PNNL-14852 Volume 1

Pacific Northwest National Laboratory
Operated by Battelle for the U.S. Department of Energy

User Instructions for the Systems Assessment Capability, Rev. 1, Computer Codes

Volume 1: Inventory, Release, and Transport Modules

P.W. Eslinger
T.B. Miley
D.W. Engel
W.E. Nichols
L.H. Gerhardstein
D.L. Strenge
C.A. Lo Presti
S.K. Wurstner

September 2004

Prepared for the U.S. Department of Energy
under Contract DE-AC06-76RL01830

Disclaimer

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC06-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062;
ph: (865) 576-8401
fax: (865) 576-5728
email: reports@adonis.osti.gov

Available to the public from the National Technical Information Service,
U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161
ph: (800) 553-6847
fax: (703) 605-6900
email: orders@ntis.fedworld.gov
online ordering: http://www.ntis.gov/ordering.htm

This document was printed on recycled paper.

PNNL-14852, Volume 1

User Instructions for the Systems Assessment Capability, Rev. 1, Computer Codes

Volume 1: Inventory, Release, and Transport Modules

P. W. Eslinger
T. B. Miley
D. W. Engel
W. E. Nichols
L. H. Gerhardstein
D.L. Strenge
C. A. Lo Presti
S. K. Wurstner

September 2004

Prepared for
the U.S. Department of Energy
under Contract DE-AC06-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99352

Contents

Figures

Tables

1.0 Introduction and Background

In 1999, the U.S. Department of Energy (DOE) initiated the development of an assessment tool that will enable users to model the movement of contaminants from all waste sites at Hanford through the vadose zone, groundwater and Columbia River and estimate the impact of contaminants on human health, ecology and the local cultures and economy. This tool was named the System Assessment Capability (SAC). This tool is an integrated system of computer models and databases used to assess the impact of waste remaining on the Hanford Site. The SAC will help decision makers and the public evaluate the cumulative effects of contamination from Hanford.

The design of the SAC resulted from extensive interactions with Hanford projects, regulators, Tribal Nations, and stakeholders. The approach taken in the assessment follows that advanced by regulatory agencies such as the U.S. Enviroinmental Protection Agency (EPA) in their guidance on uncertainty analyses (Firestone et al. 1997) and ecological risk assessments (EPA 1998), and DOE in its radioactive waste management manual (DOE 1999a) and implementation guide (DOE 1999b), which support DOE Order 435.1 on radioactive waste management. The SAC also was designed with the intent to use it to perform the next composite analysis, an assessment first performed to satisfy Defense Nuclear Facility Safety Board recommendation 94-2. The approach taken is also consistent with the methods, characteristics, and controls associated with acceptable analyses as described by the Columbia River Comprehensive Impact Assessment (CRCIA) team (DOE 1998).

1.1 Overview of the SAC Systems Code

The SAC Systems Code is a tool used to simulate the migration of contaminants (analytes) present on the Hanford Site and assess the potential impacts of the analytes, including dose to humans, socio-cultural impacts, economic impacts, and ecological impacts. The system of codes includes existing computer programs, new computer programs, electronic data libraries, and data formatting processors (or data translators). The relationships among code modules that make up the SAC Systems Code are illustrated in Figure 1.1.

Major modules appearing on the left side of the diagram perform inventory and transport calculations providing estimates of the concentrations of analytes in various media. Modules shown on the right perform calculations related to impacts from the contaminated media. Impacts include potential effects on humans, the ecology of the area, the economy of the region, the proximity of contaminants to social and cultural resources.

The general approach to handling uncertainty in SAC, Rev. 1, is a Monte Carlo approach. Conceptually, one generates a value for every stochastic parameter in the code (the entire sequence of modules from inventory through transport and impacts) and then executes the simulation, obtaining an output value, or result. This process is often called one *realization*. The entire process is then repeated, obtaining another result that is different from the first, but as equally likely to occur as the first result. After repeating this process a number of times, one has a set of equally likely consequences that represent the statistical distribution of all outcomes. Several specialized sampling techniques have been developed to reduce the number of realizations required in a Monte Carlo analysis to obtain a satisfactory description of the output distribution. One of the techniques, called Latin Hypercube Sampling (Iman and Conover 1982), has proven successful for mass transport applications in groundwater systems. The general Monte Carlo approach still applies, and the specific values of the input parameters are chosen from the same statistical distributions, but the sampling scheme spreads the values in such a way as to reduce sampling variability while also supporting a correlation structure between input variables.

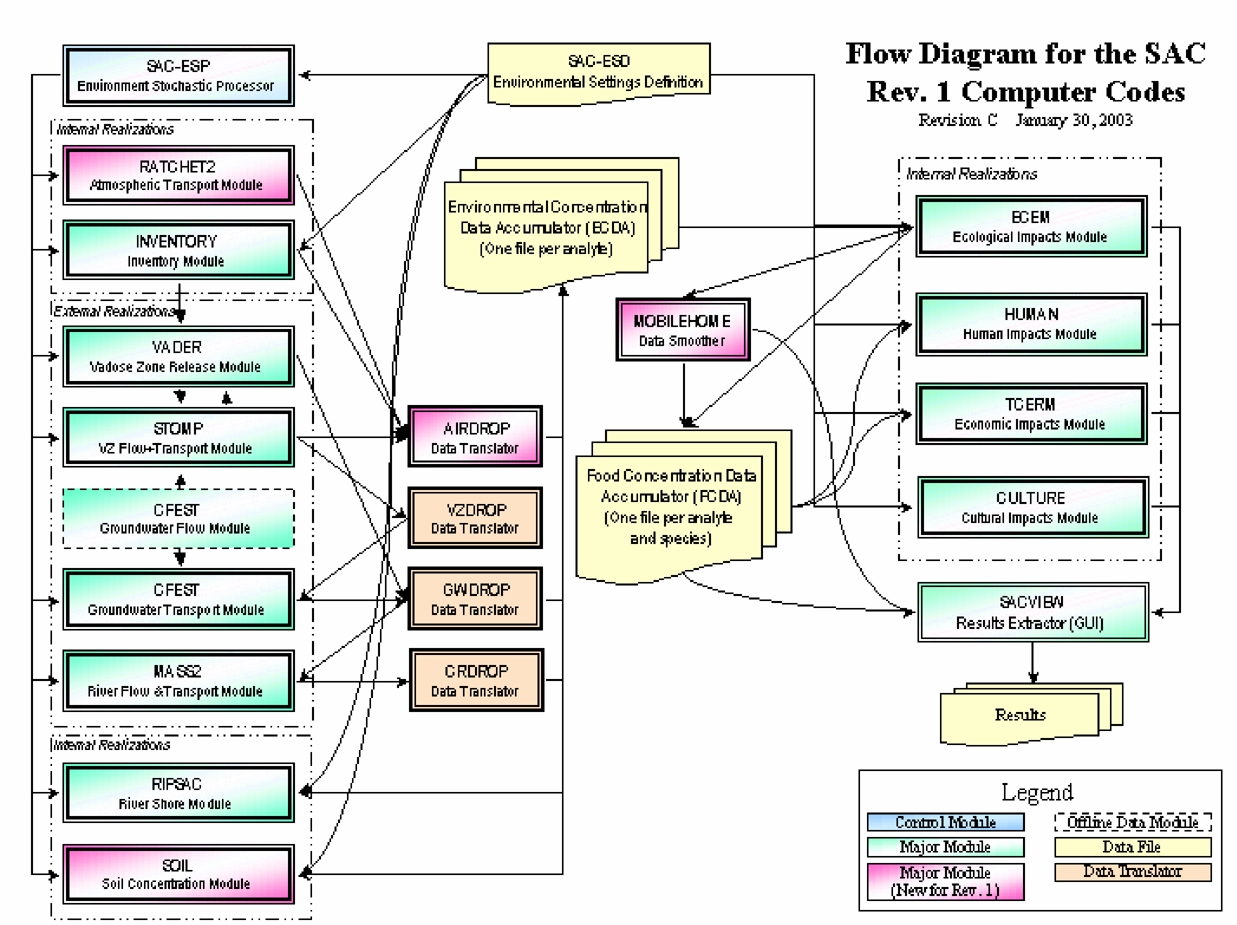

Figure 1.1 Module Information Flow for the SAC Rev. 1 Systems Code

1.2 Purpose of This Document

The SAC codes on the left side of Figure 1.1 address inventory tracking, release of contaminants to the environment, and transport of contaminants through the atmosphere, unsaturated zone, saturated zone, and the Columbia River. This document contains detailed user instructions for the computer codes described in Table 1.1. Instructions for some of the codes on the left side of Figure 1.1 and a number of utility codes are not provided in this document. The status of user instructions for these codes is shown in Table 1.2.

Table 1.1. Overview of Transport Codes

Name	Purpose
SAC ESP	Environmental stochastic processor. This processor controls the execution of all the codes on the left side of Figure 1.1.
INVENTORY	Inventory tracking and aggregation code.
VADER	Vadose zone release module. The function of this code includes release of contaminants from the waste form and tracking movement of waste during remediation activities.
VZDROP	Utility code to pass mass flux from the unsaturated zone transport code (STOMP) to the groundwater transport code (CFEST96)
GWDROP	Utility code to pass mass flux from the groundwater transport code (CFEST96) to the river transport code (MASS2). In addition, this code saves groundwater concentrations for use by the impacts codes.
CRDROP	Utility code to extract concentration data from the river module (MASS2) and save it for use by the impacts codes.
AIRDROP	Utility code to extract results from the atmospheric tranpsort module (RATCHET), scale them by inventory and vadose zone releases to the atmosphere, and save them for use by the impacts codes.
RIPSAC	Riparian zone module. The function of this code is to calculate concentrations of contaminants in seep water and soil near the edge of the river.
SOIL	Soil concentration module. The function of this code is to calculate soil concentrations in the upper soil layer for locations outside waste disposal areas.

Table 1.2. Description of Codes Not Provided in this Document

Name	Purpose
STOMP	A standalone user's guide is provided in White and Oostrom (2000). Section 3.4.1.7 documents the few code changes made to integrate STOMP into the SAC framework.
CFEST96	A user's guide for the CFEST96 code has been published separately (CFEST 2000).
MASS2	Theory and numerical methods for the MASS2 code are provided in Perkins and Richmond (2004a), and a user's guide is provided in Perkins and Richmond (2004b).
RATCHET	A user's guide is provided in Ramsdell, et. al (1994). Section 3.4.1.15 documents the changes made to integrate RATCHET into the SAC framework.

The suite of computer codes for Rev. 1 of SAC is much broader than just the inventory, release, and transport functions. The codes also address socio-cultural impact assessment, ecological impacts assessment, human impacts assessment, and regional economic impacts assessment. User instructions for the impacts codes are provided in Eslinger et al. (2004a). User instructions for a suite of utility codes are documented in Eslinger et al. (2004b).

The assumption is made that users of this document are knowledgeable computer users. In this case, the computer system runs under the Linux operating system. The user will have to create directories, create and edit files, copy files to subdirectories, and change directories in the system.

1.3 General Keyword Syntax

Many of the programs for SAC Rev. 1 are controlled through the use of data files containing text entries called keywords. In the keyword descriptions, some data are optional for a particular problem definition and some are required. Data that are required are enclosed in square brackets. For example, if AB were required, it would be denoted by [AB]. If only one of the three items AB, BC, CD were required, it would be written as [AB|BC|CD]. The vertical bars indicate that the user must choose one of the items in the list. Optional items are enclosed in normal brackets; for example, if DE were an optional entry, it would be denoted by {DE}. The {} or [] brackets do not need to be entered when the keyword is constructed. The keyword name can contain one or more characters; however, only the first eight characters are used (for example, REALIZAT has the same effect as REALIZATION). In some instances, numerical values or quote strings are associated with a modifier. In this description, the association is indicated by using the = symbol. The = symbol is not required but may be used when the keyword is constructed. When a numerical value or quote string is associated with a modifier, it must be entered on the input line directly after the modifier. Quote strings must be enclosed in double quotation marks. For example:

```
FILE C_ECDA   ANALYTE="U"   NAME="/home/CPO/ecda/U_CPO.dat"    CREATE
```

1.4 Stochastic Variable Generation

Many of the codes in SAC, Rev. 1, generate values for stochastic variables. All of the codes use the same suite of statistical routines to do this generation. The following are some major considerations for this process:

- Each distribution is generated using the Probability Integral Transformation method (Mood et al. 1974, p. 202).
- The uniform number generator uses a linear congruential method (Lewis et al. 1969).
- Stratified sampling is used when the number of values to be generated is greater than 1.
- Most distributions may be truncated between two limits that are specified as limits in the uniform domain on the interval 0 to 1.
- The user may specify a cumulative distribution function in the form of a table of values.
- Information about a stochastic variable is linked to a unique character ID. Access to all information about the variable is available through use of the variable ID.

The following statistical distributions are available:

- Constant value.
- Uniform distribution between two limits.
- Discrete uniform distribution on a set of contiguous integers.
- Loguniform (base 10) distribution between two limits.
- Loguniform (base e) distribution between two limits.
- Triangular distribution defined using a lower limit, mode, and an upper limit.
- Normal distribution with a mean and standard deviation.

- Lognormal (base 10) distribution specified by the mean and standard deviation of the logarithms of the data.
- Lognormal (base e) distribution specified by the mean and standard deviation of the logarithms of the data.
- User specified cumulative distribution function input as a table of probabilities and exceedance values.
- Beta distribution that can be shifted and scaled from the standard (0,1) interval.
- Log-ratio from a normal distribution.
- Hyperbolic arcsine from a normal distribution.

In the following discussion, the description is presented such that the keyword name for entering stochastic variable information is STOCHASTIC. In reality, a variety of keyword names are used, including KDSOIL and DILUTE, for example. The keyword STOCHASTIC will be used in the following discussion in order to simplify the presentation. This keyword facilitates entering the statistical distribution for stochastic variables. The general syntax for the STOCHASTIC record is the following:

```
STOCHASTIC ["Quote1"] [Dist_Index Parameters] {TRUNCATE U1 U2} {"Quote2"}
```

The entry for Quote_1 must be a unique character string of up to 20 characters that will be used to identify this stochastic variable in subsequent uses. It is case sensitive and embedded spaces are significant. It is sometimes useful to make the character string some combination of a variable name and other data such that it can be recreated easily when stochastic data are needed. The entry for Quote_2 is a description for the stochastic variable that can be up to 64 characters long that is used for output labeling purposes. The entry for Quote_2 is optional.

The entry for Index must be an integer in the range 1 to 13 that identifies the index of a statistical distribution. Table 1.3 defines the statistical distributions. The word Parameters in the general syntax statement indicates the numerical values of parameters required for defining the statistical distribution. The additional modifier TRUNCATE can be used for all distribution types except 1, 3, and 10. If TRUNCATE is entered, it must be followed by two values in the interval 0 to 1, inclusive of the endpoints. The lower value must be less than the upper value. These two values specify the tail probabilities at which to impose range truncation for the distribution. Truncation data must be entered after all of the other parameters that define the distribution.

Table 1.3. Statistical Distributions Available in All Codes

Index	Distribution	Truncate	Parameters Required
1	Constant	No	Single value.
2	Uniform	Yes	Lower limit, upper limit.
3	Discrete Uniform	No	Smallest integer, largest integer.
4	Loguniform (base 10)	Yes	Lower limit, upper limit.
5	Loguniform (base e)	Yes	Lower limit, upper limit.
6	Triangular	Yes	Lower limit, mode, upper limit.
7	Normal	Yes	Mean, standard deviation.

Index	Distribution	Truncate	Parameters Required
8	Lognormal (base 10)	Yes	Mean, standard deviation of logarithms.
9	Lognormal (base e)	Yes	Mean, standard deviation of logarithms.
10	User Defined	Yes	Number of pairs, data for pairs of values (Prob(X_i),X_i).
11	Beta	Yes	Alpha, beta, lower limit, upper limit. The mean of the distribution would be alpha/(alpha+beta) if the limits were 0 and 1.
12	Log ratio	Yes	Mean, Standard deviation (of normal), lower limit, upper limit.
13	Hyperbolic arcsine	Yes	Mean, Standard deviation (of normal).

The following is an example stochastic keyword for a variable assigned a constant of 234.432:

```
STOCHASTIC “Unique1” 1 234.432 “Define a constant distribution”
```

The constant can take any value.

The following is an example stochastic keyword for a variable assigned a uniform distribution on –2 to 7:

```
STOCHASTIC “Unique2” 2 –2.0 7
   “Define a uniform distribution on –2 to 7”
```

The two limits can take any values as long as the second value is strictly greater than the first value.

The following is an example stochastic keyword for a variable assigned a discrete uniform distribution on the integers 6 to 70:

```
STOCHASTIC “Unique3” 3 6 70
   “Define a discrete uniform distribution on 6 to 70”
```

The two limits must be integers where the second integer is strictly greater than the first integer.

The following is an example stochastic keyword for a variable assigned a loguniform (base 10) distribution on the interval 100.E-7 to 10.0E-3:

```
STOCHASTIC “Unique4” 4 1.0E-7 1.0E-3
   “Define a loguniform (base 10) distribution on 0.0000001 to 0.001”
```

The two limits must both be greater than zero and the second limit must be greater than the first limit.

The following is an example stochastic keyword for a variable assigned a loguniform (base e) distribution on the interval 10.0E+3 to 10.0E+6:

```
STOCHASTIC “Unique5” 5  1.0E3  1E+6
   “Define a loguniform (base e) distribution on 1000 to 1000000”
```

The two limits must both be greater than zero and the second limit must be greater than the first limit.

The following is an example stochastic keyword for a variable assigned a triangular distribution with a minimum of 2, a mode of 3, and a maximum of 7:

```
STOCHASTIC “Unique6” 6 2 3 7 “Define a triangular distribution on (2,3,6)”
```

The three values that define the triangular must all be different, and they must be entered in increasing order.

The following is an example stochastic keyword for a bioconcentration factor that is normally distributed with a mean of 125 and a standard deviation of 5 for a frog exposed to ^{14}C:

```
STOCHASTIC “BCFC14Frog” 7 125.0 5.0 “Example normally distributed frog”
```

The mean value can be any number, but the standard deviation must be greater than zero.

The following keyword would define a different stochastic variable than the one just entered because the identification string (Quote1) is case sensitive:

```
STOCHASTIC “BCFC14FROG” 7  125.0  5.0  “Example normally distributed frog”
```

The following keyword entry would define a lognormal (base 10) distribution where the mean and standard deviation (of the logarithms) are –2.0 and 0.5:

```
STOCHASTIC “Unique8”  8  –2  0.5
   “Example for a lognormal (base 10) variable”
```

The mean value can be any number, but the standard deviation must be greater than zero.

The following keyword entry would define a lognormal (base e) distribution where the mean and standard deviation (of the logarithms) are –2.0 and 0.5. In addition, the lognormal distribution will be truncated between the lower 0.025 and upper 0.99 probabilities.

```
STOCHASTIC “Unique9” 9 –2 .5 TRUNCATE 0.025 0.99
   “Example for a truncated lognormal variable”
```

The mean value can be any number, but the standard deviation must be greater than zero.

The following keyword entry illustrates the use of the user-defined distribution (distribution type 10). This example entry uses seven pairs of values. The first pair of numbers uses a probability of 0 to define the lower limit of the distribution at 8.4 E-7. The last pair of numbers uses a probability of 1 to define the upper limit of the distribution at 1.73E-6. The other values are associated with the probability levels of .025, .167, .5, .833, and .975. The probability data and distribution percentiles must be entered in strictly increasing order.

```
STOCHASTIC "Sr90Con" 10  7
   0  8.40E-7
   2.50E-02  9.20E-7
   1.67E-01  1.06E-6
   5.00E-01  1.21E-6
   8.33E-01  1.37E-6
   9.75E-01  1.58E-6
   1  1.73E-6
```

The first pair of numbers uses a probability of 0 to define the lower limit of the distribution. The last pair of numbers uses a probability of 1 to define the upper limit of the distribution. The intervening pairs define probability levels o and the associated data values. The probabilities and data values must be entered in strictly increasing order.

The following keyword entry would define a beta distribution with parameters 1.1 and 2.1 on the interval (0,1):

```
STOCHASTIC "Unique11-1"  11  1.1  2.1  0.0  1.0
   "Beta (1.1,2.1) on the interval 0,1"
```

Let the first parameter be denoted by α and the second parameter be denoted by β. The mean of the beta distribution would be $\alpha/(\alpha+\beta)$ if the limits were 0 and 1. Both α and β must be greater than zero. The lower limit must be less than the upper limit.

The following keyword entry would define a beta distribution with parameters 1.1 and 2.1 but on the interval –2 to 4:

```
STOCHASTIC "Unique11-2"  11  1.1  2.1  -2.0  4.0
   "Beta (1.1,2.1) on the interval (-2,4)"
```

The following keyword entry would define a log ratio distribution from a normal (-1.459,1.523) distribution on the interval -5.756 to 4.33.

```
STOCHASTIC "Test1203" 12 -1.459  1.523 -5.756 4.330
   "Log ratio from Normal(-1.4,1.5) on (-5.756,4.330)"
```

The entry for the normal standard deviation (a value of 1.523 in this example) must be greater than zero. The last two numerical values define the interval for the generated values, so the lower limit must be smaller than the upper limit.

The following keyword entry would define a hyperbolic arcsine distribution from a normal (0.189,0.146) distribution:

```
STOCHASTIC "Test1302" 13 0.189  0.146
   "Hyperbolic Arcsine from Normal(0.189,0.146)"
```

The entry for the normal standard deviation (a value of 0.189 in this example) must be greater than zero.

2.0 Environmental Settings Definition (ESD) Files

As seen in Figure 1.1, the SAC Rev. 1 systems code contains a number of component models that are executed independently. Some information, such as the start time and stop time of a simulated problem, are needed by more than one component of the systems code. The environmental settings definition (ESD) keyword file contains this common information. Generally, if information is needed by more than one module of the suite of codes, it will be entered in the ESD keyword file.

The river transport model can be run with or without background concentrations from upriver sources. A typical method of analyzing a Hanford-related problem is to run the river code with two data sets. First, a transport run is made using background values, but no source term from the Hanford Site is introduced. Then, another transport run is made that is identical to the first run except that a Hanford source term is introduced. If the impact codes are run using these two data sets, their results can be differenced to determine the contribution from Hanford sources. The groundwater model in SAC, Rev. 1, is not set up to model background concentrations; thus, differencing does not currently apply to impacts based on groundwater concentrations. A major effect of using background concentrations is that concentration data must be saved for both runs: the one where only background concentrations are modeled and the one where both background and Hanford concentrations are modeled. The current implementation uses two separate ESD keyword files to control these runs, and the concentration data are saved in totally separate files.

2.1 ESD Keywords

The ESD keyword file is read by a number of different programs. Data required by one program may not be used in another program; thus, specifications of data being optional or required are not made in this section.

2.1.1 ANALYTE Keyword in the ESD Keyword File

The ANALYTE keyword is used to define the analytes to be used in the simulation. The following is this keyword's syntax:

```
ANALYTE ID="quote 1" TYPE="quote 2" NAME="quote 3" AIR= "quote 4"
   HENRY=N2 DFIMM=N3 DFSED=N4} GAMMA=N5 HALFLIFE=N6 MOLWGT=N7
   SPECIFIC=N8 GASDIFF=N9 MOLDIFF=N1 COMPUTE
```

A separate ANALYTE keyword must be entered for every analyte to be included in the simulation. Table 2.1 describes the modifiers for the ANALYTE keyword.

Table 2.1 Modifiers for the ANALYTE Keyword in the ESD File

Modifier	Description
AIR	The quote string associated with the AIR modifier is a string up to 8 characters in length defining the analyte type for atmospheric transport and deposition. The following are the valid entries for this string: • NOBLE – if the analyte is to be treated as a noble gas in the atmospheric transport module (atmospheric transport only, no deposition) • PARTICLE – if the analyte is to be treated in the atmospheric transport module as one that deposits as a particle (atmospheric transport and deposition) • IODINE – if the analyte is iodine (atmospheric transport and deposition characteristics unique to iodine)
COMPUTE	The optional modifier COMPUTE can be entered. If COMPUTE is not present, the analyte will not be included in the run although information is included in the environmental settings file.
DFIMM	The numerical entry associated with the DFIMM modifier is the immersion dose factor for radioactive analytes. This value has units of mrad/yr per µCi/m^3. Entry of this modifier is optional. If it is not present, the value of DFIMM defaults to zero.
DFSED	The numerical entry associated with the DFSED modifier is the sediment external dose factor for radioactive analytes. This value has units of Sv-m^3/sec-Bq. Entry of this modifier is optional. If it is not present, the value of DFSED defaults to zero.
GAMMA	The numerical entry associated with the GAMMA modifier is the gamma decay energy for radioactive analytes. This value has units of MeV/nt. Entry of this modifier is optional. If it is not present, the value of GAMMA defaults to zero.
GASDIFF	The numerical entry associated with the GASDIFF modifier is the gas phase molecular diffusivity. This value has units of cm^2/sec. Entry of this modifier is optional. If it is not present, the value of GASDIFF defaults to zero.
HALFLIFE	The numerical entry associated with the HALFLIFE modifier is the half-life of the analyte. This value has units of years. Entry of this modifier is necessary when defining a radioactive analyte but should be omitted for nonradioactive analytes. If it is not present, the value of HALFLIFE defaults to infinity (decay constant value of zero).
HENRY	The numerical entry associated with the HENRY modifier is the Henry's law coefficient for organic analytes. This value has units of Pa-m^3/mole. Entry of this modifier is optional. If it is not present, the value of HENRY defaults to zero.

Modifier	Description
ID	The quote string associated with the ID modifier is an analyte identification string up to six characters in length. The analyte identification string is case sensitive, and spaces or hyphens change the definition. All data in the analyte identification strings must satisfy the following conventions: • Only the first entry in the analyte identification string is capitalized • No embedded spaces or hyphens are used, even for radionuclides • Individual elements are defined using the standard element abbreviation
MOLDIFF	The numerical entry associated with the MOLDIFF modifier is the molecular diffusivity of the analyte. This value has units of cm^2/sec and is used only in the ecological modules. Entry of this modifier is optional. If it is not present, the value of MOLDIFF defaults to zero.
MOLWGT	The numerical entry associated with the MOLWGT modifier is the molecular weight of the analyte. This value has units of g/mole.
NAME	The quote string associated with the NAME modifier is an analyte name or description up to 72 characters in length.
SPECIFIC	The numerical entry associated with the SPECIFIC modifier is the specific activity of the analyte. This value has units of curies per gram. Entry of this modifier is required if the analyte is radioactive but should be omitted for nonradioactive analytes.
TYPE	The quote string associated with the TYPE modifier string is a two-character analyte type indicator. The following are the valid entries for this string: • NR – if the analyte is a radioactive element or an inorganic compound containing a radionuclide • NS – if the analyte is a stable (nonradioactive) element or inorganic compound • OR – if the analyte is an organic compound containing a radionuclide • OS – if the analyte is an organic compound, containing a stable (nonradioactive) elemental analyte or compound

It is expected that common chemical formulae or acronyms would be used for the analyte identification string. The analyte identification string is case sensitive and all symbols, including spaces and hyphens, are significant. The analyte identification string is also used as a directory name, so naming conventions in Windows or Linux must also be considered when assigning the identification string. The following conventions apply:

- No embedded spaces or hyphens are used, even for radionuclides (for example, *Np237* should be used for the nuclide neptunium-237 rather than *Np 237* or *Np-237*).
- Individual elements should be defined using the standard abbreviation (for example, use *U* for uranium).

The following ANALYTE keywords select the analytes carbon tetrachloride, strontium-90 and carbon-14 for analysis.

```
ANALYTE ID="CCl4"   NAME="Carbon Tetrachloride" TYPE="NS"  COMPUTE
   HENRY  = 2.99E+03   MOLDIFF=5.0E-08    MOLWGT = 153.8  HALFLIFE = 0.0

ANALYTE ID="Sr90"  NAME="Strontium-90"  TYPE="NR"  COMPUTE
   MOLWGT=8.990774E+01  HALFLIFE=28.78  SPECIFIC=1.382778E+02
   DFIMM=1.703575E-03 DFSED=3.72E-21  GAMMA=0.0
   MOLDIFF=1.05E-05  GASDIFF=0.0 HENRY=0.0

ANALYTE ID="C14"  NAME="Carbon-14"  TYPE="NR" AIR="NOBLE"      COMPUTE
   MOLWGT=1.400324E+01  HALFLIFE=5.715E+03  SPECIFIC=4.470906E+00
   DFIMM=5.122392E-05 DFSED=7.200000E-23  GAMMA=0.0
   MOLDIFF=1.05E-05  GASDIFF=0.0 HENRY=0.0
```

2.1.2 BACKGROUND Keyword in the ESD Keyword File

The BACKGROUND keyword tells the main processor (ESP) that the background case will be modeled. The following is this keyword's syntax:

```
BACKGROUND COMPUTE
```

The background case will not be modeled unless the modifier COMPUTE is present on this keyword.

2.1.3 BALANCE Keyword in the ESD Keyword File

The BALANCE keyword identifies the years for which mass balance data will be generated. The following is this keyword's syntax:

BALANCE [ALL | [$year_1$] {$year_2$} {$year_3$} … {$year_n$}]

If the modifier ALL is present, then mass balance information is saved for all years defined in the PERIOD keyword record (Section 2.1.20) from the start year though the end year. One or more years must be listed explicitly if the modifier ALL is not present. If a year is listed twice, an error message is written.

The following is an example BALANCE keyword that saves mass balance information for the years 1950, 1960, 1970, 1980, 1990, and 2000:

```
BALANCE 1950 1960 1970 1980 1990 2000
```

To include a mass balance calculation for all years in the simulation, the following BALANCE keyword can be used:

```
BALANCE ALL
```

2.1.4 CREATDIR Keyword in the ESD Keyword File

The CREATEDIR keyword tells ESP to create the subdirectory structure for the analysis. The following is this keyword's syntax:

```
CREATEDIR COMPUTE
```

When this keyword is entered and the modifier COMPUTE is present, the directory structure for the analysis will be created, and then execution of ESP will terminate. If the modifier COMPUTE is not present, then no directory actions are taken. The directory structure is defined by the contents of the following ESD keywords: MODULE (see Section 2.1.18), ANALYTE (see Section 2.1.1), and SITE (see Section 2.1.27).

2.1.5 DEBUG Keyword in the ESD Keyword File

The DEBUG keyword initiates output of optional data. This keyword has two uses for two separate codes. The following is this keyword's syntax:

```
DEBUG STOCHASTIC="quote"
DEBUG ECDA
```

The quote string associated with the STOCHASTIC modifier identifies a file name (up to 200 characters in length) in which to store the stochastic values generated by the ESP for use in other modules or processes. For example, the groundwater recharge rate in the vadose zone is used by both VADER and STOMP. The generated values will be output to the file identified using the DEBUG keyword. For example, when storing the generated stochastic values in the file esp_stoch.out under the stoch subdirectory (relative to the assessment directory), the following keyword would be used:

```
DEBUG STOCHASTIC="stoch/esp_stoch.out"
```

If the ECDA modifier is present it initiates output of extra information to the report file when the ECDA code is executed. The keyword entry for that action is as follows:

```
DEBUG ECDA
```

With the exception of the ECDA code, the debug keyword in the ESD keyword file is only used by the ESP program. Debug options in other codes are activated by specific inputs for those codes contained in other files.

2.1.6 DILUTE Keyword in the ESD Keyword File

The DILUTE keyword is used to enter the definition of a statistical distribution for stochastic water dilution variables used in the riparian zone water mixing model. The following is this keyword's syntax:

```
DILUTE [ID="quote1"] [Dist_Index Parameters] {TRUNCATE U1 U2}
   {LABEL="quote2"}  [UNITS="quote3"]
```

The quote string associated with the ID modifier is a unique character string of up to 20 characters that will be used to identify this stochastic variable for subsequent uses. It is case sensitive and embedded spaces are significant. The quote string associated with the optional modifier LABEL contains a description for the stochastic variable that can be up to 64 characters long. An entry for quote2 is not required, although it is used for labeling purposes if present. However, if the modifier LABEL is present the associated quote string must be entered as well. The quote string associated with the UNITS modifier contains a units descriptor for the data. The strings “none” or “unitless” should be used if the variable is unitless. Section 1.4 contains further information about generating stochastic values.

More than one DILUTE keyword may be entered. The set of DILUTE keywords is used to generate a library of stochastic values that can be accessed by any SAC code. At this time only the RIPSAC code uses these values. The name of the library file is defined using the DILUTE modifier on the FILE keyword (Section 2.1.11).

A dilution factor that is triangular on the triple (0.2, 0.5, 0.99) could use the following keyword entry:

```
DILUTE ID=“ID#1” 6 0.2 0.5 0.99 LABEL=“Example dilution factor”
   UNITS=“none”
```

2.1.7 ECHO Keyword in the ESD Keyword File

The ECHO keyword is used to initiate output of summary information by the ECDA utility code when processing the statistical distributions defined by KDSOIL and DILUTE keywords. The following is this keyword’s syntax:

```
ECHO {KDSOIL} {DILUTE}
```

If the KDSOIL modifier is present then variable definitions and summary statistics on generated values will be written to the report file for every distribution specified on a KDSOIL keyword. If the DILUTE modifier is present then variable definitions and summary statistics on generated values will be written to the report file for every distribution specified on a DILUTE keyword. Example uses of this keyword are the following:

```
ECHO KDSOIL
ECHO DILUTE
ECHO DILUTE KDSOIL
```

2.1.8 END Keyword in the ESD Keyword File

The END keyword signifies the end of all keyword data. It should be the last keyword in the keyword file. Any data in the keyword file after the END keyword will be ignored. The following is this keyword’s syntax:

```
END
```

2.1.9 EVAPORATE Keyword in the ESD Keyword File

The EVAPORATE keyword defines the evaporation rates applied to liquid disposal sites. The following is this keyword's syntax:

```
EVAPORATE [SITE=“quote1”] [START=“quote2”] [END=“quote3”]
   [VOLUME=N1] [UNITS=“quote4”] {COMPUTE}
```

The modifiers for this keyword are described in Table 2.2.

Table 2.2 Modifiers for the EVAPORATE Keyword in the ESD File

Modifier	Description
SITE	The quote string associated with the SITE modifier contains the site ID for the specified site.
START	The quote string associated with the START modifier is the date (in format “mm/dd/yyyy”) when evaporation begins.
END	The quote string associated with the END modifier is the date (in format “mm/dd/yyyy”) when evaporation ends.
VOLUME	The numerical entry associated with the VOLUME modifier is the volume to be evaporated from the liquid disposal. The units of this value are those declared in the UNITS modifier.
UNITS	The quote string associated with the UNITS modifier contains a units descriptor for the data.
COMPUTE	The optional modifier COMPUTE can be entered. If COMPUTE is not present, the evaporation will not be included in the run.

Multiple EVAPORATE keywords can be entered. They can define different time periods for the same site or they can define data for different sites. The order of record dates must be arranged in ascending order for multiple keywords defining different time periods for a specific site. Example uses of this keyword are the following:

```
EVAPORATE SITE="216-B-3" START="01/01/1945" END="01/02/1945"
   VOLUME=14881.09 UNITS="m^3" compute
EVAPORATE SITE="216-B-3" START="01/02/1945" END="01/03/1945"
   VOLUME=21125.85 UNITS="m^3" compute
EVAPORATE SITE="216-B-3" START="01/03/1945" END="01/04/1945"
   VOLUME=36644.92 UNITS="m^3" compute
EVAPORATE SITE="216-B-3" START="01/04/1945" END="01/05/1945"
   VOLUME=51527.04 UNITS="m^3" compute
EVAPORATE SITE="216-B-3" START="01/05/1945" END="01/06/1945"
   VOLUME=66307.8 UNITS="m^3" compute
```

In the event that the declared quantity to evaporate exceeds the available water from infiltration and liquid disposals, the quantity will be truncated to match the available water.

2.1.10 EXEDIR Keyword in the ESD Keyword File

The EXEDIR keyword identifies the directory location of the program executable files. The following is this keyword's syntax:

```
EXEDIR ["quote"]
```

The EXEDIR keyword is used to identify the directory where executable programs reside that get called from the main processor ESP or from a script prepared by ESP. Example programs include the following:

ecda-1.exe for setup of the concentration data files for later use by the impacts codes
ripsac-1.exe for calculation of riparian zone concentrations
soil-1.exe for calculation of upper soil layer soil concentrations
back_mass2-1.exe for setup of background concentration data files for use in the river transport model
inventory-1.exe for calculation of the inventory by site and analyte.

These programs must reside under the specified directory, and the file names must be identical to those shown above (all in lowercase). The following is a sample keyword:

```
EXEDIR "/home/CODES/bin"
```

The location of other programs that are executed (run by a script generated by the ESP) is also identified by the EXEDIR keyword.

If the EXEDIR keyword is not entered or if the path is blank, then the executable programs must reside under a directory that is specified by the path environmental variable for the user who is running ESP. For instance, on a Unix system, the user can specify a path name in the user's definition file (.cshrc).

2.1.11 FILE Keyword in the ESD Keyword File

The FILE keyword is used to enter the names of many of the files used in a simulation run. The following is this keyword's syntax:

```
FILE [NAME="quote1"] {ANALYTE="quote2"} {SEED=N1}
   [ HEADER | KDSOIL | DILUTE | INFILT | C_ECDA | I_ECDA ] {CREATE}
```

The file names must be entered with complete path information. Every file definition requires the entry of a separate FILE keyword. One FILE keyword is required for every analyte for which concentrations are to be generated. The modifiers for this keyword are described in Table 2.3.

Table 2.3 Modifiers for the FILE Keyword in the ESD File

Modifier	Description
ANALYTE	If the file name is associated with the C_ECDA modifier, the ANALYTE keyword must also be entered. The quote string associated with the ANALYTE modifier contains the ID for the analyte associated with the ECDA concentration data file (see the ANALYTE keyword in Section 2.1.1).
CREATE	If this modifier is present the file will be created during a code execution. If the CREATE modifier is not present no file creation actions occur for the file.
C_ECDA	This modifier indicates that the FILE keyword is defining an ECDA file. Data for each analyte are contained in separate files. The ANALYTE modifier is used to associate an analyte with this file (see the ANALYTE keyword in Section 2.1.1).
DILUTE	This modifier indicates that the FILE keyword is defining a file to contain stochastic realizations of all of the random variables defined using the DILUTE keyword.
HEADER	This modifier indicates that the FILE keyword is defining a header file for use by the SACView graphical user interface that allows extraction of human-readable concentration data.
INFILT	This modifier indicates that the FILE keyword is defining a file to contain stochastic realizations of all of the random variables defined using the INFILTRATION keyword.
I_ECDA	This modifier indicates that the FILE keyword is defining a record index file for mapping into all ECDA concentration files.
KDSOIL	This modifier indicates that the FILE keyword is defining a file to contain stochastic realizations of all of the random variables defined using the KDSOIL keyword.
NAME	The quote string associated with the NAME modifier contains a file name (including path information) up to 200 characters in length.
SEED	If the file name is associated with the KDSOIL or DILUTE modifiers, the SEED modifier must also be entered. The numerical value associated with the SEED modifier is the value for the random number generator and must have an entry in the range 1 to 999999.

The following are example entries that define the concentration files for a suite of analytes:

```
FILE C_ECDA ANALYTE="H3" NAME="/test/ecda/H3_median.dat"  CREATE
FILE C_ECDA ANALYTE="C14" NAME="/test/ecda/C14_median.dat"  CREATE
FILE C_ECDA ANALYTE="I129" NAME="/test/ecda/I129_median.dat" CREATE
FILE C_ECDA ANALYTE="Tc99" NAME="/test/ecda/Tc99_median.dat" CREATE
```

If present, the CREATE flag causes the following actions by the ECDA code:

- Deletion of any existing file by that name and creation of a new file.
- If the file is associated with the HEADER, DILUTE, KDSOIL, C_ECDA or I_ECDA modifiers, new data are written to the file.

The following example entries define the files for soil-water Kd values, the water dilution factors for the river-shore module, and the groundwater infiltration rates for the soil module:

```
FILE KDSOIL NAME="KDSOIL.CSV" SEED=23232.0 CREATE
FILE DILUTE NAME="DILUTE.CSV" SEED=12345.0 CREATE
FILE INFILT NAME="infilt.dat"
```

The following example entries define the SACView Header file and ECDA record number index files:

```
FILE HEADER NAME="/test/ecda/Sacview_median.hdr" CREATE
FILE I_ECDA  NAME="/test/ecda/ECDA_median.map"  CREATE
```

2.1.12 FILLECDA Keyword in the ESD Keyword File

The FILLECDA keyword tells the ECDA code which values to use for initialization when an ECDA file is created. The following is this keyword's syntax:

```
FILLECDA [GWAT|SWAT|AIRC|AIRD|PWAT|SEDI|SORP|SEEP|SODR|SOGW|SOSW]
   [FIXED=N1|RANDOM]
```

The modifiers for this keyword are described in Table 2.4.

Table 2.4 Modifiers for the LOCATION Keyword in the ESD File

Modifier	Description
AIRC	Presence of this optional modifier indicates that atmospheric concentrations will be initialized with the specified value.
AIRD	Presence of this optional modifier indicates that atmospheric deposition rates will be initialized with the specified value.
FIXED	The value N_1 associated with the FIXED modifier is the value to be used to initialize the specified media in the ECDA file. Concentrations are typically initialized to zero, or an invalid (-1.0) value.
GWAT	Presence of this optional modifier indicates that groundwater concentrations will be initialized with the specified value.
PWAT	Presence of this optional modifier indicates that river bottom pore water concentrations will be initialized with the specified value.
RANDOM	The RANDOM option is used only for testing purposes. When RANDOM is selected the media concentrations are all filled with values from a uniform distribution on the (0,1) interval.
SEDI	Presence of this optional modifier indicates that sediment concentrations (on the river bottom) will be initialized with the specified value.
SEEP	Presence of this optional modifier indicates that seep water concentrations (on the land surface) will be initialized with the specified value.

Modifier	Description
SODR	Presence of this optional modifier indicates that upland soil concentrations from non-irrigated soils will be initialized with the specified value.
SOGW	Presence of this optional modifier indicates that upland soil concentrations using groundwater for irrigation will be initialized with the specified value.
SORP	Presence of this optional modifier indicates that riparian zone soil concentrations (on the land surface) will be initialized with the specified value.
SOSW	Presence of this optional modifier indicates that upland soil concentrations using surface water for irrigation will be initialized with the specified value.
SWAT	Presence of this optional modifier indicates that surface water concentrations will be initialized with the specified value.

The following example entries assign values to all 11 media for initialization of the ECDA files:

```
FILLECDA GWAT FIXED =  0.0
FILLECDA PWAT FIXED =  0.0
FILLECDA SWAT FIXED =  0.0
FILLECDA SEEP FIXED = -1.0
FILLECDA SEDI FIXED =  0.0
FILLECDA SORP FIXED = -1.0
FILLECDA SODR FIXED = -1.0
FILLECDA SOGW FIXED = -1.0
FILLECDA SOSW FIXED = -1.0
FILLECDA AIRC FIXED =  0.0
FILLECDA AIRD FIXED =  0.0
```

2.1.13 INFILTRATION Keyword in the ESD Keyword File

The INFILTRATION keyword defines the stochastic distribution for groundwater infiltration rates. If the vadose zone is being modeled, an infiltration rate or rates must be specified for each site (as specified by the SITE keyword) sufficient to cover all simulated time. Infiltration classes are also defined for impact locations with this keyword as well. The keyword may have one of two general syntaxes; the first syntax is used to define an “infiltration class” and an associated infiltration rate or stochastic distribution of rates. The second general syntax is used to apply an infiltration class to a specific vadose zone site.

The two syntaxes are distinguished by the presence of the modifier “DEFINE” or the modifier “SITE”. One, and only one, of these two modifiers must be used with the INFILTRATION keyword. There are two ways to define an infiltration class. The first way to define an infiltration class explicitly declares an infiltration class and rate using the following syntax:

```
INFILTRATION DEFINE [CLASS=“quote 1”]  [UNITS=“quote 2”]
   [Dist_Index Parameters]  {TRUNCATE U1 U2}  {“quote 3”}
```

The quote string associated with the CLASS modifier contains a user specified ID for the infiltration class. The data for this class will be referenced by the ID. The quote string associated with the UNITS modifier must contain the data units. Typically, units of mm/yr are used. The quote string identified as quote3 contains a descriptive label for this infiltration class and must be the third quote string in the keyword entry. The remaining entries for this syntax are used to define a statistical distribution for the infiltration rate. Additional information about these entreies is provided in Section 1.4. Some example uses of this syntax are the following:

```
INFILTRATION DEFINE UNIT="mm/yr" CLASS="Hanford" 1 0.1 "Hanford Barrier"
INFILTRATION DEFINE UNIT="mm/yr" CLASS="Hs-dn" 1 55
   "Hanford Sand (Hs), disturbed (d) - with no (n) vegation"
INFILTRATION DEFINE UNIT="mm/yr" CLASS="RCRA C" 1 0.1 "Modified RCRA C"
```

The second way to define an infiltration class specifies the rate as a function of two infiltration class rates already defined using the following syntax:

```
INFILTRATION DEFINE [CLASS="quote 1"]  [FRACTION=N1]
   [PRIOR="quote2"]  [NEXT="quote 3"]  "quote 4"
```

The quote string associated with the CLASS modifier contains a user specified ID for the infiltration class. The data for this class will be referenced by the ID. The quote string associated with the PRIOR modifier must contain the ID of an infiltration class already defined. The quote string associated with the PRIOR modifier must also contain the ID of an infiltration class already defined. The quote string identified as quote4 contains a descriptive label for this infiltration class and must be the fourth quote string in the keyword entry. The numerical value associated with the FRACTION modifier is used in computing the infiltration rate as a function of the PRIOR and NEXT classes as follows:

```
RATE = MIN(PRIOR,NEXT) + FRACTION×ABS(NEXT-PRIOR)
```

This alternative definition allows barrier degradation to be modeled in steps from full barrier effectiveness to a post-degradation long-term rate. For example, if a barrier period uses infiltration class S3 with a rate of 1.5 mm/yr, and the long-term post-barrier infiltration rate uses infiltration class S7 with a rate of 4.0 mm/yr , we can simulate the barrier degradation in four steps as follows:

```
INFILTRATION DEFINE CLASS="S3" UNITS="mm/yr" 1 1.5 "Barrier"
INFILTRATION DEFINE CLASS="S7" UNITS="mm/yr" 1 4.0 "Long-Term"
INFILTRATION DEFINE CLASS="S4" FRACTION=0.25 PRIOR="S3" NEXT="S7"
   "Degrade 1"
INFILTRATION DEFINE CLASS="S5" FRACTION=0.50 PRIOR="S3" NEXT="S7"
   "Degrade 2"
INFILTRATION DEFINE CLASS="S6" FRACTION=0.75 PRIOR="S3" NEXT="S7"
   "Degrade 3"
```

By way of example, the rate for dependent classes S4, S5, and S6 will be computed as follows:

```
S4 = MIN(S3,S7) + FRACTION×ABS(S7-S3) = 1.5 + 0.25×ABS(4.0-1.5) = 2.125
S5 = MIN(S3,S7) + FRACTION×ABS(S7-S3) = 1.5 + 0.50×ABS(4.0-1.5) = 2.750
S6 = MIN(S3,S7) + FRACTION×ABS(S7-S3) = 1.5 + 0.75×ABS(4.0-1.5) = 3.375
```

The second general syntax applies an infiltration class to a specific vadose zone site for a specific period of time:

```
INFILTRATION [SITE="quote 1"]  [CLASS="quote 2"] [START=Year1] [END=Year2]
   {ENHANCE=N1} {" quote 3"}
```

The quote string associated with the modifier SITE must match a site specified by the SITE keyword. The SITE keywords must be placed before the INFILTRATION keywords in the ESD keyword file. The numerical values associated with the START and END modifiers identify the time period (range of integer calendar years) for which this recharge rate will be used. The entire time period specified by the PERIOD keyword (Section 2.1.20) must be covered. The quote string associated with the CLASS modifier must contain the ID of an infiltration class specified using an INFILTRATION DEFINE keyword that precedes the INFILTRATION SITE keyword in the ESD keyword file. The quote string identified as quote3 contains a descriptive label for this infiltration class and must be the third quote string in the keyword entry. The optional ENHANCE modifier indicates the user wishes to apply a recharge rate in vadose zone transport model that differs from that applied in the vadose zone release model by a constant factor, N1. The ENHANCE factor is provided to allow treatment of barrier side-slopes that may serve to enhance recharge for the vadose zone as a whole, but not in the waste disposal area.

For example, one might define an infiltration data set for the “S” Hydrogeologic Province with an operational period cover (unvegetated gravel), which we identify as class “S2”. The INFILTRATION keyword might then be entered as follows:

```
INFILTATION DEFINE Class=“S1” Units=“mm/yr” 6 0.2 0.5 0.99 “Example class”
```

In this example, infiltration class S1 is declared and will have infiltration rates sampled from a triangular distribution with minimum 0.2 mm/yr, mode 0.5 mm/yr, and maximum 0.99 mm/yr. One might want to apply the infiltration class just declared to the vadose zone site 241-SX-104 for the period from 1955 through 2040. In this case, the INFILTRATION keyword will be enterd as follows:

```
INFILTRATION SITE “241-SX-104” CLASS=”S1” FROM 1955 TO 2040
```

2.1.14 IOONLY Keyword in the ESD Keyword File

The IOONLY keyword instructs the main processor ESP to create input files without running any analysis. The following is this keyword’s syntax:

```
IOONLY {COMPUTE}
```

No action is taken if the modifier COMPUTE is not present. The modules for which input files are to be created are specified using the MODULE keyword (see Section 2.1.18).

2.1.15 IRRIGATE Keyword in the ESD Keyword File

The IRRIGATE keyword defines the irrigation rates to be used by the SOIL model. The following is this keyword’s syntax:

```
IRRIGATE [SPRING=N1] [FALL=N2] [RATE=N3] [NET=N4] [START=N5]
   [THETAIRG=N6] [THETADRY=N7]
```

The modifiers for this keyword are defined in Table 2.5. The same irrigation period and water application rate applies to all locations in the model domain. No irrigation occurs before the start year. Once irrigation starts, it continues for every subsequent year in the simulation period.

Table 2.5 Modifiers for the IRRIGATE Keyword in the ESD File

Modifier	Description
FALL	The numerical entry associated with the FALL modifier is last day of the year (in julian days) that irrigation occurs.
NET	The numerical entry associated with the NET modifier is the net fraction of irrigation that percolates below 15 cm in the soil.
RATE	The numerical entry associated with the RATE modifier is the irrigation rate that is applied. This value has units of cm/yr.
SPRING	The numerical entry associated with the SPRING modifier is the day of the year (in julian days) that irrigation begins in the spring.
START	The numerical entry associated with the START modifier is the calendar year that irrigation begins. The assumption is that no irrigation occurs before that date, and that once started, irrigation continues every year after that through the end of the simulation.
THETADRY	The numerical entry associated with the THETADRY modifier is the surface soil volumetric water content during the dry (non-irrigated) period. This value has units of ml/cm^3.
THETAIRG	The numerical entry associated with the THETAIRG modifier is the surface soil volumetric water content during the irrigation period. This value has units of ml/cm^3.

The following example begins yearly irrigation in 2004 at a rate of 75 cm per year with 20% of the water infiltrating below 15 cm in soil depth.

```
IRRIGATE SPRING=121 FALL=256 RATE=75.0 NET=0.20 START=2004 THETAIRG=0.5
   THETADRY=0.2
```

2.1.16 KDSOIL Keyword in the ESD Keyword File

The KDSOIL keyword is used to enter statistical distribution definitions for solid-aqueous distribution coefficients (K_d) to be used for calculating soil and sediment concentrations from water concentrations. The following is this keyword’s syntax:

```
KDSOIL [ID=“quote1”] [Dist_Index Parameters] {TRUNCATE U1 U2}
   {LABEL=“quote2”}  [UNITS=“quote3”]
```

The quote string associated with the ID modifier is a unique character string of up to 20 characters that will be used to identify this stochastic variable for subsequent uses. It is case sensitive and embedded spaces are significant. The quote string associated with the optional modifier LABEL contains a description for the stochastic variable that can be up to 64 characters long. An entry for quote2 is not required, although it is used for labeling purposes if present. If the modifier LABEL is present the associated quote string must be entered as well. The quote string associated with the UNITS modifier contains a units descriptor for the data. The strings "none" or "unitless" should be used if the variable is unitless. Further information about defining statistical distributions is provided in Section 1.4.

More than one KDSOIL keyword may be entered. The set of KDSOIL keywords is used to generate a library of stochastic values that can be accessed by any SAC code. The name of the library file is defined using the KDSOIL modifier on the FILE keyword (Section 2.1.11).

A K_d that is triangular on the triple (0.2, 0.5, 0.99) would use the following keyword entry:

```
KDSOIL ID="ID#1" 6 0.2 0.5 0.99 LABEL="Example Kd" UNITS= "L/g"
```

The data entered by this keyword are used in the river model MASS2, the riparian zone model RIPSAC, and the soil concentration model SOIL.

2.1.17 LOCATION Keyword in the ESD Keyword File

The LOCATION keyword identifies the locations where concentrations will be generated for use in the impacts modules. The following is this keyword's syntax:

```
LOCATION [ID="quote1"] [EASTING=N1] [NORTHING=N2] {AREA=N3}
   {COMPUTE} {POP=N4} {GWAT} {SWAT} {AIRC} {AIRD} {PWAT} {SEDI} {SORP}
   {SEEP} {SODR} {SOGW} {SOSW} [NAME="quote2"] [TYPE="quote3"]
   [LOCUS="quote4"] {APSD=N5} {POROSITY= N6} {FOC=N7} {VEGCOV=N8}
   {NECF=N9} {RHOS=N10} {SRH=N11} {TEMP=N12} {MSWIND=N13} {MZWIND=N14}
   {COXYGEN=N15} {MILE=N16} {IRG_SWAT="quote5"}
```

The modifiers and associated data for this keyword are described in Table 2.6.

Table 2.6 Modifiers for the LOCATION Keyword in the ESD File

Modifier	Description
AIRC	Presence of this optional modifier indicates that atmospheric concentrations will be stored in the ECDA files for this location.
AIRD	Presence of this optional modifier indicates that atmospheric deposition rates will be stored in the ECDA files for this location.
APSD	The numerical value associated with the APSD modifier is the aggregate particle size. This value has units of mm and is used only in the ECEM code. It is valid only for land locations.
AREA	The numerical value associated with the AREA modifier identifies the land surface area associated with the location. This value has units of m^2.
COXYGEN	The numerical value associated with the COXYGEN modifier is the concentration of oxygen in surface water. This value has units of g/L and is used only in the ECEM code. It is valid only for aquatic locations.

Modifier	Description
EASTING	The numerical value associated with the EASTING modifier is the easting coordinate for this location. These coordinates are defined in terms of the Lambert projection of the Washington State Plane North American Datum of 1983, expressed in meters.
FOC	The numerical value associated with the FOC modifier is the fraction of organic carbon content in the soil. This value is unitless and is used only in the ECEM code. It is valid only for land locations.
GWAT	Presence of this optional modifier indicates that groundwater concentrations will be stored in the ECDA files for this location. This option can only be used for upland or aquatic locations.
ID	The quote string associated with the ID modifier contains the ID for this location. This string is limited to 15 characters and must be unique. It is used to associate other data with a specific location.
IRG_SWAT	The quote string associated with the modifier indicates the source of surface water for use in computing soil concentrations. The quote string must contain the ID of a location where a surface water solution is computed.
LOCUS	The quote string associated with the LOCUS modifier is used to limit the home range calculations for a mobile species to one side of the Columbia River. This modifier is used by the MOBILEHOME code. It has two valid values, “HANFORD” and “FARSHORE”.
MILE	The numerical value associated with the MILE modifier is the river mile for the location. This value has units of miles and is used only for riparian or aquatic locations.
MSWIND	The numerical value associated with the MSWIND modifier is the mean annual wind speed. This value has units of m/sec and is used only in the ECEM code. It is valid only for land locations.
MZWIND	The numerical value associated with the MZWIND modifier is the mixing zone wind speed. This value has units of m/sec and is used only in the ECEM code. It is valid only for land locations.
NAME	The quote string associated with the NAME modifier is a name or description up to 72 characters in length for this location. This value is used for labeling purposes only.
NECF	The numerical value associated with the NECF modifier is the non-erodible elements correlation factor. This value is unitless and is used only in the ECEM code. It is valid only for land locations.
NORTHING	The numerical value associated with the NORTHING modifier is the northing coordinate for this location. These coordinates are defined in terms of the Lambert projection of the Washington State Plane North American Datum of 1983, expressed in meters.

Modifier	Description
POROSITY	The numerical value associated with the POROSITY modifier is the soil porosity. This value is unitless and is used only in the ECEM code. It is valid only for land locations.
POP	The numerical values associated with the POP modifier is the human population at this location. This value is used only in the HUMAN code for calculating population dose.
PWAT	Presence of this optional modifier indicates that river bottom pore water concentrations will be stored in the ECDA files for this location. This option can only be used for aquatic locations.
RHOS	The numerical value associated with the RHOS modifier is the soil density. This value has units of gm/cm^3 and is used in the ECEM and SOIL codes. It is valid only for land locations.
SEDI	Presence of this optional modifier indicates that sediment concentrations (on the river bottom) will be stored in the ECDA files for this location. This option can only be used for aquatic locations.
SEEP	Presence of this optional modifier indicates that seep water concentrations (on the land surface) will be stored in the ECDA files for this location. This option can only be used for upland or aquatic locations.
SODR	Presence of this optional modifier indicates that soil concentrations without irrigation will be stored in the ECDA files for this location. This option can only be used for upland locations.
SOGW	Presence of this optional modifier indicates that soil concentrations calculated by using groundwater for irrigation will be stored in the ECDA files for this location. This option can only be used for upland locations.
SORP	Presence of this optional modifier indicates that riparian zone soil concentrations (on the land surface) will be stored in the ECDA files for this location. This option can only be used for riparian locations.
SOSW	Presence of this optional modifier indicates that soil concentrations calculated by using surface water for irrigation will be stored in the ECDA files for this location. This option can only be used for upland locations.
SRH	The numerical value associated with the SRH modifier is the surface roughness height. This value has units of m and is used only in the ECEM code. It is valid only for land locations.
SWAT	Presence of this optional modifier indicates that surface water concentrations will be stored in the ECDA files for this location. This option can only be used for aquatic locations.
TEMP	The numerical value associated with the TEMP modifier is the temperature. This value has units of Kelvin and is used only in the ECEM model. It is valid only for land locations.

Modifier	Description
TYPE	The quote string associated with the TYPE modifier is the location type. The following are the valid entries for this string: UPLAND, RIPARIAN, or AQUATIC.
VEGCOV	The numerical value associated with the VEGCOV modifier is the fraction of vegetation cover at the location. This value is unitless and is used only in the ECEM model. It is valid only for land locations.

The following example keywords define three upland locations:

```
LOCATION ID="UH0001" NAME="UnsuitableForAgricul" EASTING=576022
   NORTHING=154367 GWAT SOGW SOSW SODR AIRC AIRD, TYPE = "UPLAND",
   LOCUS = "HANFORD" APSD = 4.00E-02  POROSITY= 3.50E-01  FOC = 1.0E+00
   VEGCOV   = 5.00E-01  NECF    = 1.00E+00  RHOS   = 1.50E+00
   SRH  = 1.8E-02 TEMP     = 2.85E+02  MSWIND  = 3.44E+00
   MZWIND = 3.44E+00 IRG_SWAT ="QHP025"   AREA    = 1457376.1

LOCATION ID="UH0042" NAME="AgExclusionBuffer" EASTING=573022
   NORTHING=149867 GWAT SOGW SOSW SODR AIRC AIRD, TYPE = "UPLAND",
   LOCUS = "HANFORD" APSD = 4.00E-02  POROSITY= 3.50E-01  FOC \\= 1.0E+00
   VEGCOV   = 5.00E-01  NECF    = 1.00E+00  RHOS   = 1.50E+00
   SRH  = 1.8E-02 TEMP     = 2.85E+02  MSWIND  = 3.44E+00
   MZWIND = 3.44E+00 IRG_SWAT ="QHP025"   AREA    =  562500.0

LOCATION ID="UH0015" NAME="Upland" EASTING=576022 NORTHING=152117
   GWAT SOGW SOSW SODR AIRC AIRD, TYPE = "UPLAND", LOCUS = "HANFORD"
   APSD     = 4.00E-02  POROSITY= 3.50E-01  FOC    = 1.0E+00
   VEGCOV   = 5.00E-01  NECF    = 1.00E+00  RHOS   = 1.50E+00
   SRH  = 1.8E-02 TEMP     = 2.85E+02  MSWIND  = 3.44E+00
   MZWIND = 3.44E+00 IRG_SWAT ="QHP025"   AREA    =  562500.0
```

The following example keywords define three aquatic locations:

```
LOCATION ID="Q02I01" NAME="ISLAND 02,BLM" EASTING=574295.6
   NORTHING=155058.0 SWAT SEDI PWAT TYPE="AQUATIC"  LOCUS="HANFORD"
   MILE=375.49 COXYGEN = 1.10E-02 AREA = 2.5E+03

LOCATION ID="Q06I04" NAME="ISLAND,06,LOCKE ISLAND,UPLANDS-DOE,SHORE-BLM"
   EASTING=577962.1 NORTHING=154231.9 SWAT SEDI PWAT TYPE="AQUATIC"
   LOCUS="HANFORD"  MILE=373.09 COXYGEN = 1.10E-02 AREA = 2.5E+03

LOCATION ID="QTV008" NAME="Transect VERNITA-1 HRM 0.3" EASTING=559027
   NORTHING=146042 SWAT SEDI PWAT TYPE="AQUATIC"  LOCUS="FARSIDE"
   MILE=387.83 COXYGEN = 1.10E-02 AREA = 2.5E+03
```

The following example keywords define three riparian locations:

```
LOCATION ID="RFP021" NAME="RiparianGrantUpriver100BC" EASTING=564431.8
   NORTHING=145847.4 GWAT SEEP SORP AIRC AIRD TYPE="RIPARIAN"
   LOCUS="FARSIDE"  MILE=384.40 APSD    = 4.00E-02  COXYGEN = 1.10E-02
   POROSITY= 3.50E-01  FOC  = 1.0E+00   VEGCOV  = 5.00E-01
```

```
      NECF    = 1.00E+00  RHOS   = 1.50E+00  SRH   = 1.8E-02  TEMP = 2.85E+02
      MSWIND  = 3.44E+00 MZWIND  = 3.44E+00  AREA    = 9360

   LOCATION ID="RFP022" NAME="RiparianGrant100BC" EASTING=564841.7
      NORTHING=145818.1 GWAT SEEP SORP AIRC AIRD TYPE="RIPARIAN"
      LOCUS="FARSIDE"  MILE=384.10 APSD    = 4.00E-02  COXYGEN = 1.10E-02
      POROSITY= 3.50E-01  FOC  = 1.0E+00   VEGCOV  = 5.00E-01
      NECF    = 1.00E+00  RHOS   = 1.50E+00  SRH  = 1.8E-02  TEMP = 2.85E+02
      MSWIND  = 3.44E+00 MZWIND  = 3.44E+00  AREA    = 9360

   LOCATION ID="RFP031" NAME="RiparianGrantCoyoteRapids" EASTING=568198.5
      NORTHING=147140.3 GWAT SEEP SORP AIRC AIRD TYPE="RIPARIAN"
      LOCUS="FARSIDE"  MILE=381.70 APSD    = 4.00E-02  COXYGEN = 1.10E-02
      POROSITY= 3.50E-01  FOC  = 1.0E+00   VEGCOV  = 5.00E-01
      NECF    = 1.00E+00  RHOS  = 1.50E+00  SRH  = 1.8E-02  TEMP = 2.85E+02
      MSWIND  = 3.44E+00 MZWIND  = 3.44E+00  AREA    = 9360
```

2.1.18 MODULE Keyword in the ESD Keyword File

The MODULE keyword identifies the processes to be used in the current run of SAC_ESP. The following is this keyword's syntax:

```
   MODULE [ID="quote"] {FLOW|TRAN} {HIRES} {COMPUTE}
```

Multiple MODULE keywords are required to specify execution of more than one module. The module will be computed only if the modifier COMPUTE is present. The modifiers for this keyword are defined in Table 2.7.

Table 2.7 MODULE Options in the ESD File

Quote String	Description
AIRDROP	The quote string "AIRDROP" identifies the program that processes INVENTORY, STOMP, and RATCHET output and inserts air concentrations and deposition rates into the ECDA files.
CFEST	The quote string "CFEST" identifies the groundwater transport module. The modifier HIRES indicates that the high resolution groundwater transport model is to be run. Groundwater flow runs are performed outside the control of ESP.
CRDROP	The quote string "CRDROP" identifies the program that processes MASS2 outputs and inserts surface water, groundwater and sediment concentrations into the ECDA files.
ECDA	The quote string "ECDA" identifies the module that sets up the environmental concentration data files for later use by other modules.
GWDROP	The quote string "GWDROP" identifies the utility program to convert VADER and CFEST outputs into inputs for the river module. It also processes CFEST outputs to insert groundwater concentrations into the ECDA files.
INVENTORY	The quote string "INVENTORY" identifies the inventory module.
MASS2	The quote string "MASS2" identifies the river flow and transport module.

Quote String	Description
RATCHET	The quote string "RATCHET" identifies the atmospheric transport module.
RIPSAC	The quote string "RIPSAC" identifies the riparian zone concentration module. It computes riparian zone seep water, and wetted soil concentrations and inserts them into the ECDA files.
SOIL	The quote string "SOIL" identifies the upland soil concentration module. It computes soil concentrations under three irrigation assumptions (no irrigation, irrigation using groundwater, and irrigation using surface water) and inserts them into the ECDA files.
STOMP	The quote string "STOMP", when accompanied by the modifier FLOW, identifies a flow-only run for the vadose zone module. The optional modifier HIRES indicates that the high resolution groundwater model is to be run. The quote string "STOMP", when accompanied by the modifier TRAN, identifies a transport-only run for the vadose zone module. The optional modifier HIRES indicates that the high resolution groundwater model is to be run.
VADER	The quote string "VADER" identifies the vadose zone release module.

The following set of keywords executes all available modules:

```
MODULE ID="INVENTORY"        compute
MODULE ID="VADER"            compute
MODULE ID="STOMP" FLOW HIRES compute
MODULE ID="STOMP" TRAN HIRES compute
MODULE ID="CFEST" HIRES      compute
MODULE ID="GWDROP"           compute
MODULE ID="MASS2"            compute
MODULE ID="RIPSAC"           compute
MODULE ID="ECDA"             compute
MODULE ID="SOIL"             compute
MODULE ID="CRDROP"           compute
MODULE ID="RATCHET"          compute
MODULE ID="AIRDROP"          compute
```

2.1.19 OS Keyword in the ESD Keyword File

The OS keyword identifies the operating system under which the current run is being processed. The following is this keyword's syntax:

```
OS [“Unix” | “Windows”]
```

The default operating system is Unix; the other allowable operating system is Windows. To run on a Unix or Linux system, the keyword would appear as follows:

```
OS “Unix”
```

To run on a Windows system, the keyword would appear as follows:

```
OS “Windows”
```

2.1.20 PERIOD Keyword in the ESD Keyword File

The PERIOD keyword identifies the start and stop times for the entire simulation. The following is this keyword's syntax:

```
PERIOD [START=year1] [STOP=year2] [CLOSURE=year3]
```

The modifier START and the associated value year1 identify the calendar year the simulation starts. The start of the simulation period must be 1944 or later or the inventory code will error terminate. The modifier STOP and the associated value $year_2$ identify the calendar year the simulation ends. Start and stop years should be entered as whole numbers with the stop year no smaller than the start year. The modifier CLOSURE and the associated value $year_3$ identify the year that site closure occurs (and all inventory actions have been completed). The year of site closure cannot be smaller than the start year. The following is an example PERIOD keyword that simulates from 1944 through 3050 with site closure occurring at 2050:

```
PERIOD START=1944 STOP=3050 CLOSURE 2050
```

2.1.21 PROCESSOR Keyword in the ESD Keyword File

The PROCESSOR keyword identifies processors to be used for the current simulation. The following is this keyword's syntax:

```
PROCESSOR [MACHINE="quote 1"] [RUNDIR="quote 2"] {TYPE="quote 3"}
   {PRIORITY=N1} {COMPUTE}
```

The main processor ESP distributes the computational load of the environmental release, flow, and transport portion of the analysis to external processors. These processors can reside on the same computer or be external (networked) to the computer on which ESP is being run. The COMPUTE modifier must be present for the PROCESSOR keyword to have any effect.

The quote string associated with the MACHINE modifier must be a unique character string of up to 20 characters that will be used to identify the computational node. The names of the computational nodes are the names by which the computers are identified on the network. Dual-CPU computers should be identified with two identical PROCESSOR keywords in order to access both CPU's.

The quote string associated with the RUNDIR modifier identifies the base run directory for this particular analysis. The entry should include a complete directory path and can be up to 200 character in length.

The quote string associated with the optional TYPE modifier identifies the processor type for the remote machine. The entry should be "P3" for a Pentium 3 CPU and "P4" for a Pentium 4 CPU. This modifier allows ESP to select executable codes that are optimized for different types of processors. The type defaults to "P3" if this entry is not provided.

The numerical entry associated with the optional PRIORITY modifier identifies a priority for the ESP to use in assigning processors. Allowable values are 1, 2, and 3. When assigning jobs, the ESP will check all priority 1 CPU's for availability before checking priority 2 CPU's. Similarly, priority 3 CPU's will not have jobs assigned unless all priority 1 and 2 CPU's are busy. The priority defaults to 1 if this entry is not entered.

The SAC project runs a Linux cluster containing 128 Intel Pentium III 1.0-GHz processors, 2 Intel Pentium IV 2.0-GHz Processors and 13 Intel Pentium IV 2.7-GHz Processors. Each machine is a dual processor, so each machine can be declared twice. The following set of keywords defines a suite of processors to use for a simulation:

```
PROCESSOR MACHINE="c1-0"  PRIORITY=1 TYPE="P4" compute
   RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky"
PROCESSOR MACHINE="c1-0"  PRIORITY=1 TYPE="P4" compute
   RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky"
PROCESSOR MACHINE="c1-1"  PRIORITY=1 TYPE="P4" compute
   RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky"
PROCESSOR MACHINE="c2-0"  PRIORITY=2 TYPE="P4" compute
   RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky"
PROCESSOR MACHINE="c2-1"  PRIORITY=2 TYPE="P4" compute
   RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky"
PROCESSOR MACHINE="c0-0"  PRIORITY=3
   RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky" compute
PROCESSOR MACHINE="c0-1"  PRIORITY=3
   RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky" compute
```

2.1.22 REALIZAT Keyword in the ESD Keyword File

The REALIZAT keyword identifies the number of realizations to be simulated. The following is this keyword's syntax:

```
REALIZAT N1
```

The number of realizations is given by the single numerical entry N_1. The valid number of realizations is 1 to 9999. Run times and disk storage requirements are directly proportional to the number of realizations. The following REALIZAT keyword requests 25 realizations:

```
REALIZAT 25
```

2.1.23 REMEDIAT Keyword in the ESD File

The REMEDIAT keyword identifies remediation actions between waste sites. These actions are computed by VADER and STOMP so both codes use this keyword. The following is this keyword's syntax:

```
REMEDIAT [YEAR=N1] [FROM="quote1"] [TO="quote2"] {UPPER=N2} {LOWER=N3}
   {SOIL=N4} {CAKE=N5} {RIVER=N6} {CEMENT=N7} {CORE=N8} {LIQUID=N9}
   {DEBRIS=N10} {REACTOR=N11} {FORM="quote3"}
```

All information needed by STOMP or VADER exists on one keyword although each code may use different data from the keyword. The VADER code remediates waste in the VADER working site inventory and sends it to another site for use in STOMP. The STOMP code remediates soil from the vadose zone and sends it to another site for later use in VADER. This creates the possibility of transfers to sites of quantities from previous runs of both VADER and STOMP. Table 2.8 describes the modifiers for the REMEDIAT keyword for remedial action exports. One or more waste forms may be set for exports. The export fractions for unspecified waste forms are set to 0.

Table 2.8 Modifiers for the REMEDIAT Keyword in the ESD File

Modifier	Description
CAKE	The numerical value associated with the CAKE modifier identifies the fraction of saltcake/sludge waste (nominally found only in waste tanks) to be exported by VADER. Valid values are in the range 0 to 1. The SLUDGE and SALT modifiers can be used in place of the CAKE modifier.
CEMENT	The numerical value associated with the CEMENT modifier identifies the fraction of cement waste to be exported by VADER. Valid values are in the range 0 to 1. The CMNT modifier can be used in place of the CEMENT modifier.
CORE	The numerical value associated with the CORE modifier identifies the fraction of the remaining reactor/component waste to be exported by VADER. Valid values are in the range 0 to 1. The REACTOR modifier can be used in place of the CORE modifier.
DEBRIS	The numerical value associated with the DEBRIS modifier identifies the fraction of debris to be exported by VADER. Valid values are in the range 0 to 1.
LIQUID	The numerical value associated with the LIQUID modifier identifies the fraction of remaining liquid waste to be exported by VADER. Valid values are in the range 0 to 1.
LOWER	The numerical value associated with the LOWER modifier identifies the bottom of the column of soil to be excavated from the FROM site by STOMP and moved to the TO site. This value has units of m. STOMP uses a convention that the soil surface is location 0 and locations below the surface are larger than 0. This value must be larger that the value associated with the UPPER modifier.
FORM	(optional; default is SOIL DEBRIS if not specified) the waste form the remediated mass will be assigned by VADER at the TO site receiving the remediated mass.
FROM	The quote string associated with the FROM modifier identifies the release site from which waste or soil will be exported.
REACTOR	The numerical value associated with the REACTOR modifier identifies the fraction of reactor/component waste to be exported by VADER. Valid values are in the range 0 to 1. The CORE modifier can be used in place of the REACTOR modifier.
RIVER	The numerical value associated with the RIVER modifier identifies the fraction of remaining liquid waste to be exported by VADER. Valid values are in the range 0 to 1.
SOIL	The numerical value associated with the SOIL modifier identifies the fraction of soil-debris waste to be exported by VADER. Valid values are in the range 0 to 1.

Modifier	Description
TO	The quote string associated with the TO modifier identifies the site to which material will be exported. The FROM and TO sites must be different.
UPPER	The numerical value associated with the UPPER modifier identifies the top of the column of soil to be excavated from the FROM site by STOMP and moved to the TO site. This value has units of m. STOMP uses a convention that the soil surface is location 0 and locations below the surface are larger than 0. This value must be larger that the value associated with the UPPER modifier.
YEAR	The numerical value associated with the YEAR modifier identifies the year in which the remediation action takes place. A separate keyword must be entered for each year a remediation event occurs.

The following example REMEDIAT keyword causes VADER to export portions of the SOIL, CAKE, and CMNT wastes from the site MEMFIS to the site Kairo in the year 1960:

```
REMEDIAT YEAR=1960 FROM=”MEMFIS” TO=”Kairo” SOIL=.11 CAKE=.28 CMNT=.15
```

The following example keywords cause STOMP to excavate soil from 4 different sites send all the to site 600-148:

```
REMEDIATION YEAR=1991 FROM="618-9" TO="600-148" UPPER=0.0 LOWER=4.5
   SOIL=1.0 DEBRIS=1.0 CAKE=1.0 SLUDGE=1.0 GLASS=1.0 CEMENT=1.0 compute
REMEDIATION YEAR=1995 FROM="4843" TO="600-148" UPPER=0.0 LOWER=1 SOIL=1.0
   DEBRIS=1.0 CAKE=1.0 SLUDGE=1.0 GLASS=1.0 CEMENT=1.0 compute
REMEDIATION YEAR=1997 FROM="116-B-4" TO="600-148" UPPER=0.0 LOWER=4.5
   SOIL=1.0 DEBRIS=1.0 CAKE=1.0 SLUDGE=1.0 GLASS=1.0 CEMENT=1.0 compute
REMEDIATION YEAR=1997 FROM="116-B-5" TO="600-148" UPPER=0.0 LOWER=4.5
   SOIL=1.0 DEBRIS=1.0 CAKE=1.0 SLUDGE=1.0 GLASS=1.0 CEMENT=1.0 compute
```

2.1.24 REPORT Keyword in the ESD Keyword File

The REPORT keyword defines the file that will contain diagnostic messages and log the progress for the current run of ESP. The following is this keyword’s syntax:

```
REPORT [“quote”]
```

The REPORT keyword must be the first keyword in the file. There can be comment lines before this keyword, but if it is not the first keyword, then the ESP program will error terminate. Programs other than ESP do not use the REPORT keyword in the ESD keyword file. The name of the report file is entered in a quote string. File names up to 200 characters long are supported, and path information can be included. The following is an example REPORT keyword:

```
REPORT “/home/ANALYSIS/Initial2/ESD_Initial2.rpt”
```

2.1.25 RESITE Keyword in the ESD Keyword File

The RESITE keyword defines a subset of the sites defined by the SITE keyword (see Section 2.1.27) to be processed under the control of ESP. The following is this keyword's syntax:

```
RESITE ["site₁"] {"site₂"} … {"siteₙ"} {COMPUTE}
```

No action will be taken for the RESITE keyword unless the modifier COMPUTE is present. One or more sites can be identified on the RESITE keyword. The site IDs from the SITE keyword are the source of data for entry in the quote strings "$site_1$", "$site_2$", etc. The following RESITE keyword identifies four sites to be calculated in the run of ESP:

```
RESITE "216-B-14" "241-A-102" "UPR-300-R6-4" "US_Ecology" COMPUTE
```

2.1.26 RESTART Keyword in the ESD Keyword File

The RESTART keyword defines a subset of the realizations defined by the REALIZATION keyword (see Section 2.1.22) to be processed under the control of ESP. The following is this keyword's syntax:

```
RESTART [real₁] {real₂} … {realₙ} {COMPUTE }
```

This keyword has no effect unless the COMPUTE modifier is present. The following is an example keyword that identifies that realizations 1, 17, and 24 be computed:

```
RESTART 1 17 24
```

The realization numbers can be entered in any order.

2.1.27 SITE Keyword in the ESD Keyword File

The SITE keyword identifies the disposal sites where disposal actions will be modeled in SAC. The following is this keyword's syntax:

```
SITE [ID="quote1"] {COMPUTE} [TITLE= "quote2"] {OFFSITE}
   [POINTS=N1 (N2,N3) … (N4,N5)] {AREAX=N6}
   {AREAXMIN=N7} {UPSCALE=N8} {STACK=N9}
```

The modifiers and associated data for the SITE keyword are described in Table 2.9.

Table 2.9 Modifiers for the SITE Keyword in the ESD File

Modifier	Description
AREAX	The numerical values associated with the AREAX modifier is the multiplier (unitless) to apply to the vadose zone area for purposes of vadose zone transport simulations. A negative value will invoke the K_s-dependent area method (see below for details). This modifier is not used when the OFFSITE modifier is used.
AREAXMIN	The numerical value associated with the AREAXMIN modifier is the minimum multiplier (unitless) allowed when the K_s-dependent vadose zone modeling area method is used. This value is used only when the numerical value associated with the AREAX modifier is negative.
COMPUTE	If the COMPUTE modifier is entered, this aggregation site will be included in the simulation. If COMPUTE is not entered, this location will be ignored.
ID	The quote string associated with the ID modifer contains a location ID. This string is limited to 30 characters and must be unique. This ID is used to associate other data with a specific waste site. The ESP, inventory code and the pre- and post-processors for the inventory code can handle site IDs up to 30 characters in length. However, some of the other codes can only handle waste site IDs up to 15 characters in length. The net result is that site IDs for sites with the OFFSITE modifier can contain up to 30 characters while site IDs for sites within the modeling domain are limited to 15 characters.
OFFSITE	If the optional OFFSITE modifier is present then the site denotes a location outside the modeling domain. These sites are useful to account for waste that is exported from the modeling domain. Only the site ID and TITLE need to be defined if the OFFSITE modifer is present.
POINTS	The first numerical value (N1) associated with the POINTS modifier is the number of coordinate pairs needed to define the corners or ends of the waste disposal site. This value is followed by N1 coordinate pairs (Easting, Northing) defining the site vertices. Easting and northing coordinates are defined in terms of the Lambert projection of the Washington State Plane North American Datum of 1983, expressed in meters. This modifier is not used when the OFFSITE modifier is used.
STACK	The optional STACK modifier identifies that the waste site is a stack for atmospheric calculations. The associated numerical gives the stack height in meters. If stack height is greater than 20 m the site is modeled as a stack release in RATCHET, otherwise, it is modeled as a surface release. If the specified stack height is less than 20 m the STACK modifier has no effect.
TITLE	The quote string associated with the TITLE modifer contains a descriptive title for the waste site. This quote string is limited to 72 characters. It is used only for output labeling purposes.

Modifier	Description
UPSCALE	The numerical value associated with the UPSCALE modifier is a scaling factor (unitless, value of 1 or larger) that allows the ESP to try different input values if STOMP does not converge. This factor acts as a multiplier on the AREAX multiplier and has the effect of increasing the cross-sectional area of the transport column in STOMP in an effort to achieve a stable solution. If not specified, UPSCALE defaults to 1 (no upscaling).

The coordinates define the locations of the corners of the waste site and are usually taken from the Waste Information Database (WIDS). These coordinates are expressed in terms of the Lambert projection of the Washington State Plane North American Datum of 1983, expressed in meters. They are used to define an area for the vadose zone release (VADER) model and the vadose zone flow and transport (STOMP) model. For SAC, the vadose zone site is presumed to be represented by an area polygon with vertices defined by the coordinates specified.

The result files of INVENTORY (.res files) will include an entry for sites that have both COMPUTE and OFFSITE in the SITE keyword, but no release values will be written. There will be one line in the file stating, "Offsite location, no data provided for this location."

For the vadose zone transport calculations, an optional AREAX factor can be used to specify a multiplier for the vadose zone area. This is used to represent lateral spreading that would occur where large artificial liquid discharges happen, wherein the actual area of vadose zone transport is larger than the facility footprint given by WIDS coordinates. When ESP creates input files for STOMP runs, the area derived from the specified coordinates will be multiplied by AREAX; then the x- and y- extents written to the STOMP input file for one-dimensional cases will be the square of the resulting scaled area.

A negative value may be specified for AREAX to direct the code to compute the AREAX factor for each site and each realization using the K_s-dependent approach. For this approach, ESP will compute the effective wetted area of a vadose zone site using the equation

$$A_X = \frac{|Q_{\max}|}{K_{s|\min} A_0}, A_X \geq 1$$

where A_X = the multiplier of the facility footprint area (dimensionless)
Q_{max} = the maximum artificial liquid discharge rate (m/s)
$K_{s|min}$ = the minimum hydraulic conductivity of all layers for the given site (m/s) and realization
A_0 = the nominal area (m^2) defined by the coordinates in the SITE keyword.

AREAX is constrained by ESP to be equal to or greater than AREAXMIN. This may sometimes be smaller than the facility footprint area, but in order to accurately represent certain tank leaks this is appropriate. The horizontal extent of the vadose zone wetted area written in the STOMP input (STOMP ~Grid Card) file is

$$x = y = \sqrt{A_X A_0}$$

An example SITE keyword for specification of an offsite disposition to the Waste Isolation Pilot Plant (WIPP) to be included in the analyses is as follows:

```
SITE ID=“WIPP” TITLE=“Waste Isolation Pilot Plant” OFFSITE COMPUTE
```

Three example SITE keywords for specification of onsite disposals are the following:

```
SITE ID="100-B-15" POINTS=4 (565437.787,145477.787),
   (565482.213,145477.787), (565482.213,145522.213),
   (565437.787,145522.213)  TITLE="Radioactive Process Sewer"
   AreaX=2.0 COMPUTE
SITE ID="200-E-77" POINTS=4 (575273.468,135644.468),
   (575274.532,135644.468), (575274.532,135645.532),
   (575273.468,135645.532)  TITLE="Injection/Reverse Well"
   AreaX=10000.0 COMPUTE
SITE ID="241-A-101" POINTS=4 (575312.245,136033.167),
   (575332.505,136033.167), (575332.505,136053.427),
   (575312.245,136053.427)  TITLE="Single-Shell Tank"
   AreaX=-1.0 upScale=9.600 AreaXMin=0.01 compute
```

2.1.28 SPECIES Keyword in the ESD Keyword File

The SPECIES keyword is used to enter definitions for ecological species to be simulated. The ECEM code uses species from this master list and passes computed information on to the HUMAN, TCERM, and MOBILEHOME codes. The following is this keyword’s syntax:

```
SPECIES [ID=“quote 1”] [TYPE=“quote 2”] [NAME=“quote 3”]
   {Modifier=N1} … {Modifier=N34} {EMERGENT}
```

A separate SPECIES keyword must be entered for every species being defined. Table 2.10 describes the modifiers associated with the SPECIES keyword.

Table 2.10 Modifiers for the SPECIES Keyword in the ESD File

Modifier	Description
ID	The quote string associated with the ID modifier is a unique species identification string up to six characters in length.
TYPE	The quote string associated with the TYPE modifier string is a two-character analyte type indicator. The following are the valid entries for this string: TA – if the species is a terrestrial animal TP – if the species is a terrestrial plant QA – if the species is an aquatic animal QP – if the species is an aquatic plant
NAME	The quote string associated with the NAME modifier is a species name or description up to 72 characters in length.

Modifier	Description
AE	The numerical entry associated with the AE modifier is the assimilation efficiency of the species. This value is unitless. This modifier applies only to terrestrial plants and animals. If it is not present, the value of AE defaults to zero.
AWD	The numerical entry associated with the AWD modifier is the wet-to-dry weight ratio of the species. This value has units of g wet/g dry. This modifier applies only to aquatic plants and animals. If it is not present, the value of AWD defaults to zero.
DIFFHT	The numerical entry associated with the DIFFHT modifier is the diffusion height of the species. This value has units of meters. This modifier applies only to terrestrial plants and animals. If it is not present, the value of DIFFHT defaults to zero.
EMERGENT	There is no numerical entry associated with the EMERGENT modifier. Use of this modifier indicates that the species is an emergent plant. An emergent plant is one that has roots in water but grows in the air. This type of plant does not have rain splash of contaminated soil onto the plant's leaves.
ETWATER	The numerical entry associated with the ETWATER modifier is the exposure time to water of the species. This value has units of hr/day. This modifier applies only to terrestrial plants and animals. If it is not present, the value of ETWATER defaults to zero.
FLIPID	The numerical entry associated with the FLIPID modifier is the fraction of the mass of the species that is lipid. This value has units of g lipid/g wet. This modifier applies only to aquatic plants and animals. If it is not present, the value of FLIPID defaults to zero.
FMR	The numerical entry associated with the FMR modifier is the free metabolic rate of predator species. This value has units of kcal/day. This modifier applies only to terrestrial animals. If it is not present, the value of FMR defaults to zero.
FOC	The numerical entry associated with the FOC modifier is the fraction organic carbon of the species. This value has units of g organic carbon/g dry weight. This modifier applies only to aquatic plants and animals. If it is not present, the value of FOC defaults to zero.
FPA	The numerical entry associated with the FPA modifier is the volume fraction of plant tissue that is air for the species. This value is unitless. This modifier applies only to terrestrial plants. If it is not present, the value of FPA defaults to zero.
FPL	The numerical entry associated with the FPL modifier is the volume fraction of plant tissue that is lipid for the species. This value is unitless. This modifier applies only to terrestrial plants. If it is not present, the value of FPL defaults to zero.
FPW	The numerical entry associated with the FPW modifier is the volume fraction of plant tissue that is water for the species. This value is unitless. This modifier applies only to terrestrial plants. If it is not present, the value of FPW defaults to zero.

Modifier	Description
FW	The numerical entry associated with the FW modifier is the water weight fraction of plant tissue for the species. This value is unitless. This modifier applies only to terrestrial plants. If it is not present, the value of FW defaults to zero.
FWATER	The numerical entry associated with the FWATER modifier is the fraction exposure to water for the species. This value is unitless. This modifier applies only to terrestrial plants and animals. If it is not present, the value of FWATER defaults to zero.
FDW	The numerical entry associated with the FDW modifier is the conversion from dry weight to wet weight for the species. This value has units of kg dry/kg wet. This modifier applies only to terrestrial animals. If it is not present, the value of FDW defaults to zero.
GE	The numerical entry associated with the GE modifier is the gross energy for the species. This value has units of kcal/kg wet weight. This modifier applies only to terrestrial plants and animals. If it is not present, the value of GE defaults to zero.
INHRATE	The numerical entry associated with the INHRATE modifier is the resting inhalation rate for the species. This value has units of m^3/day. This modifier applies only to terrestrial animals. If it is not present, the value of INHRATE defaults to zero.
OCAR	The numerical entry associated with the OCAR modifier is the organic carbon assimilation rate for the species. This value has units of g organic carbon assimilated/g ingested. This modifier applies only to aquatic animals. If it is not present, the value of OCAR defaults to zero.
PCS	The numerical entry associated with the PCS modifier is the fraction of surface area in contact with soil for the species. This value has units of 1/day. This modifier applies only to terrestrial animals. If it is not present, the value of PCS defaults to zero.
PCW	The numerical entry associated with the PCW modifier is the fraction of surface area available to water soil for the species. This value is unitless. This modifier applies only to terrestrial animals. If it is not present, the value of PCW defaults to zero.
PSI	The numerical entry associated with the PSI modifier is the seasonality factor for the species. This value is unitless. This modifier applies only to terrestrial animals. If it is not present, the value of PSI defaults to zero.
RADIUS	The numerical entry associated with the RADIUS modifier is the radius of the species. This value has units of cm. This modifier applies to all ecological species. If it is not present, the value of RADIUS defaults to zero.
RHOP	The numerical entry associated with the RHOP modifier is the plant tissue density of the species. This value has units of kg/m^3. This modifier applies only to terrestrial plants. If it is not present, the value of RHOP defaults to zero.
SA	The numerical entry associated with the SA modifier is the surface area of the species. This value has units of cm^2. This modifier applies only to terrestrial animals. If it is not present, the value of SA defaults to zero.

Modifier	Description
SADHER	The numerical entry associated with the SADHER modifier is the skin adherence factor for the species. This value has units of mg/cm^2. This modifier applies only to terrestrial animals. If it is not present, the value of SADHER defaults to zero.
THETA	The numerical entry associated with the THETA modifier is the area use factor for the species. This value is unitless. This modifier applies only to terrestrial animals. If it is not present, the value of THETA defaults to zero.
WATERING	The numerical entry associated with the WATERING modifier is the water ingestion rate for the species. This value has units of liters/day. This modifier applies only to terrestrial animals. If it is not present, the value of WATERING defaults to zero.
WBMASS	The numerical entry associated with the WBMASS modifier is the wet body mass for the species. This value has units of grams. This modifier applies only to aquatic plants and animals. If it is not present, the value of WBMASS defaults to zero.
WEIGHT	The numerical entry associated with the WEIGHT modifier is the wet body weight for the species. This value has units of kg (wet). This modifier applies only to terrestrial plants and animals. If it is not present, the value of WEIGHT defaults to zero.

The following six examples illustrate the use of the SPECIES keyword for a variety of locations and species characteristics:

```
! Aquatic plant
SPECIES ID="QPERIP" TYPE="QP" NAME="periphyton" HABITAT="AQUATIC"
   AWD=10  FLIPID=0.0418   FOC=0.35  RADIUS=1.4   WBMASS=0.000035

! Aquatic animal
SPECIES ID="QCARPS" TYPE="QA" NAME="CARP  " HABITAT="AQUATIC"
   CAR=8.0000E-01   AWD=5.0000E+00 FLIPID=7.4000E-02 FOC=4.5000E-01
   BMASS=1.0500E+03 RADIUS=7.0000E+00

! Terrestrial animal - riparian location
SPECIES ID="RACOOT" TYPE="TA" NAME="American coot" HABITAT="RIPARIAN"
   AE=8.1000E-01   DIFFHT=2.0000E-01  ETWATER=2.0000E+01
   FMR=1.6309E+02  FWATER=5.0000E-01  FDW=0.4
   GE=1.9000E+03   INHRATE=6.1593E-01 PCS=2.5000E-01
   PCW=5.0000E-01     PSI=1.0000E+00      RADIUS=5.0000E+00
   SADHER=1.4500E+00  THETA=1.0000E+00   SA=7.8406E+02
   WATERING=4.6102E-02  WEIGHT=6.9200E-01

! Terrestrial animal - upland location
SPECIES ID="UBDGER" TYPE="TA" NAME="American badger" HABITAT="UPLAND"
   AE=8.1000E-01   DIFFHT=2.0000E-02  ETWATER=0.0000E+00
   FMR=8.2073E+02  FWATER=0.0000E+00  FDW=0.4
   GE=1.7000E+03   INHRATE=6.0032E+00 PCS=2.2000E-01
   PCW=0.0000E+00  PSI=1.0000E+00     RADIUS=1.0000E+01
   SADHER=1.4500E+00  THETA=1.0000E+00   SA=4.1972E+03
   WATERING=7.2063E-01  WEIGHT=9.0000E+00
```

```
! Terrestrial plant - riparian location
SPECIES ID="RCTWOD" TYPE="TP" NAME="black cottonwood" HABITAT="RIPARIAN"
   SURFACE=7.8050E+01 AE=3.4000E-01 DIFFHT=5.0000E+00 FPA=5.0000E-01
   FPL=1.0000E-02  FPW=4.0000E-01    FW=5.5000E-01 GE=3.2000E+03
   FWATER=0.0000E+00 RADIUS=3.0000E+01 RHOP=1.0000E+03
   WEIGHT=2.0000E+02 ETWATER=0.0000E+00

! Terrestrial plant - upland location
SPECIES ID="UFUNGI" TYPE="TP" NAME="fungi" HABITAT="UPLAND"
   SURFACE=1.0000E-03 AE=7.3000E-01 DIFFHT=5.0000E-02 FPA=5.0000E-01
   FPL=1.0000E-02  FPW=4.0000E-01    FW=9.1000E-01 GE=6.3000E+02
   FWATER=0.0000E+00 RADIUS=1.4000E+00 RHOP=1.0000E+03
   WEIGHT=3.0000E-03 ETWATER=0.0000E+00
```

2.1.29 TIMES Keyword in the ESD Keyword File

The TIMES keyword identifies the times at which concentration data will be saved in the ECDA files to support the calculations by the impacts codes. The following is this keyword's syntax:

TIMES [T_1] {T_2} … {T_n}

The numerical entries T_1, T_2, …, T_n are the times (whole number calendar years) when concentration data are to be saved. The following example TIMES keyword requests concentration data for the three years 2020, 2075, and 3014:

```
TIMES 2020 2075 3014
```

2.1.30 TITLE Keyword in the ESD Keyword File

The TITLE keyword is used to define a single-line problem title for the ESD run. The problem title will be written to output files. If the title is not supplied, the program will error terminate. The following is this keyword's syntax:

```
TITLE ["quote"]
```

The title is entered in a quote string, which must be enclosed in double quotes. Titles up to 200 characters long are supported. The following example defines a title for a run of the code:

```
TITLE "Example title line for the environmental settings control file."
```

2.1.31 USER Keyword in the ESD Keyword File

The USER keyword is used to identify the user of the ESP program. The user name will be written to output files. The following is this keyword's syntax:

```
USER [“quote”]
```

The user name is entered in a quote string. User names up to 16 characters long are supported. The following example defines John Q. Public as the user running the code:

```
USER “John Q. Public”
```

2.2 Environmental Concentration Data Accumulator

The purpose of the Environmental Concentration Data Accumulator is to provide a central storage for all concentrations of analytes at environmental locations and times needed to perform impact calculations.

2.2.1 Format of the ECDA Files

The file structure for the ECDA files has the following characteristics:

- Data storage for different analytes are provided in separate files (9 analytes would result in 9 concentration files). This design feature allows addition of an analyte to a data set without recalculation of the other analytes.
- The files have a binary, fixed record length, direct-access format.
- A mapping scheme is used to store only actual data (no placeholders with wasted storage space).

After the first 12 header lines, each data record in a ECDA file contains the following information: Year, Location ID, Media ID, media data (realizations 1 to the maximum). The relative order in which media data appear in the file for a given time and location is always the following:

- GWAT: concentrations in groundwater (Ci/m^3 or kg/m^3)
- SEEP: concentrations in seep water (Ci/m^3 or kg/m^3)
- SWAT: concentrations in surface water (river) (Ci/m^3 or kg/m^3)
- PWAT: concentrations in river bottom pore water (Ci/m^3 or kg/m^3)
- SEDI: concentrations in river bottom sediment (Ci/$kg_{sediment}$ or $kg_{analyte}/kg_{sediment}$)
- SORP: concentrations in riparian zone soil (land surface) (Ci/kg_{soil} or $kg_{analyte}/kg_{soil}$)
- SODR: concentrations in upland soil (land surface) with no irrigation (Ci/kg_{soil} or $kg_{analyte}/kg_{soil}$)
- SOGW: concentrations in upland soil (land surface) with groundwater irrigation (Ci/kg_{soil} or $kg_{analyte}/kg_{soil}$)
- SOSW: concentrations in upland soil (land surface) with surface water irrigation (Ci/kg_{soil} or $kg_{analyte}/kg_{soil}$)
- AIRC: concentrations in air (Ci/m^3 or kg/m^3)
- AIRD: air deposition rates (Ci/m^2/yr or kg/m^2/yr)

Table 2.11 provides an overview of the structure of a ECDA file assuming that 25 realizations were used.

Table 2.11 Structure of an ECDA File

Analyte ID		
11 lines, each containing the **units** for one medium as a character string		
Time 1	Location 1	Media L1-1, 25 realizations
		…
		Media L1-L1MAX, 25 realiz.
	Location 2	Media L2-1, 25 realizations
		…
		Media L2-L2MAX, 25 realiz.

	…	…
	Max Locations	Media LL-1, 25 realizations
		…
		Media LL-LLMAX, 25 realiz.
Time 2	Location 1	Media L1-1, 25 realizations
		…
		Media L1-L1MAX, 25 realiz.
	Location 2	Media L2-1, 25 realizations
		…
		Media L2-L2MAX, 25 realiz.
	…	…
	Max Locations	Media LL-1, 25 realizations
		…
		Media LL-LLMAX, 25 realiz.
…	…	…
Max Time	Location 1	Media L1-1, 25 realizations
		…
		Media L1-L1MAX, 25 realiz.
	Location 2	Media L2-1, 25 realizations
		…
		Media L2-L2MAX, 25 realiz.
	…	…
	Max Locations	Media LL-1, 25 realizations
		…
		Media LL-LLMAX, 25 realiz.

2.2.2 Format of the Record Map File for ECDA Data

The record map file is an ASCII file that provides indexing information for the storage locations in the binary ECDA files. One record map file is used for all ECDA files for a given simulation problem because the structure of the files are identical for all analytes. This file is generated by the ECDA code. The user should never modify this file.

The record map file has several header lines that are followed by indexing data. All character data are enclosed in double quotation marks, and records that contain more than one value have the data separated by commas. The definition of the header lines is as follows:

- **Problem Title.** The title from the ESD keyword file.
- **Code Name.** The name of the code that generated the file.
- **Code Version.** The version number of the code that generated the file.
- **Code Date:** The modification date of the code that generated the file.
- **User Name:** The user name from the ESD keyword file.
- **Run ID.** The run ID from the code run that generated the file.
- **Block Size.** The number of records with data in a time block in the binary data file.

- **Record Length.** Record length for the binary data file.
- **Number of Realizations.** The number of realizations requested for this run.
- **Number of Times.** The number of solution times requested for this run.
- **List of Times.** A list of solution times with one time per line.
- **Number of Locations.** The number of locations requested for this run.
- **List of Locations.** A list of locations IDs, with one ID per line.

The header data are followed by record map index data. There are as many lines of index data as there are locations. The index data for locations come in the same order as the location IDs provided earlier in the file. Each line of index data contains the location ID followed by as many indices as there are media types. The indices give an offset record number. If a particular media is not saved at a location, then the index value is set to negative 1. Table 2.12 provides excerpts from a record map file for a large data set.

Table 2.12 Excerpts Record Index Map File

```
"2004 Composite Analysis 10,000-year (Median Inputs) Assessment"
"ECDA"
"2.00.A.4"
"16 Oct 2003"
"Eslinger-DWE-WEN"
"20040802132252"
23155,"Records in a time block"
18,"Record length in the ECDA file"
1,"Number of realizations"
274,"Number of times"
1945
1950
<deleted lines>
12000
12050
4351,"Number of locations"
"UH0001"
"UH0002"
<deleted lines>
"QTV010"
"QHRWI1"
11,"Number of media"
"GWAT"
"SEEP"
"SWAT"
"PWAT"
"SEDI"
"SORP"
"SODR"
"SOGW"
"SOSW"
"AIRC"
"AIRD"
"UH0001",13,-1,-1,-1,-1,-1,14,15,16,17,18
"UH0002",19,-1,-1,-1,-1,-1,20,21,22,23,24
<deleted lines>
"QTV010",-1,-1,23162,23163,23164,-1,-1,-1,-1,-1,-1
"QHRWI1",-1,-1,23165,23166,23167,-1,-1,-1,-1,-1,-1
```

2.2.3 ECDA Header file for SACView

The SACView header file is an output file containing information used by the interactive SACView program to allow easy extraction of subsets of data stored in the ECDA files. These file is written by the ECDA code. The file contains the following sections of information:

- A header section of basic run information.
- Number of realizations selected.
- Media used.
- Times used.
- Locations used.
- Analytes used and analyte data file names.
- Record map file name.

Excerpts from a SACView header file is given in Table 2.13. The user only specifies the name of this file; users should not modify the file after it is generated by the ECDA code. See the instructions in (Eslinger et al. (2004a)) regarding the use of the SACView program. Because this file typically has at most a few thousand entries, the file size is always smaller than 200 Kb. File names for options not selected are set to the string "null".

Table 2.13 Example ECDA Header File for use in SACView

```
type: "ECDA"
title: "2004 Composite Analysis 10,000-year (Median Inputs) Assessment"
user: "Eslinger-DWE-WEN"
name: "ECDA"
version: "2.00.A.4"
date: "16 Oct 2003"
id: "20040802132252"
envfile: "ESD_CA1_median.key"
realizations: 1
media: 11
"GWAT","Groundwater concentrations"
"SEEP","Seep water (riparian zone) concentrations"
"SWAT","Surface water (river) concentrations"
"PWAT","Pore water (river bottom) concentrations"
"SEDI","Sediment (river bottom) concentrations"
"SORP","Soil (riparian zone) concentrations"
"SODR","Non-irrigated upland soil concentrations"
"SOGW","Groundwater irrigated upland soil concentrations"
"SOSW","Surface water irrigated upland soil concentrations"
"AIRC","Air concentrations"
"AIRD","Air deposition rates"
times: 274
 1945
 1950

<deleted lines>

 12000
```

```
 12050
locations: 4351
"UH0001","UnsuitableForAgricul"
"UH0002","UnsuitableForAgricul"

<deleted lines>

"QTV010","Transect VERNITA-1 HRM 0.3"
"QHRWI1","City of Richland Municipal Water Intake"
analytes: 15
"C14","NR","Carbon-14"
"/home/ANALYSIS4/CA1_median/ecda/C14_CA1_median.dat"
"Cs137","NR","Cesium-137"
"/home/ANALYSIS4/CA1_median/ecda/Cs137_CA1_median.dat"
"Cl36","NR","Chlorine-36"
"/home/ANALYSIS4/CA1_median/ecda/Cl36_CA1_median.dat"
"I129","NR","Iodine-129"
"/home/ANALYSIS4/CA1_median/ecda/I129_CA1_median.dat"
"Np237","NR","Neptunium-237"
"/home/ANALYSIS4/CA1_median/ecda/Np237_CA1_median.dat"
"Pa231","NR","Protactinium-231"
"/home/ANALYSIS4/CA1_median/ecda/Pa231_CA1_median.dat"
"Ra226","NR","Radium-226"
"/home/ANALYSIS4/CA1_median/ecda/Ra226_CA1_median.dat"
"Se79","NR","Selenium-79"
"/home/ANALYSIS4/CA1_median/ecda/Se79_CA1_median.dat"
"Sr90","NR","Strontium-90"
"/home/ANALYSIS4/CA1_median/ecda/Sr90_CA1_median.dat"
"Tc99","NR","Technetium-99"
"/home/ANALYSIS4/CA1_median/ecda/Tc99_CA1_median.dat"
"H3","NR","Tritium"
"/home/ANALYSIS4/CA1_median/ecda/H3_CA1_median.dat"
"Eu152","NR","Europium-152"
"/home/ANALYSIS4/CA1_median/ecda/Eu152_CA1_median.dat"
"U233","NR","Uranium-233"
"/home/ANALYSIS4/CA1_median/ecda/U233_CA1_median.dat"
"U235","NR","Uranium-235"
"/home/ANALYSIS4/CA1_median/ecda/U235_CA1_median.dat"
"U238","NR","Uranium-238"
"/home/ANALYSIS4/CA1_median/ecda/U238_CA1_median.dat"
recordmap: "/home/ANALYSIS4/CA1_median/ecda/ECDA_CA1_median.map"
```

2.3 Shared Environmental Stochastic Data

Some stochastic data are shared among several programs, including that defined by the KDSOIL (Section 2.1.16) and DILUTE (Section 2.1.6) keywords in the ESD keyword file. The following sections provide descriptions of the format of the data files containing the generated stochastic data.

2.3.1 Format of the DILUTE Data File

Entry of one or more DILUTE keywords in the ESD keyword file (see Section 2.1.6) causes the ECDA code to generate a data file of DILUTE values for use in other codes. The table below provides an example of this file for one DILUTE keyword and two realizations. The file is a text file that starts with nine header lines. The entry on line 8 is the number of DILUTE definitions – 1 in this example. The next line contains the number of realizations – 3 in this example. Each succeeding line contains data for a single DILUTE definition in the form of index, identification string, data units, and generated values for each realization. Multiple data on a single line are separated by commas, and all text data are enclosed in double quotation marks.

Table 2.14 Example DILUTE File

```
"DILUTE"
"2004 Composite Analysis 1,000-year (Stochastic Inputs) Assessment"
"ECDA"
"2.00.A.4"
"16 Oct 2003"
"20040617150724"
"Eslinger-DWE-WEN"
1
3
1,"DF5m","None", 6.01588E-01, 5.10553E-01, 5.12033E-01
```

2.3.2 Format of the KDSOIL Data File

Entry of one or more KDSOIL keywords in the ESD keyword file (see Section 2.1.16) causes the ECDA code to generate a data file of stochastic KDSOIL values for use in other codes. Table 2.15 provides an example of this file for 10 KDSOIL keywords and three realizations. The file is a text file that starts with 9 header lines. The entry on line 8 is the number of KDSOIL definitions – 10 in this example. The next line contains the number of realizations – 3 in this example. Each succeeding line contains data for a single KDSOIL definition in the form of index, identification string, data units, and generated values for each realization. Multiple data on a single line are separated by commas, and all text data are enclosed in double quotation marks.

Table 2.15 Example KDSOIL File

```
"KDSOIL"
"2004 Composite Analysis 1,000-year (Stochastic) Assessment"
"ECDA"
"2.00.A.4"
"16 Oct 2003"
"20040617150724"
"Eslinger-DWE-WEN"
10
3
1,"KDC","L/g", 8.51504E-02, 1.70702E-02, 5.09645E-02
2,"KDCl","L/g", 0.00000E+00, 0.00000E+00, 0.00000E+00
3,"KDCs","L/g", 6.42750E-01, 3.89112E+00, 2.62510E+00
4,"KDH","L/g", 0.00000E+00, 0.00000E+00, 0.00000E+00
```

```
5,"KDI","L/g", 1.37567E-05, 7.58985E-05, 6.64624E-05
6,"KDNp","L/g", 1.39180E-02, 2.26965E-02, 1.23003E-02
7,"KDSe","L/g", 5.29678E-03, 4.82762E-03, 5.18776E-03
8,"KDSr","L/g", 2.35305E-02, 2.12109E-02, 2.08005E-02
9,"KDTc","L/g", 6.34344E-06, 0.00000E+00, 2.57418E-05
10,"KDU","L/g", 8.38553E-04, 3.77616E-04, 9.35109E-04
```

2.3.3 Format of the INFILT Data File

Entry of one or more INFILTRATION keywords with the DEFINE modifier in the ESD keyword file (see Section 2.1.13) causes the ESP code to generate a data file of stochastic infiltration values for use in other codes. Table 2.15 provides an example of this file for 11 INFILTRATION keywords and three realizations. The file is a text file that starts with 9 header lines. The entry on line 8 is the number of INFILTRATION definitions – 11 in this example. The next line contains the number of realizations – 3 in this example. Each succeeding line contains data for a single INFILTRATION definition in the form of index, identification string, data units, and generated values for each realization. Multiple data on a single line are separated by commas, and all text data are enclosed in double quotation marks.

Table 2.16 Example INFILT File

```
"INFILT"
"2004 Composite Analysis 1,000-year (Stochastic Inputs) Assessment"
"ESP-Unix"
"1.5"
"21 Jul 2004"
"20040809075542"
"Eslinger-DWE-WEN"
11
3
1,"River","mm/yr", 1.00000E+00, 1.00000E+00, 1.00000E+00
2,"Eb-s","mm/yr", 9.53286E-01, 2.35171E+00, 2.12358E+00
3,"El-s","mm/yr", 1.61586E+00, 1.94937E+00, 2.24883E+00
4,"Ba-s","mm/yr", 3.84200E+00, 3.99799E+00, 3.55278E+00
5,"Rpe-s","mm/yr", 9.72041E-01, 1.31674E+00, 1.05688E+00
6,"Rp-s","mm/yr", 6.07450E+00, 2.90303E+00, 3.05800E+00
7,"Eb-ds","mm/yr", 5.34899E+00, 5.12679E+00, 3.06532E+00
8,"Eb-dg","mm/yr", 8.38464E+00, 1.12903E+01, 9.14490E+00
9,"Eb-dn","mm/yr", 1.68621E+01, 1.96773E+01, 1.75756E+01
10,"El-ds","mm/yr", 6.43085E+00, 4.41790E+00, 3.39158E+00
11,"El-dg","mm/yr", 8.03304E+00, 1.19927E+01, 8.45288E+00
```

3.0 ESP – Environmental Stochastic Preprocessor

3.1 Code Purpose

All of the environmental codes shown on the left side of Figure 1.1 are run as deterministic models except INVENTORY, RIPSAC, and SOIL. This means they require a single set of inputs and produce a single set of outputs for every realization. To model this system in a stochastic framework, ESP was developed to prepare inputs for each code, create a set of input files for each realization, and repeatedly execute the codes.

The INVENTORY, RIPSAC and SOIL codes were developed as standalone stochastic codes which generate all of their own input variables. The INVENTORY module simulates all of the waste to be released and typically will be executed outside of the ESP framework before the ESP is run.

The ESP has been developed to run under a Linux system (Linux Red Hat 7.3). Many of the other codes shown in Figure 1.1 have been developed to run under either a Windows environment or a Unix environment. However, the ESP has been optimized for the Linux operating system under and, thus, was not developed to run under both systems.

The ESP is, by design, extremely flexible in how the vadose zone simulations are handled. The complexity of simulating the vadose zone at hundreds of individual sites in a SAC assessment made it necessary to make the ESP code able to handle everything from the rerun of a single VADER or STOMP simulation to a simulation of all sites for all realizations and all analytes. Consequently, the description of the control functions provided here is not exhaustive; rather, it provides some basic guidelines to managing calculations in SAC and illustrates typical approaches.

3.2 Algorithms and Assumptions

ESP serves three basic purposes: simulate the stochastic input variables for the deterministic codes, create input files for each code, and run the codes in the proper sequence. The simulations are done using the statistical routines described in Section 1.4.

During an analysis (which will be referred to as an assessment throughout this section), ESP executes the different modules for all combinations of waste sites, analytes, and realizations. A directory structure is developed for the assessment to keep all this information in a logical order. The user will create a master directory (e.g., /Disk1/Projects/MasterRun, which is referred to as …/assessment in this document). ESP is then used to create subdirectories under the main directory, as shown below:

```
.../assessment/cfest/analyte/realization
.../assessment/inventory
.../assessment/mass2/analyte/realization
.../assessment/processors
.../assessment/ripsac
.../assessment/vadose/site/analyte/realization
.../assessment/ratchet/realization/analyte
```

The boldface names in the directory path refer to subdirectories created by the ESP that are fixed names (the user cannot modify these names). The non-boldface names refer to subdirectories whose names are based on entries in the ESD keyword file (see Section 2.1). The variable subdirectories (site, analyte, and realization) represent a potentially large number of lower-level subdirectories. For example, if the assessment has two sites (216-H-8 and 600-148), two analytes (H3 and C14) and two realizations, then the vadose subdirectory structure will contain the following entries:

```
.../assessment/vadose/216-H-8/H3/1
.../assessment/vadose/216-H-8/H3/2
.../assessment/vadose/216-H-8/C14/1
.../assessment/vadose/216-H-8/C14/2
.../assessment/vadose/600-148/H3/1
.../assessment/vadose/600-148/H3/2
.../assessment/vadose/600-148/C14/1
.../assessment/vadose/600-148/C14/2
```

The number of characters that comprise the realization subdirectory name is a function of the total number of realizations. The number is equal to the number of places in the total number of realizations, with leading zeros inserted as needed. For instance, if there were 150 realizations, then the subdirectory for 600-148, H3, and the second realization would look like the following:

```
.../assessment/vadose/600-148/H3/002
```

ESP will execute the modules VADER, STOMP, RATCHET, CFEST, and MASS2 once for every lower-level subdirectory. In the example above, both VADER and STOMP will be executed eight times ({number of sites} x {number of analytes} x {number of realizations}), and RATCHET, CFEST and MASS2 will be executed four times ({number of analytes} x {number of realizations}).

ESP will execute INVENTORY only once. The INVENTORY module can also be executed in a standalone mode external to control by ESP. The RIPSAC and SOIL modules only require one execution. They can be controlled by the ESP or executed in a standalone mode. The ecda subdirectory will contain the ECDA database files. The ECDA module gets called once by ESP to create this database.

ESP was developed to use distributed processing. Each execution of a module (VADER or STOMP, for example) will be run on an external process. ESP will distribute the runs based on priority – completion of other modules – and the list of processors described in the PROCESSORS keyword in the ESD keyword file (see Section 2.1.21).

3.2.1 ESP Major Functions

ESP is the central control processor of the environmental modules. As such, ESP is the main program for scheduling models of inventory, release, and transport in the atmosphere, the groundwater and the Columbia River. The following is a typical sequence for performing a Hanford plus background analysis:

1. Create the ESD keyword file.

2. Create the …/assessment/ecda directory and run the ECDA module to set up the ECDA, KDSOIL, and DILUTE files.

3. Create the directory structure (run ESP with the CREATEDIR keyword enabled, and all of MODULE, ANALYTE, and SITE keywords enabled). These keywords are enabled by entering a COMPUTE modifier on the keyword line.

4. Run the INVENTORY module (external) and ensure that the inventory result files (inv1.res, inv2.res, …) are located in the inventory subdirectory (…/assessment/inventory).

5. Create the input template and stochastic definition files for all the modules and copy them to the appropriate subdirectory. Most of the input files have a specified name. In the following list of required files, the files that have a variable name appear in italics:

   - …/assessment/inventory/inv###.dat – realization-specific INVENTORY result files (filename is based on total number of realizations, e.g., inv5.dat could be realization 5 of 9, inv05.dat could be realization 5 of 25, and inv005.dat could be realization 5 of 150)
   - …/assessment/vadose/stochastic_stomp.key – STOMP stochastic definitions
   - …/assessment/vadose/site/stochastic_vader.key – site specific VADER stochastic definitions
   - …/assessment/vadose/site/template_stomp.key – site specific STOMP template input file
   - …/assessment/vadose/site/template_vader.key – site specific VADER template input file
   - …/assessment/ratchet/stochastic.key – RATCHET stochastic definitions
   - …/assessment/ratchet/metdata/metYYYY.dat – RATCHET metadata definitions
   - …/assessment/ratchet/metdata/def_mix.file – RATCHET mixing layer height definitions
   - …/assessment/ratchet/metdata/z0_10cm.file – RATCHET surface roughness definitions
   - …/assessment/ratchet/metdata/ratchet.ctl – RATCHET control file
   - …/assessment/cfest/stochastic.key – CFEST stochastic definitions
   - …/assessment/cfest/analyte/cfest.key – analyte specific CFEST template input file
   - …/assessment/cfest/analyte/cfest.l3i – analyte specific CFEST template input file
   - …/assessment/cfest/analyte/cfest.lp1 – analyte specific CFEST template input file
   - …/assessment/cfest/analyte/input.hbc – analyte specific CFEST template input file
   - …/assessment/cfest/analyte/ZTOP.DAT – analyte specific CFEST template input file
   - …/assessment/cfest/analyte/bincf – subdirectory that needs to contain the flow field calculated from a separate run of CFEST
   - …/assessment/cfest/analyte/cf_tmpbinary – subdirectory that needs to contain the flow field calculated from a separate run of CFEST
   - …/assessment/mass2/biota.stoch – MASS2 stochastic definitions for biotic transport
   - …/assessment/mass2/mass2.key – MASS2 stochastic definitions (Kds)
   - …/assessment/mass2/index.key – input definition file for CRDROP_INDEX (create cross index for MASS2 grid to the ECDA locations)
   - *…/assessment/mass2/CRDROP_grid.dat* – MASS2 grid file (specified in the index.key file)
   - …/assessment/mass2/gwdrop.key – GWDROP template input file
   - *…/assessment/mass2/col-river-elem.dat* – GWDROP river elements (specified in gwdrop.key)

- *…/assessment/mass2/col-river-node.dat* – GWDROP river nodes (specified in gwdrop.key)
- *…/assessment/mass2/hanfnad83m-pt-000* – GWDROP river cells (specified in gwdrop.key)
- …/assessment/ripsac/ripsac.key – RIPSAC control and stochastic definitions
- …/assessment/ripsac/soil.key – SOIL control and stochastic definitions

6. Create the following realization-specific input files (run ESP with CREATEDIR keyword without COMPUTE, IOONLY keyword enabled, and all of the MODULE, ANALYTE, and SITE keywords enabled):

   - …/assessment/vadose/site/analyte/realization/vader.key
   - …/assessment/vadose/site/analyte/realization/input-esp
   - …/assessment/ratchet/realization/analyte/airdrop.ctl
   - …/assessment/ratchet/metYYYY.dat/surf/location/ratchet.ctl
   - …/assessment/ratchet/metYYYY.dat/stack/location/ratchet.ctl
   - …/assessment/cfest/analyte/realization/cfest.ctl
   - …/assessment/mass2/analyte/realization/biota.key
   - …/assessment/mass2/analyte/realization/realize.dat
   - …/assessment/mass2/analyte/realization/cfest/gwdrop.esp.

7. Model the release and transport through the vadose zone (run ESP, disable IOONLY keyword, enable and modify the RESTART keyword to run the realizations of interest, enable only the VADER, STOMP Flow, and STOMP Transport modules of the MODULES keyword), creating the following result files:

   - …/assessment/vadose/site/analyte/realization/vader.table
   - …/assessment/vadose/site/analyte/realization/input
   - …/assessment/vadose/site/analyte/realization/release.

8. Check the vadose results (run VZGRAB for all realizations).

9. If errors occurred in vadose modeling, fix and rerun the problem runs (run ESP, enable and modify the RESITE keyword to include the sites to rerun, modify the RESTART keyword to only rerun needed realizations, and disable ANALYTE keywords for analytes not needed to rerun).

10. Model the transport through the atmosphere (run ESP, disable RESITE keyword, enable realizations of interest on the RESTART keyword, enable only the RATCHET module of the MODULES keyword, and enable all ANALYTE keywords), creating the following result files:

    - …/assessment/ratchet/metYYYY.dat/surf/location/particle.bin
    - …/assessment/ratchet/metYYYY.dat/surf/location/noble.bin
    - …/assessment/ratchet/metYYYY.dat/surf/location/iodine.bin

11. Model the analyte/realization specific transport through the atmosphere (run ESP, disable RESITE keyword, enable realizations of interest on the RESTART keyword, enable only the AIRDROP module of the MODULES keyword, and enable all ANALYTE keywords), and place air concentration and air deposition data in the ECDA files.

12. Model the transport through the groundwater (run ESP, disable RESITE keyword, enable realizations of interest on the RESTART keyword, enable only the CFEST and GWDROP modules of the MODULES keyword, and enable all ANALYTE keywords), creating the following result files:

    - …/assessment/cfest/analyte/realization/hheldm001.tab
    - …/assessment/cfest/analyte/realization/hheldq001.tab
    - …/assessment/mass2/analyte/realization/cfest/COV-000.DAT
    - …/assessment/mass2/analyte/realization/cfest/TMS-node-FLOW.DAT
    - …/assessment/mass2/analyte/realization/cfest/TMS-node-analyte.DAT.

13. Check the groundwater results (run GRGRAB for all realizations).

14. If errors occurred in groundwater modeling, fix and rerun the problem runs (run ESP, modify the RESTART keyword to only rerun needed realizations, and disable ANALYTE keywords for analytes not needed to rerun).

15. Model the river transport (run ESP, enable realizations of interest in the RESTART keyword, enable only the MASS2 module of the MODULES keyword), creating the following result files:

    - …/assessment/mass2/analyte/realization/crdrop/analyte_realization.dat.

16. Model the river/groundwater interface zone (run ESP, enable realizations of interest in the RESTART keyword, enable only the RIPSAC module of the MODULES keyword)), and place concentrations in the ECDA files.

17. Model the contaminant accumulation in soil (run ESP, enable realizations of interest in the RESTART keyword, enable only the SOIL module of the MODULES keyword), and place concentrations in the ECDA files.

An example of a typical background-only analysis is provided to illustrate the above sequence of instructions. The main difference in the background-only modeling is with the GWDROP module. Because the groundwater module isn't run in this case, previously run CFEST result files (hheldq001.tab and hheldm001.tab) will be needed. ESP will read only the first time step and first realization of the groundwater concentration file. ESP will then create a new file (.tab file) containing zero concentrations. However, the flow results from CFEST for all time steps and realizations will be used in the calculation of the background river transport (specified on the CFDIRECT keyword of gwdrop.key).

The following is a typical background-only analysis:

1. Create a new main directory (…/assessment/bg).

2. Create the ESD keyword file.

3. Create the directory structure (run ESP, enable CREATEONLY and BACKGROUND keywords, enable the ECDA, MASS2, and RIPSAC modules of the MODULES keyword).

4. Create the input template and stochastic files:

   - …/assessment/bg/mass2/biota.stoch – MASS2 stochastic definitions for biotic transport
   - …/assessment/bg/mass2/mass2.key – MASS2 stochastic definitions (Kds)
   - …/assessment/bg/mass2/index.key – input definition file for CRDROP_INDEX (create cross index for MASS2 grid to the ECDA locations)
   - *…/assessment/bg/mass2/CRDROP_grid.dat* – MASS2 grid file (specified in the index.key file)
   - …/assessment/bg/mass2/gwdrop.key – GWDROP template input file, specify existing directories containing previously modeled CFEST result files (i.e., hheldm001.tab and hheldq001.tab) for the CFDIRECT keyword (e.g., …/assessment/cfest/Cs137/01)
   - *…/assessment/bg/mass2/col-river-elem.dat* – GWDROP river elements (specified in gwdrop.key)
   - *…/assessment/bg/mass2/col-river-node.dat* – GWDROP river nodes (specified in gwdrop.key)
   - *…/assessment/bg/mass2/hanfnad83m-pt-000* – GWDROP river cells (specified in gwdrop.key)
   - …/assessment/bg/ripsac/ripsac.key – RIPSAC stochastic definitions.

5. Create the input files and ECDA database (run ESP, disable CREATEONLY keyword, enable the IOONLY keyword, enable the ECDA, GWDROP, and MASS2 modules of the MODULES keyword).

6. Create the MASS2 node files (run ESP, enable realizations of interest in the RESTART keyword, enable only the GWDROP module of the MODULES keyword).

7. Model the river transport (run ESP, enable realizations of interest in the RESTART keyword, enable only the MASS2 module of the MODULES keyword).

8. Model the river/groundwater interface zone (run ESP, enable realizations of interest in the RESTART keyword, enable only the RIPSAC module of the MODULES keyword).

9. Model the soil contaminant accumulation (run ESP, enable realizations of interest in the RESTART keyword, enable only the SOIL module of the MODULES keyword).

3.2.2 Where Vadose Zone Flow and Transport Files Reside in SAC

In SAC, the vadose zone is modeled separately for each vadose zone site identified to ESP through the ESD keyword input file's SITE keyword. SAC defines a special subdirectory, always named /vadose, to manage vadose zone calculations. Both the VADER and STOMP codes share the /vadose subdirectory and all lower-level subdirectories therein. Figure 3.1 depicts the structure of the /vadose subdirectory.

```
.../assessment/vadose
  │
  ├stochastic_stomp.key
  └.../vadose/site
      │
      ├template_stomp.key
      ├.../vadose/site/analyte
      │    │
      │    └.../vadose/site/analyte/realization
      │         │
      │         ├input
      │         └restart
      └.../vadose/site/flow
           │
           └.../vadose/site/realization
                │
                ├input
                └restart
```

Figure 3.1 Structure of the /vadose directory

In this illustration of the /vadose subdirectory structure, fixed (hard-coded) names are bold, and variable names are plain text. For example, "**/vadose**" is literal, but "/site" could be /216-H-8 or /memfis, where the names of sites are defined by the SITE keyword in the ESD keyword input file. Similarly, "/analyte" could be /H3, /CCl4, or /Cs137, where the names are defined by the ANALYTE keyword in the ESD keyword input file. Realizations are numbered, and the number of digits used depends on the total number of realizations. Thus, for a 25-realization run, /realization could be /01, /02, or /25, but for a five-realization run it could be /1, /2, or /5. The /site, /analyte, and /realization directories are repeated for as many sites, analytes, and realizations as are specified in the ESD keyword input file.

The depiction in Figure 3.1 also indicates the appropriate location of two critical types of input files for the vadose zone flow and transport module: the stochastic_stomp.key and the template_stomp.key files. The stochastic_stomp.key file defines the values of all parameters treated stochastically in the vadose zone in a SAC assessment. The template_stomp.key file is a baseline STOMP input file that defines the deterministic vadose zone flow and transport simulation inputs for a site; one of these is required for each simulated vadose zone site. These files are discussed in more detail in Section 3.4.1.5. To illustrate the /vadose subdirectory structure further, consider an example SAC application, the "big" assessment, involving two sites (alpha and beta), two analytes (nitrate and molybdenum), and two realizations. The following would be the defining keywords in the ESD keyword file for this assessment (see Section 2.1):

```
REALIZAT 2 compute
ANALYTE ID="Ni" NAME="Nitrate" … compute
ANALYTE ID="Mo" NAME="Molybdenum" … compute
SITE "alpha" … compute
SITE "beta" … compute
```

In this assessment, the /vadose zone subdirectory would appear as shown in Figure 3.2.

```
.../big/vadose
 │
 ├stochastic_stomp.key
 ├/alpha
 │ │
 │ ├template_stomp.key
 │ ├/Mo
 │ │ │
 │ │ ├/01
 │ │ └/02
 │ ├/Ni
 │ │ │
 │ │ ├/01
 │ │ └/02
 │ └/flow
 │   │
 │   ├/01
 │   └/02
 └/beta
   │
   ├template_stomp.key
   ├/Mo
   │ │
   │ ├/01
   │ └/02
   ├/Ni
   │ │
   │ ├/01
   │ └/02
   └/flow
     │
     ├/01
     └/02
```

Figure 3.2 Example /vadose zone Subdirectory

This very simple example shows that SAC manages a very large directory structure for vadose zone calculations. The actual STOMP input and output files reside at the lowest level, the /realization level. Notice the extra directory at the /site level in the above depiction. The /site/flow directory is where ESP prepares a special STOMP simulation to solve the initial flow conditions for a given site and realization. The end product of each flow solution is a STOMP restart file that is copied from /site/flow/realization directories into each corresponding /site/analyte/realization directory. Thus, in the above example "big" assessment, the file /vadose/alpha/flow/1/restart is copied to /vadose/alpha/Mo/1 and /vadose/alpha/Mo/2. This provides the initial conditions for the transport solutions for Mo and Ni.

3.3 Code Environment

Running a SAC simulation is a multi-step, iterative process. The user will create directories and files, run ESP, create more files and copy files into the subdirectory structure, rerun ESP, modify more files, rerun ESP, and so on. This section discusses a typical execution path. The ESP can run an entire assessment in two passes. The first pass creates the subdirectory structure using the CREATEDIR keyword of the ESD file (see Section 2.1.4). The second pass runs the entire simulation. Alternatively, each process can be run separately using the MODULE keyword of the ESD file (see Section 2.1.18).

In practice, the user will typically run one process at a time (for example, inventory, then vadose, then groundwater transport, then river transport, and then finally river shore), inspect the results of the specific process, and then either rerun the process or proceed to the next process.

3.3.1 How Code Is Invoked

Under the Linux operating system, ESP is executed from a command window through the following Bourne Shell or C Shell command:

```
esp.exe ESD_Initial.key
```

This method will cause many diagnostic results to be written to the screen. To reduce the amount of output to the screen and log it for later examination, the following command could be used:

```
(esp.exe ESD.key > esp.log) >& errors.log &
```

This method will place the execution of ESP in the background (freeing the screen), log all diagnostics that would have come to the screen to the file esp.log, and log any system messages to the error.log file.

ESP distributes the majority of the work across a network of processors. All of the processors are defined using the PROCESSOR keyword of the ESD file (see Section 2.1.21). The directory for the work to be done on the external processor is also contained on the PROCESSOR keyword. An example of a PROCESSOR keyword is shown below:

```
PROCESSOR MACHINE=“c0-0” RUNDIR=“…/assessment” compute
```

In this example, the processor is named “c0-0” and the analysis will be done under the …/assessment subdirectory. This distributed modeling technique will only work if the external processors can access the same disk with the same path names as those used by the main processor.

During execution, ESP will create batch procedures for performing portions of the analysis. These batch procedures will be submitted using a remote (secure) shell command. The following is an example of this command:

```
ssh  -f c0-0 source …/assessment/processors/p121.com
```

As a result of this command, the batch procedure (…/assessment/processors/p121.com) will be submitted on the c0-0 processor. The 121 in the batch procedure file name (p121.com) is derived from the list of processors. Processor c0-0 must be the 121st entry in the processor list (as specified by the PROCESSOR keyword of the ESD file). Table 3.1 shows a sample batch procedure for modeling release and transport in the vadose.

Table 3.1 Batch Procedure …/processors/p121.com

```
#!/bin/csh
# execute vader and stomp for: 20020412090209
cd /home/ANALYSIS/Initial2/
if (-e processors/p121.done ) rm -f processors/p121.done
if (-e processors/p121.fail ) rm -f processors/p121.fail
echo 0 > processors/p121.run
echo 0 > processors/p121.start

cd /home/ANALYSIS/Initial2/vadose/307_RB/U238/23/
# run vader
if (-e vader.done ) rm -f vader.done
if (-e vader.fail ) rm -f vader.fail
vader.exe /home/ANALYSIS/Initial2/vadose/307_RB/U238/23
if ( -e vader.fail ) then
   cd /home/ANALYSIS/Initial2/
   echo "vader failed: /home/ANALYSIS/Initial2/vadose/307_RB/U238/23/" > processors/p121.fail
   goto done
endif

# modify input for AreaX
prestomp.exe input  -1.0000E+00
if ( -e prestomp.fail ) then
   cd /home/ANALYSIS/Initial2/
   echo "prestomp failed: /home/ANALYSIS/Initial2/vadose/307_RB/U238/23/" > processors/p121.fail
   goto done
endif

# run stomp
rm -f plot.*
rm -f restart.*
if (-e stomp.done ) rm -f stomp.done
if (-e stomp.fail ) rm -f stomp.fail
stomp.exe > stomp.log
if ( -e stomp.fail ) then
   cd /home/ANALYSIS/Initial2/
   echo "stomp failed: /home/ANALYSIS/Initial2/vadose/307_RB/U238/23/" > processors/p121.fail
   goto done
endif
rm -f restart.*

cd /home/ANALYSIS/Initial2/
echo 0 > processors/p121.done
done:
rm -f processors/p121.run
echo done with vadose modeling: vadose/307_238/23/
```

Table 3.2 provides a sample procedure for modeling the groundwater transport (running the VZDROP, CFEST, and GWDROP codes).

Table 3.2 Batch Procedure …/assessment/processors/p002.com

```
#!/bin/csh
# script to run GW transport for:20020414091255
cd /home/ANALYSIS/Median2/
if (-e processors/p002.done ) rm -f processors/p002.done
if (-e processors/p002.fail ) rm -f processors/p002.fail
```

```
echo 0 > processors/p002.run
echo 0 > processors/p002.start

cd /home/ANALYSIS/Median2/cfest/CrVI/1/
# run vzdrop.exe
if (-e vzdrop.done ) rm -f vzdrop.done
vzdrop.exe vzdrop.key
if ( -e vzdrop.fail ) then
   cd /home/ANALYSIS/Median2/
   echo "vzdrop failed: cfest/CrVI/1/" > processors/p002.fail
   goto done
endif

echo 0 > cfest.run
echo 0 > cfest.start
cp ../bincf/cfest.* bincf
cp ../cf_tmpbinary/*.* cf_tmpbinary
cp ../cf_runs.out .
lprog3i_001.exe
lprog3_001.exe
lgethheld_001.exe
cp hheldq/hheldq001.tab .
cp hheldm/hheldm001.tab .

cd /home/ANALYSIS/Median2/
rm  /home/ANALYSIS/Median2/mass2/CrVI/1/cfest/COV*.DAT
rm  /home/ANALYSIS/Median2/mass2/CrVI/1/cfest/TMS*.DAT
rm  /home/ANALYSIS/Median2/mass2/CrVI/1/cfest/polygons.dat
gwdrop.exe /home/ANALYSIS/Median2/mass2/CrVI/1/cfest/gwdrop.inp

cd /home/ANALYSIS/Median2/
echo 0 > processors/p002.done
done:
rm -f processors/p002.run
echo done with GW modeling: cfest/CrVI/1/
```

Table 3.3 provides a sample procedure for modeling the river transport (running MASS2 and CRDROP).

Table 3.3 Batch Procedure…/assessment/processors/p005.com

```
#!/bin/csh
# script to run Mass2 transport for:20020327161427
cd /home/ANALYSIS/Median2/Bg/
if (-e processors/p005.done ) rm -f processors/p005.done
if (-e processors/p005.fail ) rm -f processors/p005.fail
echo 0 > processors/p005.run
echo 0 > processors/p005.start

cd /home/ANALYSIS/Median2/Bg/mass2/Cs137/1/
# run mass2
if (-e mass2.done ) rm -f mass2.done
if (-e mass2.fail ) rm -f mass2.fail
(exec /home/CODES/mass2/bin/runmass2NB -l /home/ANALYSIS/Median2/Bg/mass2/Cs137/1/)
if ( -e mass2.fail ) then
   cd /home/ANALYSIS/Median2/Bg/
   echo "mass2 failed" > processors/p005.fail
   goto done
endif
```

```
# run crdrop
cd crdrop
if (-e crdrop.done ) rm -f crdrop.done
if (-e crdrop.fail ) rm -f crdrop.fail
cp   /home/ANALYSIS/Median2/Bg/ESD_Median2_Bg.key .
cp   /home/ANALYSIS/Median2/Bg/mass2/CrossIndex.grd .
crdrop.exe ESD_Median2_Bg.key 20020327161427    1 1 Cs137
rm  ESD_Median2_Bg.key
rm  CrossIndex.grd
if ( -e crdrop.fail ) then
   cd /home/ANALYSIS/Median2/Bg/
   echo "crdrop failed" > processors/p005.fail
   goto done
endif

cd /home/ANALYSIS/Median2/Bg/
echo 0 > processors/p005.done
done:
rm -f processors/p005.run
echo done with GW modeling: /home/ANALYSIS/Median2/Bg/mass2/Cs137/1/
```

3.3.2 Code Control and Keyword Descriptions

ESP is controlled entirely by the ESD keyword file. Section 2.1 describes each keyword of this file. Table 3.4 provides an example ESD keyword file.

Table 3.4 Example ESD Keyword File for Use by ESP

```
!  SAC Rev.1 2004 Composite Analysis 1,000-year (Stochastic) Assessment
!
!-------------------------------------| Simulation Control
!
REPORT "ESD_CA2_stoch_1ky-CrDir.rpt"
TITLE  "2004 Composite Analysis 1,000-year (Stochastic Inputs) Assessment"
USER   "Eslinger-DWE-WEN"
!
!- Declare operating system
OS "Unix"
!
!- Declare where executables reside
EXEDIR "/home/CODES/bin"
!
!- Output simulated stochastic values to file
DEBUG STOCHASTIC="stoch/esp_stoch.out"
!
!- Number of realizations to run
REALIZAT 100
!
PERIOD START=1944 STOP=3050 CLOSURE=2070 !
BALANCE 2000 2035 2070 2100 2150 2550 3050
!
!- Create problem subdirectory structure (1st time)
CREATEDIR compute
!
!- Create input files only - don't simulate
!IOONLY compute
!
!- Restart mode only (to preserve 2-pass STOMP setup)
```

```
!RESTART 1 compute
!
!- Resite mode (for individual VZ site handling)
!RESITE "216-B-14" compute

!-------------------------------------| SAC Module Options |
!BACKGROUND compute
!
!MODULE ID="INVENTORY"       compute
MODULE ID="VADER"           compute
MODULE ID="STOMP" FLOW HIRES compute
MODULE ID="STOMP" TRAN HIRES compute
MODULE ID="CFEST" HIRES     compute
MODULE ID="GWDROP"          compute
MODULE ID="MASS2"           compute
MODULE ID="RIPSAC"          compute
MODULE ID="ECDA"            compute
MODULE ID="SOIL"            compute
MODULE ID="CRDROP"          compute
MODULE ID="RATCHET"         compute
MODULE ID="AIRDROP"         compute

!-----------------------------------| Distributed Processors
PROCESSOR MACHINE="c0-0"  PRIORITY=3
  RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky" compute
PROCESSOR MACHINE="c0-1"  PRIORITY=3
  RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky" compute
PROCESSOR MACHINE="c0-62" PRIORITY=3
  RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky" compute
PROCESSOR MACHINE="c0-63" PRIORITY=3
  RUNDIR="/home/ANALYSIS4/CA2_stoch_1ky" compute

!------------------------------------------| Analytes
ANALYTE ID="C14"  NAME="Carbon-14"  TYPE="NR" AIR="NOBLE"      COMPUTE
  MOLWGT=1.400324E+01  HALFLIFE=5.715000E+03  SPECIFIC=4.470906E+00
  DFIMM=5.122392E-05  DFSED=7.200000E-23  GAMMA=0.000000E+00
  MOLDIFF=1.050000E-05  GASDIFF=0.000000E+00  HENRY=0.000000E+00
...
ANALYTE ID="U238"  NAME="Uranium-238"  TYPE="NR"               COMPUTE
  MOLWGT=2.380508E+02  HALFLIFE=4.470000E+09  SPECIFIC=3.362507E-07
  DFIMM=9.276314E-04  DFSED=5.520000E-22  GAMMA=1.000000E-03
  MOLDIFF=1.050000E-05  GASDIFF=0.000000E+00  HENRY=0.000000E+00

!-------------------------------| ECDA Solution Times
TIMES
  1945  1950  1955  1960  1965  1970  1975  1980  1985  1990
  2960  2980  3000  3050

!--------------------------| SORP-Water Partition Coefficients (Kd's)
ECHO KDSOIL ! Echo SORP-water Kd values to the ECDA report file
FILE KDSOIL
  NAME="/home/ANALYSIS4/CA2_stoch_1ky/ecda/KDSOIL_CA2_stoch_1ky.dat"
  SEED=232323.0 CREATE
!   Kd's for bed sediment are from the stomp stochastic file, 4G class
KDSOIL ID="KDC"    2    0.0     0.1    LABEL="Soil-water Kd" UNITS="L/g"
KDSOIL ID="KDU"    9   -7.131   0.596  LABEL="Soil-water Kd" UNITS="L/g"
  TRUNCATE   0.010   0.990
!
!    Kd's for suspended sediment are 10 larger than bed sediment Kd's
KDSOIL ID="KDSUSC"    2    0.0       1.0
  LABEL="Soil-water Kd - suspended sediment" UNITS="L/g"
```

```
KDSOIL ID="KDSUSU"    9   -4.828     0.596
  LABEL="Soil-water Kd - suspended sediment" UNITS="L/g"
  TRUNCATE   0.010   0.990

!--------------------------------| Dilution Factor Data for RIPSAC
ECHO DILUTE ! Echo dilution data to the ECDA report file
FILE DILUTE
  NAME="/home/ANALYSIS4/CA2_stoch_1ky/ecda/DILUTE_CA2_stoch_1ky.Dat"
  SEED=123445.0 CREATE
DILUTE ID="DF5m" 6  0.36 0.47 0.63
  LABEL="Dilution factor for 5 meters from the river" UNITS="None"
!
!---------------------------------------| Infiltration Data
FILE INFILT NAME="/home/ANALYSIS4/CA2_stoch_1ky/ecda/infilt.dat"
!-------------------------------| ECDA Concentration Related Files
! FILLECDA keyword initializes the ECDA files
FILLECDA GWAT FIXED =  0.0
FILLECDA PWAT FIXED =  0.0
FILLECDA SWAT FIXED =  0.0
FILLECDA SEEP FIXED = -1.0
FILLECDA SEDI FIXED =  0.0
FILLECDA SORP FIXED = -1.0
FILLECDA SODR FIXED = -1.0
FILLECDA SOGW FIXED = -1.0
FILLECDA SOSW FIXED = -1.0
FILLECDA AIRC FIXED =  0.0
FILLECDA AIRD FIXED =  0.0
!
! SACVIEW header file for ECDA concentration files
FILE HEADER NAME="/home/ANALYSIS4/CA2_1ky/ecda/Sacview_CA2_1ky.hdr" CREATE
!
! ECDA record number index map file for concentration data
FILE I_ECDA  NAME="/home/ANALYSIS4/CA2_stoch_/ecda/ECDA_CA2_1ky.map" CREATE
!
! ECDA Concentration data files for each analyte
FILE C_ECDA ANALYTE="C14"
  NAME="/home/ANALYSIS4/CA2_stoch_1ky/ecda/C14_CA2_stoch_1ky.dat"  create
FILE C_ECDA ANALYTE="U238"
  NAME="/home/ANALYSIS4/CA2_stoch_1ky/ecda/U238_CA2_stoch_1ky.dat" !create

!-------------------------------------| Irrigation Control
IRRIGATE SPRING=121 FALL=256 RATE=75.0 NET=0.20 START=2004
  THETAIRG=0.5 THETADRY=0.2

!---------------------------| Vadose Zone Release and Transport Sites
SITE ID="100-B-5" POINTS=4 (565432.181,144563.181),
  (565441.819,144563.181),  (565441.819,144572.819),
 (565432.181,144572.819)  TITLE="Trench"
  AreaX=-1.0 compute
SITE ID="100-F-23" POINTS=4 (580870.5,147790.281), (580871.5,147790.281),
  (580871.5,147791.281), (580870.5,147791.281)  TITLE="French Drain"
  AreaX=-1.0 compute
SITE ID="600-148" POINTS=4 (568708.787,134376.6), (569023.337,134376.6),
  (569023.337,134691.15), (568708.787,134691.15)
  TITLE="Landfill (Lined)" AreaX=-1.0 compute
SITE ID="600-211" POINTS=4 (566414.412,138066.896), (566460.838,138066.896),
  (566460.838,138113.322), (566414.412,138113.322)
  TITLE="Drain/Tile Field" AreaX=-1.0 compute
SITE ID="US_Ecology" POINTS=4 (571880.117,134000.117),
  (572519.883,134000.117),  (572519.883,134639.883),
  (571880.117,134639.883)  TITLE="Burial Ground" AreaX=-1.0 compute
```

```
!-------------------------------| Vadose Zone Infiltration Classes
INFILTRATION SEED=2591.67
!
! declare all unique infiltration classes for vadose zone release sites
INFILTRATION DEFINE UNIT="mm/yr" CLASS="River" 1 1
  "Columbia River outfall locations (placeholder data)"
INFILTRATION DEFINE UNIT="mm/yr" CLASS="Wa-s" 6 0.055 0.11 0.22
  "Warden Silt Loam (Wa) - (Represents final degradation of barrier top)"
!
!- Declare all transitional classes for vadose zone release sites
INFILTRATION DEFINE CLASS="Eb-ds_Eb-s_1"  FRACTION=0.8
  PRIOR="River" NEXT="Wa-s""Barrier Degradation 1"
INFILTRATION DEFINE CLASS="Wa-s_Rp-s_4"
  FRACTION=0.2 PRIOR="Wa-s" NEXT="Eb-ds_Eb-s_1""Barrier Degradation 4"
!
! assign infiltration classes to all vadose zone release sites
INFILTRATION SITE="100-B-5" CLASS="River" START=1944 END=2000 compute
INFILTRATION SITE="100-B-5" CLASS="Wa-s" START=2001 END=3050  compute
!
!--------------------------------| Vadose Zone Site Evaporation
EVAPORATE SITE="216-B-3" START="01/01/1945" END="01/02/1945"
  VOLUME=14881.09 UNITS="m^3" compute
EVAPORATE SITE="216-U-10" START="01/12/1985" END="01/01/1986"
  VOLUME=7559.96 UNITS="m^3" compute

!--------------------------------| Vadose Zone Site Remediations
REMEDIATION YEAR=1991 FROM="618-9" TO="600-148" UPPER=0.0 LOWER=4.5 SOIL=1.0
  DEBRIS=1.0 CAKE=1.0 SLUDGE=1.0 GLASS=1.0 CEMENT=1.0 compute
REMEDIATION YEAR=1995 FROM="4843" TO="600-148" UPPER=0.0 LOWER=1 SOIL=1.0
  DEBRIS=1.0 CAKE=1.0 SLUDGE=1.0 GLASS=1.0 CEMENT=1.0 compute
REMEDIATION YEAR=1997 FROM="116-B-4" TO="600-148" UPPER=0.0 LOWER=4.5
  SOIL=1.0 DEBRIS=1.0 CAKE=1.0 SLUDGE=1.0 GLASS=1.0 CEMENT=1.0 compute
!
!--------------------------------------| Impacts Locations
LOCATION ID="UH0001" NAME="UnsuitableForAg" EASTING=576022 NORTHING=154367
  GWAT SOGW SOSW SODR AIRC AIRD, TYPE = "UPLAND", LOCUS = "HANFORD"
  APSD     = 4.00E-02  POROSITY= 3.50E-01  FOC    = 1.0E+00
  VEGCOV   = 5.00E-01  NECF    = 1.00E+00  RHOS   = 1.50E+00  SRH  = 1.8E-02
  TEMP     = 2.85E+02  MSWIND  = 3.44E+00  MZWIND = 3.44E+00
  IRG_SWAT ="QHP025"   AREA    = 1457376.1
LOCATION ID="RHP234" NAME="RiparianBenton1100Area"
  EASTING=595288.4 NORTHING=111893.7
  GWAT SEEP SORP AIRC AIRD
  TYPE="RIPARIAN"  LOCUS="HANFORD"  MILE=341.80
  APSD    = 4.00E-02  COXYGEN = 1.10E-02  POROSITY= 3.50E-01
  FOC  = 1.0E+00   VEGCOV  = 5.00E-01
  NECF    = 1.00E+00  RHOS    = 1.50E+00  SRH     = 1.8E-02
  TEMP = 2.85E+02  MSWIND  = 3.44E+00
  MZWIND  = 3.44E+00  AREA    = 9360
LOCATION ID="QFP003" NAME="AquaticGrantUpriverVernitaBridge"
  EASTING=557964.6 NORTHING=145457.0  SWAT SEDI PWAT
  TYPE="AQUATIC"  LOCUS="FARSIDE"  MILE=388.60
  COXYGEN = 1.10E-02 AREA = 2.5E+03
!-------------------------------| Ecological Models Additional Data
RADIUS ID="Radius" 1.400E+00  2.000E+00  3.000E+00  5.000E+00
  7.000E+00  1.000E+01  2.000E+01  3.000E+01
RADIUS ID="C14"    5.000E-02  5.000E-02  5.000E-02  5.000E-02
  5.000E-02  5.000E-02  5.000E-02  5.000E-02
RADIUS ID="U238"   4.300E+00  4.300E+00  4.300E+00  4.300E+00
  4.300E+00  4.300E+00  4.300E+00  4.300E+00
```

```
!-----------------------------------| Biotic Species Information
! Type QP: Aquatic plants
SPECIES ID="QPERIP" TYPE="QP" NAME="periphyton" HABITAT="AQUATIC"
 AWD=10  FLIPID=0.0418   FOC=0.35  RADIUS=1.4   WBMASS=0.000035
! Type QA: Aquatic animals
SPECIES ID="QCARPS" TYPE="QA" NAME="CARP  " HABITAT="AQUATIC"
 OCAR=8.0000E-01   AWD=5.0000E+00 FLIPID=7.4000E-02 FOC=4.5000E-01
 WBMASS=1.0500E+03 RADIUS=7.0000E+00
! Type TA: Terrestrial animals - riparian locations
SPECIES ID="RACOOT" TYPE="TA" NAME="American coot" HABITAT="RIPARIAN"
  AE=8.1000E-01      DIFFHT=2.0000E-01  ETWATER=2.0000E+01  FMR=1.6309E+02
  FWATER=5.0000E-01  FDW=0.4            GE=1.9000E+03    INHRATE=6.1593E-01
  PCS=2.5000E-01     PCW=5.0000E-01     PSI=1.0000E+00   RADIUS=5.0000E+00
  SADHER=1.4500E+00  THETA=1.0000E+00   SA=7.8406E+02    WATERING=4.6102E-02
  WEIGHT=6.9200E-01
! Type TA: Terrestrial animals - upland locations
SPECIES ID="UBDGER" TYPE="TA" NAME="American badger" HABITAT="UPLAND"
  AE=8.1000E-01      DIFFHT=2.0000E-02  ETWATER=0.0000E+00  FMR=8.2073E+02
  FWATER=0.0000E+00  FDW=0.4            GE=1.7000E+03    INHRATE=6.0032E+00
  PCS=2.2000E-01     PCW=0.0000E+00     PSI=1.0000E+00   RADIUS=1.0000E+01
  SADHER=1.4500E+00  THETA=1.0000E+00   SA=4.1972E+03    WATERING=7.2063E-01
  WEIGHT=9.0000E+00
! Type TP: Terrestrial plants - riparian locations
SPECIES ID="RCTWOD" TYPE="TP" NAME="black cottonwood" HABITAT="RIPARIAN"
  SURFACE=7.8050E+01 AE=3.4000E-01  DIFFHT=5.0000E+00 FPA=5.0000E-01
  FPL=1.0000E-02  FPW=4.0000E-01  FW=5.5000E-01 GE=3.2000E+03
  FWATER=0.0000E+00  RADIUS=3.0000E+01 RHOP=1.0000E+03
  WEIGHT=2.0000E+02 ETWATER=0.0000E+00
! Type TP: Terrestrial plants - upland locations
SPECIES ID="UFUNGI" TYPE="TP" NAME="fungi" HABITAT="UPLAND"
  SURFACE=1.0000E-03 AE=7.3000E-01  DIFFHT=5.0000E-02 FPA=5.0000E-01
  FPL=1.0000E-02  FPW=4.0000E-01    FW=9.1000E-01 GE=6.3000E+02
  FWATER=0.0000E+00 RADIUS=1.4000E+00 RHOP=1.0000E+03 WEIGHT=3.0000E-03
  ETWATER=0.0000E+00
!----------------------------------------|  End of ESD File
END
```

3.4 Data Files and Simulation Control

ESP is used to build or modify input files for several deterministic codes. The following sections describe these files and provide some examples.

3.4.1 Input Files and Simulation Control

Each code in the SAC suite of codes relies on either the ESD keyword file or their own input files. This section contains descriptions of input files for each module. Several utility codes used to generate keywords for the main simulation codes take information from a master waste site definition spreadsheet. This spreadsheet is used to maintain a single source of site information. The spreadsheet is too complex to include in printed form.

3.4.1.1 Input Files for ECDA

The creation of the ECDA files can be controlled by ESP or they can be created outside the main processing stream. The typical method is to create the ECDA files outside SAC-ESO control. The program ecda-1.exe creates the ECDA files for all analytes (see Section 2.2) and a header file used by the SACView program when viewing the contents of the ECDA files. The ecda-1.exe program relies solely on the ESD keyword file for problem control.

3.4.1.2 Input Files for INVENTORY

The inventory module typically will be modeled external to ESP control. Section 1.0 discusses the input files and the process required to model the inventory.

3.4.1.3 Input Files for VADER

ESP creates a VADER input file (vader.key) for each execution of VADER (vader.exe). See Table 3.5 for an example file. VADER will be executed once for every combination of sites, analytes, and realizations. ESP creates these files, all named vader.key but located in different directories, using a template file named template_vader.key and a stochastic definition file named stochastic_vader.key that are located in each release site subdirectory.

The VADER template file contains information on the analytes and release models, including parameters, for all release models being used at the specific waste site. Table 3.5 shows an example of this file.

Table 3.5 Example File …/assessment/vadose/600-148/template_vader.key

```
! VADER template keyword file (template.key)
!
TITLE "VADER template keyword file for the 600-148 site"
!
! Define models for each analyte
MODEL ANALYTE="H3" STARTREL=1944 STOPREL=3050 SOIL KD=0.0
  CS=9.7000E+07 TW=5.9400E-02
  B=1.5350E+00 A=9.8942E+08 H=5.3000E+02 COMPUTE
MODEL ANALYTE="Sr90" STARTREL=1944 STOPREL=3050 SOIL KD=0.0
  CS=1.3700E+06 TW=5.9400E-02
  B=1.5350E+00 A=9.8942E+08 H=5.3000E+02 COMPUTE
!
END
```

The stochastic definition file stochastic_vader.key contains information for all analytes, all release models being used for this waste site, and the simulation definitions. In the example of the VADER stochastic definition file (…/assessment/vadose/600-148/stochastic_vader.key) shown in Table 3.6, the soil release model is being used for the 600-148 site and for the two analytes H3 and Sr90.

Table 3.6 Example File …/assessment/vadose/600-148/stochastic_vader.key

```
! VADER stochastic keyword file (stochastic.key)
TITLE "VADER stochastic keyword file for the 600-148 site"
SEED 2323.23
DEBUG STOCHASTIC="stoch/vader_stoch.out"
!
! Define stochastic values for each analyte
STOCHASTIC ANALYTE="H3" "SOIL-KD" "6H"
STOCHASTIC ANALYTE="H3" "SOIL-CS" 1 9.7000E+07
STOCHASTIC ANALYTE="H3" "SOIL-TW" 11 3.2828E+00 5.2817E+01
```

```
  1.000E-03 9.9900E-01
STOCHASTIC ANALYTE="H3" "SOIL-B"  7 1.535E+00 1.085E-02
  TRUNCATE 1.000E-02 9.9000E-01
STOCHASTIC ANALYTE="H3" "SOIL-A"  1 9.8942E+08
STOCHASTIC ANALYTE="H3" "SOIL-H"  1 5.3000E+02
!
STOCHASTIC ANALYTE="Sr90" "SOIL-KD" "6H"
STOCHASTIC ANALYTE="Sr90" "SOIL-CS" 1 1.3700E+06
STOCHASTIC ANALYTE="Sr90" "SOIL-TW" 11 3.2828E+00 5.28171E+01
  1.000E-03 9.9900E-01
STOCHASTIC ANALYTE="Sr90" "SOIL-B"  7 1.535E+00 1.085E-02
  TRUNCATE 1.000E-02 9.9000E-01
STOCHASTIC ANALYTE="Sr90" "SOIL-A"  1 9.8942E+08
STOCHASTIC ANALYTE="Sr90" "SOIL-H"  1 5.3000E+02
!
END
```

For this simulation, using the file shown in Table 3.6, the sorption value (SOIL-KD) will be simulated using the 6H sorption distribution as defined in the STOMP stochastic definition file stochastic_stomp.key. The solubility (SOIL-CS) will be constant (same for all realizations) for both H3 and Sr90 (9.7000E+07 Ci/cm^3 and 1.3700E+06 Ci/cm^3, respectively). The fractional volumetric water content (SOIL-TW) will be simulated for H3 and Sr90 using the same Beta distribution (distribution type 11, with parameters 3.2828E+00, 5.2817E+01, 1.000E-03, and 9.9900E-01). The bulk density (SOIL-B) will be simulated for H3 and Sr90 using the same normal distribution (distribution type 7, with parameters 1.535E+00, 1.085E-02, TRUNC 1.000E-02, and 9.9000E-01). The cross-sectional area (SOIL-A) and thickness (SOIL-H) will be constant for both H3 and Sr90 (9.8942E+08 cm^2 for both analytes and 5.3000E+02 cm for both analytes, respectively).

3.4.1.4 Preparation of VADER Input Files Using the Utility Code vadertemplate.exe

A program named vadertemplate.exe creates the VADER template and stochastic files used in creating the vader.key files (by running ESP with the IOONLY keyword enabled in the ESD keyword file). To create VADER stochastic files containing median values rather a full set of stochastic generated values, run the vadertempmedian.exe program, instead of vadertemplate.exe. Use the following procedure to run vadertemplate.exe.

1. Access the master waste site definition spreadsheet (GVLsSiteList-2002-01-15-DWE.xls, for example).

2. Save the main data (the Full SAC Rev. 0 List) as a csv-format file (named vadose.csv in this example). Several of the comment columns need to be deleted, including
   - F – Other Site Names
   - J – Location Description / UPR-source-Code
   - Y – Waste Type (from HSWMUR)
   - AD – Comments / References
   - AI – Description from HSWMR

3. Save the release table of the master spreadsheet (containing analytes, solubility, fractional release rate, and diffusion coefficient) as a csv-format file (call it solubility.dat). The following is an example of the contents of this file:

```
Analyte     Csat       Fraction Release   Diffusion Coeff
H3    9.70E+07    3.65E-04    1.58
Sr90  1.37E+06    1.10E-02    1.58E-03
```

4. Create a control file (call it vadertemplate.dat). The following is an example of the contents of this file:

```
1944 3050         -- start date, end date
2323.23                 -- seed
2                 -- number of analytes
H3
Sr90
```

5. Create a file that contains the sites to process (call it sites.dat). The following is an example of the contents of this file:

```
Site Template
116-B-5
116-B-7
118-B-8
216-B-17
241-A-101
600-148
```

6. Create the STOMP stochastic definition file (called stochastic_stomp.key).

7. Make sure that these five files are located in the directory to which all the sites will be created – typically, this will be in the vadose subdirectory of the assessment. If the site directories do not exist, then the program will create them. It is easier to do this step after the creation of the directory structure, which can be accomplished by running ESP with the CREATEDIR keyword enabled. The following five data files are needed for this program:

   - vadose.csv ! user supplies the name
   - vadertemplate.dat ! name is set
   - sites.dat ! name is set
   - solubility.dat ! name is set
   - stochastic_stomp.key ! name is set

8. Open a command window in Linux and change to the directory where the five files reside, typically …/assessment/vadose.

9. Run the program provided to create the ESP/VADER input files. The data file can be placed on the command line. If it is missing, then the program will prompt for it.

10. The program should create a number of files. A file named template _vader.key should be located in each waste site subdirectory. Table 3.5 provides an example of this file. A file named stochastic_vader.key should be located in each waste site subdirectory. Table 3.6 provides an example of this file. In addition, a file called models.csv should be generated. This file contains lists of sites, release models, and release parameters. The following is an example of the contents of this file:

```
116-B-5,Liquid,1.0
116-B-7,River,1.0
118-B-8,Reactor Block,3.6500E-04
216-B-17,Liquid,1.0
241-A-101,Liquid,1.0
241-A-101,Salt Cake,0.36,7(1.58;0.2;0.01;0.99),4104330
600-148,Soil Debris,6H,7(1.5;0.1;0.01;0.99),11(3.28;52.81;0.001;0.999),9.8942E+08,5.3000E+02
```

3.4.1.5 Input Files for STOMP

For every execution of VADER, STOMP will also be run. In fact, VADER and the STOMP transport are always run together. ESP creates a different STOMP input file (input-esp) for each execution of STOMP (stomp.exe), and places it in the appropriate subdirectory. Table 3.7 provides an example input-esp file. STOMP will be executed once for every combination of site, analyte, and realization. ESP uses a template file (named template_stomp.key) for each site and a stochastic definition file (named stochastic_stomp.key) when creating the STOMP input files.

Table 3.7 Example STOMP Control File input-esp

```
#SAC STOMP input created by ESP   05/06/2002 - 14:45:07
#SAC Case ID          : SAC Rev. 0 Initial2 Assessment
#SAC Template         : vadose/618-11/template_stomp.key
#SAC Site ID          : 618-11
#SAC Site NW Easting  :   588884.0
#SAC Site NW Northing :   127356.7
#SAC Site SE Easting  :   589070.6
#SAC Site SE Northing :   127170.0
#SAC Analyte          : Tc99
#SAC Realization      :     1

~Simulation Title Card
1,
316R-6,
W E Nichols,
Pacific Northwest National Laboratory,
January 9 2002,
3 PM PST,
2,
Template 316R - For shallow disposal sites
300 Area, Shallow Burial, Waste Chemistry Designation 6

~Solution Control Card
Restart,
Water Transport Courant,
1,
0.0,yr,1110.0,yr,1.0,s,1100.0,yr,1.5,8,1.0e-6,
10000,
```

```
0,

~Grid Card
Cartesian,
1,1,495,
0.00000E+00,m, 1.86641E+02,m,
0.00000E+00,m, 1.86641E+02,m,
0.0,ft,305@0.2,ft,220@0.5,ft,

~Rock/Soil Zonation Card
3,
Hg(6H),1,1,1,1,448,495,
Rg(6I2),1,1,1,1,306,447,
PPlz(6I2),1,1,1,1,1,305,

~Mechanical Properties Card
hg(6h),2650,kg/m^3, 1.75030E-01, 1.75030E-01,,,millington and quirk,
rg(6i2),2650,kg/m^3, 6.80613E-02, 6.80613E-02,,,millington and quirk,
pplz(6i2),2666,kg/m^3, 4.54146E-01, 4.54146E-01,,,millington and quirk,

~Hydraulic Properties Card
hg(6h), 5.42278E-04,hc cm/s, 5.42278E-04,hc cm/s, 5.42278E-04,hc cm/s,,
rg(6i2), 1.46331E-03,hc cm/s, 1.46331E-03,hc cm/s, 1.46331E-03,hc cm/s,,
pplz(6i2), 7.60036E-07,hc cm/s, 7.60036E-07,hc cm/s, 7.60036E-07,hc cm/s,,

~Saturation Function Card
hg(6h),Nonhysteretic van Genuchten, 5.53759E-02,1/cm ,1.38727E+0,4.34833E-02,,
rg(6i2),Nonhysteretic van Genuchten, 1.17821E-2,1/cm ,1.66846E+0,2.70291E-02,,
pplz(6i2),Nonhysteretic van Genuchten,6.07363E-3,1/cm ,1.98132E+0,1.00728E-1,,

~Aqueous Relative Permeability Card
Hg(6H),Mualem,,
Rg(6I2),Mualem,,
PPlz(6I2),Mualem,,

~Solute/Fluid Interaction Card
1,
Tc99,Conventional, 1.00000E-09,cm^2/s,, 2.11100E+05,yr,
0,

~Solute/Porous Media Interaction Card
hg(6h),0.09,m,,,
Tc99, 0.00000E+00,mL/g,1.0,
rg(6i2),0.09,m,,,
Tc99, 0.00000E+00,mL/g,1.0,
pplz(6i2),0.031,m,,,
Tc99, 0.00000E+00,mL/g,1.0,

~Initial Conditions Card
Gas Pressure,Aqueous Saturation,
4,
Gas Pressure,101325.0,Pa,0.0,1/m,0.0,1/m,0.0,1/m,1,1,1,1,1,495,
Aqueous Saturation ROCK,0.2230,,Hg(6H),
Aqueous Saturation ROCK,0.2259,,Rg(6I2),
Aqueous Saturation ROCK,0.2178,,PPlz(6I2),

~Boundary Conditions Card
```

```
2,
#Top,Neumann,Zero Flux,
#1,1,1,1,495,495,1,
#0.0,yr,-1.0,mm/yr,,,
Bottom,Dirichlet,Outflow,
1,1,1,1,1,1,1,
0.0,yr,101325.0,Pa,,,
Top,Neumann Aqueous,Zero Flux,
1,1,1,1,495,495,8,
0.000000000000E+00,s,-4.00000E+00,mm/yr,,,
5.680800000000E+08,s,-5.54000E+01,mm/yr,,,
2.303769600000E+09,s,-4.00000E+00,mm/yr,,,
3.281990400000E+09,s,-4.00000E+00,mm/yr,,,
3.534451200000E+09,s,-4.00000E+00,mm/yr,,,
3.755376000000E+09,s,-4.00000E+00,mm/yr,,,
4.007836800000E+09,s,-4.00000E+00,mm/yr,,,
4.228761600000E+09,s,-4.00000E+00,mm/yr,,,

~Output Control Card
1,
1,1,495,
1,1,yr,m,6,6,6,
6,
aqueous pressure,pa,
aqueous saturation,null,
z aqueous vol,mm/yr,
aqueous courant number,null,
solute volumetric conc.,Tc99,1/m^3,
solute source integral,Tc99,null,
12,
1.609459200000E+09,s,
1.767225600000E+09,s,
3.345148800000E+09,s,
4.922985600000E+09,s,
8.078659200000E+09,s,
1.123433280000E+10,s,
1.439000640000E+10,s,
1.754576640000E+10,s,
2.070144000000E+10,s,
2.701278720000E+10,s,
3.332422080000E+10,s,
3.490205760000E+10,s,
1,
solute volumetric conc.,Tc99,1/m^3,

~SAC Release Plane Card
1,
0.0,yr,1,

~Balance Card
12,
1.609459200000E+09,s,
1.767225600000E+09,s,
3.345148800000E+09,s,
4.922985600000E+09,s,
8.078659200000E+09,s,
1.123433280000E+10,s,
```

```
1.439000640000E+10,s,
1.754576640000E+10,s,
2.070144000000E+10,s,
2.701278720000E+10,s,
3.332422080000E+10,s,
3.490205760000E+10,s,

~SAC Remediation Card
  1,
2.303769600000E+09,s,Tc99, 4.500000000000E+00,m, 1.000000000000E+00,600-148,
```

The template file needs to appear under each site subdirectory (e.g., …/assessment/vadose/600-148). The template file contains all the information needed for the STOMP analysis. Table 3.8 shows a sample template file.

Table 3.8 Example File …/assessment/vadose/600-148/template_stomp.key

```
~Simulation Title Card
1,
216S-6,
name,
Pacific Northwest National Laboratory,
February 26 2001,
1 PM PST,
2,
Template 216S - For shallow disposal sites (e.g., Cribs, Burial Grounds)
200W Area (South), Cribs/Burial Grounds, Waste Chemistry Designation 6

~Solution Control Card
Restart,
Water Transport Courant,
1,
0.0,yr,1110.0,yr,1.0,s,1100.0,yr,1.5,8,1.0e-6,
4,hr,4,hr,10000,
0,

~Grid Card
Cartesian,
1,1,654,
0.0,m,39.74,m,
0.0,m,39.74,m,
0.0,ft,204@0.5,ft,200@0.2,ft,250@0.5,ft,

~Rock/Soil Zonation Card
4,
Hg(6H),1,1,1,1,525,654,
Hss(6H),1,1,1,1,405,524,

~Mechanical Properties Card
Hg(6H),2070,kg/m^3,0.166,0.166,,,Millington and Quirk,
Hss(6H),1760,kg/m^3,0.346,0.346,,,Millington and Quirk,

~Hydraulic Properties Card
Hg(6H),,,,,5.0e-3,hc cm/s,
Hss(6H),,,,,6.0e-3,hc cm/s,

~Saturation Function Card
Hg(6H),Nonhysteretic van Genuchten,0.083,1/cm,1.660,0.131,,
Hss(6H),Nonhysteretic van Genuchten,0.108,1/cm,2.111,0.086,,
```

```
~Aqueous Relative Permeability Card
Hg(6H),Mualem,,
Hss(6H),Mualem,,

~Solute/Fluid Interaction Card
1,
Tritium,Conventional Tortuosity Model,1.e-09,m^2/s,Continuous,1.0e20,yr,
0,

~Solute/Porous Media Interaction Card
Hg(6H),0.09,m,,,
Tritium,0.0,ml/g,1.0,
Hss(6H),0.203,m,,,
Tritium,0.0,ml/g,1.0,

~Initial Conditions Card
Gas Pressure,Aqueous Saturation,
5,
Gas Pressure,101325.0,Pa,0.0,1/m,0.0,1/m,0.0,1/m,1,1,1,1,1,654,
Aqueous Saturation ROCK,0.2230,,Hg(6H),
Aqueous Saturation ROCK,0.1288,,Hss(6H),

~Boundary Conditions Card
2,
Top,Neumann,Zero Flux,
1,1,1,1,654,654,1,
0.0,yr,-1.0,mm/yr,,,
Bottom,Dirichlet,Outflow,
1,1,1,1,1,1,1,
0.0,yr,101325.0,Pa,,,

~Output Control Card
1,
1,1,654,
1,1,yr,m,6,6,6,
6,
Aqueous Pressure,Pa,
Aqueous Saturation,,
Z Aqueous Vol,mm/yr,
Aqueous Courant Number,,
Solute Volumetric Conc.,Tritium,1/m^3,
Solute Source Integral,Tritium,,
0,
1,
Solute Volumetric Conc.,Tritium,1/m^3,

~SAC Release Plane Card
1,
0.0,yr,1,
```

The stochastic definition file (stochastic_stomp.key) needs to appear only under the vadose subdirectory. This file contains distribution information for each rock type for porosity, saturation, and the hydraulic properties. It also contains distribution information for each sorption class. A sample stochastic definition file is shown in Table 3.9:

Table 3.9 Example File …/assessment/vadose/stochastic_stomp.key

```
!=========================================================================
! Title and Random Number Generator Seed.
!=========================================================================
```

```
TITLE "STOMP stochastic definitions for SAC Rev. 0 Initial2 Assessment"
SEED 6749.24
DEBUG STOCHASTIC="stoch/stomp_stoch.out"

!-----------------------------------------------------------------------
! Hfs (Hanford fine sand) HYDRAULIC PROPERTIES
!-----------------------------------------------------------------------
POROSITY   ROCK="Hfs"   "TOTAL"            7   3.65E-01   5.00E-02
   TRUNCATE   2.39E-02   9.40E-01
POROSITY   ROCK="Hfs"   "DIFFUSIVE"        7   3.65E-01   5.00E-02
   TRUNCATE   3.13E-02   9.40E-01
SATURATION ROCK="Hfs"   "RESIDUAL"         7   8.57E-02   4.64E-02
   TRUNCATE   3.24E-02   9.83E-01
SATURATION ROCK="Hfs"   "AIR"  UNIT="1/cm" 9  -3.79E+00   1.37E+00
   TRUNCATE   1.03E-01   9.90E-01
SATURATION ROCK="Hfs"   "EXPONENT"         7   2.36E+00   9.81E-01
   TRUNCATE   1.18E-01   9.90E-01
HYDRAULIC  ROCK="Hfs"   UNIT="hc cm/s"     9  -7.45E+00   2.05E+00
   TRUNCATE   1.00E-01   9.82E-01

!-----------------------------------------------------------------------
! Hcs (Hanford coarse sand) HYDRAULIC PROPERTIES
!-----------------------------------------------------------------------
POROSITY   ROCK="Hcs"   "TOTAL"            7   3.36E-01   8.06E-02
   TRUNCATE   4.22E-02   9.88E-01
POROSITY   ROCK="Hcs"   "DIFFUSIVE"        7   3.36E-01   8.06E-02
   TRUNCATE   7.93E-02   9.88E-01
SATURATION ROCK="Hcs"   "RESIDUAL"         7   7.62E-02   4.81E-02
   TRUNCATE   5.66E-02   9.90E-01
SATURATION ROCK="Hcs"   "AIR"  UNIT="1/cm" 9  -2.53E+00   1.02E+00
   TRUNCATE   1.00E-01   9.90E-01
SATURATION ROCK="Hcs"   "EXPONENT"         7   2.05E+00   7.15E-01
   TRUNCATE   1.37E-01   9.90E-01
HYDRAULIC  ROCK="Hcs"   UNIT="hc cm/s"     9  -6.28E+00   1.96E+00
   TRUNCATE   1.00E-01   9.60E-01

!-----------------------------------------------------------------------
! Sorption Class (1H) [Kd (mL/g)]
!-----------------------------------------------------------------------
SORPTION ANALYTE="H3"    CLASS="6H"   1   0.000
SORPTION ANALYTE="Sr90"  CLASS="6H"   9    2.303   0.297
   TRUNCATE  0.010  0.914
!-----------------------------------------------------------------------
! Sorption Class (1I1) [Kd (mL/g)]
!-----------------------------------------------------------------------
SORPTION ANALYTE="H3"    CLASS="1I1"  1    0.000
SORPTION ANALYTE="Sr90"  CLASS="1I1"  9    3.091   0.338
   TRUNCATE  0.010  0.990

!-----------------------------------------------------------------------
EFFDIF - Defines STOMP effective diffusion option.
!-----------------------------------------------------------------------
EFFDIF ANALYTE="H3"       "Conventional"
EFFDIF ANALYTE="Sr90"     "Conventional"

!-----------------------------------------------------------------------
```

```
! SOLPAR - Defines STOMP solute partition option.
!-----------------------------------------------------------------------
SOLPAR ANALYTE="H3"       "Continuous"
SOLPAR ANALYTE="Sr90"     "Continuous"

!-----------------------------------------------------------------------
! End of stochastic input for the STOMP model in SAC.
!-----------------------------------------------------------------------
END
```

3.4.1.6 Typical Steps to Prepare STOMP Input for SAC

The user who prepares the vadose zone flow and transport inputs for SAC must be a competent user of the STOMP simulator (White and Oostrom 2000). To prepare for vadose zone flow and transport calculations in SAC, the user must take the following steps:

1. Ensure that the REALIZATION, ANALYTE, AGGREGATE, RECHARGE, and REMEDIATE keywords, at minimum, are completed in the ESD file. These collectively define the essential inputs to be used in creating STOMP input files for the vadose zone flow and transport simulations.

2. Prepare the /*assessment*/vadose subdirectory structure. To do this, modify the ESD file to include the following active keywords:
   - CREATEDIR compute
   - MODULE STOMP FLOW TRANSPORT compute

3. Run the ESP program with the modified ESD file; it will create the necessary /assessment/vadose subdirectory structure. (Note that ESP need only be run once with the CREATEDIR for the STOMP module.)

4. Prepare the following template files:
   - stochastic_stomp.key – a single file containing stochastic parameters for all STOMP simulations in the assessment
   - template_stomp.key – one vadose zone *site template* for each site identified by and AGGREGATE keyword in the ESD file for the assessment

5. Copy the template files to the appropriate location in the assessment directory:
   - /*assessment*/vadose/stochastic_stomp.key
   - /*assessment*/vadose/*site*/template_stomp.key

6. Create all STOMP input files for the assessment. To do this in advance of a production run, modify the ESD file to include the following active keywords:
   - IOONLY compute
   - MODULE STOMP FLOW TRANSPORT compute

7. Run the ESP program with the modified ESD file; it will create all of the stochastic inputs for STOMP simulations and generate all the input files for the assessment.

8. Run either the full assessment or the vadose zone portion of it. Note that all VADER inputs (Section 5.5.1) must also be ready before the vadose zone transport calculations.

3.4.1.7 Preparing STOMP Templates

A STOMP input file template is a fully functional STOMP input file used to direct a single STOMP simulation for a given vadose zone site. SAC will use the template as a base from which to prepare an individual realization of STOMP input files for a site. The process is illustrated in Figure 3.3. It is good practice to prepare the STOMP input file templates using mean values of those parameters that will be treated stochastically in a SAC assessment. This way, testing the template in advance will give you a reasonably good indication of how the simulation will perform for any vadose zone site or sites using a given template.

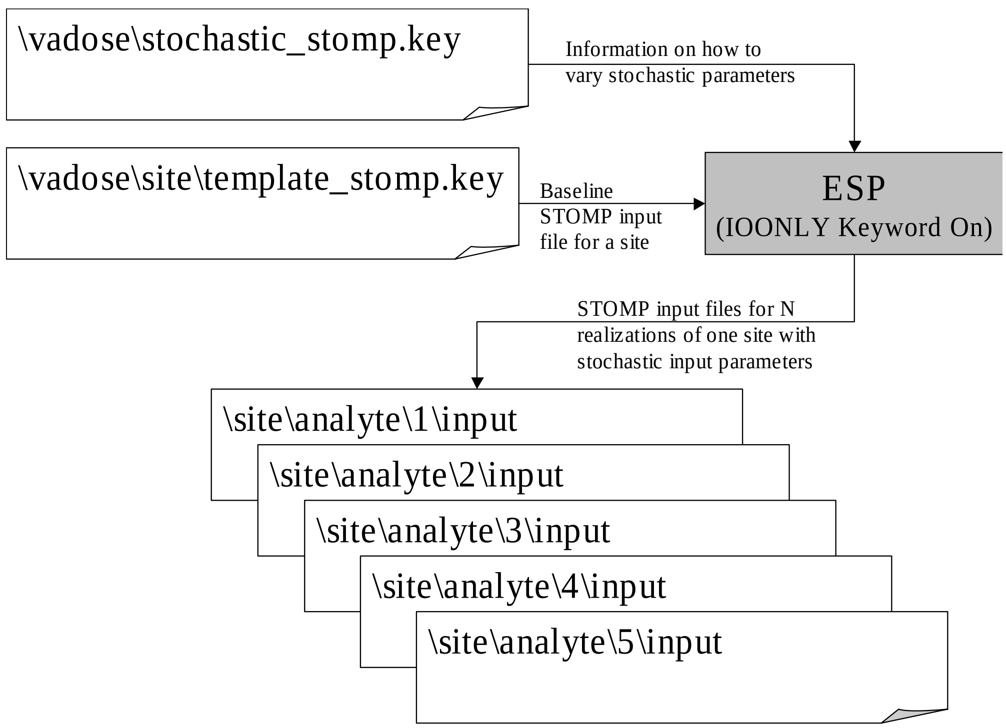

Figure 3.3 Using the ESP to Generate STOMP Input Files

The ESP is used to combine information from the stochastic_stomp.key and template_stomp.key input files to generate individual STOMP input files for each stochastic realization. There are certain necessary restrictions placed on the preparation of the STOMP input file template to make it suitable for inclusion in SAC:

- The input file ~Solution Control card must specify a “Water Transport” or “Water Transport Courant” mode
- The input file ~Grid card must be one-dimensional (or two-dimensional axi-symmetric) and use the Cartesian coordinate system
- The input file must include one dilute species (no more, no less); the actual species included and its properties are not important, as these will be overwritten in a SAC assessment.

These restrictions are imposed by limiting assumptions of the SAC, consistent with the SAC software requirements. The STOMP simulator itself is capable of simulating much more sophisticated problems, but the additional capability is not supported within the SAC framework at present.

Ideally, a STOMP template is simulated in a standalone fashion. In fact, it is highly recommended that a test liquid volume and dilute species soucrce keyword card be used to test any template before it is actually included in a SAC assessment. Such testing will ensure that the template is fully functional (does not include input formatting errors) and will considerably reduce the problems encountered in conducting a SAC assessment.

3.4.1.8 Suggestions for Managing Vadose Zone Site Templates

A typical SAC assessment will involve hundreds of vadose zone sites. While an individual template could be prepared for each of these, it is more common that you will take advantage of similarities between vadose zone sites in the same general area and of the same general nature to prepare *base templates* that apply to more than one vadose zone site.

It is recommended that a collection of base templates be prepared and tested in a separate disk location from any assessment and that a batch file be used to copy the base templates into site template locations in a SAC assessment subdirectory structure. This approach will protect the base templates from any activity in the assessment itself and will make standalone testing of the base templates easier to manage.

A simple example will be used to illustrate the general recommended approach. Consider an assessment with five vadose zone sites, of which the first three (X, Y, and Z) are very alike (same general stratigraphy and hydraulic properties with waste disposed at same general depth), while the remaining two (A and B) are also alike but different from X, Y, and Z. Thus, two base templates can be used to represent these five individual sites. Templates would then be prepared in the directory structure shown in Figure 3.4:

```
/vz_templates
 |
 |-- /baseXYZ
 |    |
 |    |-- template_XYZ
 |    |-- /test
 |         |
 |         |-- input
 |         |-- output
 |-- /baseAB
      |
      |-- template_AB
      |-- /test
           |
           |-- input
           |-- output
```

Figure 3.4 Sample Template Directory Structure

In this case, make test versions of the templates under /test subdirectories, and when satisfied, place a final version of the each base template in the locations …./baseXYZ/input_baseXYZ and …/baseAB/input_baseAB. Next, a site template must be placed in each location for this "simple" assessment. To do this, prepare a batch file containing commands similar to the following:

```
cp /vz_templates/baseXYZ/template_XYZ /simple/vadose/X/template_stomp.key
cp /vz_templates/baseXYZ/template_XYZ /simple/vadose/Y/template_stomp.key
cp /vz_templates/baseXYZ/template_XYZ /simple/vadose/Z/template_stomp.key
cp /vz_templates/baseAB/template_XYZ  /simple/vadose/A/template_stomp.key
cp /vz_templates/baseAB/template_XYZ  /simple/vadose/B/template_stomp.key
```

This batch file, when executed, will place a copy of the desired base templates for sites X, Y, and Z in the appropriate location in the assessment subdirectory structure with the required site template file name "template_stomp.key'. Similarly, a copy of the base template file to be used for both sites A and B will be placed in the appropriate locations for those sites. This method efficiently propagates two templates into five vadose zone sites in our assessment.

The advantages to managing templates in the manner described above include the following:

- Easy to track testing of base templates.
- Base templates are preserved apart from any assessments and can be "pegged" to the data rather than the assessments.
- The base templates can easily be reused for several assessments by adapting the batch file used to propagate base templates to site templates.
- If a correction is needed for a base template, it is easy to rerun the batch file after making the correction to ensure that site templates are updated for all sites that use a corrected base template.

3.4.1.9 Input Files for VZDROP

Vadose zone transport results computed by STOMP are used as boundary conditions for the groundwater transport, modeled using the CFEST code. The VZDROP code takes results from STOMP and inserts them into the proper input files for CFEST. ESP will create the needed VZDROP input file (vzdrop.key) to run VZDROP. Section 6.0 further describes VZDROP.

3.4.1.10 Input and Control for Groundwater Transport Using CFEST

Once VADER and STOMP have completed their modeling of the vadose zone release and transport for all sites, analytes, and realizations, the groundwater transport module (CFEST) can be run. CFEST requires many input files, which need to be placed in the appropriate subdirectories. The user who prepares the groundwater flow and transport inputs for SAC must be a competent user of the CFEST simulator (CFEST 2000). ESP will simulate CFEST stochastic variables and place them in the appropriate files. Table 3.10 provides a sample CFEST control file.

Table 3.10 Sample CFEST Control File …/assessment/cfest/H3/cfest.key

```
'cfest_001',  Important this version number should match code version
0,nprint_ctl
'hanford.qa'
'bincf/cfest'                      Line 1 - path and name for binary & ascii output files
input.lp1                          Line 2 - LPROG1 input file name
input.l3i                          Line 3 - LPROG3I input filename
input.hbc                          Line 4 - time dependent head B.C. file name
concbc.dat                         Line 5 - time dependent head B.C. file name
MKD                                Line 6 - Model units (valid options are "FPS" or "MKS")
0, 0, 0, 1, 0, 0, 1, 1,            Line 7 - print options:
1,0,                               Line 8 - NTSUBD,INTERM Time step sub-division
0,0,                               Line 9 - ITT (initial or restart time step),ITSTOP(0=ALL)
0, 1, 1, 0,                        Line 10 - NHEAD_NEW,IOPT_HEAD,NCYES,NDYES(1=y,0=n)
2, 4,                              Line 11 - NDIGH,NDIGC ** digits in head & conc output
1.00,                              Line 12 - Implicit (=1.0) CRANK-NICOLSON (0.5-1.0)
1.00,                              Line 13 - UPSTRM (0.0 to 1.0) Upstream factor
1.0,4492.47,                       Line 14 - RETARD,HALF-LIFE in simulation units
1,                                 Line 15 - NCROSS (0=norm,1=enhanced,2=same as 1 except exit,
3=cross deriv. to 0
2,0.00100,                         Line 16 - ITRHO (Max. rho iter.), RHO convergence (0.01-.0001)
1.E-06,1.E-05,                     Line 17 - RPARM1_H, RPARM1_C iterative convergence factors
1,                                 Line 18 - ITRANS (0=confined, 1=unconfined, 2=both)
15,0.750000E-01,0.090000,2,        Line 19 - IWT_TIMES,ERR_TRANS,THICK_MIN,IWRITE_DRY
0,                                 Line 20 - IOPT_K (0=normal,1=unsat_k.tab)
1,0,0.100000E-05,0.100000E-05,     Line 21 - IOPT_M,IM_ERR,FL_NODAL_ERRMIN, SOL_NODAL_ERRMIN
1.00000,                           Line 22 - FACT_VOL factor for printing fluid volume
cu. meters                         Line 23 - FUNITS_VOL text string for fluid volume output
Kg                                 Line 24 - Mass units
day                                Line 25 - time units(e.g. sec, days)
1.00000,                           Line 26 - FACT_AREA used for printing only
sq. meters                         Line 27 - area Units used for printing only
*----------------------------------------------------------------------------------------
* NSPGC ACCELRATOR (BCGS is most tested. LANMIN=used if BCGS not converged, others see manual)
* IACCEL_C=0(BCGS),1(LANMIN),2(CGCR), 3(LANDIR), 4(OMIN), 5(ODIR)
* ITMAX_Factor=1-5 (5 used as default - user can include lower values if system locks in)
*----------------------------------------------------------------------------------------
1,5,
```

The only variable being simulated for CFEST is the retardation coefficient for each analyte. Table 3.11 provides a sample stochastic definition file.

Table 3.11 Sample Stochastic Definition File for CFEST …/assessment/cfest/stochastic.key

```
! CFEST stochastic keyword file (stochastic.key)
!
TITLE "CFEST Stochastic Definition Keyword File"
SEED 3434.34
DEBUG STOCHASTIC="stoch/cfest_stoch.out"
!
!--- Sorption Class F1 (Groundwater)
RETARDATION ANALYTE="H3"     1   1.0             compute
RETARDATION ANALYTE="Tc99"   1   1.0             compute
RETARDATION ANALYTE="I129"   9   1.6054  1.0306  compute
RETARDATION ANALYTE="U238"   9   3.2149  1.6418  compute
RETARDATION ANALYTE="Sr90"   9   5.0752  1.3345  compute
RETARDATION ANALYTE="Cs137"  9   7.7774  0.6512  compute
RETARDATION ANALYTE="Pu239"  9   7.3721  0.7417  compute
RETARDATION ANALYTE="CCl4"   9   0.9517  0.2609  compute
RETARDATION ANALYTE="CrVI"   1   1.0             compute
!
END
```

3.4.1.11 Location of Groundwater Flow and Transport Files in SAC

In SAC, the groundwater is modeled separately for each analyte and realization. SAC defines a special subdirectory, always named /cfest, to manage groundwater calculations. The structure of the /cfest subdirectory is depicted in **Figure 3.5**.

```
.../assessment/cfest
   │
   ├stochastic.key
   └.../cfest/Analyte-Flow
        │
        ├cfest_runs.out
        ├cfest.key
        ├cfest.l3i
        ├cfest.lp1
        ├input.hbc
        ├ZTOP.DAT
        ├/cfest/analyte/bincf
        ├/cfest/analyte/cf_tmpbinary
        └/cfest/analyte/realization
             │
             ├cfest_runs.out
             ├cfest.ctl
             ├input.hbc
             ├input.l3i
             ├input.lp1
             ├ZTOP.DAT
             ├/cfest/analyte/realization/bincf
             ├/cfest/analyte/realization/cf_tmpascii
             └/cfest/analyte/realization/cf_tmpbinary
```

Figure 3.5 Structure of the /cfest Subdirectory

In the Figure 3.5 illustration, the /cfest subdirectory structure shows fixed (hard-coded) names in bold and variable names in plain text. For example, "**/cfest**" is literal, but "/analyte" could be /CCl4 or /nasty, where the names of analytes are defined by the ANALYTE keyword in the ESD keyword input file. Similarly, realizations are numbered, and the number of digits used depends on the total number of realizations. Thus, for a 25-realization run /realization could be /01, /02, or /25, but for a five-realization run it could be /1, /2, or /5. The /analyte and /realization directories are repeated for as many sites, analytes, and realizations as are specified in the ESD keyword input file.

Figure 3.5 also indicates the appropriate location of critical input files for the groundwater transport module. The first of these is the stochastic.key file for CFEST runs that defines the values of retardation, treated stochastically in the groundwater in a SAC assessment. When ESP is run to create groundwater input files, the actual values for each realization of retardation are reported to the stochastic.rpt file in the same directory level. The user must prepare the stochastic.key file and place it at the /assessment/cfest subdirectory level before using ESD to create groundwater input files.

Other input files for groundwater transport are the templates used to provide the pre-computed CFEST flow solution and baseline input files. A full template set is placed at each /assessment/cfest/analyte subdirectory level and consists of the following:

- cfest_runs.out – A history file created by CFEST during the flow solution; this will be the starting file for additional history output by CFEST during the transport solutions.
- cfest.key – A template for the cfest.ctl file used to control a CFEST simulation. This file must provide the specifics for controlling a transport-only simulation of CFEST. The actual values of retardation and half-life are not important, as ESD will replace these when preparing cfest.ctl files for each realization for the analyte, but all other inputs will be used as provided in the cfest.key file.
- cfest.lp1 – A cfest .LP1 file designed for this analyte.
- cfest.l3i – A CFEST .L3I file designed for this analyte. Values of nodal concentration or dry mass injection source/sink strengths are unimportant, as these will be overwritten in a SAC assessment by the VZDROP code. However, all other values in the .L3I file will be used as provided.
- input.hbc – A file created by CFEST during the flow solution; this will be used "as-is" by the transport solution.
- ZTOP.DAT – A file only needed if CFEST is used in 2D mode; however, ESP at present still expects this file to be present (even though it is not used in a 3D simulation), so a placeholder file with this name must still be provided for 3D simulations.
- /bincf – A copy of the /bincf subdirectory and all binary files therein created by CFEST in the flow solution.
- /cf_tmpbinary – A copy of the /cf_tmpbinary subdirectory and all binary files therein created by CFEST in the flow solution.

Groundwater flow is not treated stochastically in STOMP (only groundwater transport has a stochastic aspect). Hence, it is unnecessary and inefficient to simulate groundwater flow inside the SAC framework. Moreover, CFEST flow simulations for a Hanford Site problem take a substantial length of computer time to solve (on the order of days) and would be identical for each analyte (lacking a stochastic aspect). Therefore, the SAC software presumes groundwater flow will be pre-computed and the flow solution provided as an input to SAC. Typically, the user should set up and simulate the flow solution outside the /assessment directory structure and then copy the needed files from the completed flow solution into the /assessment/cfest directory structure. A script or batch file can be extremely useful to manage this process, particularly when a single flow solution will be used for two or more analytes. For example, if a flow solution is prepared for unretarded analytes Tc99 and H3 in directory /home/DATA/Database.2/groundwater/Flow.Fast, and for retarded analyte Sr90 in /home/DATA/Database.2/groundwater/Flow.Slow, one could use the script shown in Table 3.12 to populate the Median2 Assessment directory (/home/ANALYSIS/Median2) with CFEST flow solutions. The file names on the copy commands in the table are so long that they wrap to the second line. The syntax for the copy command is actually "cp file1 file2", all on the same line.

Table 3.12 Sample script file to populate an assessment with CFEST flow solutions

```
#
# H3
rm -R /home/ANALYSIS/Median2/cfest/H3/bincf/*
rm -R /home/ANALYSIS/Median2/cfest/H3/cf_tmpbinary/*
cp /home/DATA/Database.2/groundwater/Template.Fast/cfest.key
/home/ANALYSIS/Median2/cfest/H3-Flow/cfest.key
cp /home/DATA/Database.2/groundwater/Template.Fast/cf_runs.out
/home/ANALYSIS/Median2/cfest/H3-Flow /cf_runs.out
cp /home/DATA/Database.2/groundwater/Template.Fast/cfest.lp1
/home/ANALYSIS/Median2/cfest/H3-Flow /cfest.lp1
cp /home/DATA/Database.2/groundwater/Template.Fast/cfest.l3i
/home/ANALYSIS/Median2/cfest/H3-Flow /cfest.l3i
cp /home/DATA/Database.2/groundwater/Template.Fast/input.hbc
/home/ANALYSIS/Median2/cfest/H3-Flow /input.hbc
cp /home/DATA/Database.2/groundwater/Template.Fast/ZTOP.DAT
/home/ANALYSIS/Median2/cfest/H3-Flow /ZTOP.DAT
cp /home/DATA/Database.2/groundwater/Template.Fast/bincf/*
/home/ANALYSIS/Median2/cfest/H3-Flow /bincf/
cp /home/DATA/Database.2/groundwater/Template.Fast/cf_tmpbinary/*
/home/ANALYSIS/Median2/cfest/H3-Flow /cf_tmpbinary/
#
```

3.4.1.12 Typical Steps to Prepare CFEST Input for SAC

To prepare for groundwater transport calculations in SAC, the user must take the following steps:

- Ensure that, at a minimum, the REALIZATION and ANALYTE keywords are completed in the ESD file. These collectively define the essential inputs to be used in creating CFEST input files for the groundwater transport simulations.
- Prepare the /*assessment*/cfest subdirectory structure. To do this, modify the ESD file to include the following active keywords:
  - CREATEDIR compute
  - MODULE CFEST compute
- Run the ESP program with the modified ESD file; it will create the necessary /assessment/cfest subdirectory structure. (Note that ESP need only be run once with the CREATEDIR for the CFEST module.)
- Prepare the following template file: stochastic.key – a single file containing stochastic parameters for retardation for all CFEST simulations in the assessment.
- Prepare flow solutions appropriate to each analyte to be simulated. Note that one flow solution may be used for more than one analyte, if appropriate (see script example in Table 3.12).
- Copy the completed flow solution files to the appropriate location in the assessment directory (a script or batch file is highly recommended for this step; see example in Table 3.12).

- Create all CFEST input files for the assessment. To do this in advance of a production run, modify the ESD file to include the following active keywords:
  - IOONLY compute
  - MODULE CFEST compute
- Run the ESP program with the modified ESD file; it will create all of the stochastic inputs for CFEST simulations and generate all the ‘cfest.ctl’ files for the assessment.
- Run either the full assessment or the groundwater transport of it.

3.4.1.13 Preparing CFEST Templates

CFEST input file templates are merely the completed flow solution files but with a new cfest.key file that specifies what the CFEST control file (‘cfest.ctl’) will be for a transport-only solution. ESP will use the cfest.key template as a base from which to prepare individual realizations of CFEST cfest.ctl control files for each analyte and realization. Figure 3.6 illustrates the process. It is good practice to prepare the CFEST input file templates using the mean value of retardation, which will be treated stochastically in a SAC assessment. This way, testing the template in advance will give you a reasonably good indication of how the simulation will perform for any groundwater case using a given template.

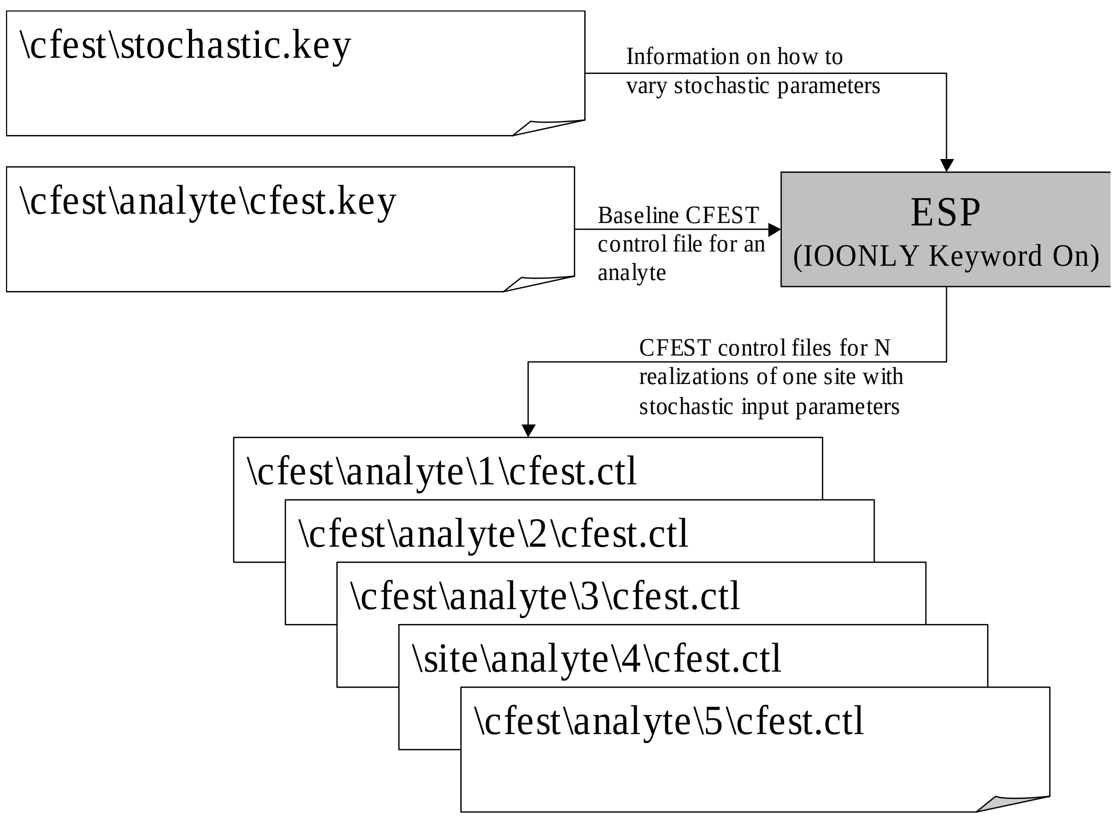

Figure 3.6 Using the ESP to Generate CFEST Control Files

Table 3.13 provides recommended parameter values for the cfest.ctl file for pre-computed flow solutions and for SAC transport solutions.