c :: soilexp
!DEC$ ATTRIBUTES value :: method
!DEC$ ATTRIBUTES reference :: state
!DEC$ ATTRIBUTES reference :: in
!DEC$ ATTRIBUTES reference :: out

! VolcanoPeriod              - volcano period [ yr]
! i                          - time step index
! in()                       - inputs from GoldSim to the dll
! k                          - soil removal rate [ 1/yr]
! n                          - number of time steps
! out()                      - outputs from the dll to GoldSim
! sum                        - series solution for soil removal factor
! timeVec()                  - time vector
! unit1                      - unit number of file 'soilexp.ou'


! define static variables
real(8) etime, k, VolcanoPeriod, VolcanoStartTime
integer(4) i, n, nStart, unit1

! define dynamic variables
real(8), allocatable :: timeVec(:)

real(8) in(*), out(*)            ! define in and out array for dll

if (method.eq.0) then            ! initialize
 return

elseif (method.eq.2) then  ! report version
 out(1) = 1.0
 return

elseif (method.eq.3) then  ! report arguments
 out(1) = 4         ! four inputs
 out(2) = 1         ! one output
 return

elseif (method.eq.1) then  ! calculation

 unit1=100   ! set unit1 unit number
```

```
! input from GoldSim to the DLL
 etime=in(1)                       ! time
 VolcanoPeriod=in(2)               ! volcano period [ yr]
 VolcanoStartTime=in(3)            ! volcano start time [ yr]
 k=in(4)                                  ! soil removal rate [ 1/yr]

 n=etime/VolcanoPeriod+1

 allocate(timeVec(1:n))  ! set the element length of timeVec

 ! set index at which volcanos start
 nStart=VolcanoStartTime/VolcanoPeriod+1

 ! calculate timeVec values
 do i=1,n
  timeVec(i)=(i-1)*VolcanoPeriod
 end do

 if (etime .lt. VolcanoStartTime) then

  out(1)=0.0  ! no volcano, return a zero for the soil removal factor

 else

  sum = 0.0          ! initialize sum to 0.0

  ! series solution for soil removal factor
  do i=nStart,n
   sum=sum+exp(-k* (timeVec(n)-timeVec(i)))
  end do

  out(1)=sum  ! soil removal factor passed back to GoldSim

 end if

 ! write the current number of time steps and the time vector to 'unit1'
 open(unit1, file='soilexp.ou')
 write(unit1,*) n
 do i=1,n
  write(unit1,*) timeVec(i)
 end do
 close(unit1)

else if (method .eq. 99) then      ! ending dll

close(200)

else

end if

end subroutine soilexp
```

III.3.3 COMPUTER LISTING OF TEST DATA INPUT AND OUTPUT

Test case inputs and outputs are listed in section 2.

ATTACHMENT IV

SOFTWARE ROUTINE T2_BINNING V. 1.0

SOFTWARE ROUTINE T2_BINNING V. 1.0

T2 BINNING v.1.0 is used to generate repository release bins based on surface infiltration information. It operates on a PC in a Windows NT environment. This section provides sample input and output files, as well as a verification test of the software routine.

INPUT FILE

T2_BINNING requires an input file called 'binning-input.txt,' which contains the names of associated files for the routine. Below is a listing of this input file:

```
sr-repo-nodes.txt         !Name of file specifying Repository Elements (DTN:
SN9910T0581699.002; same file as 'SR-repo-nodes').

mpa_pch1.v1               !Name of file containing Element and Connection
information (DTN: LB990701233129.001; same file as '3d2kpa_pc1.mesh').

pa_glal1.dat              !Name of file containing infiltration information
(DTN: LB990801233129.007; same as 'pa_glaL1.dat')

glal1_EBS.zone           !Name of output file
```

The name of the file containing infiltration information and the name of the output file are different for the three runs considered in this analysis (low, medium, and upper infiltration scenarios in the glacial-transition climate). The files shown in the sample input file are for the low infiltration, glacial-transition climate ('glal1'). The medium and upper infiltration scenarios are denoted by 'glam1' and 'glau1,' respectively.

SAMPLE OUTPUT FILE

A sample output file is provided below, and it corresponds to the low infiltration, glacial-transition climate. The name of the file is 'glal1_EBS.zone." There are a total of 5 repository release bins numbered from bin 501 through 505 for fracture nodes and bin 601 through 605 for the corresponding matrix nodes. Nodes within each bin are listed by node index. Since in all TSPA simulations, radionuclide are released only from fracture nodes, the matrix nodes are not used in FEHM. "glal1_EBS.zone"

```
501    #     BIN 1 = 0 TO 3 MM/YR INF. FRACTURE NODES
 nnum
   216
     31351    31394    31439    31482    31525    31571    31615    31658
     31703    31748    31793    31839    31883    31927    31970    32014
     32059    32102    32145    32189    32233    32276    32323    32367
     32409    32453    32537    32629    32713    32804    32846    32888
     32979    33062    33108    33195    33278    33320    33408    33452
     33494    33535    33622    33707    33747    33839    33966    34056
     34184    34270    34314    34356    34397    34487    34531    34574
     34619    34664    34706    34793    34838    34881    34920    35011
     35054    35094    35185    35266    35437    35567    35605    35648
     35693    35735    35773    35816    35861    35947    35991    36117
     36160    36203    36289    36332    36455    36500    36586    36631
     36675    36717    36758    36803    36845    36886    36927    36971
     37014    37055    37182    37221    37348    37387    37474    37515
     37554    37641    37681    37720    37807    37849    37893    37979
```

```
     38020     38062     38190     38230     38314     38356     38396     38479
     38519     38599     38641     38682     38720     38760     38801     38843
     38885     38924     38967     39009     39046     39090     39134     39176
     39218     39262     39306     39346     39388     39474     39514     39556
     39637     39676     39718     39759     39798     39881     39924     39967
     40010     40956     40999     41041     41083     41125     41166     41209
     41252     41294     41335     41377     41419     41463     41508     41552
     41595     41640     41684     41728     41773     41818     41863     41908
     41952     41996     42039     42082     42126     42170     42212     42256
     42300     42343     42387     42465     42622     42701     42780     42820
     42858     42896     42934     43009     44312     44385     44424     44465
     44504     44545     44585     44623     46287     46395     46574     46664
     46704     47070     47385     47426     47466     47506     47549     47639
502   #       BIN 2 = 3 TO 10 MM/YR INF. FRACTURE NODES
nnum
       053
     32495     32583     32671     32760     32934     33021     33153     33235
     33360     33665     33882     33926     34100     34142     34747     34965
     35140     35227     35311     35356     35398     35481     35526     35905
     36031     36072     36244     36373     36413     36543     37096     37140
     37265     37307     37431     37599     37764     37937     38107     38148
     38275     38441     38560     39432     39598     39841     42426     42542
     42582     42661     42740     46620     47594
503   #       BIN 3 = 10 TO 20 MM/YR INF. FRACTURE NODES
nnum
       004
     33575     34009     34442     42504
504   #       BIN 4 = 20 TO 60 MM/YR INF. FRACTURE NODES
nnum
       002
     33792     34224
505   #       BIN 5 = >60 MM/YR INF. FRACTURE NODES
nnum
       000
601   #       BIN 1 = 0 TO 3 MM/YR INF. MATRIX NODES
nnum
       216
     79015     79058     79103     79146     79189     79235     79279     79322
     79367     79412     79457     79503     79547     79591     79634     79678
     79723     79766     79809     79853     79897     79940     79987     80031
     80073     80117     80201     80293     80377     80468     80510     80552
     80643     80726     80772     80859     80942     80984     81072     81116
     81158     81199     81286     81371     81411     81503     81630     81720
     81848     81934     81978     82020     82061     82151     82195     82238
     82283     82328     82370     82457     82502     82545     82584     82675
     82718     82758     82849     82930     83101     83231     83269     83312
     83357     83399     83437     83480     83525     83611     83655     83781
     83824     83867     83953     83996     84119     84164     84250     84295
     84339     84381     84422     84467     84509     84550     84591     84635
     84678     84719     84846     84885     85012     85051     85138     85179
     85218     85305     85345     85384     85471     85513     85557     85643
     85684     85726     85854     85894     85978     86020     86060     86143
     86183     86263     86305     86346     86384     86424     86465     86507
     86549     86588     86631     86673     86710     86754     86798     86840
     86882     86926     86970     87010     87052     87138     87178     87220
     87301     87340     87382     87423     87462     87545     87588     87631
     87674     88620     88663     88705     88747     88789     88830     88873
     88916     88958     88999     89041     89083     89127     89172     89216
     89259     89304     89348     89392     89437     89482     89527     89572
     89616     89660     89703     89746     89790     89834     89876     89920
     89964     90007     90051     90129     90286     90365     90444     90484
     90522     90560     90598     90673     91976     92049     92088     92129
     92168     92209     92249     92287     93951     94059     94238     94328
```

```
     94368     94734     95049     95090     95130     95170     95213     95303
602    #       BIN 2 = 3 TO 10 MM/YR INF. MATRIX NODES
nnum
       053
     80159     80247     80335     80424     80598     80685     80817     80899
     81024     81329     81546     81590     81764     81806     82411     82629
     82804     82891     82975     83020     83062     83145     83190     83569
     83695     83736     83908     84037     84077     84207     84760     84804
     84929     84971     85095     85263     85428     85601     85771     85812
     85939     86105     86224     87096     87262     87505     90090     90206
     90246     90325     90404     94284     95258
603    #       BIN 3 = 10 TO 20 MM/YR INF. MATRIX NODES
nnum
       004
     81239     81673     82106     90168
604    #       BIN 4 = 20 TO 60 MM/YR INF. MATRIX NODES
nnum
       002
     81456     81888
605    #       BIN 5 = >60 MM/YR INF. MATRIX NODES
nnum
       000
```

VERIFICATION OF T2_BINNING V.1.0 BY HAND CALCULATION AND VISUAL INSPECTION.

The nodes in each of the bins have been spot-checked and verified in the following manner: (1) select a node and determine the corresponding element in the TOUGH2 mesh file by "counting" lines in the ELEME card (the nodes are sequentially ordered and matrix elements are skipped); (2) find the corresponding infiltrating element in the TOUGH2 input file by searching for the last three characters of the given element in the GENER card (which contains the infiltrating elements). The last three characters are unique to each column of elements in the TOUGH2 model. The infiltration rate is given for each element in GENER as kg/s; (3) determine the infiltrating area by searching in the CONNE card for the three characters preceded by a 'TP' (which stands for top boundary). The area associated with the element connected to the 'TPXXX' element is the infiltrating area; and (4) calculate the infiltration in mm/year by dividing the infiltration mass flow rate in kg/s by the liquid density (1000 kg/m^3) and the infiltrating area. Determine that the infiltration corresponds to the bin that the node is placed in. The following are examples of these spot checks:

<u>Spot Check #1: Node 34397 (Outlined Fracture Node in Bin 1 of Sample Output File)</u>

(1) Node 34397 corresponds to the TOUGH2 element, 'Foh72,' in the ELEME card of the TOUGH2 mesh file, 'mpa_pch1.v1.' This element is on the 68,794th line of the mesh file (34,397 × 2 = 68,794), which accounts for matrix elements on every other line.

(2) The corresponding infiltration element in the 'pa_glal1.dat' file is 'Fah72' and the infiltration rate for this element is 0.8609E-03 kg/s.

(3) The infiltrating area corresponding to the connection area for element 'Fah72' is found in the CONNE card of the TOUGH2 MESH FILE, 'mpa_pch1.v1' to be 0.3042E+05 m^2.

(4) The infiltration in mm/year is calculated as follows: 0.8609E-03 kg/s ÷1000 kg/m^3 ÷ 0.3042E+05 m^2 × 1000 mm/m x 3.15576E07 s/year = 0.8931 mm/year. Thus, the correct bin for this node is bin 1 (0-3 mm/year). This verifies the correct placement of this node.

Spot Check #2: Node 33665 (Outlined Fracture Node in Bin 2 of Sample Output File)

(1) Node 33665 corresponds to the TOUGH2 element, 'Fqh55,' in the ELEME card of the TOUGH2 mesh file, 'mpa_pch1.v1.' This element is on the 67,330th line of the mesh file (33665 × 2 = 67,330), which accounts for matrix elements on every other line.

(2) The corresponding infiltration element in the 'pa_glal1.dat' file is 'Fah55' and the infiltration rate for this element is 0.3041E-02 kg/s.

(3) The infiltrating area corresponding to the connection area for element 'Fah55' is found in the CONNE card of the TOUGH2 MESH FILE, 'mpa_pch1.v1' to be 0.2190E+05 m2.

(4) The infiltration in mm/year is calculated as follows: 0.3041-02 kg/s ÷1000 kg/m^3 ÷ 0.2190E+05 m^2 × 1000 mm/m x 3.15576E07 s/year = 4.382 mm/year. Thus, the correct bin for this node is bin 2 (3-10 mm/year). This verifies the correct placement of this node.

Spot Check #3: Node 42504 (Outlined Fracture Node in Bin 3 of Sample Output File)

(1) Node 42504 corresponds to the TOUGH2 element, 'FqC22,' in the ELEME card of the TOUGH2 mesh file, 'mpa_pch1.v1.' This element is on the 85,008th line of the mesh file (42504 × 2 = 85,008), which accounts for matrix elements on every other line.

(2) The corresponding infiltration element in the 'pa_glal1.dat' file is 'FaC22' and the infiltration rate for this element is 0.1183E-02 kg/s.

(3) The infiltrating area corresponding to the connection area for element 'FaC22' is found in the CONNE card of the TOUGH2 MESH FILE, 'mpa_pch1.v1' to be 0.2690E+04 m2.

(4) The infiltration in mm/year is calculated as follows: 0.1183-02 kg/s ÷1000 kg/m3 ÷ 0.2690E+04 m^2 × 1000 mm/m x 3.15576E07 s/year = 13.88 mm/year. Thus, the correct bin for this node is bin 3 (10-20 mm/year). This verifies the correct placement of this node.

These sample hand calculations verify that T2_BINNING v. 1.0 is performing correctly for the range of inputs used.

LISTING OF T2_BINNING V. 1.0

```
c     program t2_binning_v1.f
c_____________________________________________________________________
c  This program will extract infiltration maps from TOUGH2 input files
c  and mesh files belonging to the LBNL 3-D TOUGH2 UZ flow model.  The
c  program will then convert infiltration values from kg/s to mm/year
c  based on the vertical connection area between the top boundary (TP)
c  and the infiltrating element.  The converted infiltration values are
c  then binned into 5 different bins corresponding to the following
c  values: 0-3, 3-10, 10-20, 20-60, and greater than 60 mm/yr.  This is
```

```
c  done for the columns containing repository elements only.  The
c  repository columns are derived from a user-prescribed file that
c  contains repository elements.
c
c     M.J.Kelley,   January, 2000
c      SANDIA NATIONAL LABORATORIES
c      GEOHYDROLOGY DEPARTMENT, ORG. 6115
c      BOX 5800, MS 0735
c      ALBUQUERQUE, NM 87185-0735
c      BUILDING 823; ROOM 20099
c      (505) 284-6566
c      (505) 844-4426 (FAX)
c      mkelle@sandia.gov
c
c_______________________________________________________________________
c23456789012345678901234567890123456789012345678901234567890123456789012
c
c      !      INPUT/OUTPUT FILES
c
c      !             10 INPUT - 'binning-input.txt' LIST OF ALL I/O FILES
c      !             15 INPUT - FILE CONTAINING REPOSITORY LOCATION INFO.
c      !             20 INPUT - FILE WITH T2 ELEME AND CONNE INFORMATION
c      !             25 INPUT - FILE CONTAINING INFILTRATION INFORMATION
c
c      !             100 OUTPUT - FILE CONTAINING ZONE INFORMATION
c_______________________________________________________________________
c23456789012345678901234567890123456789012345678901234567890123456789012
c
c      !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=!
c      !#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*!
c      !                                                                 !
c      !                          MAIN PROGRAM                           !
c      !                                                                 !
c      !#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*!
c      !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=!

       PROGRAM t2_binning_v1

       IMPLICIT NONE

c      !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c      !                DECLARE GENERAL VARIABLES FOR CODE            !
c      !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      INTEGER I,J,K,F,NUMELEM,NMAX,C1,C2,C3,C4,C5
       INTEGER ANODE(1000),MANODE(1000),BNODE(1000),MBNODE(1000)
       INTEGER CNODE(1000),MCNODE(1000),DNODE(1000),MDNODE(1000)
       INTEGER ENODE(1000),MENODE(1000),NODENUMBER(1000)

       DOUBLE PRECISION FLUXX,SECPERYEAR,MMPERMETER
       DOUBLE PRECISION AREAX,FLUX(1000),REPOFLUX(1000),AREA(1000)
       DOUBLE PRECISION CUBICMETERPERKG,TERM1,TERM2

       CHARACTER*20 REPOELEMENTS,INFILTRATION,ELEMECONNE,ZONEOUTPUT
       CHARACTER*10 BLOCK2,BLOCK
       CHARACTER*5 REPOELEM(1000),ELEMINF
       CHARACTER*5 CORRELEME,COMPAREVALUE

c      !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c      !                DECLARE CONVERSIONS                           !
c      !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

       SECPERYEAR=31557600
```

```
      MMPERMETER=1000
      CUBICMETERPERKG=0.001
c     The assumption of 1000kg/m^3 was used for the above parameter
c
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !                   SET UP INPUT AND OUTPUT FILES                     !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      WRITE(*,*)''
      WRITE(*,*)'STARTING GENERAL INPUT/SETUP FOR BINNING....'

      OPEN(10,FILE='binning-input.txt')

      READ(10,9000) REPOELEMENTS
      READ(10,9000) ELEMECONNE
      READ(10,9000) INFILTRATION
      READ(10,9000) ZONEOUTPUT

9000  FORMAT(A)

      CLOSE(10)

      WRITE(*,*)'
      ....DONE'
      WRITE(*,*)''

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !                   READ IN NAMES OF REPOSITORY ELEMENTS    !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      WRITE(*,*)'READING REPOSITORY ELEMENTS....'
      WRITE(*,*)''

      OPEN(15,FILE=REPOELEMENTS)

      READ(15,*)NUMELEM

            DO J=1,NUMELEM
                  READ(15,'(A5)')REPOELEM(J)
            ENDDO

      CLOSE(15)

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !     FIND CORRESPONDING NODE NUMBERS FOR REPO ELEMENTS     !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      WRITE(*,*)'   CALCULATING NODE NUMBER....'
      WRITE(*,*)''

      OPEN(20,FILE=ELEMECONNE)

      F=1
      I=1
      READ(20,*)BLOCK

200   READ(20,9100)CORRELEME

            IF(CORRELEME(1:1).EQ.'M') GOTO 200

                  IF(CORRELEME.EQ.REPOELEM(F)) THEN

                        NODENUMBER(F)=I
```

```
                        F=F+1

                    ENDIF

                    IF(CORRELEME(1:1).EQ.'T') GOTO 220

                    I=I+1

        GOTO 200

 9100   FORMAT(A5)

 220    CONTINUE

        NMAX=I-1

c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c       !     FIND AREA CORRESPONDING TO THE FRACTURE/TOP BOUNDARY !
c       !     CONNECTION                                            !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

        WRITE(*,*)'           FINDING AREA....'
        WRITE(*,*)''

 300    READ(20,9200)BLOCK2

                    IF(BLOCK2(1:5).NE.'CONNE') GOTO 300

        J=1

 305    COMPAREVALUE=REPOELEM(J)(3:5)

 310    READ(20,9300)BLOCK2,AREAX

                IF(BLOCK2(3:5).EQ.COMPAREVALUE) THEN

                    IF(BLOCK2(6:7).EQ.'TP') THEN

                        AREA(J)=AREAX
                        J=J+1

                        IF(J.EQ.NUMELEM+1) GOTO 320

                    ENDIF
                ENDIF

        GOTO 305

 9200   FORMAT(A10)
 9300   FORMAT(A10,40X,E10.4)

 320    CONTINUE

        CLOSE(20)

c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c       !READ IN INFILTRATION INFORMATION FOR EACH CORRESPONDING    !
c       !REPOSITORY ELEMENT                                         !
c       !                                                           !
c       !NOTE: these values are in kg/s and need to be converted    !
c       !     to mm/yr                                              !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
```

```
      WRITE(*,*)'                 READING INFILTRATION VALUES....'
      WRITE(*,*)''

      OPEN(25,FILE=INFILTRATION)

400   READ(25,9400)BLOCK

            IF(BLOCK(1:5).NE.'GENER') GOTO 400

      J=1

410   COMPAREVALUE=REPOELEM(J)(3:5)

      READ(25,9500)ELEMINF,FLUXX

            IF(ELEMINF(3:5).EQ.COMPAREVALUE) THEN

                  FLUX(J)=FLUXX
                  J=J+1

                  IF(J.EQ.NUMELEM+1) GOTO 420

            ENDIF
      GOTO 410

9400  FORMAT(A10)
9500  FORMAT(A5,35X,E10.4)

420   CONTINUE

      CLOSE(25)

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !CONVERT INFILTRATION INFORMATION FROM KG/S TO MM/YR FOR      !
c     !EACH OF THE REPOSITORY ELEMENTS                               !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      WRITE(*,*)'                       CONVERTING INFILTRATION....'
      WRITE(*,*)''

500   DO I=1,NUMELEM

            TERM1=FLUX(I)*(1/AREA(I))
            TERM2=CUBICMETERPERKG*MMPERMETER*SECPERYEAR

            REPOFLUX(I)=TERM1*TERM2

      ENDDO

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !CREATE BINS FOR ABOVE INFORMATION & PREPARE OUTPUT           !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      WRITE(*,*)'                 CREATING BINS FOR INFILTRATION....'
      WRITE(*,*)''

      C1=0
      C2=0
      C3=0
      C4=0
      C5=0

600   DO I=1,NUMELEM
```

```
          IF(REPOFLUX(I).LE.3) THEN
          ANODE(C1+1)=NODENUMBER(I)
          MANODE(C1+1)=(NODENUMBER(I)+NMAX)
          C1=C1+1
          GOTO 610
          ENDIF

          IF(REPOFLUX(I).LE.10) THEN
          BNODE(C2+1)=NODENUMBER(I)
          MBNODE(C2+1)=(NODENUMBER(I)+NMAX)
          C2=C2+1
          GOTO 610
          ENDIF

          IF(REPOFLUX(I).LE.20) THEN
          CNODE(C3+1)=NODENUMBER(I)
          MCNODE(C3+1)=(NODENUMBER(I)+NMAX)
          C3=C3+1
          GOTO 610
          ENDIF

          IF(REPOFLUX(I).LE.60) THEN
          DNODE(C4+1)=NODENUMBER(I)
          MDNODE(C4+1)=(NODENUMBER(I)+NMAX)
          C4=C4+1
          GOTO 610
          ENDIF

          IF(REPOFLUX(I).GT.60) THEN
          ENODE(C5+1)=NODENUMBER(I)
          MENODE(C5+1)=(NODENUMBER(I)+NMAX)
          C5=C5+1
          GOTO 610
          ENDIF

610   CONTINUE

      ENDDO

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !    WRITE RESULTS TO OUTPUT FILE. THIS FILE IS THE *.ZONE !
c     !    FILE REQUIRED FOR FEHM                                 !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      WRITE(*,*)'         WRITING RESULTS TO OUTPUT FILE....'
      WRITE(*,*)''

700   OPEN(100,FILE=ZONEOUTPUT)

710   WRITE(100,*)'501     #      BIN 1 = 0 TO 3 MM/YR INF. FRACTURE NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C1
      IF(C1.EQ.0) GOTO 720
      WRITE(100,9700)(ANODE(I),I=1,C1)

720   WRITE(100,*)'502     #      BIN 2 = 3 TO 10 MM/YR INF. FRACTURE NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C2
      IF(C2.EQ.0) GOTO 730
      WRITE(100,9700)(BNODE(I),I=1,C2)

730   WRITE(100,*)'503     #      BIN 3 = 10 TO 20 MM/YR INF. FRACTURE NODES'
```

```
      WRITE(100,*)'nnum'
      WRITE(100,9600)C3
      IF(C3.EQ.0) GOTO 740
      WRITE(100,9700)(CNODE(I),I=1,C3)

740   WRITE(100,*)'504    #      BIN 4 = 20 TO 60 MM/YR INF. FRACTURE NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C4
      IF(C4.EQ.0) GOTO 750
      WRITE(100,9700)(DNODE(I),I=1,C4)

750   WRITE(100,*)'505    #      BIN 5 = >60 MM/YR INF. FRACTURE NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C5
      IF(C5.EQ.0) GOTO 760
      WRITE(100,9700)(ENODE(I),I=1,C5)

760   WRITE(100,*)'601    #      BIN 1 = 0 TO 3 MM/YR INF. MATRIX NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C1
      IF(C1.EQ.0) GOTO 770
      WRITE(100,9700)(MANODE(I),I=1,C1)

770   WRITE(100,*)'602    #      BIN 2 = 3 TO 10 MM/YR INF. MATRIX NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C2
      IF(C2.EQ.0) GOTO 780
      WRITE(100,9700)(MBNODE(I),I=1,C2)

780   WRITE(100,*)'603    #      BIN 3 = 10 TO 20 MM/YR INF. MATRIX NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C3
      IF(C3.EQ.0) GOTO 790
      WRITE(100,9700)(MCNODE(I),I=1,C3)

790   WRITE(100,*)'604    #      BIN 4 = 20 TO 60 MM/YR INF. MATRIX NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C4
      IF(C4.EQ.0) GOTO 800
      WRITE(100,9700)(MDNODE(I),I=1,C4)

800   WRITE(100,*)'605    #      BIN 5 = >60 MM/YR INF. MATRIX NODES'
      WRITE(100,*)'nnum'
      WRITE(100,9600)C5
      IF(C5.EQ.0) GOTO 810
      WRITE(100,9700)(MENODE(I),I=1,C5)


9600  FORMAT(I10.3)
9700  FORMAT(I10.5,I10.5,I10.5,I10.5,I10.5,I10.5,I10.5,I10.5)

810   CLOSE(100)

      WRITE(*,*)'  BINNING COMPLETE....'
      WRITE(*,*)''

      STOP
      END
```

ATTACHMENT V

SOFTWARE ROUTINE WT_BINNING V. 1.0

ATTACHMENT V.

SOFTWARE ROUTINE WT_BINNING V. 1.0

Software Routine WT_BINNING v. 1.0 was used to generate UZ radionuclide collect bins at the UZ-SZ interface. The program runs on a PC in the Windows NT operating environment. This software routine (WT_BINNING v.1.0) prompts the user for the name of the file containing the coordinates of the nodes in the UZ model. For this analysis, the file 'fm_glam1.grid' was used (DTN: SN9910T0581699.002 [126110]). This file is large, so only the related lines used for verification are shown:

```
coor
  47664
       1    169398.60     236623.64       1626.10
       2    169398.60     236623.64       1606.47
       3    169398.60     236623.64       1569.84
       *    *********     *********       *******
       *    *********     *********       *******
       *    *********     *********       *******
     300    169126.08     234549.08        795.20
       *    *********     *********       *******
       *    *********     *********       *******
       *    *********     *********       *******
     520    172726.29     234098.81        755.25
       *    *********     *********       *******
       *    *********     *********       *******
       *    *********     *********       *******
```

The first line, 'coor', is the macro name. The second line lists the number of fracture nodes. The subsequent lines list the node number and the x-, y-, and z-coordinates. The routine then evaluates whether each node is below a user-prescribed water-table elevation (850 m in this analysis) and assigns it to one of the four bins (Figure 105). In the case, an elevation of 0 is the input, the code will extract nodes at the bottom of the model. This case was not used, thus, not verified, in the analysis since our water table was set at an elevation of 850 m. A portion of the output file is shown below:

```
701    #  BIN 1 = NSP Easting(m)<171200 and NSP Northing(m)>233590. Frac Nodes
 nnum
      1979
     00033     00034     00035     00299     00300     00385     00386     00414
     00415     00416     00489     00587     00588     00914     00915     00916
     00943     00967     00968     00969     00970     00971     00972     01167
     01168     01222     01223     01224     01225     01311     01312     01341
     01342     01343     01399     01400     01401     01599     01600     01624
```

702 # BIN 2 = NSP Easting(m)>171200 and NSP Northing(m)>233590. Frac Nodes

```
 nnum
      1745
     00115     00116     00117     00135     00136     00137     00138     00139
     00140     00166     00167     00168     00169     00170     00171     00172
     00173     00174     00175     00198     00199     00200     00201     00202
     00229     00230     00231     00232     00233     00234     00235     00318
     00319     00320     00321     00322     00351     00352     00353     00354
     00355     00356     00357     00469     00470     00514     00515     00516
```

00517	00518	00519	00520	00521	00547	00548	00549
00550	00551	00552	00553	00554	00555	00608	00609

The fracture nodes are placed into bins 701-704 and the corresponding matrix nodes are placed into bins 801-804 (not shown). The placement of these nodes can easily be checked by visual inspection and comparison to the input grid file containing node coordinates.

VERIFICATION OF WT_BINNING V. 1.0 BY HAND CALCULATION AND VISUAL INSPECTION

One node is randomly picked from collect bins 701 and 702, respectively. The picked nodes are checked for node coordinates to verify that the selected nodes do fall below water table (850 m) within the designed collect bins (Figure 105).

Spot check #1: Node 300 from bin 701. The listed coordinate for node 300 in grid file fm_glam1.grid (DTN: SN9910T0581699.002 [126110]) is (169126.08 (easting), 234549.08(northing), 795.20(elevation)). As it is clear that this node is below water table and within the defined collecting bin 701 (Figure 105).

Spot check #2: Node 520 from bin 702. The listed coordinate for node 520 in grid file fm_glam1.grid (DTN: SN9910T0581699.002 [126110]) is (172726.29 (easting), 234098.81(northing), 755.25(elevation)). Obviously, the selected node is below the water table of 850 m and within the defined collect bin 702 (Figure 105).

The verification showed that the software routine WT_BINNING v. 1.0 performed correctly as designed.

LISTING OF WT_BINNING V. 1.0

```
c     program WT_binning.f
c__________________________________________________________________________
c  This program will create bins for the source regions for the
c  Saturated Zone Model.
c
c     M.J.Kelley,    January, 2000
c      SANDIA NATIONAL LABORATORIES
c      GEOHYDROLOGY DEPARTMENT, ORG. 6115
c      BOX 5800, MS 0735
c      ALBUQUERQUE, NM 87185-0735
c      BUILDING 823; ROOM 20099
c      (505) 284-6566
c      (505) 844-4426 (FAX)
c      mkelle@sandia.gov
c
c__________________________________________________________________________
c23456789012345678901234567890123456789012345678901234567890123456789012
c
c     !       INPUT/OUTPUT FILES
c
c     !              10 INPUT - INPUT FILE CONTAINING NODE COORDINATES
c
c     !              100 OUTPUT - FILE CONTAINING ZONE INFORMATION
c__________________________________________________________________________
```

```
c2345678901234567890123456789012345678901234567890123456789012
c
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=!
c       !#* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* !
c       !                                                                !
c       !                     MAIN PROGRAM                               !
c       !                                                                !
c       !#* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* #* !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=!

        PROGRAM WT_binning

        IMPLICIT NONE

c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c       !      DECLARE GENERAL VARIABLES FOR CODE                    !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

       INTEGER I,J,K,F,NUMNODES,NMAX,C1,C2,C3,C4
        INTEGER BIN1(50000),MBIN1(50000),BIN2(50000),MBIN2(50000)
        INTEGER BIN3(50000),MBIN3(50000),BIN4(50000),MBIN4(50000)
        INTEGER NODENUMBER(50000)

        DOUBLE PRECISION WTELEV,X(50000),Y(50000),Z(50000)

        CHARACTER*20 NODEFILE,HEADING,OUTPUT

c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c       !      SET UP COUNTERS FOR BINNING                           !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

        C1=0
        C2=0
        C3=0
        C4=0

c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c       !READ IN NAMES OF COORDINATE FILE.  PROVIDE THE PROGRAM      !
c       !WITH THE PRESCRIBED WATER TABLE ELEVATION FOR BINNING       !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

        WRITE(*,*)'What is the name of the Grid file containing the
      &coordinate information?'
        READ(*,9000)NODEFILE
        WRITE(*,*)''
        WRITE(*,*)''
        WRITE(*,*)'What is the water table elevation?'
        WRITE(*,*)'(0 = Only the very bottom layer of model)'
        READ(*,*)WTELEV
        WRITE(*,*)''
        WRITE(*,*)''
        WRITE(*,*)'What is the name of the output file?'
        READ(*,9000)OUTPUT

c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c       !READ IN NODE NUMBERS AND ASSOCIATED COORDINATES             !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

        OPEN(10,STATUS='OLD',FILE=NODEFILE)

100     READ(10,9000)HEADING
        READ(10,*)NUMNODES
```

```
      DO I=1,NUMNODES
             READ(10,9100)NODENUMBER(I),X(I),Y(I),Z(I)
      ENDDO

      CLOSE(10)

9000  FORMAT(A20)
9100  FORMAT(I8,4X,E9.4,4X,E9.4,4X,E9.4)

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !IF THE WATER TABLE ELEVATION IS OTHER THAN THE VERY         !
c     !BOTTOM MODEL LAYER, GOTO BINNING SECTION THAT EXPLICITLY    !
c     !CONSIDERS THE CHANGE IN WATER TABLE ELEVATION               !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

      IF(WTELEV.GT.0) GOTO 500

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !IF THE WATER TABLE ELEVATION IS THE VERY BOTTOM MODEL       !
c     !LAYER, KEEP ONLY THAT LAYER IN THE BINS                     !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

200   DO I=1,NUMNODES

            IF(NODENUMBER(I).EQ.NUMNODES) GOTO 210
            IF(Z(I).GT.Z(I+1)) GOTO 250

210               IF(X(I).LT.171200.AND.Y(I).GT.233590) THEN
                  BIN1(C1+1)=NODENUMBER(I)
                  MBIN1(C1+1)=(NODENUMBER(I)+NUMNODES)
                  C1=C1+1
                  ENDIF

220               IF(X(I).GT.171200.AND.Y(I).GT.233590) THEN
                  BIN2(C2+1)=NODENUMBER(I)
                  MBIN2(C2+1)=(NODENUMBER(I)+NUMNODES)
                  C2=C2+1
                  ENDIF

230               IF(X(I).LT.171200.AND.Y(I).LT.233590) THEN
                  BIN3(C3+1)=NODENUMBER(I)
                  MBIN3(C3+1)=(NODENUMBER(I)+NUMNODES)
                  C3=C3+1
                  ENDIF

240               IF(X(I).GT.171200.AND.Y(I).LT.233590) THEN
                  BIN4(C4+1)=NODENUMBER(I)
                  MBIN4(C4+1)=(NODENUMBER(I)+NUMNODES)
                  C4=C4+1
                  ENDIF

250   CONTINUE

      ENDDO

      GOTO 600

c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c     !IF THE WATER TABLE ELEVATION IS GIVEN BY THE USER, KEEP     !
c     !ALL NODES BELOW THAT PRESCRIBED ELEVATION IN THE BINS       !
c     !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

500   DO I=1,NUMNODES
```

```
              IF(NODENUMBER(I).EQ.NUMNODES) GOTO 510
              IF(Z(I).GT.WTELEV) GOTO 550

510                  IF(X(I).LT.171200.AND.Y(I).GT.233590) THEN
                     BIN1(C1+1)=NODENUMBER(I)
                     MBIN1(C1+1)=(NODENUMBER(I)+NUMNODES)
                     C1=C1+1
                     ENDIF

520                  IF(X(I).GT.171200.AND.Y(I).GT.233590) THEN
                     BIN2(C2+1)=NODENUMBER(I)
                     MBIN2(C2+1)=(NODENUMBER(I)+NUMNODES)
                     C2=C2+1
                     ENDIF

530                  IF(X(I).LT.171200.AND.Y(I).LT.233590) THEN
                     BIN3(C3+1)=NODENUMBER(I)
                     MBIN3(C3+1)=(NODENUMBER(I)+NUMNODES)
                     C3=C3+1
                     ENDIF

540                  IF(X(I).GT.171200.AND.Y(I).LT.233590) THEN
                     BIN4(C4+1)=NODENUMBER(I)
                     MBIN4(C4+1)=(NODENUMBER(I)+NUMNODES)
                     C4=C4+1
                     ENDIF

550     CONTINUE

        ENDDO

c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!
c       !WRITE RESULTS TO OUTPUT FILE. THIS FILE IS THE *.ZONE        !
c       !FILE REQUIRED FOR FEHM                                        !
c       !-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-!

600     OPEN(100,FILE=OUTPUT)

610     WRITE(100,*)'701     #  BIN 1 = NSP Easting(m)<171200 and
       &NSP Northing(m)>233590. Frac Nodes'
        WRITE(100,*)'nnum'
        WRITE(100,9400)C1
        WRITE(100,9500)(BIN1(I),I=1,C1)

620     WRITE(100,*)'702     #  BIN 2 = NSP Easting(m)>171200 and
       &NSP Northing(m)>233590. Frac Nodes'
        WRITE(100,*)'nnum'
        WRITE(100,9400)C2
        WRITE(100,9500)(BIN2(I),I=1,C2)

630     WRITE(100,*)'703     #  BIN 3 = NSP Easting(m)<171200 and
       &NSP Northing(m)<233590. Frac Nodes'
        WRITE(100,*)'nnum'
        WRITE(100,9400)C3
        WRITE(100,9500)(BIN3(I),I=1,C3)

640     WRITE(100,*)'704     #  BIN 4 = NSP Easting(m)>171200 and
       &NSP Northing(m)<233590. Frac Nodes'
        WRITE(100,*)'nnum'
        WRITE(100,9400)C4
        WRITE(100,9500)(BIN4(I),I=1,C4)
```

```
650   WRITE(100,*)'801     #  BIN 1 = NSP Easting(m)<171200 and
     &NSP Northing(m)>233590. Matrix Nodes'
      WRITE(100,*)'nnum'
      WRITE(100,9400)C1
      WRITE(100,9500)(MBIN1(I),I=1,C1)

660   WRITE(100,*)'802     #  BIN 2 = NSP Easting(m)>171200 and
     &NSP Northing(m)>233590. Matrix Nodes'
      WRITE(100,*)'nnum'
      WRITE(100,9400)C2
      WRITE(100,9500)(MBIN2(I),I=1,C2)

670   WRITE(100,*)'803     #  BIN 3 = NSP Easting(m)<171200 and
     &NSP Northing(m)<233590. Matrix Nodes'
      WRITE(100,*)'nnum'
      WRITE(100,9400)C3
      WRITE(100,9500)(MBIN3(I),I=1,C3)

680   WRITE(100,*)'804     #  BIN 4 = NSP Easting(m)>171200 and
     &NSP Northing(m)<233590. Matrix Nodes'
      WRITE(100,*)'nnum'
      WRITE(100,9400)C4
      WRITE(100,9500)(MBIN4(I),I=1,C4)

9400  FORMAT(I10.3)
9500  FORMAT(I10.5,I10.5,I10.5,I10.5,I10.5,I10.5,I10.5,I10.5)

690   CLOSE(100)

      STOP
      END
```

ATTACHMENT VI

SOFTWARE ROUTINE MAKEPTRK V. 2.0

SOFTWARE ROUTINE MAKEPTRK V. 2.0

Name and version of software routine: MAKEPTRK v. 2.0

Name and version of software to develop routine: FORTRAN77 on Sun Ultra Sparc running Sun OS 5.7

DESCRIPTION

The software routine MAKEPTRK v. 2.0 provides input information required by the code FEHM to define transport models and nodal assignments. The required inputs for this routine are files containing the TOUGH2 'ROCKS' card and the 'ELEME' card. The large size of these files prohibits showing them here, but they can be readily downloaded from the Technical Data Management System. The two sample files that are used in this verification are DTN: LB990801233129.009 [118717] ('pa_glam1.dat' which contains the 'ROCKS' card) and DTN: LB990701233129.001 [106785] ('3d2kpa_pc1.mesh' which contains the 'ELEME' card). Another user-specified input file can be used to enter the names of these TOUGH2 files and other relevant parameters. It is important to note that the transport parameters (e.g., Kd's, diffusion coefficients, aperture parameters, etc.) that are listed in this input file for MAKEPTRK v. 2.0 are not used in the output file that is eventually used by FEHM in TSPA calculations. The only output used from this routine are the listing of the materials as identified in the TOUGH2 ROCKS card and the nodal assignments to the corresponding materials. The following shows a sample input file for MAKEPTRK v. 2.0. The text beneath the dashed line is copied from the source file to define the requested information on each line of the input file.

SAMPLE INPUT FILE FOR MAKEPTRK V. 2.0

```
/home/ckho/tspaSR-LA/LBNL_flow/glacial_transition/lb990801233129.009/pa_glam1.dat
/home/ckho/tspaSR-LA/LBNL_grid/lb990701233129.001/3d2kpa_pc1.mesh
glam1_new.mptr
1
6
-1
-2
-3
0.
20.
-4
1.
0.01
-----------
      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ROCKS card?'
      read(*,*) rocks
      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ELEME and CONNE cards?'
      read(*,*) mesh
      write(*,*)'What would you like to name the output file?'
      read(*,*) out
      write(*,*)'What transport mechanisms apply for the matrix?'
      write(*,*)'1 - advection only (no dispersion or matrix diff)'
      write(*,*)'2 - advection and dispersion (no matrix diff)'
      write(*,*)'3 - advection and matrix diff (no dispersion)'
      write(*,*)'4 - advection, dispersion, and matrix diff'
```

```
write(*,*)'6 - advection, dispersion, and matrix diff with '
write(*,*)'    finite fracture spacing'
read(*,*) iflagm
write(*,*)'What transport mechanisms apply for the fracture?'
write(*,*)'1 - advection only (no dispersion or matrix diff)'
write(*,*)'2 - advection and dispersion (no matrix diff)'
write(*,*)'3 - advection and matrix diff (no dispersion)'
write(*,*)'4 - advection, dispersion, and matrix diff'
write(*,*)'6 - advection, dispersion, and matrix diff with '
write(*,*)'    finite fracture spacing'
read(*,*) iflagf
write(*,*)'What is the Kd (cc/g) for vitric units?'
read(*,*)xkdv
write(*,*)'What is the Kd (cc/g) for zeolitic units?'
read(*,*)xkdz
write(*,*)'What is the Kd (cc/g) for devitrified units?'
read(*,*)xkdd
write(*,*)'What is the matrix dispersivity (m)?'
read(*,*) dispm
write(*,*)'What is the fracture dispersivity (m)?'
read(*,*) dispf
write(*,*)'What is the molecular diffusion coefficient?'
read(*,*) do
write(*,*)'What is the retardation factor for fracture'
write(*,*)'sorption?  (1 = no fracture sorption)'
read(*,*) rdfrac
write(*,*)'What is the residual fracture saturation?'
read(*,*) slr
```

VERIFICATION

The output generated from this software routine can be visually inspected and verified. The following describes examples of these visual inspections.

To verify that the listing and ordering of the materials in the output file from MAKEPTRK v. 2.0 is correct, we can compare the listing of the materials shown below (output from MAKEPTRK v. 2.0) and the materials listed in DTN: LB990801233129.009 [118717] ('pa_glam1.dat' which contains the 'ROCKS' card). The first material shown in 'pa_glam1.dat' is 'tcwM1,' which is the same as the material identifier for the first line in the output below (as indicated in the last column). The tenth material in 'pa_glam1.dat' is 'tswM1,' which is the same as the material identifier for the tenth line below. The last material (excluding the boundaries 'topbd' and 'botbd') in 'pa_glam1.dat' is 'chnFf,' which is the same as the last material listed below. The 96 total materials listed below correspond to the total number of materials in 'pa_glam1.dat' (excluding the boundaries). It is emphasized again that the actual values of the transport parameters in the first 11 columns are not used from MAKEPTRK v. 2.0.

Partial Output from MAKEPTRK v. 2.0 Showing Material Listing

```
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.253E+00      0 0.000E+00  0.000E+00 #  1 tcwM1
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.820E-01      0 0.000E+00  0.000E+00 #  2 tcwM2
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.203E+00      0 0.000E+00  0.000E+00 #  3 tcwM3
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.387E+00      0 0.000E+00  0.000E+00 #  4 ptnM1
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.439E+00      0 0.000E+00  0.000E+00 #  5 ptnM2
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.254E+00      0 0.000E+00  0.000E+00 #  6 ptnM3
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.411E+00      0 0.000E+00  0.000E+00 #  7 ptnM4
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.499E+00      0 0.000E+00  0.000E+00 #  8 ptnM5
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.492E+00      0 0.000E+00  0.000E+00 #  9 ptnM6
1     -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.530E-01      0 0.000E+00  0.000E+00 # 10 tswM1
```

1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.157E+00	0	0.000E+00	0.000E+00	#	11	tswM2
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.154E+00	0	0.000E+00	0.000E+00	#	12	tswM3
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.110E+00	0	0.000E+00	0.000E+00	#	13	tswM4
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.131E+00	0	0.000E+00	0.000E+00	#	14	tswM5
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.112E+00	0	0.000E+00	0.000E+00	#	15	tswM6
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.940E-01	0	0.000E+00	0.000E+00	#	16	tswM7
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.370E-01	0	0.000E+00	0.000E+00	#	17	tswM8
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.173E+00	0	0.000E+00	0.000E+00	#	18	tswM9
1	-1	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.273E+00	0	0.000E+00	0.000E+00	#	19	ch1Mv
1	-1	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.345E+00	0	0.000E+00	0.000E+00	#	20	ch2Mv
1	-1	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.345E+00	0	0.000E+00	0.000E+00	#	21	ch3Mv
1	-1	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.345E+00	0	0.000E+00	0.000E+00	#	22	ch4Mv
1	-1	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.345E+00	0	0.000E+00	0.000E+00	#	23	ch5Mv
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.288E+00	0	0.000E+00	0.000E+00	#	24	ch1Mz
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.331E+00	0	0.000E+00	0.000E+00	#	25	ch2Mz
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.331E+00	0	0.000E+00	0.000E+00	#	26	ch3Mz
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.331E+00	0	0.000E+00	0.000E+00	#	27	ch4Mz
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.331E+00	0	0.000E+00	0.000E+00	#	28	ch5Mz
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.266E+00	0	0.000E+00	0.000E+00	#	29	ch6Mz
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.325E+00	0	0.000E+00	0.000E+00	#	30	pp4Mz
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.303E+00	0	0.000E+00	0.000E+00	#	31	pp3Md
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.263E+00	0	0.000E+00	0.000E+00	#	32	pp2Md
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.280E+00	0	0.000E+00	0.000E+00	#	33	pp1Mz
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.115E+00	0	0.000E+00	0.000E+00	#	34	bf3Md
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.259E+00	0	0.000E+00	0.000E+00	#	35	bf2Mz
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.115E+00	0	0.000E+00	0.000E+00	#	36	tr3Md
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.259E+00	0	0.000E+00	0.000E+00	#	37	tr2Mz
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.370E-01	0	0.000E+00	0.000E+00	#	38	pcM38
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.173E+00	0	0.000E+00	0.000E+00	#	39	pcM39
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.288E+00	0	0.000E+00	0.000E+00	#	40	pcM1z
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.331E+00	0	0.000E+00	0.000E+00	#	41	pcM2z
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.331E+00	0	0.000E+00	0.000E+00	#	42	pcM5z
1	-2	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.266E+00	0	0.000E+00	0.000E+00	#	43	pcM6z
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.325E+00	0	0.000E+00	0.000E+00	#	44	pcM4p
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.860E-01	0	0.000E+00	0.000E+00	#	45	tcwMf
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.446E+00	0	0.000E+00	0.000E+00	#	46	ptnMf
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.127E+00	0	0.000E+00	0.000E+00	#	47	tswMf
1	-3	0.000E+00	0.000E+00	0.000E+00	-4	1.0	0.259E+00	0	0.000E+00	0.000E+00	#	48	chnMf
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.253E+00	-5	0.100E-01	0.302E+00	#	49	tcwF1
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.820E-01	-6	0.100E-01	0.302E+00	#	50	tcwF2
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.203E+00	-7	0.100E-01	0.302E+00	#	51	tcwF3
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.387E+00	-8	0.100E-01	0.905E-01	#	52	ptnF1
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.439E+00	-9	0.100E-01	0.905E-01	#	53	ptnF2
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.254E+00	-10	0.100E-01	0.905E-01	#	54	ptnF3
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.411E+00	-11	0.100E-01	0.905E-01	#	55	ptnF4
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.499E+00	-12	0.100E-01	0.905E-01	#	56	ptnF5
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.492E+00	-13	0.100E-01	0.905E-01	#	57	ptnF6
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.530E-01	-14	0.100E-01	0.647E-01	#	58	tswF1
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.157E+00	-15	0.100E-01	0.410E+00	#	59	tswF2
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.154E+00	-16	0.100E-01	0.410E+00	#	60	tswF3
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.110E+00	-17	0.100E-01	0.410E+00	#	61	tswF4
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.131E+00	-18	0.100E-01	0.410E+00	#	62	tswF5
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.112E+00	-19	0.100E-01	0.410E+00	#	63	tswF6
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.940E-01	-20	0.100E-01	0.410E+00	#	64	tswF7
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.370E-01	-21	0.100E-01	0.410E+00	#	65	tswF8
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.173E+00	-22	0.100E-01	0.410E+00	#	66	tswF9
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.273E+00	-23	0.100E-01	0.133E+00	#	67	ch1Fv
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.345E+00	-24	0.100E-01	0.133E+00	#	68	ch2Fv
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.345E+00	-25	0.100E-01	0.133E+00	#	69	ch3Fv
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.345E+00	-26	0.100E-01	0.133E+00	#	70	ch4Fv
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.345E+00	-27	0.100E-01	0.133E+00	#	71	ch5Fv
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.288E+00	-28	0.100E-01	0.954E-01	#	72	ch1Fz
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.331E+00	-29	0.100E-01	0.954E-01	#	73	ch2Fz
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.331E+00	-30	0.100E-01	0.954E-01	#	74	ch3Fz
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.331E+00	-31	0.100E-01	0.954E-01	#	75	ch4Fz
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.331E+00	-32	0.100E-01	0.954E-01	#	76	ch5Fz
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.266E+00	-33	0.100E-01	0.954E-01	#	77	ch6Fz
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.325E+00	-34	0.100E-01	0.954E-01	#	78	pp4Fz
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.303E+00	-35	0.100E-01	0.462E+00	#	79	pp3Fd
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.263E+00	-36	0.100E-01	0.462E+00	#	80	pp2Fd
6	0	0.200E+02	0.200E+02	0.200E+02	-4	1.0	0.280E+00	-37	0.100E-01	0.954E-01	#	81	pp1Fz

```
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.115E+00   -38 0.100E-01  0.462E+00 # 82 bf3Fd
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.259E+00   -39 0.100E-01  0.954E-01 # 83 bf2Fz
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.115E+00   -40 0.100E-01  0.462E+00 # 84 tr3Fd
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.259E+00   -41 0.100E-01  0.954E-01 # 85 tr2Fz
1    -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.370E-01     0 0.000E+00  0.000E+00 # 86 pcF38
1    -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.173E+00     0 0.000E+00  0.000E+00 # 87 pcF39
1    -2 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.288E+00     0 0.000E+00  0.000E+00 # 88 pcF1z
1    -2 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.331E+00     0 0.000E+00  0.000E+00 # 89 pcF2z
1    -2 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.331E+00     0 0.000E+00  0.000E+00 # 90 pcF5z
1    -2 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.266E+00     0 0.000E+00  0.000E+00 # 91 pcF6z
1    -3 0.000E+00  0.000E+00  0.000E+00    -4  1.0  0.325E+00     0 0.000E+00  0.000E+00 # 92 pcF4p
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.860E-01   -42 0.100E-01  0.300E+00 # 93 tcwFf
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.446E+00   -43 0.100E-01  0.100E+00 # 94 ptnFf
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.127E+00   -44 0.100E-01  0.500E+00 # 95 tswFf
6     0 0.200E+02  0.200E+02  0.200E+02    -4  1.0  0.259E+00   -45 0.100E-01  0.300E+00 # 96 chnFf
```

Finally, the nodal assignments can be verified in a similar manner. Three nodal assignments are extracted from the output file from MAKEPTRK v. 2.0:

```
1     1 1    49
30    30 1    80
95328 95328 1     34
```

The first two columns indicate the node number, the third column will always be one (nodal increment), and the fourth column is the material number assigned to that node. In FEHM the fractures are always listed first, followed by the matrix nodes. In TOUGH2, the elements are listed alternately, so the 2nd element listed in the TOUGH2 ELEME card is actually the 1st matrix node defined by FEHM. Looking at LB990701233129.001 [106785] ('3d2kpa_pc1.mesh' which contains the 'ELEME' card), the first element in '3d2kpa_pc1.mesh' belongs to material 'tcwF1', which is correctly identified as the 49th material (see list above). The 30th node in FEHM is actually the 30th fracture node in TOUGH2, which would be the 61st element listed in '3d2kpa_pc1.mesh.' The 61st element in '3d2kpa-pc1.mesh' belongs to materials 'pp2Fd,' which is correctly identified as the 80th material in the list above. Finally, the last node identified in the MAKEPTRK file (95328) corresponds to the last matrix element in the TOUGH2 file (excluding boundaries). This element belongs to material 'bf3Md,' which is correctly identified as the 34th material in the list above.

These visual inspection verify that MAKEPTRK v. 2.0 is performing correctly for the range of inputs that are used (TOUGH2 input files) for generating material listings and nodal assignments for use in the FEHM input files.

SOURCE FILE FOR MAKEPTRK V. 2.0

```
c makeptrk_v2.f
c______________________________________________________________________
c This program will create the transport models that are used in the
c FEHM ptrk macro. The required input files are the TOUGH2 ROCKS card,
c ELEME card, and CONNE card. This program will also ask the user for
c parameters including fracture and matrix diffusion, dispersivity, and
c Kd. The primary output is, for each ROCKS material, the Kd,
c dispersivity, molecular diffusion, fracture sorption, matrix porosity,
c and aperture parameter (for fracture->matrix diffusion).
c
c  C.K.Ho
c  3/12/99
c
c Several modifications have been made:
c 1) Kd's are not assigned to fracture materials
c 2) Format for dispersivity value has been changed from f5.2 to e10.3
c 3) User is given an option to use fracture/matrix reduction factor in
c    calculating aperture parameter.
c  C.K.Ho
c  4/20/99
c
c This version (makeptrk_v2.f) now accommodates the active fracture
c model (Liu et al. 1998) by including additional flags and
c parameters (a gamma parameter (g) is read in from the ROCKS card).
c It also assigns EACH node to a transport model (material).
c The previous use of g is deleted because the formulation in FEHM
c for matrix diffusion automatically accounts for the reduced
c fracture/matrix area,(CRWMS M&O 2000 [141418])
c Section 6.2.1).
c  C.K.Ho
c  1/31/2000
c______________________________________________________________________
c23456789012345678901234567890123456789012345678901234567890123456789012
c   implicit double precision (a-h,o-z)
      character*100 block,rocks,mesh,out
      character*5 matname(999),mat(99999),elemn
      real por(999),g(999)
      dimension imatname(99999),ibf(999)

      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ROCKS card?'
      read(*,'(a)') rocks
      write(*,*)'What is the name of the file containing the TOUGH2'
      write(*,*)'ELEME and CONNE cards?'
      read(*,'(a)') mesh
      write(*,*)'What would you like to name the output file?'
      read(*,'(a)') out
      write(*,*)'What transport mechanisms apply for the matrix?'
      write(*,*)'1 - advection only (no dispersion or matrix diff)'
      write(*,*)'2 - advection and dispersion (no matrix diff)'
      write(*,*)'3 - advection and matrix diff (no dispersion)'
      write(*,*)'4 - advection, dispersion, and matrix diff'
      write(*,*)'6 - advection, dispersion, and matrix diff with '
      write(*,*)'   finite fracture spacing'
      read(*,*) iflagm
      write(*,*)'What transport mechanisms apply for the fracture?'
      write(*,*)'1 - advection only (no dispersion or matrix diff)'
      write(*,*)'2 - advection and dispersion (no matrix diff)'
      write(*,*)'3 - advection and matrix diff (no dispersion)'
      write(*,*)'4 - advection, dispersion, and matrix diff
```

```
      write(*,*)'6 - advection, dispersion, and matrix diff with '
      write(*,*)'   finite fracture spacing'
      read(*,*) iflagf
      write(*,*)'What is the Kd (cc/g) for vitric units?'
      read(*,*)kdv
      write(*,*)'What is the Kd (cc/g) for zeolitic units?'
      read(*,*)kdz
      write(*,*)'What is the Kd (cc/g) for devitrified units?'
      read(*,*)kdd
      write(*,*)'What is the matrix dispersivity (m)?'
      read(*,*) dispm
      write(*,*)'What is the fracture dispersivity (m)?'
      read(*,*) dispf
      write(*,*)'What is the molecular diffusion coefficient?'
      read(*,*) ido
      write(*,*)'What is the retardation factor for fracture'
      write(*,*)'sorption?  (1 = no fracture sorption)'
      read(*,*) rdfrac
      write(*,*)'What is the residual fracture saturation?'
      read(*,*) slr

      open(1,file=mesh,status='old')
      open(3,file=rocks,status='old')
      open(12,file=out,status='unknown')

c...Data
c...Assign a dummy aperture parameter for matrix materials.
c...Matrix diffusion is not used for matrix materials.
      ibfm=0
      slm=0.

c...Read in ROCKS information from TOUGH2 input file

18    read(3,1000) block
      if(block(1:5).ne.'ROCKS') go to 18

      i=1
      nfmat=0
      nmmat=0
408   read(3,410) matname(i),drok,por(i)
410   format(a5,5x,2e10.4)
      if(matname(i).eq.'REFCO'.or.matname(i)(1:3).eq.'top'.or.
     &  matname(i)(1:3).eq.'bot') go to 408
      if(matname(i).eq.'     ') then
c...ntotmat is the total number of materials in the ROCKS card
       ntotmat=i-1
       go to 27
      end if
c...nfmat is the total number of fracture materials
      if(matname(i)(3:3).eq.'F'.or.matname(i)(4:4).eq.'F')nfmat=nfmat+1
c...nmmat is the total number of matrix materials
      if(matname(i)(3:3).eq.'M'.or.matname(i)(4:4).eq.'M') nmmat=nmmat+1

c...Read in gamma parameter (g) for each material
      read(3,*)
      read(3,*)
      read(3,33) g(i)
33    format(60x,e10.4)
      i=i+1
      go to 408

27    continue
```

```
      write(*,75) nmmat
75    format('Number of matrix materials in ROCKS = ',i5)
      write(*,77) nfmat
77    format('Number of fracture materials in ROCKS = ',i5)

c...Read in element information from MESH
      n=1
      read(1,1000) block
1000  format(a22)
99    read(1,65) elemn,mat(n)
65    format(a5,10x,a5,e10.4)
c...End of active elements is signified by boundary elements or a
c...blank space
      if(elemn.eq.'     ') go to 98
      if(elemn(1:2).eq.'TP'.or.elemn(1:2).eq.'BT') go to 98

      N=N+1
      GO TO 99
98    CONTINUE
      NMAX = n - 1

c...NMAX is the total number of elements read from MESH
      write(*,107) nmax
107   format('Have read in ',i8,' elements from MESH...')

c...Find material number corresponding to element. Note that the node
c...numbering used in FEHM lists all the fracture nodes first, followed
c...by the matrix nodes
c...Identify materials corresponding to fracture nodes first
      do j=1,nmax,2
        do i=1,ntotmat
          if(mat(j).eq.matname(i)) imatname((j+1)/2)=i
        end do
      end do
c...Now identify materials corresponding to the matrix nodes
      do j=2,nmax,2
        do i=1,ntotmat
          if(mat(j).eq.matname(i)) imatname(j/2+nmax/2)=i
        end do
      end do

c...Determine matrix porosities corresponding to each fracture material
c...Because the number of fracture and matrix materials may not be equal,
c...I compare the characters of the element names to assign a matrix
c...porosity to the corresponding fracture element (for matrix diffusion).
c...I first determine where the 'F' is, and then I compare all other
c...characters with the matrix material to get a match.
      do i=1,ntotmat
        if(matname(i)(3:3).eq.'M'.or.matname(i)(4:4).eq.'M')goto83
        if(matname(i).eq.'topbd'.or.matname(i).eq.'botbd')goto83
          do j=1,ntotmat
            if(matname(i)(3:3).eq.'F') then
              if(matname(j)(3:3).eq.'M') then
                if(matname(j)(1:2).eq.matname(i)(1:2).and.
     &             matname(j)(4:5).eq.matname(i)(4:5)) then
                  por(i)=por(j)
                  go to 83
                end if
              end if
            elseif(matname(i)(4:4).eq.'F') then
              if(matname(j)(4:4).eq.'M') then
                if(matname(j)(1:3).eq.matname(i)(1:3).and.
```

```
   &           matname(j)(5:5).eq.matname(i)(5:5)) then
             por(i)=por(j)
             go to 83
           end if
         end if
        end if
       end do
       por(i)=0.1
c234567890123456789012345678901234567890123456789012345678901234567890l2
       write(*,113) matname(i)
113       format('Material ',a5,' does not have a matrix counterpart.'/
   &         'It has been assigned a matrix porosity of 0.1')
83   end do

c...Determine aperture parameter, ibf

     icount=-5
     do i=1,ntotmat

c...Initialize all aperture parameters to be equal to zero
      ibf(i)=ibfm

c...If the material is a fracture, assign it a flag that will specify which
c...column to choose in a prescribed file.  The flag is a negative number.
c...Treat perched water elements (elements with a 'pc' in the name) as matrix
      if(matname(i)(1:2).ne.'pc') then
       if(matname(i)(3:3).eq.'F'.or.matname(i)(4:4).eq.'F') then
        ibf(i)=icount
        icount=icount-1
       end if
      end if
87   end do

c...Write data to output file for PTRK macro

     do i=1,ntotmat

c...Assign appropriate Kd
      if(matname(i)(5:5).eq.'v') then
       kd=kdv
      elseif(matname(i)(5:5).eq.'z') then
       kd=kdz
      else
       kd=kdd
      end if

c234567890123456789012345678901234567890123456789012345678901234567890l2
      if(matname(i)(3:3).eq.'M'.or.matname(i)(4:4).eq.'M') then
       write(12,505)iflagm,kd,dispm,dispm,dispm,ido,1.,por(i),
   &           ibf(i),slm,g(i),i,matname(i)
505     format(i1,1x,i5,3(e10.3,1x),i5,1x,f4.1,1x,e10.3,1x,i5,
   &       2(e10.3,1x),'#',i3,1x,a5)
      elseif(matname(i)(1:2).eq.'pc') then
       write(12,505)iflagm,kd,dispm,dispm,dispm,ido,1.,por(i),
   &           ibf(i),slm,g(i),i,matname(i)
      else
       write(12,505)iflagf,0.,dispf,dispf,dispf,ido,rdfrac,por(i),
   &           ibf(i),slr,g(i),i,matname(i)
      end if
     end do

     write(12,*)
```

```
      do i=1,nmax
        write(12,507) i,i,1,imatname(i)
507     format(i5,1x,i5,1x,i1,1x,i5)
      end do

      stop
      end
```

INTENTIONALLY LFET BLANK

ATTACHMENT VII

PREWAP SOFTWARE ROUTINE REPORT

INTENTIONALLY LEFT BLANK

ATTACHMENT VII.

VII.1 SOFTWARE ROUTINE IDENTIFICATION

VII.1.1 SOFTWARE NAME AND VERSION NUMBER: PREWAP VERSION 1.0

VII.1.2 NAME AND VERSION OF INDUSTRY STANDARD SOFTWARE UNDER WHICH ROUTINE WAS DEVELOPED

This routine was developed using Microsoft Developer Studio 97 with Visual Fortran 6.0A, Professional Edition.

VII.1.3 SRR DOCUMENT IDENTIFICATION NUMBER: N/A

VII.1.4 SRR MEDIA NUMBER (IF APPLICABLE): N/A

VII.2. DESCRIPTION AND TESTING

VII.2.1 OVERVIEW

Corrosion of the drip shields and waste packages is accounted for in the Total System Performance Assessment-Site Recommendation (TSPA-SR) model by the WAPDEG routine, which runs as a DLL under the TSPA-SR software (Golder Associates 2000 [143556]). As input, WAPDEG requires T-H data (temperatures, relative humidities, etc.), as well as seepage chemistry information (pH, chloride concentration, etc.). T-H data are characterized *Abstraction of NFE Drift Thermodynamic Environment and Percolation Flux* (CRWMS M&O 2000 [149860]). In-drift chemistry is characterized in *Precipitate/Salts Model Results for THC Abstraction* (CRWMS M&O 2000 [151708]). In-package chemistry is characterized in *In-Package Chemistry Abstraction* (CRWMS M&O 2000 [129287]).

The PREWAP routine calculates the seepage chemistry associated with the T-H data. The T-H and seepage chemistry data are then written to output files that are used as input to the WAPDEG routine (CRWMS M&O 2000[151566]).

The PREWAP routine extracts this data from these various tables and prepares an output table that is used as input to the WAPDEG routine.

The PREWAP routine is a stand-alone executable. This allows the WAPDEG input to be prepared independent of the TSPA-SR software, reducing the run time for TSPA-SR realizations.

VII.2.2 INPUTS

The input to PREWAP consist of in-drift drip and no-drip chemistry pH and Cl data, in-package pH and Cl data, and T-H data (for low, mean, and high infiltration cases) for Commercial Spent Nuclear Fuel (CSNF) and Co-Disposed Waste Package (CDSP) waste packages. Information is also passed to PREWAP regarding input and output file names, as well as an RH corrosion limit.

VII.2.2.1 In-Drift Chemistry Data (Drip Conditions)

In-drift pH and Chloride Concentration (Cl) under dripping conditions are dependent on RH and the abstracted time period. Within a given set of RH and time period, they can also be dependent on temperature (T), invert evaporation rate (Q_e), and seepage rate (Q_s) into the drift. The break-down of cases and their independent parameters are given in Table VII-1.

Table VII-1. Classification of In-Drift pH and Cl Data Sets for Dripping Conditions

	RH	time period(s)*	additional independent parameters
case 1	RH<50.3%	all	none
case 2	50.3%<RH<85%	2, 3, 4, 5	none
case 3	RH>85%	2, 3, 5	1-Qe/Qs
case 4	RH>85%	4	1-Qe/Qs, T

*time periods:
1 0 to 50 years from initial opening of the repository
2 50 to 1000 years from initial opening of the repository
3 1000 to 2000 years from initial opening of the repository
4 2000 to 100,000 years from initial opening of the repository
5 > 100,000 years from initial opening of the repository

The in-drift chemistry data in Tables 2 through 7 are taken from *Precipitate/Salts Model Results for THC Abstraction* (CRWMS M&O 2000 [151708], Tables 2-4) (DTN: MO0002SPALOO46.010 [149168] (Unqualified)).

Case 1 conditions have no pH and Cl (Molal) data. For this case, the pH and Cl are hardwired in the PREWAP code to be equal to -9.99E-02 (the default 'does not exist' value for WAPDEG input).

Case 2 data for pH and Cl are contained in files **phTable1.dat** and **ClTable1.dat**, respectively. The contents of these files are shown in Table VII-2 and Table VII-3. The value in the first row indicates the number of rows of data to follow. The data are organized as a set of 1-D look-up tables. The 1st column contains the RH independent parameter values. Columns 2, 3, and 4 contain the dependent parameter values (pH or Cl) for time periods 2, 3/5, and 4, respectively. The remaining information in the file below the look-up table (column headings and descriptive text) is not used by PREWAP.

Case 3 data for pH and Cl are contained in files **phTable2.dat** and **ClTable2.dat**, respectively. The contents of these files are shown in Table VII-4 and Table VII-5. The value in the first row indicates the number of rows of data to follow. The data are organized as a set of 1-D look-up tables. The 1st column contains the $1\text{-}Q_e/Q_s$ independent parameter values. Columns 2, 3, and 4 contain the dependent parameter values (pH or Cl) for time periods 2 and 3/5, respectively. The remaining information in the file below the look-up table (column headings and descriptive text) is not used by PREWAP.

Case 4 data for pH and Cl are contained in files **phTable3.dat** and **ClTable3.dat**, respectively. The contents of these files are shown in Table VII-6 and Table VII-7. The values in the first row indicate the number of rows and columns that make up the dependent data set (pH or Cl values) in the 2-D look-up table that follows. The next row contains the independent parameter temperature values. In the remaining rows, the 1st column contains the 1-Q_e/Q_s independent parameter values. Columns 2, 3, and 4 contain the dependent parameter values (pH or Cl) for temperatures of 25 C, 50 C, and 75 C, respectively. The remaining information in the file below the look-up table (column headings and descriptive text) is not used by PREWAP.

Table VII-2. Case 2 pH Look-Up Table (In-Drift Dripping Conditions)

10					
50.3	9.40	7.64	7.02		
51.0	9.40	7.64	7.02		
53.1	9.40	7.64	7.02		
55.2	9.40	7.64	7.02		
60.5	9.40	7.64	7.02		
65.7	9.40	7.64	7.02		
71.0	9.40	7.64	7.02		
76.2	9.40	7.64	7.02		
81.5	9.40	7.64	7.02		
85.0	9.40	7.64	7.02		
		2		3/5	4

; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (CRWMS M&O 2000 [127818])
; Seepage Name: Abstracted THC Seepage Water
; 1st independent variable (columns) = Abstracted Period
; 2nd independent variable (rows) = relative humidity (RH)
; dependent parameter = pH

Table VII-3. Case 2 Cl Look-Up Table (In-Drift Dripping Conditions)

10					
50.3	-2.431	-2.428	-2.415		
51.0	-1.246	-1.244	-1.231		
53.1	-0.389	-0.391	-0.380		
55.2	-0.164	-0.169	-0.159		
60.5	0.225	0.211	0.216		
65.7	0.380	0.358	0.359		
71.0	0.420	0.396	0.396		
76.2	0.428	0.403	0.403		
81.5	0.418	0.394	0.394		
85.0	0.407	0.382	0.382		
		2		3/5	4

; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (CRWMS M&O 2000 [127818])
; Seepage Name: Abstracted THC Seepage Water
; 1st independent variable (columns) = Abstracted Period
; 2nd independent variable (rows) = relative humidity (RH)
; dependent parameter = log Cl (i.e., log of Cl concentration (molal))

Table VII-4. Case 3 pH Look-Up Table (In-Drift Dripping Conditions)

7		
0.000999 9.40	7.64	
0.001	9.41	7.64
0.01	9.28	7.58
0.1	9.21	7.45
0.5	8.87	7.64
0.9	8.62	7.71
1.0	8.58	7.72
	2	3/5

; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (CRWMS M&O 2000 [127818])
; Seepage Name: Abstracted THC Seepage Water
; 1st independent variable (columns) = Abstracted Period
; 2nd independent variable (rows) = 1-Qe/Qs (Qe = evaporation rate, Qs = incoming seepage rate)
; condition: relative humidity (RH) > 85 percent
; dependent parameter = pH

Table VII-5. Case 3 Cl Look-Up Table (In-Drift Dripping Conditions)

7		
0.000999 0.387	0.382	
0.001	0.190	0.373
0.01	-0.752	-0.502
0.1	-1.745	-1.496
0.5	-2.445	-2.194
0.9	-2.699	-2.449
1.0	-2.745	-2.496
	2	3/5

; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (CRWMS M&O 2000 [127818])
; Seepage Name: Abstracted THC Seepage Water
; 1st independent variable (columns) = Abstracted Period
; 2nd independent variable (rows) = 1-Qe/Qs (Qe = evaporation rate, Qs = incoming seepage rate)
; condition: relative humidity (RH) > 85 percent
; dependent parameter = log Cl (i.e., log of Cl concentration (molal))

Table VII-6. Case 4 pH Look-Up Table (In-Drift Dripping Conditions)

7		3			
25				50	75
0.0011999	7.02	7.02	7.02		
0.0012	6.78	6.86	7.02		
0.01	6.986	6.95	7.02		
0.1		7.11	7.03	6.97	
0.5		7.23	7.18	7.14	
0.9		7.09	7.22	7.18	
1.0		7.05	7.22	7.19	

; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (CRWMS M&O 2000 [127818])
; Seepage Name: Abstracted THC Seepage Water
; condition: Period 4
; 1st independent variable (columns) = temperature ('C)
; dependent parameter = pH

Table VII-7. Case 4 Cl Look-Up Table (In-Drift Dripping Conditions)

7		3			
25				50	75
0.0011999	0.38202	0.38202	0.38202		
0.0012	0.39094	0.38202	0.38202		
0.01	-0.48798	-0.48872	-0.48945		
0.1	-1.4828	-1.48216	-1.48214		
0.5	-2.18053	-2.18052	-2.18059		
0.9	-2.43581	-2.43581	-2.43581		
1.0	-2.48149	-2.48149	-2.48162		

; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (CRWMS M&O 2000 [127818])
; Seepage Name: Abstracted THC Seepage Water
; dependent parameter = log Cl (i.e., log of Cl concentration (Molal))

VII.2.2.2 In-Drift Chemistry Data (No-Drip Conditions)

In-drift pH under no-dripping conditions is dependent on CO_2 fugacity and temperature. No-drip pH data are contained in file **phTable4.data**. The contents of this file are shown in Table VII-8. The values in the first row indicate the number of rows and columns that make up the dependent data set (pH values) in the 2-D look-up table that follows. The next row contains the independent parameter temperature values. In the remaining rows, the 1st column contains the independent parameter log CO2 fugacity values. Columns 2, 3, 4, and 5 contain the dependent parameter values (pH) for temperatures of 25 C, 45 C, 75 C, and 95 C, respectively. The remaining information in the file below the look-up table (column headings) is not used by PREWAP.

There are no data for Cl under no-dripping conditions; hence the no-drip Cl is hardwired in the PREWAP code to be equal to the default 'does not exist' value of -9.99E-02.

Table VII-8. pH Look-Up Table (In-Drift No-Dripping Conditions)

7			4			
25			45		75	95
-1	4.41	4.47	4.60	4.70		
-3	5.41	5.49	5.73	6.02		
-4	5.91	6.03	6.41	6.70		
-5	6.39	6.57	6.88	6.96		
-6	6.80	6.92	6.99	7.00		
-7	6.97	6.99	7.00	7.00		
-9	7.00	7.00	7.00	7.00		
log fCO2						

VII.2.2.3 IN-PACKAGE CHEMISTRY DATA (DRIP AND NO-DRIP CONDITIONS)

In-Package chemistry is dependent upon the waste type (CSNF or CDSP) in the waste package. Bounding values for the pH and Cl are read into PREWAP from the file **InPkgChem.dat**. The 1st row contains the bounding pH values for CSNF and CDSP, respectively. The 2nd row contains the bounding Cl value used for both CSNF and CDSP.

Table VII-9. pH and Cl In-Package Chemistry Data

7.60	9.83
2.014E-04	

For CSNF, the in-package chemistry is a function of cladding coverage and seepage flow rate. Inspection of Figures 3 and 4 in the *In-Package Chemistry Abstraction* (CRWMS M&O 2000 [129287]) finds that the >1000 year post-breach period has the potential to have the highest pH within the bounds of the response surface data-set. Using the appropriate equation from Table 6 (CRWMS M&O 2000 [129287]) yields the upper bound on pH for CSNF.

$$pH = 6.0668 - 0.5395\log(cc) + 4.0479\left[\frac{yr}{mm}\right]Q$$

$$pH = 6.0668 - 0.5395\log(0.02) + 4.0479\left[\frac{yr}{mm}\right]\left(0.15\frac{mm}{yr}\right) = 7.60$$

The terms *cc* and *Q* represent cladding coverage fraction and flow rate (mm/yr), respectively. For CDSP the in-package chemistry is a function of relative glass rate and seepage flow rate. The glass rate is a relative dissolution rate and is described in further detail in *In-Package Chemistry Abstraction* (CRWMS M&O 2000 [129287]). Inspection of Figures 5 and 6 in *In-Package Chemistry Abstraction* (CRWMS M&O 2000 [129287]) finds that the >1000 year post-breach period has the potential to have the highest pH within the bounds of the response surface data-set. Using the appropriate equation from Table 12 (CRWMS M&O 2000 [129287]) yields the upper bound on pH for CSNF.

$$pH = 8.4247 - 3.4173\left[\frac{yr}{mm}\right]Q + 0.1403GR$$

$$pH = 8.4247 - 3.4173\left[\frac{yr}{mm}\right]\left(0.0015\frac{mm}{yr}\right) + 0.1403(10.0) = 9.83$$

The terms Q and GR represent the seepage and glass rate, respectively. A chloride value of 2.014E-04 mol/kg (equal to that of J-13 water) is specified for both CSNF and CDSP waste package (CRWMS M&O 2000 [129287]).

VII.2.2.4 T-H DATA

The T-H data sets are taken from DTN: SN0001T0872799.006 [147198] (backfill TH data sets) and DTN: SN0007T0872799.014 [152545] (no backfill TH data sets).

The T-H data sets are broken down into five 'bins' based on infiltration rate. Furthermore, there are separate sets of T-H data for each infiltration scenario (low, mean, or high). Table VII-10 shows the relationship between infiltration bins, infiltration scenario, and the T-H data files.

Table VII-10. Relationship Between Infiltration Bins, Infiltration Scenario, and T-H Data Files

	infiltration scenario		
infiltration bin	**low**	**mean**	**high**
bin 1 (< 3.4 mm/yr)	CSNF_low_Bin1.in HLW_low_Bin1.in	CSNF_mean_Bin1.in HLW_mean_Bin1.in	n/a n/a
bin 2 (3.4 to 10 mm/yr)	CSNF_low_Bin2.in HLW_low_Bin2.in	CSNF_mean_Bin2.in HLW_mean_Bin2.in	CSNF_high_Bin2.in HLW_high_Bin2.in
bin 3 (10 to 20 mm/yr)	n/a n/a	CSNF_mean_Bin3.in HLW_mean_Bin3.in	CSNF_high_Bin3.in HLW_high_Bin3.in
bin 4 (20 to 60 mm/yr)	n/a n/a	CSNF_mean_Bin4.in HLW_mean_Bin4.in	CSNF_high_Bin4.in HLW_high_Bin4.in
bin 5 (> 60 mm/yr)	n/a n/a	CSNF_mean_Bin5.in HLW_mean_Bin5.in	CSNF_high_Bin5.in HLW_high_Bin5.in

The format of the T-H files is illustrated in Table VII-11.

Table VII-11 T-H File CSNF_mean_Bin5.in

line(s)	T-H file information	comment
1	Infiltration Bin:	not used
2	qinf > 60.0 mm/yr	not used
3	RIP_csnf_d0010500_bin-60_mean	not used
4	data column headers (see below)	not used
5	The number of Rows = 83	numeric value read in
6	The fraction of this history = 0.000576	numeric value read in
7	Coordinate Location:	not used

line(s)	T-H file information	comment
8	The easting coordinate = 170208.78 m	not used
9	The northing coordinate = 234316.70 m	not used
10	Infiltration rate:	not used
11	qinf = 61.00266 mm/yr	not used
12 to 94	T-H data	read in
95	The number of Rows = 84	numeric value read in
96	The fraction of this history = 0.000960	numeric value read in
97	Coordinate Location:	not used
98	The easting coordinate = 170228.75 m	not used
99	The northing coordinate = 234315.60 m	not used
100	Infiltration rate:	not used
101	qinf = 60.79187 mm/yr	not used
102 to 195	T-H data	read in
196	The number of Rows = 87	numeric value read in
197	The fraction of this history = 0.001153	numeric value read in
198	Coordinate Location:	not used
199	The easting coordinate = 170256.20 m	not used
200	The northing coordinate = 234314.20 m	not used
201	Infiltration rate:	not used
202	qinf = 60.37322 mm/yr	not used
203 to 290	T-H data	read in

Each T-H data file contains time-histories from zero to one-million years for the following parameters at a given number of spatial locations:

- Waste Package Temperature [C]
- Drip Shield Temperature [C]
- Drift Wall Temperature [C]
- Invert Temperature [C]
- Waste Package RH [-]
- Drip Shield RH [-]
- Drift Wall RH [-]
- Backfill RH [-]
- Invert RH [-]
- Liquid Saturation at the Drip Shield [-]
- Liquid Saturation at the Invert [-]
- Air Mass Fraction [-]
- Water Vapor Flux at Drift Wall [kg/yr/m of drift]
- Air Flux at Drift Wall [kg/yr/m of drift]
- Drip Shield Water Evaporation Rate [m3/yr]
- Backfill Water Evaporation Rate [m3/yr]
- Invert Water Evaporation Rate [m3/yr]
- Percolation Flux at 5 m [mm/yr]

- Volume flow at the Drip Shield Top [m3/yr]
- Volume flow at the Invert [m3/yr]
- Top of the Drip Shield Temperature [C]

VII.2.2.5 INPUT/OUTPUT CONTROL FILES

The **InMaster.in** and **OutMaster.in** files pass file-name information to PREWAP. The 1st row in **InMaster.in** contains the RH corrosion limit; the 2nd row contains the number of file names. The remaining rows list the names of the T-H files that are to be read by PREWAP. **OutMaster.in** contains the names of the WAPDEG input files that PREWAP results are to be written.

Table VII-12. InMaster.in File

```
0.501
22
CSNF_low_bin1.in
CSNF_low_bin2.in
HLW_low_bin1.in
HLW_low_bin2.in
CSNF_mean_bin1.in
CSNF_mean_bin2.in
CSNF_mean_bin3.in
CSNF_mean_bin4.in
CSNF_mean_bin5.in
HLW_mean_bin1.in
HLW_mean_bin2.in
HLW_mean_bin3.in
HLW_mean_bin4.in
HLW_mean_bin5.in
CSNF_high_bin2.in
CSNF_high_bin3.in
CSNF_high_bin4.in
CSNF_high_bin5.in
HLW_high_bin2.in
HLW_high_bin3.in
HLW_high_bin4.in
HLW_high_bin5.in
```

Table VII-13. OutMaster.in File

```
CSNF_low_bin1.ou
CSNF_low_bin2.ou
HLW_low_bin1.ou
HLW_low_bin2.ou
CSNF_mean_bin1.ou
CSNF_mean_bin2.ou
CSNF_mean_bin3.ou
CSNF_mean_bin4.ou
CSNF_mean_bin5.ou
HLW_mean_bin1.ou
HLW_mean_bin2.ou
HLW_mean_bin3.ou
HLW_mean_bin4.ou
HLW_mean_bin5.ou
CSNF_high_bin2.ou
CSNF_high_bin3.ou
```

CSNF_high_bin4.ou
CSNF_high_bin5.ou
HLW_high_bin2.ou
HLW_high_bin3.ou
HLW_high_bin4.ou
HLW_high_bin5.ou

VII.2.3 DESCRIPTION OF SOFTWARE ROUTINE INCLUDING THE EXECUTION ENVIRONMENT

VII.2.3.1 DEVELOPMENT AND EXECUTION ENVIRONMENT

The PREWAP routine is a FORTRAN executable. The code was developed and tested in the Windows NT 4.0 operating system. It was compiled with Digital FORTRAN Professional 6.0A as a stand-alone executable (exe) program. The routine operates in a Windows 95/98 or Windows NT environment

VII.2.3.2 MAIN PROGRAM

The PREWAP program begins by calling a subroutine (**ReadMasterFiles**) that reads in the T-H input and WAPDEG output file names. Next it calls a subroutine (**ReadChemData**) to read in the in-drift chemistry lookup tables and in-package chemistry data. The program then initiates a loop that calls subroutines that; read in the T-H data, perform the necessary calculations, and generate the WAPDEG input files.

The program loop first calls a subroutine to count the data sets(**CountDataSets**) in the selected T-H file. It then calls a subroutine to allocate arrays (**AllocateArays**) to hold the data during processing. Next a subroutine (**ReadInputFile**) reads the T-H data. The data are then processed by a subroutine (**DoCalculations**) that performs the necessary calculations. The next subroutine (**CullDataPoints**) checks the data set resulting from the calculations and eliminates (based on a threshold RH value) those portions that will not contribute to corrosion of the EBS. This modified dataset is in turn checked by the **AddDataPoints** subroutine to determine if minimum time-step size requirements are met. If they are not, interpolated data points are added back to the data set between the times that do not meet the minimum time-step requirements. The data set is then written to an output file by the **WriteOutputFile** subroutine. Finally it calls a subroutine (**DeallocateArrays**) to deallocate the arrays allocated earlier in the loop.

VII.2.3.4 SUBROUTINE READMASTERFILES

The **ReadMasterFiles** subroutine opens the files **InMaster.in** and **OutMaster.in**. The RH corrosion limit and the number of T-H and WAPDEG input files are read in. A do-loop is then initiated that reads in the input file names (T-H files) from **InMaster.in** and the output file names (WAPDEG files) from **OutMaster.in**.

VII.2.3.5 SUBROUTINE READCHEMDATA

This subroutine reads in the Cl and pH look-up tables from files **CLtable1.dat, CLtable2.dat, Cltable3.dat, pHtable1.dat, pHtable2.dat, pHtable3.dat,** and **pHtable4.dat.** In-package chemistry data are read in from the file **InPkgChem.dat**. The data contained in these files are described Section 2.2.

VII.2.3.6 SUBROUTINE COUNTDATASETS

This subroutine counts the number of data sets in each of the T-H files. It initializes the number of data sets (**nDataSets**) counter to 1 and the maximum number of rows (**maxRows**) variable to 0. The subroutine then reads past the 1st four rows of header information to the 5th row. It then reads past the header information in row 5 and reads the number of rows listed for that data set. This value is assigned to the variable **rows**. It then sets the value of **maxRows** equal to the number of rows just read.

The subroutine then reads past the next six rows of header information to the 1st data set. It then initiates a do-loop that executes **rows** number of times to read past the 1st data set.

It then begins to read the rest of the file with a do-loop. It reads the 1st header row for the next data set. If the end of file is reached the subroutine exits the do loop. If not, the subroutine reads the number of rows in the next data set as **rows**. It then increments the counter, **nDataSets,** by 1 and tests to see if the number of rows in this data set is greater than **maxrows**. If so, **maxrows** is set equal to **rows**. It then reads through this data set and restarts the loop. This loop is repeated until the end of file is reached. When the end of file is reached, the subroutine exits the do loop and closes the data file. The subroutine is then exited back to the main program.

VII.2.3.7 SUBROUTINE ALLOCATEARAYS

This subroutine sets the bounds on dynamic arrays to match the maximum number of rows (**maxRows**) and number of data sets (**nDataSets**) counted in the subroutine **CountDataSets.**

VII.2.3.8 SUBROUTINE READINPUTFILE

This subroutine reads the data from the T-H file to the dynamic arrays established in the previous subroutine.

VII.2.3.9 SUBROUTINE DOCALCULATIONS

This subroutine calculates pH and pH^2 for the waste package and the drip shield under drip and no drip conditions. Source Code is included for calculating Cl chemistry, but it is commented out. It also sets the in-package and barrier interface pH values for drip and no drip conditions.

The subroutine begins with a do-loop that sequentially processes each data set read from the TH file. Inside this loop is another do-loop that sequentially processes each row of data in the data set to calculate pH and pH^2 for the waste package and the drip shield. First it calculates the waste package pH and pH^2 for both drip and no drip conditions by calling the **InDriftCalc** subroutine using arguments that are specific to the waste package. Next it calculates the drip shield pH and pH^2 for both drip and no drip conditions by again calling the **InDriftCalc** subroutine, but using arguments that are specific to the drip shield.

After these calculations the subroutine sets the in-package pH for drip conditions for the waste package to the appropriate bounding value (pH of 7.6 for CSNF, 9.8 for Defense High Level Waste, and 9.83 for CDSP). It then sets the in-package pH for no-drip conditions equal to the default 'does not exist' value of -9.99E-02. Values for pH^2 are calculated from the pH values.

This process is repeated for each row of data in the data set. After all rows in a data set have been processed, the code processes the next data set until all data sets have been processed.

VII.2.3.10 SUBROUTINE INDRIFTCALC

The **InDriftCalc** subroutine is called by the **DoCalculations** subroutine. It performs the pH and pH^2 calculations for drip and no drip conditions for each row of data in the data set. The subroutine begins by first checking to see if the temperature is less than zero or if the seep rate is less than –99. If either condition applies, the pH for drip and no drip conditions is set to the default 'does not exist' value of -9.99E-02. If neither condition applies, the routine calculates $1-Q_e/Q_s$ for the row of data.

An **if-then-else** statement is used to determine which of the time periods is applicable. Values of drip and no-drip pH in the >50 year time period are set equal to the default 'does not exist' value of –9.99E-02. For the remaining time periods, an **if-then-else** statement is used to determine the applicable pH data-set based on RH. Table VII-14 shows the relationships between time periods, RH ranges, the potential independent parameters, and the pH data-sets.

Table VII-14. In-Drift Chemistry

time period	RH	drip condition	log(fCO2)	$1-Q_e/Q_s$	T	applicable point-value or data-set
>50 yrs	n/a	drip	n/a	n/a	n/a	-9.99E-02
		no drip				-9.99E-02
50 to 1000 yrs	RH < 50	drip	n/a	n/a	n/a	-9.99E-02
		no drip				-9.99E-02
	50<= RH <=85	drip	n/a	n/a	n/a	9.40
		no drip	-6.5		T	phTable4
	RH > 85	drip	n/a	$1-Q_e/Q_s$	n/a	phTable2a
		no drip	-6.5	n/a	T	phTable4
1000 to 2000 yrs	RH < 50	drip	n/a	n/a	n/a	-9.99E-02
		no drip				-9.99E-02
	50<= RH <=85	drip	n/a	n/a	n/a	7.64
		no drip	-3.0	n/a	T	phTable4
	RH > 85	drip	n/a	$1-Q_e/Q_s$	n/a	phTable2b
		no drip	-3.0	n/a	T	phTable4

Table VII-14. In-Drift Chemistry (Continued)

time period	RH	drip condition	log(fCO2)	$1\text{-}Q_o/Q_s$	T	applicable point-value or data-set
2000 to 100,000 yrs	RH < 50	drip	n/a	n/a	n/a	-9.99E-02
		no drip				-9.99E-02
	50<= RH <=85	drip	n/a	n/a	n/a	7.02
		no drip	**-2.0**	n/a	**T**	phTable4
	RH > 85	drip	n/a	**$1\text{-}Q_o/Q_s$**	**T**	phTable3
		no drip	**-2.0**	n/a	**T**	phTable4
<100,000 yrs	RH < 50	drip	n/a	n/a	n/a	-9.99E-02
		no drip				-9.99E-02
	50<= RH <=85	drip		n/a	n/a	7.64
		no drip	**-3.0**	n/a	**T**	phTable4
	RH > 85	drip		**$1\text{-}Q_o/Q_s$**	n/a	phTable2b
		no drip	**-3.0**	n/a	**T**	phTable4

As an example, the subroutine **Interp1D** is used to select pH values from the pH data-sets **phTable2a** and **phTable2b**, while subroutine **Interp2D** is used to select pH values from the pH data-sets **phTable3** and **phTable4**. In Table VII-14 the independent parameters associated with the pH data sets are denoted by **bold-face** type.

After these tests and calculations are performed to determine the values for pH under drip and no drip conditions, the values for pH^2 for drip and no drip conditions are calculated.

VII.2.3.11 SUBROUTINE INTERP1D

This subroutine is called by the **InDriftCalc** subroutine to interpolate thermophysical properties such as pH values, from one-dimensional arrays (e.g., **phTable2a** and **phTable2b**) created when the in-drift chemistry data from and **phTable2.dat** file were read. The subroutine is passed the value of the independent variable, the independent and dependent variable vectors, and the number of rows in the passed vectors. The subroutine passes back the interpolated dependent variable value.

The subroutine first checks to see if the independent variable value is within the upper and lower bounds of the independent variable vector. If it is above the upper bound, the dependent variable value is set equal to its upper bound; if it is below the lower bound the dependent variable is set equal to its lower bound. If neither condition is met, the subroutine linearly interpolates the dependent variable value between the independent vector values bounding the independent variable.

VII.2.3.12 SUBROUTINE INTERP2D

This subroutine is called by the **InDriftCalc** subroutine to interpolate thermophysical properties such as pH values from two dimensional arrays (e.g., **phTable3** and **phTable4**) created when the in-drift chemistry data from the **phTable3.dat** and **phTable4.dat** files were read. The subroutine is passed the values of the two independent variable, the two independent variable vectors, the dependent variable array, and the number of rows and columns passed array. The subroutine passes back the interpolated dependent variable value.

This subroutine first checks the value of the 1st independent variable to see if it is within the range of the 1st independent variable vector. If it is outside the range of the independent vector, a flag is set denoting whether it is above or below the range of the 1st independent vector. If the value of the 1st independent variable is within the range of the 1st independent vector, the subroutine loops through the 1st independent vector to identify the first row where the value of the 1st independent vector is less than the 1st independent variable.

Next the subroutine checks the value of the 2nd independent variable to see if it is within the range of the 2nd independent variable vector. If it is outside the range of the independent vector, a flag is set denoting whether it is above or below the range of the 2nd independent vector. If the value of the 2nd independent variable is within the range of the 2nd independent vector, the subroutine loops through the 2nd independent vector to identify the first row where the value of the 2nd independent vector is less than the 2nd independent variable.

The subroutine then checks to see if the 1st independent variable lower bound flag is set. If so, it then checks to see if the 2nd independent variable lower or upper bound flag is set. If this condition is satisfied, the dependent variable is assigned the value of the applicable corner point in the 2D array. If the 2nd independent variable is within the bounds of the 2nd independent vector, the subroutine linearly interpolates the dependent variable value between the 2nd independent vector values bounding the independent variable (i.e., along the lower edge of the array).

If the 1st independent variable is not outside the lower bound, the same process is repeated to determine if it is outside the upper bound. If this condition is satisfied, the dependent variable is set to the value at the upper corner points of the array or along the upper edge of the array.

The same logic is then repeated to identify values that are outside the upper and lower bounds of the 2nd independent variable.

If the 1st and 2nd independent variables are both within the bounds of their respective vectors, the program linearly interpolates the j-th column value between the i and i+1 rows. It then linearly interpolates the i-th row value between the j-th and j+1 columns. The results of these calculations are then used to linearly interpolate the dependent variable value.

VII.2.3.13 SUBROUTINE CULLDATAPOINTS

This subroutine removes rows of data where the waste package or drip shield temperature or RH are outside predetermined values. The subroutine loops through each data set. In turn each data-set is looped through (excepting the last row). A flag (**corFlag**) is set, based on a series of tests, to indicate whether or not that row of data is to be retained.

The **corFlag** is initialized to zero, as is the counter **nnRows()** which keeps track of the number of rows that are retained from each data-set.

The subroutine first checks to see if the waste package temperature or drip shield temperature is less than zero (values less than zero denote temperatures that 'do not exist'). If the condition is satisfied, the subroutine skips the remaining tests with **corFlag** set to zero. If the conditions are not satisfied, then the next test is performed with the **corFlag** variable still equal to zero.

Next the waste package and drip shield RH are checked to see if they are greater than the **corLim** value. If either is greater than **corLim, corFlag** is set to one and the remaining tests are skipped. Otherwise, **corFlag** remains at zero and the next test is performed.

Next the waste package RH for this row of data is checked to see if it is less than **corLim,** and the waste package RH for the next row of data is checked to see if it is greater than **corLim**. If both conditions are met, **corFlag** is set to one and the remaining tests are skipped. Otherwise, **corFlag** remains at zero, and the next test is performed.

Next the drip shield RH for this row of data is checked to see if it is less than **corLim,** and the drip shield RH for the next row of data is checked to see if it is greater than **corLim**. If both conditions are met, **corFlag** is set to one, and the remaining tests are skipped. Otherwise, it remains at zero, and the next test is performed.

Next the waste package RH for this row of data is checked to see if it is less than **corLim,** and the waste package RH for the previous row of data is checked to see if it is greater than **corLim**. If both conditions are met, **corFlag** is set to one, and the remaining tests are skipped. Otherwise, **corFlag** remains at zero, and the next test is performed.

Next the drip shield RH for this row of data is checked to see if it is less than **corLim,** and the drip shield RH for the previous row of data is checked to see if it is greater than **corLim**. If both conditions are met, **corFlag** is set to one, and the remaining tests are skipped. Otherwise, **corFlag** remains at zero and the next test is performed.

Next the waste package RH for the current row of data, the preceding row of data, and the next row of data are checked to see if they are all less than **corLim**. If these conditions are met, the remaining tests are skipped, with **corFlag** remaining at zero. If these conditions are not met, the next test is performed.

Next the drip shield RH for the current row of data, the preceding row of data, and the next row of data, are checked to see if they are all less than **corLim**. If these conditions are met, the final test is skipped, with **corFlag** remaining at zero. If these conditions are not met, **corFlag** is set to one.

If **corFlag** is set to one by any of the preceding tests, the row of data is written to a temporary file (**temp.dat**) and **nnRows()** is incremented by one.

The last row of data is written to the temporary file for all of the data sets.

When all of the data-sets have been processed, the temporary file is closed.

VII.2.3.14 SUBROUTINE ADDDATAPOINTS

This subroutine steps through the time histories in the temporary file (**temp.dat**) created by the **CullDataPoints** subroutine and determines if time-step sizes above 50,000 years are sufficiently small. This is accomplished in two parts. For time periods from 50,000 to 200,000 years, the time-step interval should be no greater than 10,000 years. For time periods greater than 200,000 years, the time-step interval should be no greater than 100,000 years.

First the subroutine opens the temporary data file (**temp.dat**) created by the **CullDataPoints** subroutine and creates a new temporary data file (**temp2.dat**).

A **do-loop** is used to cycle through all of the time histories. The current time history is read from **temp.dat** and stored in the dynamically allocated **TempStorage** array.

A nested do-loop is then used to cycle through all of the rows in the current time history.

First the time for the current row of data is checked to see if it is greater than 50,000 years and less than 200,000 years. If so, the interval between it and the next time step is evaluated to determine if it is greater than 10,000 years. If so, the current row of data is written to the **temp2.dat** file, and the **AddPoints1** subroutine is called to generate a sufficient number of 10,000 year-spaced interpolated data sets such that no time interval is greater than 10,000 years. If the time interval is less than 10,000 years, the current row of data is written to the **temp2.dat** file.

Next the subroutine checks to see if the time history is greater than 200,000 years. If so, the interval between it and the next time step is checked to determine if it is greater than 100,000 years. If so, the current row of data is written to the **temp2.dat** file and the **AddPoints2** subroutine is called to generate a sufficient number of 100,000 year-spaced interpolated data sets such that no time interval is greater than 100,000 years. If the time interval is less than 100,000 years, the current row of data is written to the **temp2.dat** file.

If the time history is less than 50,000 years, the data is written to the temp2.dat file.

This process is repeated until all rows up to the last one have been checked. When the last row of data is reached, it is written to the **temp2.dat** file.

VII.2.3.15 SUBROUTINE ADDPOINTS1

This subroutine interpolates data between time steps. It begins by checking to see if the time for the next row of data is greater than 200,000 years. If not, it skips forward to generate points for 10,000 year intervals. If so, it then sets two time steps, one for less than 200,000 years (**delTime1** = 200,000 years – current time step) and one for greater than 200,000 years (**delTime2** = 800,000 years). It then calculates the number of extra time steps needed for less than 200,000 years (**numExtraPoints1**) by dividing **delTime1** by 10,000 years. The number of time steps required above 200,000 years (**numExtraPoints2**) is determined by dividing **delTime2** by 100,000 years.

If **numExtraPoints1** is greater than zero, a **do-loop** is initiated that interpolates data points at 10,000 year intervals and writes them to the **TempStorage** array.

If **numExtraPoints2** is greater than zero, a **do-loop** is initiated that interpolates data points at 100,000 year intervals and writes them to the **TempStorage** array.

If the time step checked at the beginning of the routine is less than 200,000 years this, section of the subroutine calculates the time interval between the current time history and the next time history (**delTime**). It then divides **delTime** by 10,000 years to determine **numExtraPoints.** Next a **do-loop** is initiated that interpolates data points at 10,000 year intervals and writes them to the **TempStorage** array.

VII.2.3.16 SUBROUTINE ADDPOINTS2

This subroutine interpolates data for time steps above 200,000 years. It begins by calculating the time interval between the current time step and the next time step (**delTime**). It then divides **delTime** by 100,000 years to determine **numExtraPoints**. Next a **do-loop** is initiated that interpolates data points at 100,000 year intervals and writes them to the **TempStorage** array.

VII.2.3.17 SUBROUTINE WRITEOUTPUTFILE

This subroutine writes the output file from the **PREWAP** routine. It begins by opening the current output file (**outfile**) and the **temp2.dat** file. It then writes the initial comment lines and number of data sets to **outfile.** A **do-loop** is used to write each data set to the **outfile.** Within the **do-loop** the number of rows of data, the fraction of packages this data set is applicable to, and the header line for the data set are written to the **outfile.** Then a nested **do-loop** is used to read the data-set values from the **temp2.dat** file and write them to the **outfile.**

Finally the subroutine closes the **outfile** and **temp2.dat** files.

VII.2.3.18 SUBROUTINE DEALLOCATEARRAYS

This subroutine deallocates all of the arrays allocated at the beginning of the program.

VII.2.4 DESCRIPTION OF TEST CASES

PREWAP was validated using EXCEL spreadsheets to replicate PREWAP calculations and logic functions.

The interpolation subroutines were verified by running them independently of the overall program. A separate program was written containing the interpolate subroutines. This program was then compiled and run using an input deck that exercised all of the subroutine's calculations and logic functions. The output was written to output file. These results were then compared to an EXCEL spreadsheet that replicated the subroutine's calculations. These files are presented in Section 3.3 COMPUTER LISTING OF TEST DATA INPUT AND OUTPUT. Visual inspections of these files show that the outputs from both methods agree, thus validating the operation of the interpolation subroutines.

Next the overall program was verified by comparing the output from the program using a limited input deck covering the full range of values expected for the input to the output from an EXCEL spreadsheet that replicated the programs calculations and logic functions. This was accomplished by copying the test data input file to an EXCEL spreadsheet. Additional columns were then added to the spreadsheet containing equations or logic functions performed by the PREWAP program. This included columns for intermediate and final output. The output from the PREWAP program, using the test file as input, was compared to the results obtained from the spreadsheet. These files are presented in Section 3.3 COMPUTER LISTING OF TEST DATA INPUT AND OUTPUT. Visual inspection of these files shows that the output from the PREWAP program is consistent with the results generated by the spreadsheet.

VII.2.5 DESCRIPTION OF TEST RESULTS

The results of these tests demonstrate that the output from the PREWAP program agrees with the test cases, verifying that the program correctly performs its intended functions.

VII.2.6 RANGE OF INPUT VALUES FOR WHICH RESULTS WERE VERIFIED

Inputs to PREWAP are those physical parameters contained in the pH, Cl, and T-H files. Ranges for these parameters are those that are physically plausible for the parameter. For example RH cannot exceed 100 percent, pH and Cl concentrations values cannot be negative. No other limitations exist on the range of input parameter values.

VII.2.7 LIMITATIONS ON SOFTWARE ROUTINE APPLICATIONS OR VALIDITY

This is a stand-alone executable program that can be run under the Windows 95/98 and Windows NT operating environments on any PC platform with 100 megabytes of disk space and 64 megabytes of RAM.

VII.3 SUPPORTING INFORMATION:

VII.3.1 DIRECTORY LISTING OF EXECUTABLE AND DATA FILES

The PREWAP executable and the associated input files must be contained in the same directory. There are no other restrictions on directory names or structure that will affect the operation of the code.

VII.3.2 COMPUTER LISTING OF SOURCE CODE

```
program prewap

! define dynamic variables
real(8), allocatable :: etime(:,:), wpT(:,:), dsT(:,:), dwT(:,:), iT(:,:)
```

```
real(8), allocatable :: wpRH(:,:), dsRH(:,:), dwRH(:,:), bfRH(:,:), iRH(:,:)
real(8), allocatable :: dsLS(:,:), iLS(:,:), massFracAir(:,:)
real(8), allocatable :: dwFluxWV(:,:), dwFluxAir(:,:)
real(8), allocatable :: dsEvapRate(:,:), bfEvapRate(:,:), iEvapRate(:,:)
real(8), allocatable :: PercFlux5m(:,:), tdsPercFlux(:,:), iPercFlux(:,:)
real(8), allocatable :: tdsT(:,:)

real(8), allocatable :: fract(:)

real(8), allocatable :: wpPHnd(:,:), wpCLnd(:,:), wpPHd(:,:), wpCLd(:,:)
real(8), allocatable :: dsPHnd(:,:), dsCLnd(:,:), dsPHd(:,:), dsCLd(:,:)
real(8), allocatable :: ipkPHnd(:,:), ipkCLnd(:,:), ipkPHd(:,:), ipkCLd(:,:)
real(8), allocatable :: barPHnd(:,:), barCLnd(:,:), barPHd(:,:), barCLd(:,:)

real(8), allocatable :: TempStorage(:,:)

integer(4), allocatable :: nRows(:), nnRows(:), nnnRows(:)

! define fixed variables

real(8) RHvector(10), Qvector(7), Tvector3(3), Tvector4(4), fCO2vector(7)
real(8) CLtable1a(10), CLtable1b(10), CLtable1c(10)
real(8) CLtable2a(7), CLtable2b(7)
real(8) CLtable3(7,3)
real(8) PHtable1a(10), PHtable1b(10), PHtable1c(10)
real(8) PHtable2a(7), PHtable2b(7)
real(8) PHtable3(7,3)
real(8) PHtable4(7,4)

real(8) ReadVector(22)
real(8) newValue(22)

integer(4) nRowsTable1, nRowsTable2, nRowsTable3, nColsTable3
integer(4) nRowsTable4, nColsTable4

real(8) ipkPHbounding, CorLim

real(8) pHCSNFinpk, pHCDSPinpk, Clinpk

integer(4) i, j, k
integer(4) iFile, nFile
integer(4) rows, maxRows, nDataSets, maxnnRows
integer(4) corFlag

character*6 dummy1
character*6 dummy2(6)
character*25 infile, outfile
character*25 InFileNames(100)
character*25 OutFileNames(100)

open(unit=99, file='debug.dat')        ! open debug file

! read in TH input and WAPDEG output file names
call ReadMasterFiles

! read in data for in-drift chemistry lookup tables
call ReadChemData

maxnnRows=0  ! initialize counter

! main program loop
! calls the subroutines that read in TH data, perform the necessary
! calculations, and generate the WAPDEG input files
```

```
do iFile=1,nFile

  infile=InFileNames(iFile)
  outfile=OutFileNames(iFile)

  write(*,*) "processing file: ", infile

  call CountDataSets
  call AllocateArrays
  call ReadInputFile
  call DoCalculations
  call CullDataPoints
  call AddDataPoints
  call WriteOutputFile
  call DeallocateArrays

end do

write(99,*) maxnnRows

close(99) ! close debug file

contains

! this is the end of the "prewap" main program logic

! subroutines between the "contains" line and the "end subroutine prewap"
! line are internal to the "prewap" main program

!*******************************************************************************
!*******************************************************************************
subroutine ReadMasterFiles

! open 'InMaster.in' and 'OutMaster.in' files
open(unit=11, file='InMaster.in')
open(unit=12, file='OutMaster.in')

! read in the RH corrosion limit
read(11,*) CorLim

! read in the number of file names in the files
read(11,*) nFile

! read in input and output file names
do i=1,nFile
 read(11,*) InFileNames(i)
 read(12,*) OutFileNames(i)
end do

! close files
close(unit=11)
close(unit=12)

end subroutine ReadMasterFiles
!*******************************************************************************
!*******************************************************************************
subroutine ReadChemData

! read in log[ Cl] data as a function of RH
open(unit=80, file='CLtable1.dat')
read(80,*) nRowsTable1     ! number of rows in the table
do m=1,nRowsTable1
 read(80,*) RHvector(m), CLtable1a(m), CLtable1b(m), CLtable1c(m)
```

```
end do
close(80)

! read in log[ Cl] data as a function of 1-Qe/Qs
open(unit=80, file='CLtable2.dat')
read(80,*) nRowsTable2     ! number of rows in the table
do m=1,nRowsTable2
 read(80,*) Qvector(m), CLtable2a(m), CLtable2b(m)
end do
close(80)

! log[ Cl] data as a function of 1-Qe/Qs and temperature(C)
open (unit=80, file='CLtable3.dat')

! number of rows and columns in the table
read(80,*) nRowsTable3, nColsTable3

! read in the temperature data
read(80,*) (Tvector3(n), n=1,nColsTable3)
do m=1,7
 read(80,*) Qvector(m), (CLtable3(m,n), n=1,nColsTable3)
end do
close(80)

! pH data as a function of RH
open(unit=80, file='PHtable1.dat')
do m=1,10
 read(80,*) RHvector(m), PHtable1a(m), PHtable1b(m), PHtable1c(m)
end do
close(80)

! pH data as a function of 1-Qe/Qs
open(unit=80, file='PHtable2.dat')
do m=1,nRowsTable2
 read(80,*) Qvector(m), PHtable2a(m), PHtable2b(m)
end do
close(80)

! pH data as a function of 1-Qe/Qs and temperature(C)
open (unit=80, file='PHtable3.dat')
read(80,*) (Tvector3(n), n=1,nColsTable3)
do m=1,nRowsTable3
 read(80,*) Qvector(m), (PHtable3(m,n), n=1,nColsTable3)
end do
close(80)

! pH data as a function of fCO2 and temperature(C)
open (unit=80, file='PHtable4.dat')
read(80,*) nRowsTable4, nColsTable4
read(80,*) (Tvector4(n), n=1,nColsTable4)
do m=1,nRowsTable4
 read(80,*) fCO2vector(m), (PHtable4(m,n), n=1,nColsTable4)
end do
close(80)

open (unit=80, file='InPkgChem.dat')
read(80,*) pHCSNFinpk, phCDSPinpk
read(80,*) Clinpk
close(80)


end subroutine ReadChemData
!*******************************************************************************
```

```
!*******************************************************************************
subroutine CountDataSets

open(unit=70, file=infile)

nDataSets=1          ! initialize # of data sets to 1
maxRows=0     ! initialize the max number of rows to 0

! read past 1st four rows of header information
do i=1,4
 read(70,*) dummy1
end do

! read past header info in line 5 to get number of rows of data
read(70,*) (dummy2(i), i=1,5), rows
write(99,*) rows, "  rows"

! set max number of rows equal to the # of rows in 1st data set
maxRows=rows

! read past next six rows of header information
do i=1,6
 read(70,*) dummy1
end do

! read past the 1st data set
do i=1,rows
 read(70,*) dummy
end do

! read through the rest of the file until the end of the file is reached
do

! read the 1st row header information for the next data set
! if this read occurs at the end of the file, the 'eof' error
! causes the do loop to be exited
 read(70,*,end=100) (dummy2(i), i=1,5), rows
 !write(99,*) rows, "  rows"

! if an 'eof' error did not occur, increment the data set counter
! and read through the given data set
 nDataSets=nDataSets+1

! if the # of rows in the current data set are greater than the current
! max rows value, set max rows equal to the # of rows in the current data set
 if (rows .gt. maxRows) then
  maxRows=rows
 end if

! read through the current data set
 do i=1,(6+rows)
  read(70,*) dummy1
 end do

end do

! line that the 'eof' error causes the do-loop to bails out to
100 continue

! close the data file
close(unit=70)

! write the number of data sets to debug.dat
```

```
write(99,*) nDataSets, "  # of data sets"

end subroutine CountDataSets
!*****************************************************************************
!*****************************************************************************
subroutine AllocateArrays

! set bounds on dynamic arrays whose size is dependent upon the current TH file
allocate (etime(1:maxRows, 1:nDataSets))
allocate (wpT(1:maxRows, 1:nDataSets))
allocate (dsT(1:maxRows, 1: nDataSets))
allocate (dwT(1:maxRows, 1:nDataSets))
allocate (iT(1:maxRows, 1:nDataSets))
allocate (wpRH(1:maxRows, 1:nDataSets))
allocate (dsRH(1:maxRows, 1:nDataSets))
allocate (dwRH(1:maxRows, 1:nDataSets))
allocate (bfRH(1:maxRows, 1:nDataSets))
allocate (iRH(1:maxRows, 1:nDataSets))
allocate (dsLS(1:maxRows, 1:nDataSets))
allocate (iLS(1:maxRows, 1:nDataSets))
allocate (massFracAir(1:maxRows, 1:nDataSets))
allocate (dwFluxWV(1:maxRows, 1:nDataSets))
allocate (dwFluxAir(1:maxRows, 1:nDataSets))
allocate (dsEvapRate(1:maxRows, 1:nDataSets))
allocate (bfEvapRate(1:maxRows, 1:nDataSets))
allocate (iEvapRate(1:maxRows, 1:nDataSets))
allocate (PercFlux5m(1:maxRows, 1:nDataSets))
allocate (tdsPercFlux(1:maxRows, 1:nDataSets))
allocate (iPercFlux(1:maxRows, 1:nDataSets))
allocate (tdsT(1:maxRows, 1:nDataSets))

allocate (nRows(1:nDataSets))
allocate (nnRows(1:nDataSets))
allocate (nnnRows(1:nDataSets))
allocate (fract(1:nDataSets))

end subroutine AllocateArrays
!*****************************************************************************
!*****************************************************************************
subroutine ReadInputFile

open(unit=70, file=infile)        ! open input file

j=1     ! column index for 1st data set

! read past 1st four rows of header information
do i=1,4
 read(70,*) dummy1
end do

! read past header info in line 5 to get number of rows of data
read(70,*)  (dummy2(i), i=1,5), nRows(j)

! read past header info in line 6 to get "fraction of this history" value
read(70,*)  (dummy2(i), i=1,6), fract(j)

! read past next five rows of header information
do i=1,5
 read(70,*) dummy1
end do

! read in data from 1st data set
do i=1, nRows(j)
```

```
 read(70,*) etime(i,j),          &      ! time [ yr]
            wpT(i,j),            & ! temperature - waste package [ C]
          dsT(i,j),            &   ! temperature - drip shield [ C]
          dwT(i,j),            &   ! temperature - drift wall [ C]
          iT(i,j),             &   ! temperature - invert [ C]
          wpRH(i,j),           &   ! rel. humidity - waste package [ -]
          dsRH(i,j),           &   ! rel. humidity - drip shield [ -]
          dwRH(i,j),           &   ! rel. humidity - drift wall [ -]
          bfRH(i,j),           &   ! rel. humidity - backfill [ -]
          iRH(i,j),            &   ! rel. humidity - invert [ -]
          dsLS(i,j),           &   !
            iLS(i,j),            & !
          massFracAir(i,j),    &   ! mass frac. air [
          dwFluxWV(i,j),       &   ! water vapor flux - drift wall [
          dwFluxAir(i,j),      &   ! air flux - drift wall [
          dsEvapRate(i,j),     &   ! evap. rate - drip shield [ m3/yr]
          bfEvapRate(i,j),     &   ! evap. rate - backfill [ m3/yr]
          iEvapRate(i,j),      &
          PercFlux5m(i,j),     &   ! perc flux @ 5m [ mm/yr]
          tdsPercFlux(i,j),    &   ! perc flux -  drip shield top [ mm/yr]
          iPercFlux(i,j),      &   ! perc flux - invert [ mm/yr]
          tdsT(i,j)                    ! temperature -  drip shield top [ C]
end do

! now read in data for data sets 2 to nDataSets
do j=2,nDataSets

! read past header info in line 5 to get number of rows of data
 read(70,*) (dummy2(i), i=1,5), nRows(j)

! read past header info in line 6 to get "fraction of this history" value

 read(70,*) (dummy2(i), i=1,6), fract(j)

! read past next five rows of header information
 do i=1,5
  read(70,*) dummy1
 end do

! read in data from the j-th data set
 do i=1, nRows(j)
  read(70,*) etime(i,j),                                             &
             wpT(i,j), dsT(i,j), dwT(i,j), iT(i,j),                   &
             wpRH(i,j), dsRH(i,j), dwRH(i,j), bfRH(i,j), iRH(i,j),    &
             dsLS(i,j), iLS(i,j),                                     &
             massFracAir(i,j),                                        &
             dwFluxWV(i,j), dwFluxAir(i,j),                           &
             dsEvapRate(i,j), bfEvapRate(i,j), iEvapRate(i,j),        &
           PercFlux5m(i,j), tdsPercFlux(i,j), iPercFlux(i,j),       &
           tdsT(i,j)
 end do

end do

close(unit=70)            ! close the data file

end subroutine ReadInputFile
!*******************************************************************************
!*******************************************************************************
subroutine DoCalculations

allocate (wpPHnd(1:maxRows, 1:nDataSets))
allocate (wpCLnd(1:maxRows, 1:nDataSets))
```

```
allocate (wpPHd(1:maxRows, 1:nDataSets))
allocate (wpCLd(1:maxRows, 1:nDataSets))

allocate (dsPHnd(1:maxRows, 1:nDataSets))
allocate (dsCLnd(1:maxRows, 1:nDataSets))

allocate (dsPHd(1:maxRows, 1:nDataSets))
allocate (dsCLd(1:maxRows, 1:nDataSets))

allocate (ipkPHnd(1:maxRows, 1:nDataSets))
allocate (ipkCLnd(1:maxRows, 1:nDataSets))

allocate (ipkPHd(1:maxRows, 1:nDataSets))
allocate (ipkCLd(1:maxRows, 1:nDataSets))

allocate (barPHnd(1:maxRows, 1:nDataSets))
allocate (barCLnd(1:maxRows, 1:nDataSets))

allocate (barPHd(1:maxRows, 1:nDataSets))
allocate (barCLd(1:maxRows, 1:nDataSets))

! perform calculations at each "i" time for all "j" data sets
do j=1,nDataSets
 do i=1,nRows(j)

! NOTE: CL chemistry is NOT calculated in this code version. Instead, pH^2 is
! reported in the wpCLd, wpCLnd, dsCLd, and dsCLnd variables

! calculate waste package in-drift pH and pH^2 for drip and no drip conditions
  call InDriftCalc(etime(i,j), wpT(i,j), wpRH(i,j),                &
                    dsEvapRate(i,j), tdsPercFlux(i,j),             &
                RHvector, Qvector, Tvector3, Tvector4,         &
                    CLtable1a, CLtable1b, CLtable1c,               &
                    CLtable2a, CLtable2b,                          &
                    CLtable3,                                      &
                    PHtable1a, PHtable1b, PHtable1c,               &
                    PHtable2a, PHtable2b,                          &
                    PHtable3, PHtable4,                            &
                    nRowsTable1, nRowsTable2,                      &
                nRowsTable3, nColsTable3,                      &
                nRowsTable4, nColsTable4,                      &
                wpPHd(i,j), wpCLd(i,j), wpPHnd(i,j), wpCLnd(i,j),  &
                i, j, infile)

! calculate drip shield in-drift pH and pH^2 for drip and no drip conditions
  call InDriftCalc(etime(i,j), dsT(i,j), dsRH(i,j),                &
                    dsEvapRate(i,j), tdsPercFlux(i,j),             &
                RHvector, Qvector, Tvector3, Tvector4,         &
                    CLtable1a, CLtable1b, CLtable1c,               &
                    CLtable2a, CLtable2b,                          &
                    CLtable3,                                      &
                    PHtable1a, PHtable1b, PHtable1c,               &
                    PHtable2a, PHtable2b,                          &
                    PHtable3, PHtable4,                            &
                nRowsTable1, nRowsTable2,                      &
                nRowsTable3, nColsTable3,                      &
                nRowsTable4, nColsTable4,                      &
                dsPHd(i,j), dsCLd(i,j), dsPHnd(i,j), dsCLnd(i,j),  &
                i, j, infile)

! set bounding in-package pH for CSNF or HLW
if (infile(1:4) .eq. 'CSNF') then
```

```
 ipkPHbounding=phCSNFinpk        ! CSNF bounding pH value
else
 ipkPHbounding=phCDSPinpk        ! HLW bounding pH value
end if

! in-package drip pH is set equal to bounding values

  ipkPHd(i,j)=ipkPHbounding
  ipkCLd(i,j)=ipkPHd(i,j)*ipkPHd(i,j)
!  ipkCLd(i,j)=CLinpk                    ! mol/kg

! in-package no drip pH is set equal to -9.99E-02
! (default 'don't exist' values)
  ipkPHnd(i,j)=-9.99E-02
  ipkCLnd(i,j)=ipkPHnd(i,j)*ipkPHnd(i,j)
!  ipkCLd(i,j)=-9.99E-02                 ! mol/kg

! barrier drip and no drip pH are set equal to -9.99E-02
! (default 'don't exist' values)
  barPHd(i,j)=-9.99E-02
  barCLd(i,j)=barPHd(i,j)*barPHd(i,j)
!  barCLd(i,j)=-9.99E-02
  barPHnd(i,j)=-9.99E-02
  barCLnd(i,j)=barPHnd(i,j)*barPHnd(i,j)
!  barCLnd(i,j)=-9.99E-02

 end do
end do

end subroutine DoCalculations
!******************************************************************************
!******************************************************************************
subroutine CullDataPoints

open(unit=72, file='temp.dat')          ! open temporary storage file

! loop through all of the data sets
do j=1,nDataSets

 ! initialize counter for number of rows that will get written
 ! to temporary storage file
 nnRows(j)=0

 do i=1,nRows(j)-1

  corFlag=0  ! initialize corrosion flag to 0 (no corrosion)

  ! skip row if wpT or dsT 'do not exist'
  if(wpT(i,j) .le. 0.0 .or. dsT(i,j) .le. 0.0) then
   ! write to debug file
   !write(99,*) etime(i,j), "  trapped on no wpT or dsT"

  ! write row if wpRH or dsRH is equal or above corrosion limit
  elseif( (wpRH(i,j) .ge. CorLim) .or. (dsRH(i,j) .ge. CorLim) ) then
   corFlag=1
   !write(99,*) etime(i,j), "  RH above corrosion limit"

  ! write row for wp no corrosion/corrosion transition
  elseif( (wpRH(i,j) .lt. 0.501) .and. (wpRH(i+1,j) .ge. 0.501) ) then
   corFlag=1
   !write(99,*) etime(i,j), "  wp no cor/cor transition"

  ! write row for ds no corrosion/corrosion transition
```

```
  elseif( (dsRH(i,j) .lt. 0.501) .and. (dsRH(i+1,j) .ge. 0.501)  ) then
   corFlag=1
   !write(99,*) etime(i,j), "  ds no cor/cor transition"

  ! write row for wp corrosion/no corrosion transition
  elseif( (wpRH(i,j) .lt. 0.501) .and. (wpRH(i-1,j) .ge. 0.501) ) then
   corFlag=1
   !write(99,*) etime(i,j), "  wp cor/no cor transition"

  ! write row for ds corrosion/no corrosion transition
  elseif( (dsRH(i,j) .lt. 0.501) .and. (dsRH(i-1,j) .ge. 0.501) ) then
   corFlag=1
   !write(99,*) etime(i,j), "  ds cor/no cor transition"

  ! skip row if in middle of no corrosion
  elseif( (wpRH(i,j) .lt. 0.501) .and.  &
          (wpRH(i-1,j) .lt. 0.501) .and.  &
          (wpRH(i+1,j) .lt. 0.501) ) then

   !write(99,*) etime(i,j), "  trapped on middle of no corrosion (wp)"
   ! trap

  elseif( (dsRH(i,j) .lt. 0.501) .and.  &
          (dsRH(i-1,j) .lt. 0.501) .and.  &
         (dsRH(i+1,j) .lt. 0.501) ) then

   !write(99,*) etime(i,j), "  trapped on middle of no corrosion (ds)"
   ! trap

  else
 ! middle of corrosion
   corFlag=1
   !write(99,*) etime(i,j), "  default"

  end if

  ! write the i-th row of data to the temp file if corFlag=1
  if(corFlag .eq. 1) then
   write(72,1020) etime(i,j), wpT(i,j), wpRH(i,j), dsT(i,j), dsRH(i,j),    &
             wpPHnd(i,j), wpCLnd(i,j), wpPHd(i,j), wpCLd(i,j),       &
           dsPHnd(i,j), dsCLnd(i,j), dsPHd(i,j), dsCLd(i,j),       &
           ipkPHnd(i,j), ipkCLnd(i,j), ipkPHd(i,j), ipkCLd(i,j),  &
           barPHnd(i,j), barCLnd(i,j), barPHd(i,j), barCLd(i,j),  &
           PercFlux5m(i,j)
   1020 format(22(ES10.3, "  "))

   ! increment the number of rows stored for the j-th time history
   nnRows(j)=nnRows(j)+1

  end if

 end do

 ! write the last time history to the temp file
 i=nRows(j)
 write(72,1020) etime(i,j), wpT(i,j), wpRH(i,j), dsT(i,j), dsRH(i,j),    &
             wpPHnd(i,j), wpCLnd(i,j), wpPHd(i,j), wpCLd(i,j),       &
           dsPHnd(i,j), dsCLnd(i,j), dsPHd(i,j), dsCLd(i,j),       &
           ipkPHnd(i,j), ipkCLnd(i,j), ipkPHd(i,j), ipkCLd(i,j),  &
           barPHnd(i,j), barCLnd(i,j), barPHd(i,j), barCLd(i,j),  &
           PercFlux5m(i,j)

 ! increment the number of rows stored for the j-th time history
```

```
 nnRows(j)=nnRows(j)+1

 write(99,*) j, nnRows(j)


 if(nnRows(j) .gt. maxnnRows) then
  maxnnRows=nnRows(j)
 end if

end do

close(72)

end subroutine CullDataPoints
!***********************************************************************
!***********************************************************************
subroutine AddDataPoints

open(unit=72, file='temp.dat')          ! open temporary storage files
open(unit=73, file='temp2.dat')


do j=1, nDataSets           ! loop through the time histories

  allocate (TempStorage(1:nnRows(j), 1:22))          ! set TempStorage array size

 ! initialize counter for number of rows to be written to the WAPDEG input file
 ! for the j-th time history
 nnnRows(j)=nnRows(j)

 ! read j-th time history from temp.dat file
 do i=1,nnRows(j)
  read(72,*) (TempStorage(i,m), m=1,22)
 end do

  do i=1,nnRows(j)-1       ! loop through all but the last row of data

  ! check times between 50,000 and 200,000 years to see
  ! if time steps are <= 10,000 years
  if((TempStorage(i,1) .ge. 50000.0)  .and.   &
     (TempStorage(i,1) .lt. 200000.0)) then

   ! if time step is greater than 10,000 years write current row of data
   ! to temp2.dat and call subroutine to add interpolated data and times
   ! at 10,000 year intervals between the i-th and i-th+1 rows
   if(TempStorage(i+1,1)-TempStorage(i,1) .gt. 10000.0) then
    write(73,1020) (TempStorage(i,m), m=1,22)
     1020 format(22(ES10.3, "  "))
    call AddPoints1

   else
    ! if time step is <= 10,000 years write current row of data to temp2.dat
    write(73,1020) (TempStorage(i,m), m=1,22)

   endif

  ! check times after 200,000 years to see
  ! if time steps are <= 100,000 years
  elseif(TempStorage(i,1) .ge. 200000.0) then

   ! if time step is greater than 100,000 years write current row of data
   ! to temp2.dat and call subroutine to add interpolated data and times
   ! at 100,000 year intervals between the i-th and i-th+1 rows
```

```
    if(TempStorage(i+1,1)-TempStorage(i,1) .gt. 100000.0) then
     write(73,1020) (TempStorage(i,m), m=1,22)
      call AddPoints2

    else
     ! if time step is <= 100,000 years write current row of data to temp2.dat
     write(73,1020) (TempStorage(i,m), m=1,22)

    endif

   else
   ! if time is <= 50,000 years write current row of data to temp2.dat
    write(73,1020) (TempStorage(i,m), m=1,22)

   endif

  end do

  ! write the last row of data to the temp2.dat file
  write(73,1020) (TempStorage(nnRows(j),m), m=1,22)

  deallocate (TempStorage)        ! deallocate the TempStorage array

 end do

 close(72)    ! close temporary files
 close(73)

 end subroutine AddDataPoints
!*******************************************************************************
!*******************************************************************************
subroutine WriteOutputFile

open(unit=71, file=outfile)             ! open output file
open(unit=73, file='temp2.dat')  ! open temporary storage file

! write initial comment lines
write(71,1011)
1011 format('! 1st comment line')
write(71,1012)
1012 format('! 2nd comment line')
write(71,1013)
1013 format('! 3rd comment line')

write(71,1014) nDataSets
1014 format('# ', I4, '  21')        ! # of datasets and # of columns of data
                                     ! WAPDEG guys don't want 22nd column
do j=1,nDataSets
 write(71,1015) nnnRows(j)
 1015 format('# ', I4)                ! # of rows in the j-th dataset

 write(71,1016) fract(j)
 1016 format('# ', ES10.3)        ! fraction of packages

 write(71,1018)                          ! writes header line
 1018 format('!  t          ', '  wpT        ', '  wpRH       ',   &
          '  dsT       ', '  dsRH       ', '  wpPHnd     ',    &
        '  wpCLnd     ', '  wpPHd      ', '  wpCLd      ',     &
        '  dsPHnd     ', '  dsCLnd     ', '  dsPHd      ',     &
        '  dsCLd      ', '  ipkPHnd    ', '  ipkCLnd    ',     &
        '  ipkPHd     ', '  ipkCLd     ', '  barPHnd    ',     &
        '  barCLnd    ', '  barPHd     ', '  barCLd     ',     &
         '  PercFlux5m')
```

```
  1020 format(22(ES10.3, "  "))
  do i=1,nnnRows(j)
   read(73,*) (ReadVector(m), m=1,22)
   write(71,1020) (ReadVector(m), m=1,22)
  end do


 end do

 write(99,*)

 close(unit=71)              ! close output file
 close(unit=73)              ! close temporary storage file

 end subroutine WriteOutputFile
!*******************************************************************************
!*******************************************************************************
 subroutine DeallocateArrays

 ! deallocate arrays
 deallocate (etime, wpT,    dsT, dwT, iT, wpRH, dsRH, dwRH, bfRH, iRH)
 deallocate (dsLS, iLS, massFracAir, dwFluxWV, dwFluxAir)
 deallocate (dsEvapRate, bfEvapRate, iEvapRate)
 deallocate (PercFlux5m, tdsPercFlux, iPercFlux, tdsT)

 deallocate (fract)

 deallocate (wpPHnd, wpCLnd, wpPHd,  wpCLd)
 deallocate (dsPHnd,  dsCLnd,  dsPHd, dsCLd)
 deallocate (ipkPHnd,  ipkCLnd,  ipkPHd,  ipkCLd)
 deallocate (barPHnd,  barCLnd,  barPHd,  barCLd)

 deallocate (nRows, nnRows, nnnRows)

 end subroutine DeallocateArrays
!*******************************************************************************
!*******************************************************************************
 subroutine AddPoints1

 ! check for time step spanning across 200,000 years
 if (TempStorage(i+1,1) .gt. 200000.0) then
  ! if it does set two time steps
  ! one for <= 200,000 and one for > 200,000 years
  delTime1=200000.0-TempStorage(i,1)
  delTime2=800000.0

  ! calculate the number of extra points to be added
  numExtraPoints1 = ceiling(delTime1/10000)
  numExtraPoints2 = ceiling(delTime2/100000)-1


  if (numExtraPoints1 .ne. 0) then
  ! generate interpolated data for the extra points to be added
   do ii=1,numExtraPoints1
    do jj=2,22
     delDep=TempStorage(i+1,jj)-TempStorage(i,jj)
     newValue(jj)=TempStorage(i,jj) + delDep*(10000*ii/delTime1)
    end do
    1021 format(22(ES10.3, "  "))
    ! write the interpolated data to the temp2.dat file
    write(73,1021) TempStorage(i,1)+10000*ii, (newValue(m), m=2,22)
   end do
```

```
 end if

 if (numExtraPoints2 .ne. 0) then
  ! generate interpolated data for the 1st extra point to be added
  do jj=2,22
   delDep=TempStorage(i+1,jj)-TempStorage(i,jj)
   newValue(jj)=TempStorage(i,jj) +  &
                delDep*(200000-TempStorage(i,1))/(TempStorage(i+1,1)-TempStorage(i,1))
  end do
  ! write the interpolated data to the temp2.dat file
  write(73,1021) 300000.0, (newValue(m), m=2,22)

  ! generate interpolated data for the remaining extra points to be added
  do ii=2,numExtraPoints2
   do jj=2,22
    delDep=TempStorage(i+1,jj)-TempStorage(i,jj)
    newValue(jj)=TempStorage(i,jj) + delDep*(100000*ii/delTime2)
   end do
   ! write the interpolated data to the temp2.dat file
   write(73,1021) 200000.0+100000.0*ii, (newValue(m), m=2,22)
  end do
 end if

 ! increment the number of rows of the j-ht time history by the number of points added
 nnnRows(j)=nnnRows(j)+numExtraPoints1+numExtraPoints2

else
 ! time step doesn't span 200,000 years

 ! calculate the number of extra points to be added
 delTime=TempStorage(i+1,1)-TempStorage(i,1)
 numExtraPoints = ceiling(delTime/10000)-1

 if (numExtraPoints .eq. 0) then
  return
 end if

 ! generate interpolated data for the points to be added
 do ii=1,numExtraPoints
  do jj=2,22
   delDep=TempStorage(i+1,jj)-TempStorage(i,jj)
   newValue(jj)=TempStorage(i,jj) + delDep*(10000*ii/delTime)
  end do
  ! write the interpolated data to the temp2.dat file
  write(73,1021) TempStorage(i,1)+10000*ii, (newValue(m), m=2,22)
 end do

 ! increment the number of rows of the j-ht time history by the number of points added
 nnnRows(j)=nnnRows(j)+numExtraPoints

end if

end subroutine AddPoints1
!********************************************************************************
!********************************************************************************
subroutine AddPoints2

delTime=TempStorage(i+1,1)-TempStorage(i,1)

numExtraPoints = ceiling(delTime/100000)-1

if (numExtraPoints .eq. 0) then
 return
```

```
end if

do ii=1,numExtraPoints
 do jj=2,22
  delDep=TempStorage(i+1,jj)-TempStorage(i,jj)
  newValue(jj)=TempStorage(i,jj) + delDep* (100000*ii/delTime)
 end do
 write(73,1022) TempStorage(i,1)+100000*ii, (newValue(m), m=2,22)
 1022 format(22(ES10.3, "  "))
end do

nnnRows(j)=nnnRows(j)+numExtraPoints

end subroutine AddPoints2
!****************************************************************************
!****************************************************************************
end program prewap

! subroutines past this point are external to the "prewap" main program

!****************************************************************************
!****************************************************************************
! calualate the pH and Cl under drip and no drip conditions

! NOTE: CL chemistry is NOT calculated in this code version. Instead, pH^2 is
! reported in the wpCLd, wpCLnd, dsCLd, and dsCLnd variables

subroutine InDriftCalc(etime, T, RH, EvapRate, SeepRate,          &
               RHvector, Qvector, Tvector3, Tvector4,        &
                   CLtable1a, CLtable1b, CLtable1c,               &
                   CLtable2a, CLtable2b,                          &
                   CLtable3,                                      &
                   PHtable1a, PHtable1b, PHtable1c,               &
                   PHtable2a, PHtable2b,                          &
                   PHtable3, PHtable4,                            &
               nRowsTable1, nRowsTable2,                     &
               nRowsTable3, nColsTable3,                     &
               nRowsTable4, nColsTable4,                     &
                   PHd, CLd, PHnd, CLnd,                          &
                  i, j, infile)

real(8) RHvector(nRowsTable1), Qvector(nRowsTable2), fCO2vector(nRowsTable4)
real(8) Tvector3(nColsTable3), Tvector4(nColsTable4)
real(8) CLtable1a(nRowsTable1), CLtable1b(nRowsTable1), CLtable1c(nRowsTable1)
real(8) CLtable2a(nRowsTable2), CLtable2b(nRowsTable2)
real(8) CLtable3(nRowsTable3,nColsTable3)
real(8) PHtable1a(nRowsTable1), PHtable1b(nRowsTable1), PHtable1c(nRowsTable1)
real(8) PHtable2a(nRowsTable2), PHtable2b(nRowsTable2)
real(8) PHtable3(nRowsTable3,nColsTable3)
real(8) PHtable4(nRowsTable4,nColsTable4)

real(8) etime, T, RH, Qratio, EvapRate, SeepRate
real(8) PHd, CLd, PHnd, CLnd, logCLd, logfCO2

integer(4) nRowsTable1, nRowsTable2, nRowsTable3, nColsTable3
integer(4) i, j

character*15 infile

! trap for temperatures and seep rates that "don't exist"
if( (T .lt. 0.0) .or. (SeepRate .lt. -99.0)) then

! drip and no drip pH are set equal to -9.99E-02
```

```
! (default 'don't exist' values)
 PHd=-9.99E-02
 PHnd=-9.99E-02

! place holder for CL calculations
!  CLd=-9.99E-02
!  CLnd=-9.99E-02

else

! calculate 1-Qe/Qs
 if( SeepRate .eq. 0.0) then          ! sets 1-Qe/Qs equal to 0.0 when Qs=0
  Qratio=0.0
 else
  Qratio=1.0 - abs(EvapRate/SeepRate)
 end if

! determine what range of in-drift chemistry data is applicable, then
! calculate pH and pH^2

! 1st period (< 50 years -- pre-closure) ************************************
 if(etime .lt. 50.0) then

  ! drip and no drip pH are set equal to -9.99E-02
  ! (default 'don't exist' values)
  PHd=-9.99E-02
  PHnd=-9.99E-02

! place holder for CL calculations
   ! drip and no drip CL are set equal to -9.99E-02
   ! (default 'don't exist' values)
!  CLd=-9.99E-02
!  CLnd=-9.99E-02

! 2nd time period (50 to 1000 years) ****************************************
 elseif( (etime .gt. 50.0) .and. (etime .le. 1000.0) ) then

  logfCO2=-6.5

  ! 1st range (RH <= 50)
  if (100*RH .le. 50.0) then
   !!write(99,*) "1st range"

   ! drip and no drip pH are set equal to -9.99E-02
   ! (default 'don't exist' values)
   PHd=-9.99E-02
   PHnd=-9.99E-02

! place holder for CL calculations
   ! drip and no drip CL are set equal to -9.99E-02
   ! (default 'don't exist' values)
!  CLd=-9.99E-02
!  CLnd=-9.99E-02

  ! 2nd range (50 <= RH <= 85)
  elseif( (100*RH .ge. 50.0) .and.  &
          (100*RH .le. 85.0) ) then

   !!write(99,*) "2nd range"

   PHd=9.40  ! drip pH is constant in this range
   call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                 nRowsTable4, nColsTable4, PHnd)
```

```
! place holder for CL calculations
!   call Interp1D(100*RH, RHvector, CLtable1a, nRowsTable1, logCLd)
!   CLd=10**logCLd
!   CLnd=-9.99E-02

  ! 3rd range (RH > 85)
  elseif(100*RH .gt. 85.0) then

   !!write(99,*) "3rd range"

   call Interp1D(Qratio, Qvector, PHtable2a, nRowsTable2, PHd)
   call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                 nRowsTable4, nColsTable4, PHnd)

! place holder for CL calculations
!   call Interp1D(Qratio, Qvector, CLtable2a, nRowsTable2, logCLd)
!   CLd=10**logCLd
!   CLnd=-9.99E-02

  else
   !!write(99,*) "failed all 2nd period tests"
  end if

 ! 3rd time period (1000 years to 2000 years) *************************
 elseif( (etime .gt. 1000.0) .and. (etime .le. 2000.0) ) then

  logfCO2=-3.0

 ! 1st range (RH < 50)
  if (100*RH .lt. 50.0) then
   !!write(99,*) "1st range"

   ! drip and no drip pH are set equal to -9.99E-02
   ! (default 'don't exist' values)
   PHd=-9.99E-02
   PHnd=-9.99E-02

! place holder for CL calculations
   ! drip and no drip CL are set equal to -9.99E-02
   ! (default 'don't exist' values)
!   CLd=-9.99E-02
!   CLnd=-9.99E-02

  ! 2nd range (50 <= RH <= 85)
  elseif( (100*RH .ge. 50.0) .and.  &
          (100*RH .le. 85.0) ) then

   !!write(99,*) "2nd range"


   PHd=7.64   ! drip pH is constant in this range
   call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                 nRowsTable4, nColsTable4, PHnd)

! place holder for CL calculations
!   call Interp1D(100*RH, RHvector, CLtable1b, nRowsTable1, logCLd)
!   CLd=10**logCLd
!   CLnd=-9.99E-02

  ! 3rd range (RH > 85)
  elseif(100*RH .gt. 85.0) then
```

```
    !!write(99,*) "3rd range"
    call Interp1D(Qratio, Qvector, PHtable2b, nRowsTable2, PHd)
    call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                  nRowsTable4, nColsTable4, PHnd)

 ! place holder for CL calculations
 !    call Interp1D(Qratio, Qvector, CLtable2b, nRowsTable2, logCLd)
 !    CLd=10**logCLd
 !    CLnd=-9.99E-02

   else
    !!write(99,*) "failed all 3nd period tests"
   end if

  ! 4th time period (2000 year to 100,000 years) ******************
  elseif( (etime .gt. 2000.0) .and. (etime .le. 100000.0) ) then

   logfCO2=-2.0

   ! 1st range (RH < 50)
   if (100*RH .lt. 50.0) then
    !!write(99,*) "1st range"

    PHd=-9.99E-02
    PHnd=-9.99E-02

 ! place holder for CL calculations
 !    CLd=-9.99E-02
 !    CLnd=-9.99E-02

   ! 2nd range (50 <= RH <= 85)
   elseif( (100*RH .ge. 50.0) .and.  &
           (100*RH .le. 85.0) ) then

    !!write(99,*) "2nd range"

    PHd=7.02   ! drip pH is constant in this range
    call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                  nRowsTable4, nColsTable4, PHnd)

 ! place holder for CL calculations
 !    call Interp1D(100*RH, RHvector, CLtable1b, nRowsTable1, logCLd)
 !    CLd=10**logCLd
 !    CLnd=-9.99E-02

   ! 3rd range (RH > 85)
   else

    !!write(99,*) "3rd range"

    call Interp2D(Qratio, T, Qvector, Tvector3, PHtable3, &
                 nRowsTable3, nColsTable3, PHd)
    call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                  nRowsTable4, nColsTable4, PHnd)

 ! place holder for CL calculations
 !    call Interp2D(Qratio, T, Qvector, Tvector, CLtable3, &
 !                  nRowsTable3, nColsTable3, logCLd)
 !    CLd=10**logCLd
 !    CLnd=-9.99E-02

   end if
```

```
 ! 5th time period (>100,000 years) ******************
 else

  logfCO2=-3.0

  ! 1st range (RH < 50)
  if (100*RH .lt. 50.0) then
   !!write(99,*) "1st range"

   PHd=-9.99E-02
   PHnd=-9.99E-02

! place holder for CL calculations
!   CLd=???
!   CLnd=-9.99E-02

  ! 2nd range (50 <= RH <= 85)
  elseif( (100*RH .ge. 50.0) .and.  &
          (100*RH .le. 85.0) ) then

   !!write(99,*) "2nd range"

   PHd=7.64  ! drip pH is constant in this range
   call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                 nRowsTable4, nColsTable4, PHnd)

! place holder for CL calculations
!   call Interp1D(100*RH, RHvector, CLtable1b, nRowsTable1, logCLd)
!   CLd=10**logCLd
!   CLnd=-9.99E-02

  ! 3rd range (RH > 85)
  elseif(100*RH .gt. 85.0) then

   !!write(99,*) "3rd range"
   call Interp1D(Qratio, Qvector, PHtable2b, nRowsTable2, PHd)
   call Interp2D(logfCO2, T, fCO2vector, Tvector4, PHtable4, &
                 nRowsTable4, nColsTable4, PHnd)

! place holder for CL calculations
!   call Interp1D(Qratio, Qvector, CLtable2b, nRowsTable2, logCLd)
!   CLd=10**logCLd
!   CLnd=-9.99E-02

  end if

 end if

end if

! substitute pH^2 values in place of CL values
CLd=PHd*PHd
CLnd=PHnd*PHnd

  !!write(99,*) etime, " etime"
  !!write(99,*) j, i, " j-th dataset, i-th time"
  !!write(99,*) T, " temp"
  !!write(99,*) RH, " RH"
  !!write(99,*) EvapRate, " evap rate"
  !!write(99,*) SeepRate, " seep rate"
  !!write(99,*) Qratio, " Qe/Qs"
  !!write(99,*)
```

```
end subroutine InDriftCalc
!******************************************************************************
!******************************************************************************
! 1-D interpolation routine
subroutine Interp1D(ind, IndData, DepData, nRows, dep)

! number of rows in 1-D table
integer(4) nRows

! independent and dependent variable vectors
real(8) IndData(nRows), DepData(nRows)

! independent and dependent variables
real(8) ind, dep

! check for independent variable outside of data set range
if (ind .le. IndData(1)) then
 dep=DepData(1)                    ! value is below lower bound, set equal to floor
elseif (ind .ge. IndData(nRows)) then
 dep=DepData(nRows)         ! value is above upper bound, set equal to ceiling
else

 do i=1,nRows-1             ! value is within the range of the data set
  if ((ind .ge. IndData(i)) .and. (ind .lt. IndData(i+1))) then

   ! linear interpolation
   ! y = y(i) + [x-x(i)]/[x(i+1)-x(i)] * [y(i+1)-y(i)]
   dep=DepData(i)  &
          + (ind-IndData(i))/(IndData(i+1)-IndData(i))  &
           *(DepData(i+1)-DepData(i))
  end if
 end do
end if

end subroutine Interp1D
!******************************************************************************
!******************************************************************************
subroutine Interp2D(ind1, ind2, IndData1, IndData2, DepData, nRows, nCols, dep)

! number of rows and columns in 2-D table
integer(4) nRows, nCols

! independent variable vectors and dependent variable array
real(8) IndData1(nRows), IndData2(nCols), DepData(nRows,nCols)

! independent variables, intermediate dependent variables, and dependent variable
real(8) ind1, ind2, dep1, dep, dep2

! flags for independent variable values beyond upper and lower bounds
integer(4) iflag_lb, iflag_ub, jflag_lb, jflag_ub

!!write(99,*) "in Interp2D"
!!write(99,*) ind1, " ind1"
!!write(99,*) ind2, " ind2"

! initialize flags
iflag_lb = 0
iflag_ub = 0
jflag_lb = 0
jflag_ub = 0

! determine i-index
if (ind1 .le. IndData1(1)) then
```

```
 i=1       ! ind1 less than lower bound
 iflag_lb = 1
elseif (ind1 .ge. IndData1(nRows)) then
 i=nRows     ! ind1 greater than upper bound
 iflag_ub = 1

else

 do ii=1,nRows-1
  if ((ind1 .ge. IndData1(ii)) .and. (ind1 .lt. IndData1(ii+1))) then
   i=ii   ! ind1 is between IndData1(ii) and IndData1(ii+1)
  end if
 end do

end if

!!write(99,*) i, " i"


! determine j-index
if (ind2 .le. IndData2(1)) then
 j=1       ! ind2 less than lower bound
 jflag_lb = 1

elseif (ind2 .ge. IndData1(nCols)) then
 j=nCols     ! ind2 greater than upper bound
 jflag_ub = 1

else

 do jj=1,nCols-1
  if ((ind2 .ge. IndData2(jj)) .and. (ind2 .lt. IndData2(jj+1))) then
   j=jj      ! ind2 is between IndData2(jj) and IndData2(jj+1)
  end if
 end do

end if

!!write(99,*) j, " j"

! logic trap to catch points below the lower bound of the table
if(iflag_lb .eq. 1) then   ! outside lower bound
 if( (jflag_lb .eq. 1) .or. (jflag_ub .eq. 1) ) then
 dep=DepData(i,j)    .      ! corner point
 else
 ! linearly interpolate along lower edge
  dep=DepData(i,j)  &
     + (ind2-IndData2(j))/(IndData2(j+1)-IndData2(j))  &
         * (DepData(i,j+1)-DepData(i,j))
 end if
end if

! logic trap to catch points above the upper bound of the table
if(iflag_ub .eq. 1) then   ! outside upper bound
 if( (jflag_lb .eq. 1) .or. (jflag_ub .eq. 1) ) then
  dep=DepData(i,j)          ! corner point
 else
 ! linearly interpolate along upper edge
  dep=DepData(i,j)  &
     + (ind2-IndData2(j))/(IndData2(j+1)-IndData2(j))  &
         * (DepData(i,j+1)-DepData(i,j))
 end if
end if
```

```
! logic trap to catch points beyond the left and right bound of the table
if( (jflag_lb .eq. 1) .or. (jflag_ub .eq. 1) ) then
 ! outside right or left bound
 if( (iflag_lb .eq. 1) .or. (iflag_ub .eq. 1) ) then
  ! trap for corner points (already calculated)
 else
  ! linearly interpolate along left or right edge
  dep=DepData(i,j)  &
     + (indl-IndDatal(i))/(IndDatal(i+1)-IndDatal(i))  &
        *(DepData(i+1,j)-DepData(i,j))
 end if
end if

! logic trap to catch points in the table
if( (iflag_lb .eq. 0) .and. (iflag_ub .eq. 0) .and.  &
    (jflag_lb .eq. 0) .and. (jflag_ub .eq. 0) ) then

! interpolate in j-th column between the i-th and (i+1)-th row
dep1i=DepData(i,j)  &
    + (indl-IndDatal(i))/(IndDatal(i+1)-IndDatal(i))  &
       *(DepData(i+1,j)-DepData(i,j))

! interpolate in (j+1)-th column between the i-th and (i+1)-th row
dep2i=DepData(i,j+1)  &
    + (indl-IndDatal(i))/(IndDatal(i+1)-IndDatal(i))  &
       *(DepData(i+1,j+1)-DepData(i,j+1))

! interpolate the results above results between
! the j-th and (j+1)-th columns
dep=dep1i           &
  +(ind2-IndData2(j))/(IndData2(j+1)-IndData2(j))      &
    *(dep2i-dep1i)

end if

end subroutine Interp2D
!*******************************************************************************
!*******************************************************************************
```

VII.3.3 COMPUTER LISTING OF TEST DATA INPUT AND OUTPUT

VII.3.3.1 SOURCE CODE FOR TESTING THE INTERPOLATION SUBROUTINES

```
program PWinterp

real(8) CLvector(10), CLtable1a(10)
real(8) Tvector(3), Qvector(7), PHtable3(7,3)
real(8) nInput1D, nInput2D
real(8) indVar1, indVar2
real(8) dep

integer(4) nRows1
integer(4) nRows2, nCols2

call ReadChemData

open(unit=90, file='input1D.dat')
open(unit=91, file='output1D.dat')
read(90,*) nInput1D
```

```
do i=1,nInput1D
 read(90,*) indVar1
 call Interp1D(indVar1, CLvector, CLtable1a, nRows1, dep)
 write(91,*) indVar1, dep
end do
close(90)
close(91)

open(unit=90, file='input2D.dat')
open(unit=91, file='output2D.dat')
read(90,*) nInput2D
do i=1,nInput2D
 read(90,*) indVar1, indVar2
 call Interp2D(indVar1, indVar2, Qvector, Tvector, PHtable3, nRows2, nCols2, dep)
 write(91,*) indVar1, indVar2, dep
end do
close(90)
close(91)

contains

!*******************************************************************************
subroutine ReadChemData

! CL data as a function of RH (1-D table)
open(unit=80, file='CLtable1.dat')
read(80,*) nRows1

do m=1,nRows1
 read(80,*) CLvector(m), CLtable1a(m)
end do
close(80)

! pH data as a function of 1-Qe/Qs and temperature(C) (2-D table)
open (unit=80, file='PHtable3.dat')
read(80,*) nRows2, nCols2
read(80,*) (Tvector(n), n=1,nCols2)
do m=1,nRows2
 read(80,*) Qvector(m), (PHtable3(m,n), n=1,nCols2)
end do
close(80)

end subroutine ReadChemData
!*******************************************************************************

end program PWinterp

!*******************************************************************************
!*******************************************************************************
! 1-D interpolation routine
subroutine Interp1D(ind, IndData, DepData, nRows, dep)

! number of rows in 1-D table
integer(4) nRows

! independent and dependent variable vectors
real(8) IndData(nRows), DepData(nRows)

! independent and dependent variables
real(8) ind, dep

! check for independent variable outside of data set range
if (ind .le. IndData(1)) then
```

```
 dep=DepData(1)                    ! value is below lower bound, set equal to floor
elseif (ind .ge. IndData(nRows)) then
 dep=DepData(nRows)         ! value is above upper bound, set equal to ceiling
else

 do i=1,nRows-1             ! value is within the range of the data set
  if ((ind .ge. IndData(i)) .and. (ind .lt. IndData(i+1))) then

   ! linear interpolation
   ! y = y(i) + [x-x(i)]/[x(i+1)-x(i)] * [y(i+1)-y(i)]
   dep=DepData(i)  &
          + (ind-IndData(i))/(IndData(i+1)-IndData(i))  &
                 *(DepData(i+1)-DepData(i))
  end if
 end do
end if

end subroutine Interp1D
!*****************************************************************************
!*****************************************************************************
subroutine Interp2D(ind1, ind2, IndData1, IndData2, DepData, nRows, nCols, dep)

! number of rows and columns in 2-D table
integer(4) nRows, nCols

! independent variable vectors and dependent variable array
real(8) IndData1(nRows), IndData2(nCols), DepData(nRows,nCols)

! independent variables, intermediate dependent variables, and dependent variable
real(8) ind1, ind2, dep1, dep, dep2

! flags for independent variable values beyond upper and lower bounds
integer(4) iflag_lb, iflag_ub, jflag_lb, jflag_ub

! initialize flags
iflag_lb = 0
iflag_ub = 0
jflag_lb = 0
jflag_ub = 0

! determine i-index
if (ind1 .le. IndData1(1)) then
 i=1                 ! ind1 less than lower bound
 iflag_lb = 1
elseif (ind1 .ge. IndData1(nRows)) then
 i=nRows             ! ind1 greater than upper bound
 iflag_ub = 1

else

 do ii=1,nRows-1
  if ((ind1 .ge. IndData1(ii)) .and. (ind1 .lt. IndData1(ii+1))) then
   i=ii              ! ind1 is between IndData1(ii) and IndData1(ii+1)
  end if
 end do

end if

! determine j-index
if (ind2 .le. IndData2(1)) then
 j=1                 ! ind2 less than lower bound
 jflag_lb = 1
```

```
elseif (ind2 .ge. IndData2(nCols)) then
 j=nCols             ! ind2 greater than upper bound
 jflag_ub = 1

else

 do jj=1,nCols-1
  if ((ind2 .ge. IndData2(jj)) .and. (ind2 .lt. IndData2(jj+1))) then
   j=jj                    ! ind2 is between IndData2(jj) and IndData2(jj+1)
  end if
 end do

end if

! logic trap to catch points below the lower bound of the table
if(iflag_lb .eq. 1) then   ! outside lower bound
 if( (jflag_lb .eq. 1) .or. (jflag_ub .eq. 1) ) then
  dep=DepData(i,j)         ! corner point

 else
 ! linearly interpolate along lower-bound edge
  dep=DepData(i,j)  &
     + (ind2-IndData2(j))/(IndData2(j+1)-IndData2(j))  &
           *(DepData(i,j+1)-DepData(i,j))
 end if
end if

! logic trap to catch points above the upper bound of the table
if(iflag_ub .eq. 1) then   ! outside upper bound
 if( (jflag_lb .eq. 1) .or. (jflag_ub .eq. 1) ) then
  dep=DepData(i,j)         ! corner point
 else
 ! linearly interpolate along upper edge
  dep=DepData(i,j)  &
     + (ind2-IndData2(j))/(IndData2(j+1)-IndData2(j))  &
           *(DepData(i,j+1)-DepData(i,j))
 end if
end if

! logic trap to catch points beyond the left and right bound of the table
if( (jflag_lb .eq. 1) .or. (jflag_ub .eq. 1) ) then
 ! outside right or left bound
 if( (iflag_lb .eq. 1) .or. (iflag_ub .eq. 1) ) then
  ! trap for corner points (already calculated)
 else
  ! linearly interpolate along left or right edge
  dep=DepData(i,j)  &
     + (ind1-IndData1(i))/(IndData1(i+1)-IndData1(i))  &
        *(DepData(i+1,j)-DepData(i,j))
 end if
end if

! logic trap to catch points in the table
if( (iflag_lb .eq. 0) .and. (iflag_ub .eq. 0) .and.  &
    (jflag_lb .eq. 0) .and. (jflag_ub .eq. 0) ) then

! interpolate in j-th column between the i-th and (i+1)-th row
dep1i=DepData(i,j)  &
    + (ind1-IndData1(i))/(IndData1(i+1)-IndData1(i))  &
       *(DepData(i+1,j)-DepData(i,j))

! interpolate in (j+1)-th column between the i-th and (i+1)-th row
dep2i=DepData(i,j+1)  &
```

```
    + (ind1-IndData1(i))/(IndData1(i+1)-IndData1(i))   &
       * (DepData(i+1,j+1)-DepData(i,j+1))

! interpolate the results above results between
! the j-th and (j+1)-th columns
dep=dep1i            &
  +(ind2-IndData2(j))/(IndData2(j+1)-IndData2(j))      &
    * (dep2i-dep1i)

end if

end subroutine Interp2D
!******************************************************************************
!******************************************************************************
```

VII.3.3.2 INPUT FILE FOR 1D INTERPOLATION SUBROUTINE TEST CASE

```
4
50.0
53.1
60.0
90.0

10
50.3   -2.431 -2.428 -2.415
51.0   -1.246 -1.244 -1.231
53.1   -0.389 -0.391 -0.380
55.2   -0.164 -0.169 -0.159
60.5   0.225         0.211         0.216
65.7   0.380         0.358         0.359
71.0   0.420         0.396         0.396
76.2   0.428         0.403         0.403
81.5   0.418         0.394         0.394
85.0   0.407         0.382         0.382
       2             3             4
; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (ANL-EBS-MD-000045)
; Seepage Name: Abstracted THC Seepage Water
; 1st independent variable (columns) = Abstracted Period
; 2nd independent variable (rows) = relative humidity (RH)
; dependent parameter = log Cl (i.e., log of Cl concentration (molal))
```

VII.3.3.3 OUTPUT FILE FOR 1D INTERPOLATION SUBROUTINE TEST CASE

```
50.0000000000000      -2.43100000000000
53.1000000000000      -0.389000000000000
60.0000000000000       0.188301886792453
90.0000000000000       0.407000000000000
```

VII.3.3.4 EXCEL SPREADSHEET REPLICATING 1D INTERPOLATION SUBROUTINE

1D Interpolation Subroutine				
Lookup Table			Interpolated Values	
Independent Variable	Dependent Variable		Independent Variable	Interpolated Value of Dependent Variable
50.3	-2.431		50	-2.43100
51	-1.246		53.1	-0.38900
53.1	-0.389		60	0.18830
55.2	-0.164		90	0.40700
60.5	0.225			
65.7	0.38			
71	0.42			
76.2	0.428			
81.5	0.418			
85	0.407			

VII.3.3.5 INPUT FILE FOR 2D INTERPOLATION SUBROUTINE TEST CASE

```
9
0.0    20.0
0.0    80.0
1.1    20.0
1.1    80.0
0.2    20.0
0.2    80.0
0.0    65.0
1.1    65.0
45.0
```

VII.3.3.6 2D LOOKUP TABLE

```
7      3
             25            50            75
0.0011999    7.02          7.02          7.02
0.0012 6.78         6.86          7.02
0.01         6.986         6.95          7.02
0.1          7.11          7.03          6.97
0.5          7.23          7.18          7.14
0.9          7.09          7.22          7.18
1.0          7.05          7.22          7.19
; Salts Lookup Tables
; In-Drift Precipitates/Salts AMR (ANL-EBS-MD-000045)
; Seepage Name: Abstracted THC Seepage Water
; condition: Period 4
; 1st independent variable (columns) = temperature ('C)
; dependent parameter = pH
```

VII.3.3.7 OUTPUT FILE FOR 2D INTERPOLATION SUBROUTINE TEST CASE

```
000000000000000E+000    20.0000000000000    7.02000000000000
0.000000000000000E+000  80.0000000000000    7.02000000000000
1.10000000000000        20.0000000000000    7.05000000000000
1.10000000000000        80.0000000000000    7.19000000000000
0.200000000000000       20.0000000000000    7.14000000000000
0.200000000000000       80.0000000000000    7.01250000000000
0.000000000000000E+000  65.0000000000000    7.02000000000000
1.10000000000000        65.0000000000000    7.20200000000000
0.250000000000000       45.0000000000000    7.09999990463257
```

VII-3.3.8 EXCEL SPREADSHEET REPLICATING 2D INTERPOLATION SUBROUTINE

2D Interpolation Subroutine						
2D Lookup Table				Interpolated Value of Dependent Variable		
1st Independent Variable	2nd Independent Variable			1st Independent Variable	2nd Independent Variable	Interpolated/ Truncated Value
	25	50	75	0	20	7.02000
0.0011999	7.02	7.02	7.02	0	80	7.02000
0.0012	6.78	6.86	7.02	1.1	20	7.05000
0.01	6.986	6.95	7.02	1.1	80	7.19000
0.1	7.11	7.03	6.97	0.2	20	7.14000
0.5	7.23	7.18	7.14	0.2	80	7.01250
0.9	7.09	7.22	7.18	0	65	7.02000
1	7.05	7.22	7.19	1.1	65	7.20200
				0.25	45	7.10000
				Intermediate Values For Last Data Set		
				7.15500	7.08625	

VII.3.3.9 INPUT FILE USED TO TEST PREWAP PROGRAM

```
line 1 test file

 line 2
 line 3
 Time (yr), Waste Pack Temp.(C), Drip shield temp. (C), Drift wall temp.(C), Invert
temp. (C), Waste pack RH, Drip   shield RH, Drift wall RH, Backfill RH, Invert RH,
Liquid Satr. @  Drip Shield, Liquid Satr.@Invert, Air mass Frac, Water Vapor flux at
Dwall (kg/yr/m of drift), Air flux at Dwall(kg/yr/m of drift), A Drip Shield Evapo.
rate (m3/yr), Backfill Evapo. Rate (m3/yr),  Invert Evapo. Rate (m3/yr), Percolation
Flux at 5 m (mm/yr), Vol ume flow at top dripshield (m3/yr), volume flow at invert
(m3/yr), Top of the dripshield Temp (C)
The number of Rows =  21
The fraction of this history = 0.000576
 line 7
 line 8
 line 9
 line 10
```

```
 line 11       wpT            dsT              dwT              iT               wpRH
dsRH           dwRH
dsEvapRate                                   pf-5m
     0.00   0.222933E+02  -0.999000E+02   0.222797E+02   0.223071E+02   0.999137E+00
-0.999000E+02   0.999952E+00  -0.999000E+02   0.999180E+00  -0.999000E+02
0.000000E+00  -0.999000E+02   0.000000E+00   0.000000E+00  -0.999000E+02  -
0.999000E+02  -0.999000E+02   0.153137E+02  -0.999000E+02   0.000000E+00  -
0.999000E+02
     1.00   0.846557E+02  -0.999000E+02   0.679710E+02   0.750104E+02  -0.999000E+02
0.500429E+00   0.999958E+00  -0.999000E+02   0.876529E+00  -0.999000E+02   0.196320E-
01  -0.999000E+02   0.243105E+02   0.106586E+02  -0.999000E+02  -0.999000E+02  -
0.999000E+02   0.143936E+02  -0.999000E+02  -0.116894E-01  -0.999000E+02
    50.00   0.665731E+02  -0.999000E+02   0.612045E+02   0.633398E+02   0.100000E-01
-0.999000E+02   0.999504E+00  -0.999000E+02   0.967314E+00  -0.999000E+02   0.316090E-
01  -0.999000E+02   0.624383E+01   0.291981E+01  -0.999000E+02  -0.999000E+02  -
0.999000E+02   0.142088E+02  -0.999000E+02  -0.279873E-02  -0.999000E+02
    50.20   0.236173E+03   0.230511E+03   0.109784E+03   0.188458E+03   0.100000E-01
0.840750E-01   0.655213E+00   0.969566E+00   0.829800E-01   0.000000E+00
0.000000E+00   0.208500E-02   0.293924E+04   0.834146E+00  -0.123300E-05   0.235282E-
02  -0.821340E-04   0.141540E+02   0.000000E+00   0.000000E+00   0.231596E+03
    51.00   0.270679E+03   0.266027E+03   0.130378E+03   0.229314E+03   0.100000E-01
0.369300E-01   0.499133E+00   0.646741E+00   0.347820E-01   0.000000E+00
0.000000E+00   0.792600E-02   0.386056E+02  -0.176730E-01   0.300000E-08  -0.700000E-
08   0.170000E-06   0.144618E+02   0.000000E+00   0.000000E+00   0.266970E+03
    53.00   0.271006E+03   0.266812E+03   0.143359E+03   0.239704E+03   0.600000E+00
0.298220E-01   0.365954E+00   0.489141E+00   0.281430E-01   0.000000E+00
0.000000E+00   0.145170E-01   0.102893E+02   0.420108E+00  -0.310800E-05  -0.550000E-
07   0.522000E-06   0.150733E+02   0.000000E+00   0.000000E+00   0.267646E+03
    55.00   0.261421E+03   0.257416E+03   0.144179E+03   0.240140E+03   0.600000E+00
0.308210E-01   0.359387E+00   0.480840E+00   0.278820E-01   0.000000E+00
0.000000E+00   0.148380E-01   0.103388E+02   0.472702E+00  -0.355400E-05  -0.580000E-
07   0.488000E-06   0.160079E+02   0.000000E+00   0.000000E+00   0.258208E+03
    60.00   0.225009E+03   0.221233E+03   0.132806E+03   0.194677E+03   0.600000E+00
0.352420E-01   0.365349E+00   0.491466E+00   0.607210E-01   0.000000E+00
0.000000E+00   0.105486E+00   0.291557E+00  -0.355520E+00  -0.700000E-08  -0.200000E-
08   0.390000E-07   0.179075E+02   0.000000E+00   0.000000E+00   0.221968E+03
    65.00   0.197084E+03   0.193435E+03   0.120758E+03   0.173413E+03   0.600000E+00
0.535270E-01   0.501935E+00   0.674298E+00   0.954170E-01   0.000000E+00
0.000000E+00   0.130878E+00   0.717410E+00  -0.453557E+00  -0.100000E-08   0.100000E-
08  -0.500000E-08   0.216911E+02   0.000000E+00   0.000000E+00   0.194135E+03
    70.00   0.144995E+03   0.141441E+03   0.975721E+02   0.128478E+03   0.100000E-01
0.866760E-01   0.748124E+00   0.990652E+00   0.290811E+00   0.000000E+00
0.000000E+00   0.189501E+00   0.265674E+03   0.349618E+02  -0.102500E-05  -0.286851E-
02  -0.755000E-06   0.268478E+02   0.000000E+00   0.000000E+00   0.142113E+03
    80.00   0.949581E+02   0.915515E+02   0.814937E+02   0.938274E+02   0.600000E+00
0.130910E+00   0.992015E+00   0.997802E+00   0.926671E+00   0.126481E+00   0.609780E-
01   0.223147E+00   0.220814E+03   0.193291E+02   0.128079E+00  -0.525328E-02
0.281983E+00   0.275514E+02   0.000000E+00   0.000000E+00   0.921773E+02
   100.00   0.900955E+02   0.869274E+02   0.771852E+02   0.899821E+02   0.100000E-01
0.158592E+00   0.994236E+00   0.999768E+00   0.981316E+00   0.161142E+00
0.110698E+00   0.325127E+00   0.128401E+03   0.167901E+02   0.993992E-01  -0.412338E-
02   0.248594E+00   0.189149E+02   0.000000E+00   0.000000E+00   0.874805E+02
   110.00   0.867760E+02   0.836941E+02   0.745860E+02   0.874209E+02   0.600000E+00
0.170195E+00   0.995199E+00   0.999764E+00   0.984132E+00   0.164643E+00
0.118413E+00   0.395588E+00   0.925944E+02   0.149763E+02   0.861841E-01  -0.368624E-
02   0.210613E+00   0.172476E+02   0.000000E+00   0.000000E+00   0.842257E+02
   120.00   0.825661E+02   0.795754E+02   0.712272E+02   0.838059E+02   0.100000E-01
0.190156E+00   0.997187E+00   0.999758E+00   0.987154E+00   0.167614E+00
0.122490E+00   0.484554E+00   0.622246E+02   0.125183E+02   0.700028E-01  -0.307191E-
02   0.169845E+00   0.162730E+02   0.000000E+00   0.000000E+00   0.800842E+02
   130.00   0.810327E+02   0.781357E+02   0.700483E+02   0.824621E+02   0.100000E-01
0.219832E+00   0.998197E+00   0.999755E+00   0.988637E+00   0.168528E+00
```

```
0.123351E+00   0.514527E+00   0.543765E+02   0.116858E+02   0.647288E-01  -0.262423E-
02   0.156327E+00   0.155837E+02   0.000000E+00   0.000000E+00   0.786208E+02
    140.00   0.776294E+02   0.748289E+02   0.675343E+02   0.794910E+02   0.600000E+00
0.247555E+00   0.999252E+00   0.999748E+00   0.991091E+00   0.170812E+00
0.125106E+00   0.576015E+00   0.410952E+02   0.998433E+01   0.538530E-01  -0.230470E-
02   0.129704E+00   0.150984E+02   0.000000E+00   0.000000E+00   0.752897E+02
    150.00   0.737673E+02   0.710678E+02   0.647680E+02   0.761593E+02   0.600000E+00
0.279064E+00   0.999473E+00   0.999745E+00   0.993142E+00   0.183805E+00
0.136964E+00   0.637418E+00   0.301292E+02   0.797134E+01   0.432977E-01  -0.192148E-
02   0.104080E+00   0.147780E+02   0.000000E+00   0.000000E+00   0.715033E+02
    190.00   0.712332E+02   0.687596E+02   0.629677E+02   0.739267E+02   0.100000E-01
0.343920E+00   0.999606E+00   0.999741E+00   0.993787E+00   0.186603E+00
0.139378E+00   0.674327E+00   0.247632E+02   0.682169E+01   0.372664E-01  -0.176730E-
02   0.896966E-01   0.143423E+02   0.000000E+00   0.000000E+00   0.691465E+02
    270.00   0.690627E+02   0.668984E+02   0.613302E+02   0.718319E+02   0.100000E-01
0.442273E+00   0.999772E+00   0.999737E+00   0.994472E+00   0.187777E+00
0.140144E+00   0.705799E+00   0.207220E+02   0.593611E+01   0.322745E-01  -0.153458E-
02   0.776633E-01   0.139374E+02   0.118268E+00  -0.358887E-01   0.672245E+02
    615.00   0.671101E+02   0.656206E+02   0.596731E+02   0.696723E+02   0.100000E-01
0.673121E+00   0.999966E+00   0.999733E+00   0.996401E+00   0.188846E+00
0.140840E+00   0.735027E+00   0.172995E+02   0.511785E+01   0.276815E-01  -0.114842E-
02   0.662257E-01   0.145990E+02   0.398767E-01  -0.350144E-01   0.658302E+02
1000000.00   0.187600E+02   0.187354E+02   0.186076E+02   0.187464E+02   0.998407E+00
0.999883E+00   0.999964E+00   0.999686E+00   0.999958E+00   0.224044E+00
0.166789E+00   0.984671E+00   0.132790E-01  -0.110600E-02   0.268370E-04  -0.663300E-
05   0.548690E-04   0.610027E+02   0.112866E-02   0.184060E+00   0.187314E+02
The number of Rows =  29
The fraction of this history = 0.000960
line 3
□
line 4
line 5
line 6
line 7         wpT           dsT            dwT            iT             wpRH
dsRH           dwRH
dsEvapRate                                        pf-5m
□
      0.00   0.223001E+02  -0.999000E+02   0.222865E+02   0.223136E+02   0.999138E+00
-0.999000E+02   0.999952E+00  -0.999000E+02   0.999180E+00  -0.999000E+02
0.000000E+00  -0.999000E+02   0.000000E+00   0.000000E+00  -0.999000E+02   -
0.999000E+02  -0.999000E+02   0.152617E+02  -0.999000E+02   0.000000E+00   -
0.999000E+02
      1.00   0.806202E+02  -0.999000E+02   0.633204E+02   0.707888E+02   0.477524E+00
-0.999000E+02   0.999955E+00  -0.999000E+02   0.886023E+00  -0.999000E+02   0.190470E-
01  -0.999000E+02   0.224813E+02   0.119247E+02  -0.999000E+02  -0.999000E+02   -
0.999000E+02   0.143430E+02  -0.999000E+02  -0.116604E-01  -0.999000E+02
      2.00   0.874782E+02  -0.999000E+02   0.717236E+02   0.784255E+02   0.527047E+00
-0.999000E+02   0.999416E+00  -0.999000E+02   0.869311E+00  -0.999000E+02   0.182770E-
01  -0.999000E+02   0.259342E+02   0.102259E+02  -0.999000E+02  -0.999000E+02   -
0.999000E+02   0.143530E+02  -0.999000E+02  -0.725051E-02  -0.999000E+02
      5.00   0.944264E+02  -0.999000E+02   0.808013E+02   0.866099E+02   0.589066E+00
-0.999000E+02   0.996760E+00  -0.999000E+02   0.856647E+00  -0.999000E+02   0.963900E-
02  -0.999000E+02   0.304570E+02   0.102450E+02  -0.999000E+02  -0.999000E+02   -
0.999000E+02   0.144701E+02  -0.999000E+02  -0.117121E-02  -0.999000E+02
     20.00   0.890934E+02  -0.999000E+02   0.797213E+02   0.834106E+02   0.688934E+00
-0.999000E+02   0.995898E+00  -0.999000E+02   0.900721E+00  -0.999000E+02   0.282600E-
02  -0.999000E+02   0.171358E+02   0.707731E+01  -0.999000E+02  -0.999000E+02   -
0.999000E+02   0.145367E+02  -0.999000E+02   0.000000E+00  -0.999000E+02
     25.00   0.852817E+02  -0.999000E+02   0.767855E+02   0.801724E+02   0.707787E+00
-0.999000E+02   0.996314E+00  -0.999000E+02   0.912235E+00  -0.999000E+02   0.982700E-
02  -0.999000E+02   0.144370E+02   0.576231E+01  -0.999000E+02  -0.999000E+02   -
0.999000E+02   0.144728E+02  -0.999000E+02   0.000000E+00  -0.999000E+02
```

```
     30.00   0.817664E+02  -0.999000E+02   0.740034E+02   0.771318E+02   0.724692E+00
-0.999000E+02   0.996711E+00  -0.999000E+02   0.925986E+00  -0.999000E+02   0.132880E-
01  -0.999000E+02   0.127767E+02   0.525119E+01  -0.999000E+02  -0.999000E+02  -
0.999000E+02   0.144009E+02  -0.999000E+02   0.000000E+00  -0.999000E+02
     40.00   0.749153E+02  -0.999000E+02   0.684373E+02   0.710976E+02   0.756514E+00
-0.999000E+02   0.997577E+00  -0.999000E+02   0.946729E+00  -0.999000E+02   0.171170E-
01  -0.999000E+02   0.971942E+01   0.456280E+01  -0.999000E+02  -0.999000E+02  -
0.999000E+02   0.142671E+02  -0.999000E+02   0.000000E+00  -0.999000E+02
     50.00   0.678115E+02  -0.999000E+02   0.625164E+02   0.647416E+02   0.788405E+00
-0.999000E+02   0.999037E+00  -0.999000E+02   0.963607E+00  -0.999000E+02   0.284150E-
01  -0.999000E+02   0.707103E+01   0.310949E+01  -0.999000E+02  -0.999000E+02  -
0.999000E+02   0.141580E+02  -0.999000E+02  -0.278996E-02  -0.999000E+02
     50.20   0.234737E+03   0.229075E+03   0.109300E+03   0.187167E+03   0.762630E-01
0.100000E-01   0.658942E+00   0.980873E+00   0.847710E-01   0.000000E+00
0.000000E+00   0.203100E-02   0.303839E+04   0.855563E+00  -0.129200E-05   0.307937E-
02  -0.861920E-04   0.141030E+02   0.000000E+00   0.000000E+00   0.230161E+03
     51.00   0.268555E+03   0.263902E+03   0.127610E+03   0.226167E+03   0.362760E-01
0.100000E-01   0.527864E+00   0.680440E+00   0.370450E-01   0.000000E+00
0.000000E+00   0.643000E-02   0.128020E+03  -0.416400E-02   0.200000E-08   0.400000E-
08   0.193000E-06   0.144098E+02   0.000000E+00   0.000000E+00   0.264845E+03
     53.00   0.274117E+03   0.269923E+03   0.142879E+03   0.239454E+03   0.282400E-01
0.100000E-01   0.369748E+00   0.493922E+00   0.282940E-01   0.000000E+00
0.000000E+00   0.143340E-01   0.102595E+02   0.386741E+00  -0.283100E-05  -0.520000E-
07   0.542000E-06   0.150223E+02   0.000000E+00   0.000000E+00   0.270758E+03
     55.00   0.265761E+03   0.261756E+03   0.144988E+03   0.240575E+03   0.283890E-01
0.600000E+00   0.352861E+00   0.472570E+00   0.276230E-01   0.000000E+00
0.000000E+00   0.151570E-01   0.102755E+02   0.521326E+00  -0.397400E-05  -0.620000E-
07   0.455000E-06   0.159587E+02   0.000000E+00   0.000000E+00   0.262547E+03
     60.00   0.231692E+03   0.227916E+03   0.136306E+03   0.201924E+03   0.308040E-01
0.600000E+00   0.342139E+00   0.460237E+00   0.542150E-01   0.000000E+00
0.000000E+00   0.926510E-01   0.163887E+00  -0.218317E+00  -0.400000E-08  -0.200000E-
08  -0.110000E-07   0.178319E+02   0.000000E+00   0.000000E+00   0.228651E+03
     65.00   0.198981E+03   0.195333E+03   0.122966E+03   0.177266E+03   0.467120E-01
0.600000E+00   0.473065E+00   0.635232E+00   0.877210E-01   0.000000E+00
0.000000E+00   0.124235E+00   0.474730E+00  -0.461679E+00  -0.100000E-08   0.400000E-
08   0.890000E-07   0.215856E+02   0.000000E+00   0.000000E+00   0.196033E+03
     70.00   0.139131E+03   0.135577E+03   0.976314E+02   0.128600E+03   0.781570E-01
0.600000E+00   0.746703E+00   0.989340E+00   0.289581E+00   0.000000E+00
0.000000E+00   0.190654E+00   0.265218E+03   0.348544E+02  -0.103400E-05  -0.285501E-
02  -0.763000E-06   0.268695E+02   0.000000E+00   0.000000E+00   0.136249E+03
     80.00   0.100158E+03   0.967514E+02   0.857046E+02   0.970909E+02   0.115084E+00
0.100000E-01   0.985973E+00   0.995477E+00   0.884019E+00   0.650060E-01   0.264120E-
01   0.146113E+00   0.397487E+03   0.303699E+02   0.153011E+00  -0.498900E-02
0.204364E+00   0.275664E+02   0.000000E+00   0.000000E+00   0.973771E+02
    100.00   0.934533E+02   0.902852E+02   0.818526E+02   0.942050E+02   0.140307E+00
0.600000E+00   0.990235E+00   0.997121E+00   0.919350E+00   0.124630E+00   0.604280E-
01   0.217854E+00   0.229181E+03   0.193812E+02   0.130214E+00  -0.515831E-02
0.235767E+00   0.188267E+02   0.000000E+00   0.000000E+00   0.908383E+02
    110.00   0.887102E+02   0.856283E+02   0.795500E+02   0.922187E+02   0.150632E+00
0.100000E-01   0.991787E+00   0.999695E+00   0.959159E+00   0.141891E+00   0.773060E-
01   0.267986E+00   0.172739E+03   0.181566E+02   0.112748E+00  -0.293494E-02
0.271712E+00   0.171724E+02   0.000000E+00   0.000000E+00   0.861599E+02
    120.00   0.874543E+02   0.844636E+02   0.784762E+02   0.912332E+02   0.168459E+00
0.600000E+00   0.992456E+00   0.999765E+00   0.975882E+00   0.151229E+00   0.917010E-
01   0.293625E+00   0.151039E+03   0.175141E+02   0.105458E+00  -0.453336E-02
0.266612E+00   0.162056E+02   0.000000E+00   0.000000E+00   0.849724E+02
    130.00   0.840941E+02   0.811971E+02   0.756464E+02   0.885412E+02   0.194897E+00
0.100000E-01   0.993634E+00   0.999763E+00   0.980092E+00   0.162964E+00
0.115489E+00   0.368973E+00   0.104537E+03   0.155809E+02   0.912377E-01  -0.373414E-
02   0.225596E+00   0.155216E+02   0.000000E+00   0.000000E+00   0.816822E+02
    140.00   0.807761E+02   0.779757E+02   0.729943E+02   0.857868E+02   0.220070E+00
0.100000E-01   0.995169E+00   0.999759E+00   0.982636E+00   0.165588E+00
```

```
0.120302E+00   0.439992E+00   0.758019E+02   0.136974E+02   0.778465E-01  -0.346009E-
02   0.190273E+00   0.150399E+02   0.000000E+00   0.000000E+00   0.784364E+02
    150.00   0.777314E+02   0.750318E+02   0.706811E+02   0.832177E+02   0.248718E+00
0.600000E+00   0.997062E+00   0.999755E+00   0.985363E+00   0.167470E+00
0.122409E+00   0.499941E+00   0.580119E+02   0.120507E+02   0.671403E-01  -0.294301E-
02   0.162715E+00   0.147219E+02   0.000000E+00   0.000000E+00   0.754674E+02
    180.00   0.752735E+02   0.727402E+02   0.687718E+02   0.809819E+02   0.293681E+00
0.600000E+00   0.999266E+00   0.999751E+00   0.988900E+00   0.168870E+00
0.123639E+00   0.546981E+00   0.468024E+02   0.107262E+02   0.588192E-01  -0.261450E-
02   0.142453E+00   0.143975E+02   0.000000E+00   0.000000E+00   0.731398E+02
    225.00   0.729992E+02   0.706785E+02   0.668359E+02   0.786637E+02   0.355414E+00
0.100000E-01   0.998974E+00   0.999745E+00   0.989652E+00   0.171033E+00
0.125191E+00   0.593248E+00   0.378513E+02   0.950032E+01   0.507596E-01  -0.230120E-
02   0.122659E+00   0.140623E+02   0.000000E+00   0.000000E+00   0.710345E+02
    315.00   0.707678E+02   0.687327E+02   0.649136E+02   0.763492E+02   0.442330E+00
0.100000E-01   0.999309E+00   0.999745E+00   0.991455E+00   0.182473E+00
0.135905E+00   0.635019E+00   0.304144E+02   0.802757E+01   0.435520E-01  -0.194807E-
02   0.104876E+00   0.138135E+02   0.968528E-01  -0.127453E+00   0.690353E+02
    475.00   0.686313E+02   0.669335E+02   0.628589E+02   0.737972E+02   0.548132E+00
0.100000E-01   0.999531E+00   0.999740E+00   0.992464E+00   0.185740E+00
0.138739E+00   0.677026E+00   0.243129E+02   0.672385E+01   0.366556E-01  -0.170702E-
02   0.881237E-01   0.137836E+02   0.574295E-01  -0.773144E-01   0.671779E+02
    615.00   0.671560E+02   0.656666E+02   0.614066E+02   0.719363E+02   0.630850E+00
0.600000E+00   0.999704E+00   0.999736E+00   0.993185E+00   0.186764E+00
0.139402E+00   0.704846E+00   0.207516E+02   0.593717E+01   0.323022E-01  -0.150387E-
02   0.778167E-01   0.145462E+02   0.396563E-01  -0.347207E-01   0.658761E+02
1000000.00   0.188336E+02   0.188090E+02   0.186851E+02   0.188401E+02   0.998413E+00
0.999883E+00   0.999964E+00   0.999686E+00   0.999958E+00   0.223843E+00
0.166600E+00   0.984575E+00   0.151070E-01  -0.179900E-02   0.296850E-04  -0.676300E-
05   0.632210E-04   0.607919E+02   0.112650E-02   0.183561E+00   0.188050E+02
The number of Rows =  32
The fraction of this history = 0.001153
 Coordinate Location:
The easting coordinate = 170256.20 m
The northing coordinate = 234314.20 m
 Infiltration rate:
qinf =   60.37322 mm/yr
0       2.23E+01      -9.99E+01     2.23E+01      2.23E+01      9.99E-01      -9.99E+01
        1.00E+00      -9.99E+01     9.99E-01      -9.99E+01     0.00E+00      -9.99E+01
        0.00E+00      0.00E+00      -9.99E+01     -9.99E+01     -9.99E+01     1.52E+01
        -9.99E+01     0.00E+00      -9.99E+01
□
1       7.88E+01      -9.99E+01     6.12E+01      6.88E+01      4.67E-01      -9.99E+01
        1.00E+00      -9.99E+01     8.90E-01      -9.99E+01     1.86E-02      -9.99E+01
        2.15E+01      1.24E+01      -9.99E+01     -9.99E+01     -9.99E+01     1.42E+01
        -9.99E+01     -1.16E-02     -9.99E+01
□
40      7.56E+01      -9.99E+01     6.91E+01      7.19E+01      7.56E-01      -9.99E+01
        9.97E-01      -9.99E+01     9.43E-01      -9.99E+01     1.66E-02      -9.99E+01
        1.00E+01      4.62E+00      -9.99E+01     -9.99E+01     -9.99E+01     1.42E+01
        -9.99E+01     0.00E+00      -9.99E+01
□
50.2    2.26E+02      2.20E+02      1.06E+02      1.82E+02      4.90E-01      5.50E-01
        6.78E-01      1.00E+00      9.38E-02      0.00E+00      0.00E+00      4.08E-02
        2.91E+03      4.62E+00      -1.25E-06     6.81E-03      -8.42E-05     1.40E+01
        0.00E+00      0.00E+00      2.21E+02
□
51      2.64E+02      2.60E+02      1.23E+02      2.20E+02      5.50E-01      6.50E-01
        5.56E-01      7.18E-01      4.24E-02      0.00E+00      0.00E+00      4.70E-03
        5.85E+02      1.78E-01      -5.00E-09     2.50E-08      1.26E-06      1.43E+01
        0.00E+00      0.00E+00      2.61E+02
□
```

52	2.74E+02	2.69E+02	1.34E+02	2.33E+02	6.50E-01	8.50E-01
	4.54E-01	5.95E-01	3.23E-02	0.00E+00	0.00E+00	1.02E-02
	2.39E+01	1.90E-02	0.00E+00	-1.80E-08	4.61E-07	1.46E+01
	0.00E+00	0.00E+00	2.70E+02			
□						
55	2.72E+02	2.68E+02	1.45E+02	2.40E+02	8.50E-01	4.90E-01
	3.55E-01	4.76E-01	2.77E-02	0.00E+00	0.00E+00	1.50E-02
	1.01E+01	4.94E-01	-3.75E-06	-6.00E-08	4.68E-07	1.59E+01
	0.00E+00	0.00E+00	2.69E+02			
□						
60	2.55E+02	2.51E+02	1.45E+02	2.19E+02	9.00E-01	9.50E-01
	2.87E-01	3.87E-01	3.88E-02	0.00E+00	0.00E+00	6.17E-02
	-1.31E-01	9.90E-02	3.00E-09	-3.00E-09	-1.03E-07	1.96E+01
	0.00E+00	0.00E+00	2.52E+02			
□						
65	2.26E+02	2.22E+02	1.35E+02	2.00E+02	9.50E-01	9.00E-01
	3.48E-01	4.68E-01	5.60E-02	0.00E+00	0.00E+00	9.70E-02
	1.96E-01	-2.53E-01	-4.00E-09	-2.00E-09	-8.00E-09	2.14E+01
	0.00E+00	0.00E+00	2.23E+02			
□						
1180	6.54E+01	6.44E+01	6.01E+01	7.02E+01	4.90E-01	5.50E-01
	1.00E+00	1.00E+00	9.94E-01	1.86E-01	1.39E-01	7.29E-01
	1.77E+01	5.21E+00	2.83E-02	-1.14E-03	6.87E-02	3.91E+01
	1.35E-02	9.16E-02	6.46E+01			
□						
1420	6.34E+01	6.25E+01	5.80E+01	6.75E+01	5.50E-01	6.50E-01
	1.00E+00	1.00E+00	9.98E-01	1.87E-01	1.41E-01	7.63E-01
	1.42E+01	4.33E+00	2.31E-02	-9.44E-04	5.61E-02	3.93E+01
	9.53E-03	1.01E-01	6.26E+01			
□						
1680	6.13E+01	6.05E+01	5.60E+01	6.48E+01	6.50E-01	8.50E-01
	1.00E+00	1.00E+00	9.99E-01	1.89E-01	1.43E-01	7.92E-01
	1.14E+01	3.59E+00	1.90E-02	-7.95E-04	4.71E-02	3.94E+01
	7.26E-03	1.06E-01	6.06E+01			
□						
1900	5.94E+01	5.86E+01	5.41E+01	6.22E+01	8.50E-01	4.90E-01
	1.00E+00	1.00E+00	1.00E+00	1.92E-01	1.44E-01	8.17E-01
	9.15E+00	2.97E+00	1.55E-02	-7.74E-04	3.84E-02	3.96E+01
	5.85E-03	1.10E-01	5.87E+01			
□						
1950	5.91E+01	5.84E+01	5.38E+01	6.19E+01	9.00E-01	9.50E-01
	1.00E+00	1.00E+00	1.00E+00	1.92E-01	1.44E-01	8.20E-01
	8.90E+00	2.89E+00	1.51E-02	-7.60E-04	3.75E-02	4.14E+01
	5.80E-03	1.16E-01	5.85E+01			
□						
1975	5.90E+01	5.83E+01	5.37E+01	6.17E+01	9.50E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.92E-01	1.44E-01	8.21E-01
	8.79E+00	2.86E+00	1.49E-02	-7.54E-04	3.71E-02	4.32E+01
	5.81E-03	1.21E-01	5.84E+01			
□						
2060	5.89E+01	5.82E+01	5.36E+01	6.16E+01	4.90E-01	5.50E-01
	1.00E+00	1.00E+00	1.00E+00	1.92E-01	1.44E-01	8.23E-01
	8.69E+00	2.83E+00	1.48E-02	-7.48E-04	3.67E-02	4.50E+01
	5.83E-03	1.26E-01	5.83E+01			
□						
2080	5.88E+01	5.81E+01	5.35E+01	6.14E+01	5.50E-01	6.50E-01
	1.00E+00	1.00E+00	1.00E+00	1.93E-01	1.44E-01	8.24E-01
	8.58E+00	2.80E+00	1.46E-02	-7.43E-04	3.63E-02	4.67E+01
	5.84E-03	1.32E-01	5.82E+01			
□						
2100	5.87E+01	5.80E+01	5.34E+01	6.13E+01	6.50E-01	8.50E-01
	1.00E+00	1.00E+00	1.00E+00	1.93E-01	1.44E-01	8.25E-01
	8.48E+00	2.77E+00	1.44E-02	-7.37E-04	3.59E-02	4.85E+01

	5.86E-03	1.37E-01	5.81E+01			
□						
2120	5.86E+01	5.79E+01	5.33E+01	6.12E+01	8.50E-01	4.90E-01
	1.00E+00	1.00E+00	1.00E+00	1.93E-01	1.44E-01	8.26E-01
	8.38E+00	2.75E+00	1.43E-02	-7.31E-04	3.56E-02	5.03E+01
	5.87E-03	1.42E-01	5.80E+01			
□						
2140	2.00E+01	5.78E+01	5.32E+01	6.10E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.93E-01	1.44E-01	8.28E-01
	8.28E+00	2.72E+00	1.00E+00	-7.26E-04	3.52E-02	1.00E+00
	5.89E-03	1.48E-01	5.79E+01			
□						
2160	2.00E+01	5.77E+01	5.31E+01	6.09E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.93E-01	1.44E-01	8.29E-01
	8.18E+00	2.69E+00	4.50E-01	-7.21E-04	3.48E-02	1.00E+00
	5.90E-03	1.53E-01	5.78E+01			
□						
2180	2.00E+01	5.76E+01	5.30E+01	6.08E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.93E-01	1.44E-01	8.30E-01
	8.09E+00	2.66E+00	0.00E+00	-7.05E-04	3.45E-02	1.00E+00
	5.92E-03	1.58E-01	5.77E+01			
□						
2200	2.00E+01	5.75E+01	5.29E+01	6.06E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.93E-01	1.44E-01	8.31E-01
	8.01E+00	2.62E+00	-1.00E-02	-6.83E-04	3.41E-02	1.00E+00
	5.93E-03	1.64E-01	5.76E+01			
□						
2600	4.40E+01	5.57E+01	5.12E+01	5.82E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.96E-01	1.46E-01	8.51E-01
	6.41E+00	2.10E+00	1.00E+00	-4.56E-04	2.76E-02	1.00E+00
	4.99E-03	1.69E-01	5.58E+01			
□						
3050	5.60E+01	5.40E+01	4.95E+01	5.61E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.98E-01	1.47E-01	8.67E-01
	5.34E+00	1.78E+00	4.50E-01	-4.21E-04	2.33E-02	1.00E+00
	4.46E-03	1.72E-01	5.41E+01			
□						
3600	6.70E+01	5.24E+01	4.80E+01	5.40E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	1.99E-01	1.47E-01	8.81E-01
	4.45E+00	1.48E+00	0.00E+00	-3.67E-04	1.96E-02	1.00E+00
	3.96E-03	1.73E-01	5.24E+01			
□						
4300	6.70E+01	5.06E+01	4.63E+01	5.18E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	2.00E-01	1.48E-01	8.94E-01
	3.57E+00	1.22E+00	-1.00E-02	-3.94E-04	1.61E-02	1.00E+00
	3.84E-03	1.74E-01	5.07E+01			
□						
5100	9.80E+01	4.89E+01	4.48E+01	5.00E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	2.06E-01	1.54E-01	9.04E-01
	3.08E+00	9.34E-01	1.00E+00	-2.72E-04	1.39E-02	1.00E+00
	3.44E-03	1.75E-01	4.90E+01			
□						
6000	9.80E+01	4.73E+01	4.33E+01	4.81E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	2.08E-01	1.55E-01	9.13E-01
	2.61E+00	7.93E-01	4.50E-01	-2.42E-04	1.17E-02	1.00E+00
	3.02E-03	1.88E-01	4.73E+01			
□						
7000	9.80E+01	4.56E+01	4.17E+01	4.64E+01	9.00E-01	9.00E-01
	1.00E+00	1.00E+00	1.00E+00	2.09E-01	1.56E-01	9.21E-01
	2.27E+00	7.00E-01	0.00E+00	-2.02E-04	1.02E-02	1.00E+00
	2.90E-03	2.05E-01	4.57E+01			
□						

```
8000   9.80E+01      4.42E+01      4.04E+01      4.48E+01      9.00E-01      9.00E-01
       1.00E+00      1.00E+00      1.00E+00      2.10E-01      1.56E-01      9.27E-01
       2.00E+00      6.17E-01      -1.00E-02     -1.74E-04     9.11E-03      1.00E+00
       2.64E-03      2.19E-01      4.42E+01
□
1000000       1.89E+01      1.89E+01      1.88E+01      1.89E+01      9.00E-01      9.00E-
01     1.00E+00      1.00E+00      1.00E+00      2.24E-01      1.66E-01      9.84E-01
       1.66E-02      -5.50E-03     3.20E-05      -6.84E-06     7.01E-05      6.04E+01
       1.12E-03      2.21E-01      1.89E+01
```

VII.3.3.10 OUTPUT FROM PREWAP TEST CASE

```
! 1st comment line
! 2nd comment line
! 3rd comment line
#    3  21
#   17
#  5.760E-04
!  t            wpT          wpRH         dsT          dsRH         wpPHnd       wpCLnd
wpPHd        wpCLd        dsPHnd       dsCLnd       dsPHd        dsCLd        ipkPHnd
ipkCLnd      ipkPHd       ipkCLd       barPHnd      barCLnd      barPHd       barCLd
PercFlux5m
 5.100E+01   2.707E+02   1.000E-02   2.660E+02   3.693E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.446E+01
 5.300E+01   2.710E+02   6.000E-01   2.668E+02   2.982E-02  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.507E+01
 5.500E+01   2.614E+02   6.000E-01   2.574E+02   3.082E-02  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.601E+01
 6.000E+01   2.250E+02   6.000E-01   2.212E+02   3.524E-02  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.791E+01
 6.500E+01   1.971E+02   6.000E-01   1.934E+02   5.353E-02  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
2.169E+01
 7.000E+01   1.450E+02   1.000E-02   1.414E+02   8.668E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
2.685E+01
 8.000E+01   9.496E+01   6.000E-01   9.155E+01   1.309E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
2.755E+01
 1.000E+02   9.010E+01   1.000E-02   8.693E+01   1.586E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.891E+01
 1.100E+02   8.678E+01   6.000E-01   8.369E+01   1.702E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.725E+01
 1.200E+02   8.257E+01   1.000E-02   7.958E+01   1.902E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.627E+01
```

```
 1.300E+02   8.103E+01   1.000E-02   7.814E+01   2.198E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.558E+01
 1.400E+02   7.763E+01   6.000E-01   7.483E+01   2.476E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.510E+01
 1.500E+02   7.377E+01   6.000E-01   7.107E+01   2.791E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.478E+01
 1.900E+02   7.123E+01   1.000E-02   6.876E+01   3.439E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.434E+01
 2.700E+02   6.906E+01   1.000E-02   6.690E+01   4.423E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.394E+01
 6.150E+02   6.711E+01   1.000E-02   6.562E+01   6.731E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.460E+01
 1.000E+06   1.876E+01   9.984E-01   1.874E+01   9.999E-01  -9.990E-02   9.980E-03
7.188E+00   5.166E+01  -9.990E-02   9.980E-03   7.188E+00   5.166E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
6.100E+01
#   19
#  9.600E-04
!  t            wpT         wpRH        dsT         dsRH        wpPHnd      wpCLnd
wpPHd       wpCLd       dsPHnd      dsCLnd      dsPHd       dsCLd       ipkPHnd
ipkCLnd     ipkPHd      ipkCLd      barPHnd     barCLnd     barPHd      barCLd
PercFlux5m
 5.020E+01   2.347E+02   7.626E-02   2.291E+02   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.410E+01
 5.300E+01   2.741E+02   2.824E-02   2.699E+02   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.502E+01
 5.500E+01   2.658E+02   2.839E-02   2.618E+02   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.596E+01
 6.000E+01   2.317E+02   3.080E-02   2.279E+02   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.783E+01
 6.500E+01   1.990E+02   4.671E-02   1.953E+02   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
2.159E+01
 7.000E+01   1.391E+02   7.816E-02   1.356E+02   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
2.687E+01
 8.000E+01   1.002E+02   1.151E-01   9.675E+01   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
2.757E+01
```

```
 1.000E+02   9.345E+01   1.403E-01   9.029E+01   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.883E+01
 1.100E+02   8.871E+01   1.506E-01   8.563E+01   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.717E+01
 1.200E+02   8.745E+01   1.685E-01   8.446E+01   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.621E+01
 1.300E+02   8.409E+01   1.949E-01   8.120E+01   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.552E+01
 1.400E+02   8.078E+01   2.201E-01   7.798E+01   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.504E+01
 1.500E+02   7.773E+01   2.487E-01   7.503E+01   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.472E+01
 1.800E+02   7.527E+01   2.937E-01   7.274E+01   6.000E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.440E+01
 2.250E+02   7.300E+01   3.554E-01   7.068E+01   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.406E+01
 3.150E+02   7.077E+01   4.423E-01   6.873E+01   1.000E-02  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.381E+01
 4.750E+02   6.863E+01   5.481E-01   6.693E+01   1.000E-02  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.378E+01
 6.150E+02   6.716E+01   6.309E-01   6.567E+01   6.000E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.455E+01
 1.000E+06   1.883E+01   9.984E-01   1.881E+01   9.999E-01  -9.990E-02   9.980E-03
7.187E+00   5.166E+01  -9.990E-02   9.980E-03   7.187E+00   5.166E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
6.079E+01
#   29
#   1.153E-03
!  t           wpT         wpRH        dsT         dsRH        wpPHnd      wpCLnd
wpPHd       wpCLd       dsPHnd      dsCLnd      dsPHd       dsCLd       ipkPHnd
ipkCLnd     ipkPHd      ipkCLd      barPHnd     barCLnd     barPHd      barCLd
PercFlux5m
 5.020E+01   2.260E+02   4.900E-01   2.200E+02   5.500E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.400E+01
 5.100E+01   2.640E+02   5.500E-01   2.600E+02   6.500E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.430E+01
```

```
 5.200E+01   2.740E+02   6.500E-01   2.690E+02   8.500E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.460E+01
 5.500E+01   2.720E+02   8.500E-01   2.680E+02   4.900E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.590E+01
 6.000E+01   2.550E+02   9.000E-01   2.510E+02   9.500E-01  -9.990E-02   9.980E-03
8.580E+00   7.362E+01  -9.990E-02   9.980E-03   8.580E+00   7.362E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.960E+01
 6.500E+01   2.260E+02   9.500E-01   2.220E+02   9.000E-01  -9.990E-02   9.980E-03
8.580E+00   7.362E+01  -9.990E-02   9.980E-03   8.580E+00   7.362E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
2.140E+01
 1.180E+03   6.540E+01   4.900E-01   6.440E+01   5.500E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   7.640E+00   5.837E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
3.910E+01
 1.420E+03   6.340E+01   5.500E-01   6.250E+01   6.500E-01  -9.990E-02   9.980E-03
7.640E+00   5.837E+01  -9.990E-02   9.980E-03   7.640E+00   5.837E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
3.930E+01
 1.680E+03   6.130E+01   6.500E-01   6.050E+01   8.500E-01  -9.990E-02   9.980E-03
7.640E+00   5.837E+01  -9.990E-02   9.980E-03   7.640E+00   5.837E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
3.940E+01
 1.900E+03   5.940E+01   8.500E-01   5.860E+01   4.900E-01  -9.990E-02   9.980E-03
7.640E+00   5.837E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
3.960E+01
 1.950E+03   5.910E+01   9.000E-01   5.840E+01   9.500E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
4.140E+01
 1.975E+03   5.900E+01   9.500E-01   5.830E+01   9.000E-01  -9.990E-02   9.980E-03
9.400E+00   8.836E+01  -9.990E-02   9.980E-03   9.400E+00   8.836E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
4.320E+01
 2.060E+03   5.890E+01   4.900E-01   5.820E+01   5.500E-01  -9.990E-02   9.980E-03  -
9.990E-02   9.980E-03  -9.990E-02   9.980E-03   7.020E+00   4.928E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
4.500E+01
 2.080E+03   5.880E+01   5.500E-01   5.810E+01   6.500E-01  -9.990E-02   9.980E-03
7.020E+00   4.928E+01  -9.990E-02   9.980E-03   7.020E+00   4.928E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
4.670E+01
 2.100E+03   5.870E+01   6.500E-01   5.800E+01   8.500E-01  -9.990E-02   9.980E-03
7.020E+00   4.928E+01  -9.990E-02   9.980E-03   7.020E+00   4.928E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
4.850E+01
 2.120E+03   5.860E+01   8.500E-01   5.790E+01   4.900E-01  -9.990E-02   9.980E-03
7.020E+00   4.928E+01  -9.990E-02   9.980E-03  -9.990E-02   9.980E-03  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
5.030E+01
 2.140E+03   2.000E+01   9.000E-01   5.780E+01   9.000E-01  -9.990E-02   9.980E-03
7.020E+00   4.928E+01  -9.990E-02   9.980E-03   7.020E+00   4.928E+01  -9.990E-02
9.980E-03   9.830E+00   9.663E+01  -9.990E-02   9.663E+01   9.830E+00   9.663E+01
1.000E+00
 2.160E+03   2.000E+01   9.000E-01   5.770E+01   9.000E-01  -9.990E-02   9.980E-03
7.020E+00   4.928E+01  -9.990E-02   9.980E-03   7.020E+00   4.928E+01  -9.990E-02
```

9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
2.180E+03	2.000E+01	9.000E-01	5.760E+01	9.000E-01	-9.990E-02	9.980E-03
7.190E+00	5.170E+01	-9.990E-02	9.980E-03	7.190E+00	5.170E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
2.200E+03	2.000E+01	9.000E-01	5.750E+01	9.000E-01	-9.990E-02	9.980E-03
7.020E+00	4.928E+01	-9.990E-02	9.980E-03	7.020E+00	4.928E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
2.600E+03	4.400E+01	9.000E-01	5.570E+01	9.000E-01	-9.990E-02	9.980E-03
7.020E+00	4.928E+01	-9.990E-02	9.980E-03	7.020E+00	4.928E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
3.050E+03	5.600E+01	9.000E-01	5.400E+01	9.000E-01	-9.990E-02	9.980E-03
7.020E+00	4.928E+01	-9.990E-02	9.980E-03	7.020E+00	4.928E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
3.600E+03	6.700E+01	9.000E-01	5.240E+01	9.000E-01	-9.990E-02	9.980E-03
7.190E+00	5.170E+01	-9.990E-02	9.980E-03	7.190E+00	5.170E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
4.300E+03	6.700E+01	9.000E-01	5.060E+01	9.000E-01	-9.990E-02	9.980E-03
7.020E+00	4.928E+01	-9.990E-02	9.980E-03	7.020E+00	4.928E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
5.100E+03	9.800E+01	9.000E-01	4.890E+01	9.000E-01	-9.990E-02	9.980E-03
7.020E+00	4.928E+01	-9.990E-02	9.980E-03	7.020E+00	4.928E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
6.000E+03	9.800E+01	9.000E-01	4.730E+01	9.000E-01	-9.990E-02	9.980E-03
7.020E+00	4.928E+01	-9.990E-02	9.980E-03	7.020E+00	4.928E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
7.000E+03	9.800E+01	9.000E-01	4.560E+01	9.000E-01	-9.990E-02	9.980E-03
7.190E+00	5.170E+01	-9.990E-02	9.980E-03	7.190E+00	5.170E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
8.000E+03	9.800E+01	9.000E-01	4.420E+01	9.000E-01	-9.990E-02	9.980E-03
7.020E+00	4.928E+01	-9.990E-02	9.980E-03	7.020E+00	4.928E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
1.000E+00						
1.000E+06	1.890E+01	9.000E-01	1.890E+01	9.000E-01	-9.990E-02	9.980E-03
7.187E+00	5.166E+01	-9.990E-02	9.980E-03	7.187E+00	5.166E+01	-9.990E-02
9.980E-03	9.830E+00	9.663E+01	-9.990E-02	9.663E+01	9.830E+00	9.663E+01
6.040E+01						

VII-3.3.11 EXCEL SPREADSHEET REPLICATING PREWAP TEST CASE

Time Period	Time (yr),	Waste Pack Temp.(C),	Drip shield temp. (C),		Drift wall temp.(C),	Invert temp. (C),	Waste pack RH,	RH	pH	Reason	Drip shield RH,	RH
1st Data Set Page 1												
1st	0.0	22.29	-99.90	dst<0	22.28	22.31	1.00	0.50	-0.0999	< 50 Yrs	-99.900	-100.40
1st	1.0	84.66	-99.90	dst<0	67.97	75.01	-99.90	-100.40	-0.0999	< 50 Yrs	0.500	0.00
2nd	50.0	66.57	-99.90	dst<0	61.20	63.34	0.01	-0.49	-0.0999	RH<0.5	-99.900	-100.40
2nd	50.2	236.17	230.51		109.78	188.46	0.01	-0.49	-0.0999	RH<0.5	0.084	-0.42
2nd	51.0	270.68	266.03		130.38	229.31	0.01	-0.49	-0.0999	RH<0.5	0.037	-0.46
2nd	53.0	271.01	266.81		143.36	239.70	0.60	0.10	9.4000	pH const	0.030	-0.47
2nd	55.0	261.42	257.42		144.18	240.14	0.60	0.10	9.4000	pH const	0.031	-0.47
2nd	60.0	225.01	221.23		132.81	194.68	0.60	0.10	9.4000	pH const	0.035	-0.47
2nd	65.0	197.08	193.44		120.76	173.41	0.60	0.10	9.4000	pH const	0.054	-0.45
2nd	70.0	145.00	141.44		97.57	128.48	0.01	-0.49	-0.0999	RH<0.5	0.087	-0.41
2nd	80.0	94.96	91.55		81.49	93.83	0.60	0.10	9.4000	pH const	0.131	-0.37
2nd	100.0	90.10	86.93		77.19	89.98	0.01	-0.49	-0.0999	RH<0.5	0.159	-0.34
2nd	110.0	86.78	83.69		74.59	87.42	0.60	0.10	9.4000	pH const	0.170	-0.33
2nd	120.0	82.57	79.58		71.23	83.81	0.01	-0.49	-0.0999	RH<0.5	0.190	-0.31
2nd	130.0	81.03	78.14		70.05	82.46	0.01	-0.49	-0.0999	RH<0.5	0.220	-0.28
2nd	140.0	77.63	74.83		67.53	79.49	0.60	0.10	9.4000	pH const	0.248	-0.25
2nd	150.0	73.77	71.07		64.77	76.16	0.60	0.10	9.4000	pH const	0.279	-0.22
2nd	190.0	71.23	68.76		62.97	73.93	0.01	-0.49	-0.0999	RH<0.5	0.344	-0.16
2nd	270.0	69.06	66.90		61.33	71.83	0.01	-0.49	-0.0999	RH<0.5	0.442	-0.06
2nd	615.0	67.11	65.62		59.67	69.67	0.01	-0.49	-0.0999	RH<0.5	0.673	0.17
	1000000.0	18.76	18.74		18.61	18.75	1.00	0.50		Interpolate	1.000	0.50

Skip if wp or ds Temp < 0	wpRH or dsRH > Cor Lim (.501)	wpRH i<.501 & i+1 >.501	dsRH i<.501 & i+1 >.501	wpRH i<.501 & i-1>=.501	dsRH i<.501 & I-1 >.501	Skip if wpRH I<.501 I+1<.501 , I-1<.501	Skip if dsRH I<.501 I+1<.501 , I-1<.501	SAVE LINE	Drift wall RH,	Backfill RH,	Invert RH,	Liquid Satr. @ Drip Shield,	Liquid Satr.@Invert,	Air mass Frac,
1st Data Set page 2														
TRUE	TRUE	FALSE	FALSE					cull	1.00	-99.90	1.00	-99.90	0.00	-99.90
TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.88	-99.90	0.02	-99.90
TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	cull	1.00	-99.90	0.97	-99.90	0.03	-99.90
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	cull	0.66	0.97	0.08	0.00	0.00	0.00
FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	save	0.50	0.65	0.03	0.00	0.00	0.01
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	0.37	0.49	0.03	0.00	0.00	0.01
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	0.36	0.48	0.03	0.00	0.00	0.01
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	0.37	0.49	0.06	0.00	0.00	0.11
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	0.50	0.67	0.10	0.00	0.00	0.13
FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	save	0.75	0.99	0.29	0.00	0.00	0.19
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	0.99	1.00	0.93	0.13	0.06	0.22
FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	save	0.99	1.00	0.98	0.16	0.11	0.33
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	1.00	1.00	0.98	0.16	0.12	0.40
FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	save	1.00	1.00	0.99	0.17	0.12	0.48
FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	save	1.00	1.00	0.99	0.17	0.12	0.51
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	1.00	1.00	0.99	0.17	0.13	0.58
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	save	1.00	1.00	0.99	0.18	0.14	0.64
FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	save	1.00	1.00	0.99	0.19	0.14	0.67
FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	save	1.00	1.00	0.99	0.19	0.14	0.71
FALSE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.74
NA	NA	NA	NA	NA	NA	NA	NA	SAVE	1.00	1.00	1.00	0.22	0.17	0.98

Water Vapor flux at Dwall (kg/yr/m of drift),	Air flux at Dwall(kg/yr/m of drift),	A Drip Shield Evapo. rate (m3/yr),	Backfill Evapo. Rate (m3/yr),	Invert Evapo. Rate (m3/yr),	Percolation Flux at 5 m (mm/yr),	Volume flow at top dripshield (m3/yr),	volume flow at invert (m3/yr),	Top of the dripshield Temp (C)
1st Data Set Page 3								
0.00	0.00	-99.90	-99.90	-99.90	15.31	*-99.90*	0.00	-99.90
24.31	10.66	-99.90	-99.90	-99.90	14.39	*-99.90*	-0.01	-99.90
6.24	2.92	-99.90	-99.90	-99.90	14.21	*-99.90*	0.00	-99.90
2939.24	0.83	0.00	0.00	0.00	14.15	0.00	0.00	231.60
38.61	-0.02	0.00	0.00	0.00	14.46	0.00	0.00	266.97
10.29	0.42	0.00	0.00	0.00	15.07	0.00	0.00	267.65
10.34	0.47	0.00	0.00	0.00	16.01	0.00	0.00	258.21
0.29	-0.36	0.00	0.00	0.00	17.91	0.00	0.00	221.97
0.72	-0.45	0.00	0.00	0.00	21.69	0.00	0.00	194.14
265.67	34.96	0.00	0.00	0.00	26.85	0.00	0.00	142.11
220.81	19.33	0.13	-0.01	0.28	27.55	0.00	0.00	92.18
128.40	16.79	0.10	0.00	0.25	18.91	0.00	0.00	87.48
92.59	14.98	0.09	0.00	0.21	17.25	0.00	0.00	84.23
62.22	12.52	0.07	0.00	0.17	16.27	0.00	0.00	80.08
54.38	11.69	0.06	0.00	0.16	15.58	0.00	0.00	78.62
41.10	9.98	0.05	0.00	0.13	15.10	0.00	0.00	75.29
30.13	7.97	0.04	0.00	0.10	14.78	0.00	0.00	71.50
24.76	6.82	0.04	0.00	0.09	14.34	0.00	0.00	69.15
20.72	5.94	0.03	0.00	0.08	13.94	0.12	-0.04	67.22
17.30	5.12	0.03	0.00	0.07	14.60	0.04	-0.04	65.83
0.01	0.00	0.00	0.00	0.00	61.00	0.00	0.18	18.73

2nd Data Set Page 1

1st	0.0	22.30	-99.90	dst<0	22.29	22.31	1.00	0.50	-0.0999	< 50 Yrs	-99.900	-100.40
1st	1.0	80.62	-99.90	dst<0	63.32	70.79	0.48	-0.02	-0.0999	< 50 Yrs	-99.900	-100.40
1st	2.0	87.48	-99.90	dst<0	71.72	78.43	0.53	0.03	-0.0999	< 50 Yrs	-99.900	-100.40
1st	5.0	94.43	-99.90	dst<0	80.80	86.61	0.59	0.09	-0.0999	< 50 Yrs	-99.900	-100.40
1st	20.0	89.09	-99.90	dst<0	79.72	83.41	0.69	0.19	-0.0999	< 50 Yrs	-99.900	-100.40
1st	25.0	85.28	-99.90	dst<0	76.79	80.17	0.71	0.21	-0.0999	< 50 Yrs	-99.900	-100.40
1st	30.0	81.77	-99.90	dst<0	74.00	77.13	0.72	0.22	-0.0999	< 50 Yrs	-99.900	-100.40
1st	40.0	74.92	-99.90	dst<0	68.44	71.10	0.76	0.26	-0.0999	< 50 Yrs	-99.900	-100.40
2nd	50.0	67.81	-99.90	dst<0	62.52	64.74	0.79	0.29	-0.0999	Seep< -99	-99.900	-100.40
2nd	50.2	234.74	229.08		109.30	187.17	0.08	-0.42	-0.0999	RH<0.5	0.010	-0.49
2nd	51.0	268.56	263.90		127.61	226.17	0.04	-0.46	-0.0999	RH<0.5	0.010	-0.49
2nd	53.0	274.12	269.92		142.88	239.45	0.03	-0.47	-0.0999	RH<0.5	0.010	-0.49
2nd	55.0	265.76	261.76		144.99	240.58	0.03	-0.47	-0.0999	RH<0.5	0.600	0.10
2nd	60.0	231.69	227.92		136.31	201.92	0.03	-0.47	-0.0999	RH<0.5	0.600	0.10
2nd	65.0	198.98	195.33		122.97	177.27	0.05	-0.45	-0.0999	RH<0.5	0.600	0.10
2nd	70.0	139.13	135.58		97.63	128.60	0.08	-0.42	-0.0999	RH<0.5	0.600	0.10
2nd	80.0	100.16	96.75		85.70	97.09	0.12	-0.39	-0.0999	RH<0.5	0.010	-0.49
2nd	100.0	93.45	90.29		81.85	94.21	0.14	-0.36	-0.0999	RH<0.5	0.600	0.10
2nd	110.0	88.71	85.63		79.55	92.22	0.15	-0.35	-0.0999	RH<0.5	0.010	-0.49
2nd	120.0	87.45	84.46		78.48	91.23	0.17	-0.33	-0.0999	RH<0.5	0.600	0.10
2nd	130.0	84.09	81.20		75.65	88.54	0.19	-0.31	-0.0999	RH<0.5	0.010	-0.49
2nd	140.0	80.78	77.98		72.99	85.79	0.22	-0.28	-0.0999	RH<0.5	0.010	-0.49
2nd	150.0	77.73	75.03		70.68	83.22	0.25	-0.25	-0.0999	RH<0.5	0.600	0.10
2nd	180.0	75.27	72.74		68.77	80.98	0.29	-0.21	-0.0999	RH<0.5	0.600	0.10
2nd	225.0	73.00	70.68		66.84	78.66	0.36	-0.15	-0.0999	RH<0.5	0.010	-0.49
2nd	315.0	70.77	68.73		64.91	76.35	0.44	-0.06	-0.0999	RH<0.5	0.010	-0.49
2nd	475.0	68.63	66.93		62.86	73.80	0.55	0.05	9.4000	pH const	0.010	-0.49
2nd	615.0	67.16	65.67		61.41	71.94	0.63	0.13	9.4000	pH const	0.600	0.10
5th	1000000.0	18.83	18.81		18.69	18.84	1.00	0.50		Interpolate	1.000	0.50

2nd Data Set Page 2														
TRUE	TRUE	FALSE	FALSE					cull	1.00	-99.90	1.00	-99.90	0.00	-99.90
TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.89	-99.90	0.02	-99.90
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.87	-99.90	0.02	-99.90
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.86	-99.90	0.01	-99.90
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.90	-99.90	0.00	-99.90
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.91	-99.90	0.01	-99.90
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.93	-99.90	0.01	-99.90
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.95	-99.90	0.02	-99.90
TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.96	-99.90	0.03	-99.90
FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	save	0.66	0.98	0.08	0.00	0.00	0.00
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	cull	0.53	0.68	0.04	0.00	0.00	0.01
FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	save	0.37	0.49	0.03	0.00	0.00	0.01
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	0.35	0.47	0.03	0.00	0.00	0.02
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	0.34	0.46	0.05	0.00	0.00	0.09
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	0.47	0.64	0.09	0.00	0.00	0.12
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	0.75	0.99	0.29	0.00	0.00	0.19
FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	save	0.99	1.00	0.88	0.07	0.03	0.15
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	0.99	1.00	0.92	0.12	0.06	0.22
FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	save	0.99	1.00	0.96	0.14	0.08	0.27
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	0.99	1.00	0.98	0.15	0.09	0.29
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	cull	0.99	1.00	0.98	0.16	0.12	0.37
FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	FALSE	save	1.00	1.00	0.98	0.17	0.12	0.44
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	1.00	1.00	0.99	0.17	0.12	0.50
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	save	1.00	1.00	0.99	0.17	0.12	0.55
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	cull	1.00	1.00	0.99	0.17	0.13	0.59
FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	TRUE	save	1.00	1.00	0.99	0.18	0.14	0.64
FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	0.99	0.19	0.14	0.68
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	0.99	0.19	0.14	0.70
NA	NA	NA	NA	NA	NA	NA	NA	SAVE	1.00	1.00	1.00	0.22	0.17	0.98

2nd Data Set Page 3

0.00	0.00	-99.90	-99.90	-99.90	15.26	***-99.90***	0.00	-99.90
22.48	11.92	-99.90	-99.90	-99.90	14.34	***-99.90***	-0.01	-99.90
25.93	10.23	-99.90	-99.90	-99.90	14.35	***-99.90***	-0.01	-99.90
30.46	10.25	-99.90	-99.90	-99.90	14.47	***-99.90***	0.00	-99.90
17.14	7.08	-99.90	-99.90	-99.90	14.54	***-99.90***	0.00	-99.90
14.44	5.76	-99.90	-99.90	-99.90	14.47	***-99.90***	0.00	-99.90
12.78	5.25	-99.90	-99.90	-99.90	14.40	***-99.90***	0.00	-99.90
9.72	4.56	-99.90	-99.90	-99.90	14.27	***-99.90***	0.00	-99.90
7.07	3.11	-99.90	-99.90	-99.90	14.16	***-99.90***	0.00	-99.90
3038.39	0.86	0.00	0.00	0.00	14.10	0.00	0.00	230.16
128.02	0.00	0.00	0.00	0.00	14.41	0.00	0.00	264.85
10.26	0.39	0.00	0.00	0.00	15.02	0.00	0.00	270.76
10.28	0.52	0.00	0.00	0.00	15.96	0.00	0.00	262.55
0.16	-0.22	0.00	0.00	0.00	17.83	0.00	0.00	228.65
0.47	-0.46	0.00	0.00	0.00	21.59	0.00	0.00	196.03
265.22	34.85	0.00	0.00	0.00	26.87	0.00	0.00	136.25
397.49	30.37	0.15	0.00	0.20	27.57	0.00	0.00	97.38
229.18	19.38	0.13	-0.01	0.24	18.83	0.00	0.00	90.84
172.74	18.16	0.11	0.00	0.27	17.17	0.00	0.00	86.16
151.04	17.51	0.11	0.00	0.27	16.21	0.00	0.00	84.97
104.54	15.58	0.09	0.00	0.23	15.52	0.00	0.00	81.68
75.80	13.70	0.08	0.00	0.19	15.04	0.00	0.00	78.44
58.01	12.05	0.07	0.00	0.16	14.72	0.00	0.00	75.47
46.80	10.73	0.06	0.00	0.14	14.40	0.00	0.00	73.14
37.85	9.50	0.05	0.00	0.12	14.06	0.00	0.00	71.03
30.41	8.03	0.04	0.00	0.10	13.81	0.10	-0.13	69.04
24.31	6.72	0.04	0.00	0.09	13.78	0.06	-0.08	67.18
20.75	5.94	0.03	0.00	0.08	14.55	0.04	-0.03	65.88
0.02	0.00	0.00	0.00	0.00	60.79	0.00	0.18	18.81

3rd Data Set Page 1

1st	0.0	22.30	-99.90	dst<0	22.30	22.30	1.00	0.50	-0.0999	< 50 Yrs	-99.900	-100.40
1st	1.0	78.80	-99.90	dst<0	61.20	68.80	0.47	-0.03	-0.0999	< 50 Yrs	-99.900	-100.40
1st	40.0	75.60	-99.90	dst<0	69.10	71.90	0.76	0.26	-0.0999	< 50 Yrs	-99.900	-100.40
2nd	50.2	226.00	220.00		106.00	182.00	0.49	-0.01	-0.0999	RH<0.5	0.550	0.05
2nd	51.0	264.00	260.00		123.00	220.00	0.55	0.05	9.4000	pH const	0.650	0.15
2nd	52.0	274.00	269.00		134.00	233.00	0.65	0.15	9.4000	pH const	0.850	0.35
2nd	55.0	272.00	268.00		145.00	240.00	0.85	0.35		Interpolate	0.490	-0.01
2nd	60.0	255.00	251.00		145.00	219.00	0.90	0.40		Interpolate	0.950	0.45
2nd	65.0	226.00	222.00		135.00	200.00	0.95	0.45		Interpolate	0.900	0.40
3rd	1180.0	65.40	64.40		60.10	70.20	0.49	-0.01	-0.0999	RH<0.5	0.550	0.05
3rd	1420.0	63.40	62.50		58.00	67.50	0.55	0.05	7.6400	pH const	0.650	0.15
3rd	1680.0	61.30	60.50		56.00	64.80	0.65	0.15	7.6400	pH const	0.850	0.35
3rd	1900.0	59.40	58.60		54.10	62.20	0.85	0.35	7.6400	pH const	0.490	-0.01
3rd	1950.0	59.10	58.40		53.80	61.90	0.90	0.40		Interpolate	0.950	0.45
3rd	1975.0	59.00	58.30		53.70	61.70	0.95	0.45		Interpolate	0.900	0.40
4th	2060.0	58.90	58.20		53.60	61.60	0.49	-0.01	-0.0999	RH<0.5	0.550	0.05
4th	2080.0	58.80	58.10		53.50	61.40	0.55	0.05	7.0200	pH const	0.650	0.15
4th	2100.0	58.70	58.00		53.40	61.30	0.65	0.15	7.0200	pH const	0.850	0.35
4th	2120.0	58.60	57.90		53.30	61.20	0.85	0.35	7.0200	pH const	0.490	-0.01
4th	2140.0	20.00	57.80		53.20	61.00	0.90	0.40		Interpolate	0.900	0.40
4th	2160.0	20.00	57.70		53.10	60.90	0.90	0.40		Interpolate	0.900	0.40
4th	2180.0	20.00	57.60		53.00	60.80	0.90	0.40		Interpolate	0.900	0.40
4th	2200.0	20.00	57.50		52.90	60.60	0.90	0.40		Interpolate	0.900	0.40
4th	2600.0	44.00	55.70		51.20	58.20	0.90	0.40		Interpolate	0.900	0.40
4th	3050.0	56.00	54.00		49.50	56.10	0.90	0.40		Interpolate	0.900	0.40
4th	3600.0	67.00	52.40		48.00	54.00	0.90	0.40		Interpolate	0.900	0.40
4th	4300.0	67.00	50.60		46.30	51.80	0.90	0.40		Interpolate	0.900	0.40
4th	5100.0	98.00	48.90		44.80	50.00	0.90	0.40		Interpolate	0.900	0.40
4th	6000.0	98.00	47.30		43.30	48.10	0.90	0.40		Interpolate	0.900	0.40
4th	7000.0	98.00	45.60		41.70	46.40	0.90	0.40		Interpolate	0.900	0.40
4th	8000.0	98.00	44.20		40.40	44.80	0.90	0.40		Interpolate	0.900	0.40
5th	1000000.0	18.90	18.90		18.80	18.90	0.90	0.40		Interpolate	0.900	0.40

3rd Data Set Page 2															
TRUE	TRUE	FALSE	FALSE					cull	1.00	-99.90	1.00	-99.90	0.00	-99.90
TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	cull	1.00	-99.90	0.89	-99.90	0.02	-99.90
TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	cull	1.00	-99.90	0.94	-99.90	0.02	-99.90
FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	save	0.68	1.00	0.09	0.00	0.00	0.04
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	0.56	0.72	0.04	0.00	0.00	0.00
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	0.45	0.60	0.03	0.00	0.00	0.01
FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	save	0.36	0.48	0.03	0.00	0.00	0.02
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	0.29	0.39	0.04	0.00	0.00	0.06
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	0.35	0.47	0.06	0.00	0.00	0.10
FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	save	1.00	1.00	0.99	0.19	0.14	0.73
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.76
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.79
FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.82
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.82
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.82
FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.82
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.82
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.83
FALSE	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.83
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.83
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.83
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.83
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.19	0.14	0.83
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.20	0.15	0.85
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.20	0.15	0.87
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.20	0.15	0.88
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.20	0.15	0.89
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.21	0.15	0.90
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.21	0.16	0.91
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.21	0.16	0.92
FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	save	1.00	1.00	1.00	0.21	0.16	0.93
NA	NA	NA	NA	NA	NA	NA	NA	SAVE	1.00	1.00	1.00	0.22	0.17	0.98

3rd Data Set Page 3								
0.00	0.00	-99.90	-99.90	-99.90	15.20	***-99.90***	0.00	-99.90
21.50	12.40	-99.90	-99.90	-99.90	14.20	***-99.90***	-0.01	-99.90
10.00	4.62	-99.90	-99.90	-99.90	14.20	***-99.90***	0.00	-99.90
2910.00	4.62	0.00	0.01	0.00	14.00	0.00	0.00	221.00
585.00	0.18	0.00	0.00	0.00	14.30	0.00	0.00	261.00
23.90	0.02	0.00	0.00	0.00	14.60	0.00	0.00	270.00
10.10	0.49	0.00	0.00	0.00	15.90	0.00	0.00	269.00
-0.13	0.10	0.00	0.00	0.00	19.60	0.00	0.00	252.00
0.20	-0.25	0.00	0.00	0.00	21.40	0.00	0.00	223.00
17.70	5.21	0.03	0.00	0.07	39.10	0.01	0.09	64.60
14.20	4.33	0.02	0.00	0.06	39.30	0.01	0.10	62.60
11.40	3.59	0.02	0.00	0.05	39.40	0.01	0.11	60.60
9.15	2.97	0.02	0.00	0.04	39.60	0.01	0.11	58.70
8.90	2.89	0.02	0.00	0.04	41.40	0.01	0.12	58.50
8.79	2.86	0.01	0.00	0.04	43.20	0.01	0.12	58.40
8.69	2.83	0.01	0.00	0.04	45.00	0.01	0.13	58.30
8.58	2.80	0.01	0.00	0.04	46.70	0.01	0.13	58.20
8.48	2.77	0.01	0.00	0.04	48.50	0.01	0.14	58.10
8.38	2.75	0.01	0.00	0.04	50.30	0.01	0.14	58.00
8.28	2.72	1.00	0.00	0.04	1.00	0.01	0.15	57.90
8.18	2.69	0.45	0.00	0.03	1.00	0.01	0.15	57.80
8.09	2.66	0.00	0.00	0.03	1.00	0.01	0.16	57.70
8.01	2.62	-0.01	0.00	0.03	1.00	0.01	0.16	57.60
6.41	2.10	1.00	0.00	0.03	1.00	0.00	0.17	55.80
5.34	1.78	0.45	0.00	0.02	1.00	0.00	0.17	54.10
4.45	1.48	0.00	0.00	0.02	1.00	0.00	0.17	52.40
3.57	1.22	-0.01	0.00	0.02	1.00	0.00	0.17	50.70
3.08	0.93	1.00	0.00	0.01	1.00	0.00	0.18	49.00
2.61	0.79	0.45	0.00	0.01	1.00	0.00	0.19	47.30
2.27	0.70	0.00	0.00	0.01	1.00	0.00	0.21	45.70
2.00	0.62	-0.01	0.00	0.01	1.00	0.00	0.22	44.20
0.02	-0.01	0.00	0.00	0.00	60.40	0.00	0.22	18.90

INTENTIONALLY LEFT BLANK

ATTACHMENT VIII

TSPA-SR MODEL DATA SUBMITTAL 'README' FILE

A. "SR00_037ne6.gsm" is the actual TSPA-SR GoldSim model file: Rev00B, base case, nominal scenario, no backfill, median-value realization, 1,000,000 years.

The model file and associated external files (listed below) are required to run the model. Model results/output can be viewed and exported using just the .GSM Model File and GoldSim Code.

Software used -

GoldSim Software, GoldSim V6.04.00 STN 10310-6.04.00-00
WAPDEG Software ID: 100-4.0-00 version 4.0
FEHM 2.1 Software ID: 10086-2.10-00
SZ_CONVLUTE Software ID: 10207-2.0-00 version 2.0
ASHPLUME 1.4LVdll Software ID: 10022-1.4LVdll-00
GVP Software ID 10341-SRR-1.02-00 version V1.02
MFD Software ID 10342-SRR-1.01-00 version V1.01
SCCD Sofware ID 10343-SRR-2.01-00 version V2.01

Model Constraints -

Model runs should be performed on WindowsNT or Windows95 OS

Model Limitations -

Model valid for 0 to 1,000,000 year simulations.

Model developed to simulate the release of radionuclides into the natural system and subsequent travel to the accessible environment and predict the annual dose to an average individual receptor at a 20 km boundary from the proposed repository location at Yucca Mountain, NV.

Model results are valid within the range of model input as defined by supporting AMRs.

B. Support files for Median-Value Realization

This is a list of the support files needed to run the TSPA_SR GoldSim model (for any nominal, median value realization case).

Table 1 - Support files required for a Median-Value Realization:

Date	Time	File size	File name	Description
			Seepage files	
5/17/00	11:04a	364,611	seepagedllv2.dll	seepage DLL
1/11/00	02:01p	265,450	CSNF_bf_high_pf_bin2.dat	Seepage DLL input files for the thermohydrology, for civil spent nuclear fuel, in the backfill case
1/11/00	02:02p	1,856,578	CSNF_bf_high_pf_bin3.dat	
1/11/00	02:01p	6,023,818	CSNF_bf_high_pf_bin4.dat	
1/11/00	01:58p	3,656,068	CSNF_bf_high_pf_bin5.dat	
1/11/00	02:26p	6,744,598	CSNF_bf_low_pf_bin1.dat	

Date	Time	File size	File name	Description
			Seepage files	
1/11/00	02:23p	4,673,647	CSNF_bf_low_pf_bin2.dat	Seepage DLL input files for the thermohydrology, for civil spent nuclear fuel, in the backfill case
5/18/00	12:48p	307,639	CSNF_bf_mean_pf_bin1.dat	
5/18/00	12:48p	1,332,229	CSNF_bf_mean_pf_bin2.dat	
5/18/00	12:49p	2,547,100	CSNF_bf_mean_pf_bin3.dat	
5/18/00	12:50p	4,889,020	CSNF_bf_mean_pf_bin4.dat	
5/18/00	12:50p	44,173	CSNF_bf_mean_pf_bin5.dat	
5/17/00	11:07a	59,958	CSNF_nbf_high_pf_bin2.dat	Seepage DLL input files for the thermohydrology, for civil spent nuclear fuel, in the no backfill case
5/17/00	11:08a	405,342	CSNF_nbf_high_pf_bin3.dat	
5/17/00	11:08a	1,339,158	CSNF_nbf_high_pf_bin4.dat	
5/17/00	11:09a	797,630	CSNF_nbf_high_pf_bin5.dat	
5/17/00	11:13a	1,539,566	CSNF_nbf_low_pf_bin1.dat	
5/17/00	11:14a	1,061,998	CSNF_nbf_low_pf_bin2.dat	
5/17/00	11:14a	17,318	CSNF_nbf_mean_pf_bin1.dat	
5/17/00	11:14a	435,190	CSNF_nbf_mean_pf_bin2.dat	
5/17/00	11:15a	725,142	CSNF_nbf_mean_pf_bin3.dat	
5/17/00	11:16a	1,411,646	CSNF_nbf_mean_pf_bin4.dat	
5/17/00	11:16a	13,054	CSNF_nbf_mean_pf_bin5.dat	
1/11/00	02:21p	265,450	HLW_bf_high_pf_bin2.dat	Seepage DLL input files for the thermohydrology, for high level waste, in the backfill case (HLW fuel)
1/11/00	02:21p	1,856,578	HLW_bf_high_pf_bin3.dat	
1/11/00	02:20p	6,023,818	HLW_bf_high_pf_bin4.dat	
1/11/00	02:18p	3,656,068	HLW_bf_high_pf_bin5.dat	
1/11/00	02:32p	6,744,598	HLW_bf_low_pf_bin1.dat	
1/11/00	02:29p	4,673,647	HLW_bf_low_pf_bin2.dat	
5/18/00	12:51p	307,639	HLW_bf_mean_pf_bin1.dat	
5/18/00	01:26p	1,332,229	HLW_bf_mean_pf_bin2.dat	
5/18/00	01:27p	2,547,100	HLW_bf_mean_pf_bin3.dat	
5/18/00	01:28p	4,889,020	HLW_bf_mean_pf_bin4.dat	
5/18/00	01:28p	44,173	HLW_bf_mean_pf_bin5.dat	
5/17/00	11:16a	59,958	HLW_nbf_high_pf_bin2.dat	Seepage DLL input files for the thermohydrology, for high level waste, in the no backfill case
5/17/00	11:17a	405,342	HLW_nbf_high_pf_bin3.dat	
5/17/00	11:17a	1,339,158	HLW_nbf_high_pf_bin4.dat	
5/17/00	11:17a	797,630	HLW_nbf_high_pf_bin5.dat	
5/17/00	11:18a	1,539,566	HLW_nbf_low_pf_bin1.dat	
5/17/00	11:19a	1,061,998	HLW_nbf_low_pf_bin2.dat	
5/17/00	11:19a	17,318	HLW_nbf_mean_pf_bin1.dat	
5/17/00	11:19a	435,190	HLW_nbf_mean_pf_bin2.dat	
5/17/00	11:20a	725,142	HLW_nbf_mean_pf_bin3.dat	
5/17/00	11:20a	1,411,646	HLW_nbf_mean_pf_bin4.dat	
5/17/00	11:20a	13,054	HLW_nbf_mean_pf_bin5.dat	
5/17/00	10:38a	617	master_bf.in	Seepage DLL input files for thermal hydrology
5/17/00	10:39a	639	master_nbf.in	
1/17/00	12:49p	395	SeepFlowMean.dat	
1/17/00	12:52p	388	SeepFlowSD.dat	
1/17/00	12:54p	337	SeepFrac.dat	

Date	Time	File size	File name	Description
			Seepage files	
5/21/00	10:09p	880	seep_debug.dat	Seepage DLL output files
5/21/00	10:09p	104,520	seep_output.dat	
			WAPDEG files	
5/26/00	09:56a	778,752	wapdeg.dll	the waste package degradation (WAPDEG) DLL
4/26/00	12:00p	271,872	GVP.dll	gaussian variance partitioning DLL, supporting WAPDEG
4/26/00	01:50p	388,608	MFD.dll	manufacturing defect DLL, supporting WAPDEG
4/6/00	02:06p	438,784	SCCD.dll	stress corrosion cracking DLL, supporting WAPDEG
5/25/00	12:14p	4,106	debug4.txt	WAPDEG run log
5/25/00	12:13p	41,999	WD4DLL.ina	WAPDEG model files
5/4/00	03:51p	1,292,451	WD4DLL.oua	
5/4/00	03:51p	1,760,945	WD4DLL.out	
5/25/00	12:03p	363	WD4DLL.wap	
5/25/00	12:13p	9,119	WDdA22SR00.cdf	
5/25/00	12:13p	9,119	WDdA22x0p5.cdf	
5/25/00	12:13p	9,119	WDdA22x2p5.cdf	
5/25/00	12:13p	9,119	WDdTi7Sr00.cdf	
5/12/00	11:13a	12,524	WDgA22SR00.cdf	
5/12/00	11:13a	12,524	WDgA22x0p5.cdf	
5/12/00	11:13a	12,524	WDgA22x2p5.cdf	
5/24/00	12:27p	12,526	WDgTi7SR00.cdf	
2/16/00	08:59a	304,964	WDHLW_high_bin2.ou	
5/17/00	12:40p	355,504	WDHLW_nbf_high_bin2.ou	
5/25/00	12:13p	9,119	WDiA22x2p5.cdf	
1/14/00	02:06p	1,436	WDKlinM.fil	
1/14/00	09:26p	1,439	WDKlinO.fil	
5/25/00	12:13p	13,458	WDKISCCM.fil	
5/25/00	12:13p	13,458	WDKISCCO.fil	
11/24/99	03:55p	775	WDMFDND.cdf	
5/25/00	12:13p	4,732	WDMFDNDM.cdf	
5/25/00	12:13p	5,112	WDMFDNDO.cdf	
5/25/00	12:13p	10,329	WDMFDSizeM.cdf	
5/25/00	12:13p	10,329	WDMFDSizeO.cdf	
1/17/00	10:36a	1,463	WDndTi.cdf	
5/25/00	12:13p	9,119	WDndTi7SR00.cdf	
2/24/00	07:24p	411	WDRHcrit.fil	
5/25/00	12:13p	13,448	WDStressM.fil	
5/25/00	12:13p	13,448	WDStressO.fil	
			FEHM files	
5/26/00	07:39a	4,714,496	fehmn_sr.dll	FEHMN DLL
1/5/00	12:39p	3,715	afm_pch1.dpdp	FEHMN input files
3/13/00	09:36a	77	bf2.txt	
3/13/00	09:36a	77	bf3.txt	

Date	Time	File size	File name	Description
			Seepage files	
3/13/00	09:36a	77	ch1.txt	
3/13/00	09:36a	77	ch6.txt	
3/13/00	09:36a	77	chv.txt	
3/13/00	09:36a	77	chz.txt	
2/15/00	09:48p	291	fehmn.files	
3/3/00	11:30a	596	fehmn.gold	
3/3/00	11:30a	596	fehmn.gold.exp	
5/21/00	10:39p	2,528,738	fm_pchm1.chk	
6/1/00	10:46a	1,621	fm_pchm1.dat	
5/21/00	10:39p	3,908,630	fm_pchm1.fin	
1/4/00	10:20a	2,335,583	fm_pchm1.grid	
5/21/00	11:34p	186,157	fm_pchm1.his	
1/4/00	10:20a	29,301,805	fm_pchm1.stor	
5/21/00	11:34p	28,517	fm_pchm1.trc	
1/4/00	10:22a	3,349	pch1.rock	
3/13/00	09:36a	77	pp1.txt	
3/13/00	09:36a	77	pp2.txt	
3/13/00	09:36a	77	pp3.txt	
3/13/00	09:36a	77	pp4.txt	
3/13/00	09:37a	77	tsw4.txt	
3/13/00	09:37a	77	tsw5.txt	
3/13/00	09:37a	77	tsw6.txt	
3/13/00	09:37a	77	tsw7.txt	
3/13/00	09:37a	77	tsw8.txt	
3/13/00	09:37a	77	tsw9.txt	
10/11/99	06:05p	15,937,540	ff0100.ini	Flow field input files
10/11/99	06:06p	15,937,540	ff0200.ini	
10/11/99	08:55p	15,937,540	ff0300.ini	
10/14/99	09:40p	15,938,094	ff1100.ini	
10/14/99	09:43p	15,938,094	ff1200.ini	
10/14/99	09:43p	15,938,094	ff1300.ini	
10/14/99	09:45p	15,938,094	ff2100.ini	
10/14/99	09:46p	15,938,094	ff2200.ini	
10/14/99	09:47p	15,938,094	ff2300.ini	
1/4/00	10:19a	980,781	fm_pchm1.zone	Input zone files for the FEHMN DLL
3/3/00	11:12a	1,082,426	fm_pchm1.zone2	
3/3/00	11:12a	1,082,426	fm_pchm1.zone2.0200	
5/22/00	12:00p	2,108,971	ptrk.median	FEHMN particle tracking files
8/25/00	01:45p	2,108,990	ptrk.median_with_conversion_factors	
5/17/00	01:51p	2,011,054	ptrk.multrlz	
5/26/00	04:15p	2,011,157	ptrk.multrlz.0100	
5/26/00	04:16p	2,011,151	ptrk.multrlz.0200	
5/26/00	04:18p	2,011,148	ptrk.multrlz.0300	

Date	Time	File size	File name	Description
			Seepage files	
8/30/00	05:12p	730	UZ_Params_median.sr	input file containing the unsaturated zone transport parameters
5/21/00	10:09p	688	fehmn_real.bat	batch files
5/21/00	10:09p	1,118	fehmn_ts0.bat	
			SZ files	
3/17/00	09:52a	455,680	szconv_sr.dll	saturated zone (SZ) convolution DLL
3/8/00	03:53p	98,153	SZ_01_01	Saturated zone breakthrough curves input files
3/8/00	03:55p	98,153	SZ_01_02	
3/8/00	03:55p	98,153	SZ_01_03	
3/8/00	03:56p	98,153	SZ_01_04	
3/8/00	09:33p	98,153	SZ_02_01	
3/8/00	09:33p	98,153	SZ_02_02	
3/8/00	09:34p	98,153	SZ_02_03	
3/8/00	09:34p	98,153	SZ_02_04	
3/8/00	03:57p	196,153	SZ_03_01	
3/8/00	03:57p	196,153	SZ_03_02	
3/8/00	03:58p	196,153	SZ_03_03	
3/8/00	03:58p	196,153	SZ_03_04	
3/8/00	03:58p	245,153	SZ_04_01	
3/8/00	03:58p	245,153	SZ_04_02	
3/8/00	03:58p	245,153	SZ_04_03	
3/8/00	03:58p	245,153	SZ_04_04	
3/8/00	03:59p	98,153	SZ_05_01	
3/8/00	03:59p	98,153	SZ_05_02	
3/8/00	04:00p	98,153	SZ_05_03	
3/8/00	04:00p	98,153	SZ_05_04	
3/15/00	09:49a	245,153	SZ_06_01	
3/15/00	09:49a	245,153	SZ_06_02	
3/15/00	09:49a	245,153	SZ_06_03	
3/15/00	09:49a	245,153	SZ_06_04	
4/12/00	12:44p	245,153	SZ_07_01	
4/12/00	12:44p	245,153	SZ_07_02	
4/12/00	12:44p	245,153	SZ_07_03	
4/12/00	12:45p	245,153	SZ_07_04	
3/8/00	04:09p	196,153	SZ_08_01	
3/8/00	04:10p	196,153	SZ_08_02	
3/8/00	04:10p	196,153	SZ_08_03	
3/8/00	04:10p	196,153	SZ_08_04	
6/3/00	06:05p	383	sz_convolute2.dat	saturated zone convolution output file
			Other files	
2/3/00	04:46p	473,088	ashdll.dll	ashplume DLL
4/28/00	12:38p	262,201	soilexp.dll	soil removal factor DLL

C. Support files for Multiple Realizations

Probabilistic approach (Monte Carlo Analysis) was used to run the sensitivity cases. Multiple realizations (usually 100 and 300 realizations) were conducted.

Supporting files required to run a multiple realization case are similar to the one used for the median-value realization.

Table 2 - Support files required for Multiple Realizations

Date	Time	File size	File name	Description
			Seepage files	
5/17/00	11:04a	364,611	seepagedllv2.dll	seepage DLL
1/11/00	02:01p	265,450	CSNF_bf_high_pf_bin2.dat	Seepage DLL input files for the thermohydrology, for civil spent nuclear fuel, in the backfill case
1/11/00	02:02p	1,856,578	CSNF_bf_high_pf_bin3.dat	
1/11/00	02:01p	6,023,818	CSNF_bf_high_pf_bin4.dat	
1/11/00	01:58p	3,656,068	CSNF_bf_high_pf_bin5.dat	
1/11/00	02:26p	6,744,598	CSNF_bf_low_pf_bin1.dat	
1/11/00	02:23p	4,673,647	CSNF_bf_low_pf_bin2.dat	
5/18/00	12:48p	307,639	CSNF_bf_mean_pf_bin1.dat	
5/18/00	12:48p	1,332,229	CSNF_bf_mean_pf_bin2.dat	
5/18/00	12:49p	2,547,100	CSNF_bf_mean_pf_bin3.dat	
5/18/00	12:50p	4,889,020	CSNF_bf_mean_pf_bin4.dat	
5/18/00	12:50p	44,173	CSNF_bf_mean_pf_bin5.dat	
5/17/00	11:07a	59,958	CSNF_nbf_high_pf_bin2.dat	Seepage DLL input files for the thermohydrology, for civil spent nuclear fuel, in the no-backfill case
5/17/00	11:08a	405,342	CSNF_nbf_high_pf_bin3.dat	
5/17/00	11:08a	1,339,158	CSNF_nbf_high_pf_bin4.dat	
5/17/00	11:09a	797,630	CSNF_nbf_high_pf_bin5.dat	
5/17/00	11:13a	1,539,566	CSNF_nbf_low_pf_bin1.dat	
5/17/00	11:14a	1,061,998	CSNF_nbf_low_pf_bin2.dat	
5/17/00	11:14a	17,318	CSNF_nbf_mean_pf_bin1.dat	
5/17/00	11:14a	435,190	CSNF_nbf_mean_pf_bin2.dat	
5/17/00	11:15a	725,142	CSNF_nbf_mean_pf_bin3.dat	
5/17/00	11:16a	1,411,646	CSNF_nbf_mean_pf_bin4.dat	
5/17/00	11:16a	13,054	CSNF_nbf_mean_pf_bin5.dat	
1/11/00	02:21p	265,450	HLW_bf_high_pf_bin2.dat	Seepage DLL input files for thermal hydrology, for high level waste, in the backfill case
1/11/00	02:21p	1,856,578	HLW_bf_high_pf_bin3.dat	
1/11/00	02:20p	6,023,818	HLW_bf_high_pf_bin4.dat	
1/11/00	02:18p	3,656,068	HLW_bf_high_pf_bin5.dat	
1/11/00	02:32p	6,744,598	HLW_bf_low_pf_bin1.dat	
1/11/00	02:29p	4,673,647	HLW_bf_low_pf_bin2.dat	
5/18/00	12:51p	307,639	HLW_bf_mean_pf_bin1.dat	
5/18/00	01:26p	1,332,229	HLW_bf_mean_pf_bin2.dat	
5/18/00	01:27p	2,547,100	HLW_bf_mean_pf_bin3.dat	
5/18/00	01:28p	4,889,020	HLW_bf_mean_pf_bin4.dat	
5/18/00	01:28p	44,173	HLW_bf_mean_pf_bin5.dat	

Date	Time	File size	File name	Description
			Seepage files	
5/17/00	11:16a	59,958	HLW_nbf_high_pf_bin2.dat	Seepage DLL input files for thermal hydrology, for high level waste, in the no-backfill case
5/17/00	11:17a	405,342	HLW_nbf_high_pf_bin3.dat	
5/17/00	11:17a	1,339,158	HLW_nbf_high_pf_bin4.dat	
5/17/00	11:17a	797,630	HLW_nbf_high_pf_bin5.dat	
5/17/00	11:18a	1,539,566	HLW_nbf_low_pf_bin1.dat	
5/17/00	11:19a	1,061,998	HLW_nbf_low_pf_bin2.dat	Seepage DLL input files for thermal hydrology, for high level waste, in the no-backfill case
5/17/00	11:19a	17,318	HLW_nbf_mean_pf_bin1.dat	
5/17/00	11:19a	435,190	HLW_nbf_mean_pf_bin2.dat	
5/17/00	11:20a	725,142	HLW_nbf_mean_pf_bin3.dat	
5/17/00	11:20a	1,411,646	HLW_nbf_mean_pf_bin4.dat	
5/17/00	11:20a	13,054	HLW_nbf_mean_pf_bin5.dat	
5/17/00	10:38a	617	master_bf.in	Seepage DLL input files for thermal hydrology
5/17/00	10:39a	639	master_nbf.in	
1/17/00	12:49p	395	SeepFlowMean.dat	
1/17/00	12:52p	388	SeepFlowSD.dat	
1/17/00	12:54p	337	SeepFrac.dat	
5/19/00	01:30p	880	seep_debug.dat	Seepage DLL output files
5/4/00	05:41p	284,440	seep_output.dat	
			WAPDEG files	
5/26/00	09:56a	778,752	wapdeg.dll	the waste package degradation (WAPDEG) DLL
4/26/00	12:00p	271,872	GVP.dll	gaussian variance partitioning DLL, supporting WAPDEG
4/26/00	01:50p	388,608	MFD.dll	manufacturing defect DLL, supporting WAPDEG
4/6/00	02:06p	438,784	SCCD.dll	stress corrosion cracking DLL, supporting WAPDEG
5/25/00	12:14p	4,106	debug4.txt	WAPDEG run log
5/25/00	12:13p	41,999	WD4DLL.ina	WAPDEG model files
5/4/00	05:42p	782,594	WD4DLL.oua	
5/4/00	05:42p	30,430	WD4DLL.out	
5/25/00	12:03p	363	WD4DLL.wap	
5/25/00	12:13p	9,119	WDdA22SR00.cdf	
5/25/00	12:13p	9,119	WDdA22x0p5.cdf	
5/25/00	12:13p	9,119	WDdA22x2p5.cdf	
5/25/00	12:13p	9,119	WDdTi7Sr00.cdf	
5/12/00	11:13a	12,524	WDgA22SR00.cdf	
5/12/00	11:13a	12,524	WDgA22x0p5.cdf	
5/12/00	11:13a	12,524	WDgA22x2p5.cdf	
5/24/00	12:27p	12,526	WDgTi7SR00.cdf	
2/16/00	08:59a	304,964	WDHLW_high_bin2.ou	
5/17/00	12:40p	355,504	WDHLW_nbf_high_bin2.ou	
5/25/00	12:13p	9,119	WDiA22x2p5.cdf	
1/14/00	02:06p	1,436	WDKlinM.fil	
1/14/00	09:26p	1,439	WDKlinO.fil	
5/25/00	12:13p	13,458	WDKISCCM.fil	

Date	Time	File size	File name	Description
			Seepage files	
5/25/00	12:13p	13,458	WDKISCCO.fil	
11/24/99	03:55p	775	WDMFDND.cdf	
5/25/00	12:13p	4,732	WDMFDNDM.cdf	
5/25/00	12:13p	5,112	WDMFDNDO.cdf	
5/25/00	12:13p	10,329	WDMFDSizeM.cdf	
5/25/00	12:13p	10,329	WDMFDSizeO.cdf	
1/17/00	10:36a	1,463	WDndTi.cdf	
5/25/00	12:13p	9,119	WDndTi7SR00.cdf	
2/24/00	07:24p	411	WDRHcrit.fil	
5/25/00	12:13p	13,448	WDStressM.fil	
5/25/00	12:13p	13,448	WDStressO.fil	
			FEHMN files	
5/26/00	07:39a	4,714,496	fehmn_sr.dll	FEHMN DLL
1/5/00	12:39p	3,715	afm_pch1.dpdp	FEHMN input files
3/13/00	09:36a	77	bf2.txt	
3/13/00	09:36a	77	bf3.txt	
3/13/00	09:36a	77	ch1.txt	
3/13/00	09:36a	77	ch6.txt	
3/13/00	09:36a	77	chv.txt	
3/13/00	09:36a	77	chz.txt	
2/15/00	09:48p	291	fehmn.files	
3/3/00	10:40a	1,277	fehmn.gold	
3/3/00	10:40a	1,277	fehmn.gold.multi	
5/4/00	05:42p	2,528,722	fm_pchm1.chk	
6/1/00	10:46a	1,622	fm_pchm1.dat	
5/4/00	05:42p	3,908,630	fm_pchm1.fin	
1/4/00	10:20a	2,335,583	fm_pchm1.grid	
5/4/00	05:51p	96,157	fm_pchm1.his	
1/4/00	10:20a	29,301,805	fm_pchm1.stor	
5/4/00	05:51p	14,837	fm_pchm1.trc	
1/4/00	10:22a	3,349	pch1.rock	
3/13/00	09:36a	77	pp1.txt	
3/13/00	09:36a	77	pp2.txt	
3/13/00	09:36a	77	pp3.txt	
3/13/00	09:36a	77	pp4.txt	
3/13/00	09:37a	77	tsw4.txt	
3/13/00	09:37a	77	tsw5.txt	
3/13/00	09:37a	77	tsw6.txt	
3/13/00	09:37a	77	tsw7.txt	
3/13/00	09:37a	77	tsw8.txt	
3/13/00	09:37a	77	tsw9.txt	
10/11/99	06:05p	15,937,540	ff0100.ini	Flow field input files
10/11/99	06:06p	15,937,540	ff0200.ini	
10/11/99	08:55p	15,937,540	ff0300.ini	
10/14/99	09:40p	15,938,094	ff1100.ini	
10/14/99	09:43p	15,938,094	ff1200.ini	

Date	Time	File size	File name	Description
Seepage files				
10/14/99	09:43p	15,938,094	ff1300.ini	
10/14/99	09:45p	15,938,094	ff2100.ini	
10/14/99	09:46p	15,938,094	ff2200.ini	
10/14/99	09:47p	15,938,094	ff2300.ini	
1/4/00	10:19a	980,781	fm_pchm1.zone	Input zone files for the FEHMN DLL
3/3/00	11:12a	1,082,426	fm_pchm1.zone2	
3/3/00	11:12a	1,082,424	fm_pchm1.zone2.0100	
3/3/00	11:12a	1,082,426	fm_pchm1.zone2.0200	
3/3/00	11:12a	1,082,424	fm_pchm1.zone2.0300	
5/17/00	01:51p	2,011,054	ptrk.multrlz	FEHMN particle tracking files
5/26/00	04:15p	2,011,157	ptrk.multrlz.0100	
5/26/00	04:16p	2,011,151	ptrk.multrlz.0200	
5/26/00	04:18p	2,011,148	ptrk.multrlz.0300	
5/1/00	09:25a	79,304	UZ_Params_Multi.sr	input file containing the unsaturated zone transport parameters
5/4/00	08:24p	688	fehmn_real.bat	batch file
5/4/00	08:24p	2,236	fehmn_ts0.bat	
SZ files				
3/17/00	09:52a	455,680	szconv_sr.dll	saturated zone (SZ) convolution DLL
1/28/00	04:15p	7,454,385	SZ_01_01	Saturated zone breakthrough curves input files
1/28/00	04:15p	7,454,385	SZ_01_02	
1/28/00	04:15p	7,413,715	SZ_01_03	
1/28/00	04:15p	7,413,715	SZ_01_04	
1/28/00	04:16p	7,475,137	SZ_02_01	
1/28/00	04:16p	7,475,137	SZ_02_02	
1/28/00	04:16p	7,475,137	SZ_02_03	
1/28/00	04:16p	7,475,137	SZ_02_04	
4/12/00	10:31a	16,447,812	SZ_03_01	
4/12/00	10:40a	16,447,812	SZ_03_02	
4/12/00	10:45a	16,447,812	SZ_03_03	
4/12/00	10:49a	16,447,812	SZ_03_04	
3/8/00	03:31p	7,531,121	SZ_04_01	
3/8/00	03:32p	7,531,121	SZ_04_02	
3/8/00	03:36p	7,531,121	SZ_04_03	
3/8/00	03:36p	7,531,121	SZ_04_04	
1/28/00	04:20p	10,449,584	SZ_05_01	
1/28/00	04:20p	10,625,985	SZ_05_02	
1/28/00	04:20p	9,743,980	SZ_05_03	
1/28/00	04:20p	9,743,980	SZ_05_04	
1/28/00	04:21p	7,476,265	SZ_06_01	
1/28/00	04:21p	7,476,265	SZ_06_02	
1/28/00	04:21p	7,476,265	SZ_06_03	
1/28/00	04:21p	7,476,265	SZ_06_04	
4/12/00	12:44p	6,963,312	SZ_07_01	
4/12/00	12:44p	6,963,312	SZ_07_02	

Date	Time	File size	File name	Description
			Seepage files	
4/12/00	12:45p	6,963,312	SZ_07_03	
4/12/00	12:45p	6,963,312	SZ_07_04	
1/28/00	04:17p	19,909,441	SZ_08_01	
1/28/00	04:17p	20,076,777	SZ_08_02	
1/28/00	04:17p	20,076,777	SZ_08_03	
1/28/00	04:17p	20,076,777	SZ_08_04	
6/15/00	03:23p	382	sz_convolute2.dat	saturated zone convolution output file
			Other files	
2/3/00	04:46p	473,088	ashdll.dll	ashplume DLL
4/28/00	12:38p	262,201	soilexp.dll	soil removal factor DLL

ATTACHMENT IX
DLL OUTPUT FILES FOR MODEL VALIDATION

1. WAPDEG OUTPUT FILE: WD4DLL.ina

```
This file contains input passed to DLL from EXE

Each line contains an index and the input value associated with
that index

================================================

             1   2.00000000000000
             2   1.00000000000000
             3   25.0000000000000
             4  0.750000000000000
             5   1000.00000000000
             6  0.000000000000000E+000
             7  0.000000000000000E+000
             8  0.000000000000000E+000
             9  0.000000000000000E+000
            10   1000.00000000000
            11  4.000000000000000E-004
            12  0.000000000000000E+000
            13  0.000000000000000E+000
            14  0.000000000000000E+000
            15   2.00000000000000
            16   10.0000000000000
            17  0.750000000000000
            18   1000.00000000000
            19  0.000000000000000E+000
            20  0.000000000000000E+000
            21  0.000000000000000E+000
            22  0.000000000000000E+000
            23   1000.00000000000
            24  4.000000000000000E-004
            25  0.000000000000000E+000
            26  0.000000000000000E+000
            27  0.000000000000000E+000
            28   23460000.0000000
            29  0.500000000000000
            30  0.500000000000000
            31   1000.00000000000
            32   23460.0000000000
            33  0.000000000000000E+000
            34  0.000000000000000E+000
            35  0.000000000000000E+000
            36  -1.00000000000000
            37  -1.00000000000000
```

```
    38   3.00000000000000
    39   15.0000000000000
    40  0.750000000000000
    41   1000.00000000000
    42  0.000000000000000E+000
    43  0.000000000000000E+000
    44  0.000000000000000E+000
    45  0.000000000000000E+000
    46   1000.00000000000
    47  0.000000000000000E+000
    48  0.000000000000000E+000
    49  0.000000000000000E+000
    50  0.000000000000000E+000
    51   36070000.0000000
    52   1.00000000000000
    53   1000.00000000000
    54   72140.0000000000
    55  0.000000000000000E+000
    56  0.000000000000000E+000
    57  0.000000000000000E+000
    58  -1.00000000000000
    59   1000.00000000000
    60   1.00000000000000
    61  0.000000000000000E+000
    62  0.000000000000000E+000
    63  0.000000000000000E+000
    64   1.00000000000000
  65   400.000000000000
```

...............continues.............................

```
 1076  0.000000000000000E+000
  1077   1073741824.00000
  1078   300.000000000000
  1079   1000.00000000000
  1080   5.00000000000000
  1081   1000.00000000000
  1082   5000.00000000000
  1083   10000.0000000000
  1084   100000.000000000
  1085   1000000.00000000
  1086  0.000000000000000E+000
  1087   1.00000000000000
  1088   1.00000000000000
  1089   1000000.00000000
  1090   10.0000000000000
  1091  0.000000000000000E+000
```

```
1092  0.000000000000000E+000
1093  0.000000000000000E+000
1094  0.000000000000000E+000
1095  0.000000000000000E+000
1096  0.000000000000000E+000
1097  0.000000000000000E+000
1098  0.000000000000000E+000
1099  -1.00000000000000
1100  0.000000000000000E+000
```

1. SZ CONVOLUTE LOG FILE: SZCONVOLUTE.LOG

Beginning of realization: 1
Climate change at time = 625.000000000000
Climate change at time = 2125.00000000000

2. ASHPLUME LOG FILE: ASHPLUME.OUT

```
iseed=            98
 ASHPLUME version 1.4

*****************************************************************
******
*
*
*                        realization number                    1
*
*                          wind speed (cm/s)            601.2869
*
*                      wind direction (deg)             -90.0000
*
*              mean particle diameter (cm)                0.0100
*
*                              log- std dev               2.0000
*
*                            column ht (km)               3.1587
*
*                      event duration (s)             0.6843E+06
*
*                              ash mass (g)           0.2966E+14
*
```

```
*                      event power (W)          0.4081E+12
*
*                                 beta          0.0707
*
*              vent exit velocity (cm/s)        7406.6379
*
*             particle shape parameter          0.5000
*
*                   air density (g/cc)          0.1117E-02
*
*                air viscosity (g/cm-s)         0.1758E-03
*
*       eddy diff. constant (cm2/s5/2)          400.0000
*
*                     size cutoff (cm)          10.0000
*
*                  incorporation ratio          0.3000
*
*       fuel particle minimum log-diam          -4.0000
*
*        fuel particle median log-diam          -2.6990
*
*       fuel particle maximum log-diam          -1.3010
*
*        total fuel mass available (g)          0.6075E+08
*
*                eruption volume (km3)          0.0297
*
*                  ash density (g/cm3)          1.0000
*
*
*
*****************************************************************
******

       x(km)      y(km)     xash(g/cm^2)     xfuel(g/cm^2)

       0.000     -20.000     0.6034E+00      0.1753E-05
```

2. FEHMN OUTPUT FILE: FEHMN.OUT

Running fehmn_real.bat

Not using tty output

```
File purpose - Variable - Unit number - File name

  control       - iocntl -  1 - fehmn.files
  input         - inpt   - 11 - fm_pchm1.dat
  geometry      - incoor - 12 - fm_pchm1.grid
  zone          - inzone - 13 - fm_pchm1.zone
  output        - iout   - 14 - fm_pchm1.out
  initial state - iread  - 15 - ff0200.ini
  final state   - isave  - 16 - fm_pchm1.fin
  time history  - ishis  - 17 - fm_pchm1.his
  time his.(tr) - istrc  - 18 - fm_pchm1.trc
  contour plot  - iscon  -  0 - not using
  con plot (dp) - iscon1 -  0 - not using
  fe coef stor  - isstor - 21 - fm_pchm1.stor
  input check   - ischk  - 22 - fm_pchm1.chk
Value provided to subroutine user: not using

zone read from optional input file: fm_pchm1.zone2
mptr read from optional input file: ptrk.median
n0 =      95328

Mean inf  conceptual #1 input ysw/6/3/99


**** input title : coor **** incoor =  12 ****
**** input title : elem **** incoor =  12 ****
**** input title : stop **** incoor =  12 ****
**** input title : zone **** inzone =  13 ****
**** input title : stop **** inzone =  13 ****
**** input title : dpdp **** inpt =  11 ****
dpdp read from optional input file: afm_pch1.dpdp
**** input title : perm **** inpt =  11 ****
**** input title : rlp  **** inpt =  11 ****
**** input title : rock **** inpt =  11 ****
rock read from optional input file: pch1.rock
**** input title : flow **** inpt =  11 ****
**** input title : time **** inpt =  11 ****
**** input title : itfc **** inpt =  11 ****
**** input title : ctrl **** inpt =  11 ****
**** input title : iter **** inpt =  11 ****
**** input title : sol  **** inpt =  11 ****
**** input title : rflo **** inpt =  11 ****
**** input title : air  **** inpt =  11 ****
**** input title : node **** inpt =  11 ****
**** input title : zone **** inpt =  11 ****
```

```
zone read from optional input file: fm_pchm1.zone2
**** input title : mptr **** inpt =  11 ****
mptr read from optional input file: ptrk.median
**** input title : stop **** inpt =  11 ****
WARNING: ASSUMING NEW ASCII STOR FORMAT
Coefficient file read with format ascii
!!!!!!!!!!!!!!!!!!!!!!!!!
Coefficients read from *.stor file
!!!!!!!!!!!!!!!!!!!!!!!!!

storage for geometric coefficients    391475 in common(nr)    391475


time for reading input, forming coefficients  11.8



**** analysis of input data on file fm_pchm1.chk                    ****

volumes and fe coefficients checked

storage for fe coefficients 391466 allocated   391475

Running fehmn_ts0.bat
Calling arguments are 0200 0001
zone read from optional input file: fm_pchm1.zone2
mptr read from optional input file: ptrk.median
Arrays:dum_p,insnode,ptindex,& pcnsk deallocated
dum_p()   re_allocated with size:         1
insnode() re_allocated with size:         6
ptindex() re_allocated with size:       276
pcnsk()   re_allocated with size:       276
Computing particle probability vector
**** particle tracking ****
Processing Species:         1
Processing Species:         2
Processing Species:         3
Processing Species:         4
Processing Species:         5
Processing Species:         6
Processing Species:         7
Processing Species:         8
Processing Species:         9
Processing Species:        10
Processing Species:        11
Processing Species:        12
```

```
Processing Species:     13
Processing Species:     14
Processing Species:     15
Processing Species:     16
Processing Species:     17
Processing Species:     18
Processing Species:     19
Processing Species:     20
Processing Species:     21
Processing Species:     22
Processing Species:     23
Processing Species:     24
Processing Species:     25
Processing Species:     26

Time Step     1

          Timing Information
     Years          Days        Step Size (Days)
  0.273785E-12   0.100000E-09   0.100000E-09
Heat and Mass Solution Disabled


simulation ended: days  1.000E-10 timesteps    1


total N-R iterations =      0
total solver iterations =      0


total code time(timesteps) =  1469.031250


****-------------------------------------------------------****
**** This program for                                  ****
****  Finite Element Heat and Mass Transfer in porous media ****
****-------------------------------------------------------****
****               Version  : FEHM V2.10PC00-03-28 ****
****               End Date : 09/22/2000        ****
****                  Time : 11:15:20        ****
****-------------------------------------------------------****
**** particle tracking ****
Processing Species:     1
Processing Species:     2
Processing Species:     3
Processing Species:     4
```

```
Processing Species:        5
Processing Species:        6
Processing Species:        7
Processing Species:        8
Processing Species:        9
Processing Species:       10
Processing Species:       11
Processing Species:       12
Processing Species:       13
Processing Species:       14
Processing Species:       15
Processing Species:       16
Processing Species:       17
Processing Species:       18
Processing Species:       19
Processing Species:       20
Processing Species:       21
Processing Species:       22
Processing Species:       23
Processing Species:       24
Processing Species:       25
Processing Species:       26

Time Step      1

            Timing Information
     Years          Days        Step Size (Days)
  0.156250E+02   0.570703E+04   0.570703E+04
Heat and Mass Solution Disabled


simulation ended: days  5.707E+03 timesteps     1


total N-R iterations =       0
total solver iterations =      0


total code time(timesteps) =   1474.500000

.....continues......


****--------------------------------------------------****
**** This program for                                ****
```

```
****  Finite Element Heat and Mass Transfer in porous media ****
****-------------------------------------------------------****
****                Version  : FEHM V2.10PC00-03-28 ****
****                End Date : 09/22/2000        ****
****                  Time : 11:59:13         ****
****-------------------------------------------------------****
**** particle tracking ****
Processing Species:        1
Processing Species:        2
Processing Species:        3
Ptrk is  50% complete
Processing Species:        4
Processing Species:        5
Processing Species:        6
Processing Species:        7
Processing Species:        8
Ptrk is  50% complete
Processing Species:        9
Processing Species:       10
Processing Species:       11
Ptrk is  60% complete
Ptrk is  70% complete
Ptrk is  80% complete
Processing Species:       12
Ptrk is  50% complete
Processing Species:       13
Ptrk is  70% complete
Processing Species:       14
Processing Species:       15
Ptrk is 120% complete
Ptrk is 150% complete
Ptrk is 160% complete
Ptrk is 170% complete
Ptrk is 180% complete
Processing Species:       16
Ptrk is  80% complete
Processing Species:       17
Ptrk is  60% complete
Ptrk is  70% complete
Processing Species:       18
Ptrk is  40% complete
Ptrk is  60% complete
Ptrk is  70% complete
Processing Species:       19
Ptrk is  40% complete
Ptrk is  50% complete
```

Ptrk is 60% complete
Ptrk is 70% complete
Ptrk is 80% complete
Ptrk is 90% complete
Ptrk is 100% complete
Ptrk is 110% complete
Ptrk is 120% complete
Ptrk is 130% complete
Processing Species: 20
Ptrk is 20% complete
Ptrk is 30% complete
Ptrk is 40% complete
Ptrk is 50% complete
Ptrk is 60% complete
Ptrk is 70% complete
Ptrk is 80% complete
Processing Species: 21
Ptrk is 10% complete
Ptrk is 20% complete
Ptrk is 30% complete
Ptrk is 40% complete
Ptrk is 50% complete
Ptrk is 60% complete
Processing Species: 22
Ptrk is 30% complete
Ptrk is 40% complete
Processing Species: 23
Ptrk is 160% complete
Ptrk is 170% complete
Ptrk is 180% complete
Processing Species: 24
Ptrk is 40% complete
Ptrk is 50% complete
Ptrk is 60% complete
Ptrk is 70% complete
Ptrk is 80% complete
Ptrk is 90% complete
Ptrk is 100% complete
Ptrk is 110% complete
Ptrk is 120% complete
Ptrk is 130% complete
Ptrk is 140% complete
Ptrk is 150% complete
Ptrk is 160% complete
Ptrk is 170% complete
Ptrk is 180% complete

```
Processing Species:        25
Processing Species:        26

Time Step      1

            Timing Information
      Years            Days        Step Size (Days)
   0.100000E+07     0.365250E+09     0.292200E+07
Heat and Mass Solution Disabled


simulation ended: days  3.653E+08 timesteps     1


total N-R iterations =         0
total solver iterations =        0


total code time(timesteps) =   4118.703125


****--------------------------------------------------****
**** This program for                              ****
****  Finite Element Heat and Mass Transfer in porous media ****
****--------------------------------------------------****
****                Version  : FEHM V2.10PC00-03-28 ****
****                End Date : 09/22/2000          ****
****                  Time : 11:59:35          ****
****--------------------------------------------------****
```

INTENTIONALLY LEFT BLANK