PNNL-15915 Rev. 1

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

CFEST
Coupled Flow, Energy & Solute Transport
Version CFEST005
User's Guide

V. L. Freedman
Y. Chen
A. Gilca
C. R. Cole
S. K. Gupta

July 2006

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

DISCLAIMER

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by

BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

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.

CFEST
Coupled Flow, Energy & Solute Transport
Version CFEST005
User's Guide

V. L. Freedman
Y. Chen
A. Gilca
C. R. Cole
S. K. Gupta[(a)]

July 2006

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99352

(a) CFEST , Inc., Irvine, California

Summary

The CFEST (Coupled Flow, Energy, and Solute Transport) simulator described in this User's Guide is a three-dimensional finite-element model used to evaluate groundwater flow and solute mass transport. Confined and unconfined aquifer systems, as well as constant and variable density fluid flows, can be represented with CFEST. For unconfined aquifers, the model uses a moving boundary for the water table, deforming the numerical mesh so that the uppermost nodes are always at the water table. For solute transport, changes in concentration of a single dissolved chemical constituent are computed for advective and hydrodynamic transport, linear sorption represented by a retardation factor, and radioactive decay. Although several thermal parameters described in this User's Guide are required inputs, thermal transport has not yet been fully implemented in the simulator. Once fully implemented, transport of thermal energy in the groundwater and solid matrix of the aquifer can also be used to model aquifer thermal regimes.

The CFEST simulator is written in the FORTRAN 77 language following American National Standards Institute (ANSI) standards. Execution of the CFEST simulator is controlled through three required text input files. These input files use a structured format of associated groups of input data. Example input data lines are presented for each file type, as is a description of the structured FORTRAN data format.

Detailed descriptions of all input requirements, output options, and program structure and execution are provided in this User's Guide. Required inputs for auxiliary CFEST utilities that aide in post-processing data are also described. Global variables are defined for those with access to the source code. Although CFEST is a proprietary code (CFEST, Inc., Irvine, CA), Pacific Northwest National Laboratory[(a)] retains permission to maintain its own source, and to distribute executables to Hanford subcontractors.

(a) Pacific Northwest National Laboratory is operated by Battelle for the U.S. Department of Energy under Contract DE-AC05-76RL01830.

Contents

Figures

Tables

1.0 Introduction

CFEST, which is an acronym for Coupled Flow, Energy and Solute Transport, is a three-dimensional finite-element computer program that simulates fluid movement and solute and energy transport in fully saturated porous media. CFEST can simulate flow and transport as either steady state or transient. Over the past decade, the CFEST simulator has been used to analyze groundwater flow and solute transport at the U.S. Department of Energy (DOE) Hanford Site. This user's guide has been written in support of the ongoing modeling efforts at the Hanford Site.

This document is a guide is to the model structure, input file requirements, and output file options used in the CFEST simulator as well as CFEST utilities. This document is a companion to Freedman et al. (2005), which described the model theory, solution methods, and solver descriptions. Together, this document and Freedman et al. (2005) update and replace the previous manuals (Gupta et al. 1987, 1997).

This document is designed to familiarize the user with CFEST and CFEST utilities. This document is not designed to assist the user in conceptual model development, the first step in translating a description of the physical system into a computational domain.

Section 2 of this User Guide outlines program design and concepts. This is followed by a description of major input variables. In Section 3, the input file descriptions are presented. Because CFEST requires multiple input files, input file descriptions are subdivided by type. Input options for each file type are presented in tabular format, so that the user, at-a-glance, has access to all input file options. Because CFEST input requirements include fixed formats for all entries, the FORTRAN format for each entry is also provided, as well as examples for each input line. Program execution is presented in Section 4, where assembling the source code, creating an executable, and executing the simulator are described. Directions for performing restarts on intermediate results using altered or unaltered descriptions of the system are also presented. Section 5 describes CFEST output files, and Section 6 presents CFEST utilities used to analyze flow and transport results. Cited references are presented in Section 7.

Although the input structure and governing equations for heat transport have been implemented in CFEST, the solver has not yet been updated for heat transport in the CFEST simulator. Hence, this guide presents the required inputs associated with thermal transport, even though, at the time of publication, nonisothermal flow simulations cannot be performed. In subsequent revisions of this user's guide, thermal transport, as well as flow and solute transport, will be updated according to the current state of implementation within the CFEST simulator.

2.0 Program Concepts and Design

CFEST is a finite-element code developed for three-dimensional hydrologic flow, heat transport, and single-constituent transport in subsurface confined and unconfined aquifers. CFEST simulates three-dimensional groundwater flow, and contaminant transport in groundwater. The source code is written in FORTRAN77, and utility programs that accompany CFEST are written in either FORTRAN or ANSI C.

CFEST was first developed in the late 1970s, and the code has continued to be developed to the present date. Given its long development history and its programming roots in the FORTRAN77 language, CFEST has been subject to numerous upgrades, as well as being left with the remains of its original programming structure. For example, when the direct solvers were replaced with more robust and efficient iterative equation solvers, the solver routines were never integrated with the heat transport equations because the energy component was considered to have limited use in routine groundwater and transport applications. Because the input structure of the code remained unchanged, input parameters relating to heat transport are still required in the current version of the code.

With the "legacy" nature of the code, CFEST readability of the source code can be limited. Although several comments appear throughout, large sections of the code are still not documented. A modular design to the code has been implemented through a large group of common blocks and minimal subroutine and function arguments.

2.1 Flow Path

Four modules are required to run a simulation with the CFEST software: *lprog1, lband, lprog3i,* and *lprog3.* The *lprog1, lband,* and *lprog3i* are preprocessors that set the grid, solver arrays, and initial conditions, respectively. These modules write binary files that are read by the main code, *lprog3*.

The general flow path for the main program in CFEST, *lprog3*, is variable initialization, iteration on flow and transport solutions, and closeout. Figure 2.1 is a flow chart showing the main flow path. The initialization component of the program is executed once during a simulation. The iteration component of the program contains a pair of nested loops, an outer loop for time stepping and an inner loop for iteration when the flow and transport solutions are coupled.

CFEST includes both coupled and decoupled approaches for solving the flow and solute transport equations. With the coupled approach, the fluid density term provides a link between the two governing equations. In this approach, solutions to the flow and transport equations are repeated, and concentrations and densities are updated within each time step until the maximum difference in fluid density of each element is less than a user-specified value.

The program terminates when the time-stepping loop has been completed. Closure routines are executed when the simulation has been completed.

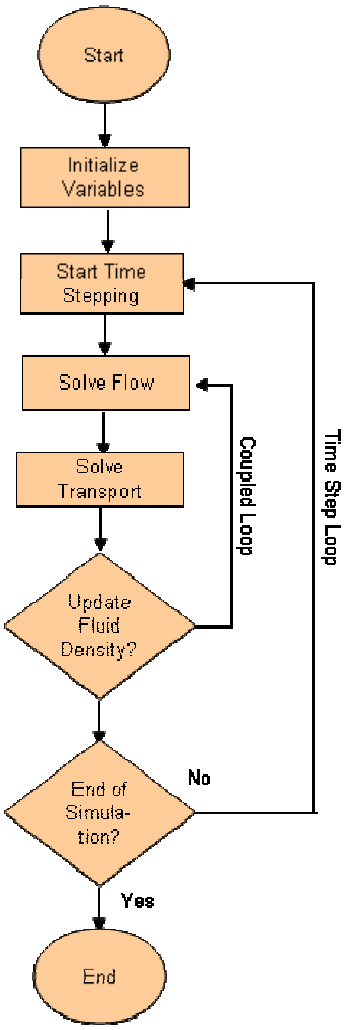

Figure 2.1. CFEST Flow Chart

2.2 Variables

The principal variables in CFEST described in Table 2.1 are defined in common blocks appearing at the beginning of each subroutine. The use of common blocks reduces the number of arguments that are passed between routines, and increases code readability due to consistent variable names. Variable names generally follow the intrinsic protocols for FORTRAN77, where integer variables begin with letters in the range *I – N*, and real variables begin with letters in the ranges *A – H* and *O – Z*.

Field variables such as head, density and concentration are defined at each node in the domain and are stored in vectors. Properties of the porous media such as hydraulic conductivity and porosity are also stored as vectors corresponding to the elements in the domain. Flux variables, which are also stored in one-dimensional arrays, are defined at element boundaries.

A list of global variables and their definitions is shown in Table 2.1. Because CFEST only simulates isothermal flow but still retains the input structure for heat flow, only those variables that relate directly to input appear in Table 2.1.

Table 2.1. Description of Global Flow and Transport Variables in CFEST

Global Variable	Definition
A(NP_UNK*NP_STF+10)	Coefficient matrix (stored as a vector) for head and concentration
ADD(8)	Convective nodal flow coefficients for an element with outflow (i.e., NSEEP>0)
ADDT(64)	Diffusive nodal mass flux coefficients for an element with outflow (i.e., NSEEP>0)
AHELD(NP_HMX*NP_STF)	Held node coefficient matrix for head, pressure, and concentration
ALPHAL(NP_LAYER)	Longitudinal dispersivity length
ALPHAT(NP_LAYER)	Transverse dispersivity length
AREA_NODE(NP_NUSER)	Total area associated with each node
B(NP_UNK)	Right B vector of AX=B for head, pressure and concentration
BBIVFF(NP_NFLX)	Stores BIVF values
BHELD(NP_HMX)	Stores B (AX=B) values of held node
BIV(NBBPTC)	Held node value for steady state; change in held node per unit time for transient. Stores head, pressure, and concentration data
BIVC(NBBPTC)	The value for the held concentration node
BIVF(NBBPTC)	Integrated flow volume or switch for dry mass injection
BIVFC(NP_NFLX)	Concentration of injection fluid
BIVH(NP_HH)	Held heads
BOV	Constant thickness of overlying rock
BOVER(NP_BUR)	Thickness of overburden by node
BUN	Constant thickness of underlying rock
BUNDER(NP_BUR)	Thicknesses of underburden by node
BWRN	Resident fluid density
C0	Reference concentration at RHO0+B334
CAPOB	Heat capacities per unit volume of overlying rock strata
CAPUNB	Heat capacities per unit volume of underlying rock strata
CB(78)	CB is used as common equivalence to dimensioned variable mass terms
CCC	The concentration to density ratio
CCCMAS	The concentration to mass ratio
CCONC(NP_NOD)	Concentration values for previous time step
CCRIV(NP_STRM)	Total mass flow at a stream node
CDCHOT	Total contaminant mass decay over time step
CDCMIT	Total over the time step for contaminant mass decay
CDCMOC	Cumulative contaminant mass decay
CDCRI	Input contaminant mass decay rate
CDCRO	Output contaminant mass decay rate
CDEL(NP_NOD)	Rate of concentration change between previous and current time step
CDFMOC	Cumulative outflow mass for dispersive-diffusive flux
CDFRO	Cumulative outflow rate for dispersive-diffusive flux
CDRHOC	Cumulative mass for dry mass extraction
CDRHOT	Total mass over time step for dry mass extraction

Table 2.1 (contd)

Global Variable	Definition
CDRMIC	Cumulative mass for dry mass injection
CDRMIT	Total dry mass injection for time step
CDRRI	Dry contaminant mass injection rate
CDRRO	Dry contaminant mass extraction rate
CELMIT	Total solute mass over time step for elemental inflow
CELMOC	Cumulative solute mass for elemental outflow
CELMOT	Total mass for elemental outflow over current time step
CELRI	Elemental solute source mass rate
CELRO	Elemental solute sink mass rate
CHNHIC	Cumulative solute mass for nodal inflow with held concentrations
CHNMIT	Total solute mass for nodal inflow over current time step for nodal inflow with held concentrations
CHNMOC	Cumulative solute mass for nodal outflow with held concentrations
CHNMOT	Total solute mass for nodal outflow over current time step for nodal inflow with held concentrations
CHNRI	Nodal inflow solute mass rate for held concentrations
CHNRO	Nodal outflow solute mass rate for held concentrations
CI(64)	Element matrix coefficient used for head, pressure and concentration
CINJ	Scalar nodal injection concentration
CMAX	Maximum concentration in domain
CMIN	Minimum concentration in domain
CMND(NP_LAYER)	The difference in mass between previous and current time step in each material (layer)
CMNI(NP_LAYER)	Total inflow mass in each material (layer)
CMNN(NP_LAYER)	Total mass in each material (layer) for current time step
CMNP(NP_LAYER)	Total mass in each material (layer) for previous time step
CNDHIC	Cumulative mass inflow through nodal source terms
CNDHIT	Total mass inflow through nodal source terms for current time step
CNDMOC	Cumulative contaminant mass outflow through nodal sources
CNDMOT	Total mass outflow through nodal source terms for current time step
CNDRI	Contaminant mass inflow rate through nodal sources
CNDRO	Contaminant mass outflow rate through nodal sources
CONC(NP_NOD)	Nodal concentration
CONC_FLUXQLINE(200)	Concentration along flux boundary
CONC_RES(NP_NOD)	Negative concentration values for correcting fluid density in MASSL subroutine
CONC_STOP(HP_NFLX)	Upper concentration limit for mass extraction
CONMAX	Maximum concentration
CONVRG	Convergence criterion for computing fluid density
CPR(NP_LAYER)	Heat capacity per unit volume of the rock matrix
CPW	Heat capacity of the resident aquifer fluid

Table 2.1 (contd)

Global Variable	Definition
CR(NP_LAYER)	Compressibility of the media
CRANK	The Crank-Nicolson coefficient
CRIV(NP_STRM)	River node mass flow for the current time step
CROCK_ELM(NP_ELM)	Compressibility (rock) of each element
CSUAI	Surface area for solute mass infiltration
CSUAO	Surface area for solute mass extraction
CSUHIC	Cumulative solute mass from surface infiltration
CSUMC_BH	Scalar for storing elemental source concentration
CSUMIC	Cumulative solute mass from surface infiltration
CSUMIT	Total solute mass from surface infiltration over current time step
CSUMOC	Cumulative solute mass outflow from surface extraction
CSURI	Solute mass rate from surface infiltration
CSURO	Solute mass rate from surface extraction
CSYMDT	Change in solute mass between current and previous time step
CSYMEC	Cumulative solute mass balance error
CSYMET	Error in solute mass balance between current and previous step
CSYMI	Initial solute mass in the domain
CSYMN	Current solute mass in the domain
CSYMNT	Solute mass in system at end of current time step
CSYMP	Total solute mass at previous time step
CSYMPT	Total solute mass between current and previous time step
CSYMRC	Cumulative solute mass balance error ratio
CSYMRT	Solute mass balance error ratio between current and previous time step
CTW	Coefficient of thermal expansion
DATETEXT*12	Date character string
DCS(NP_LAYER)	Maximum concentration that soluble material can contain (solubility limit)
DDATE*8	Date character string
DECAY	Half life
DECFRC	Solute mass remaining after decay
DELSAV	Saves change in time step
DELTAT	Time step
DENMAX	Maximum density
DETAIL	80 character title read from *lprog3i* input file title
DFS(NP_LAYER)	Fraction of material in the geohydrologic media which is soluble
DKS(NP_LAYER)	Salt dissolution rate constant
DMOLE	Molecular diffusivity, including the effects of tortuosity
DQXYZ(64)	Diffusive flux at each node of an element
DUU(8)	Element volume for each integration point
ELEM_AREA(NP_EUSER)	Element surface area

Table 2.1 (contd)

Global Variable	Definition
ELEM_DRYMASS(NP_DT,1000)	Elemental dry mass through optional file *dry_mass.elm*
ELEV(NP_SNOD)	Elevation at point of dry mass injection
ELMASS(NELEM)	Solute mass in element for current time step
EQUIVALENCE()	Mass balance terms to assist in initialization and cumulative totals using equivalence statements
ERR_TRANS	Convergence criterion for unconfined aquifer (maximum head difference between iterations)
ERRMAX	Maximum error
EZA(8)	Local integration y-coordinate
EZAZ(4)	Local integration z-coordinate for x-z model
FACT_AREA	Area factor used only for printing
FACT_FCWLND(100)	PARACODE FCWLND = factor well node
FACT_FLUXQLINE(500,2000)	Factor for adjusting flow line flux (PARACODE)
FACT_FLXQAREA(1500,20)	Factor for adjusting flow area flux (PARACODE)
FACT_VOL	Conversion factor for printing fluid volume
FCQSUR	Factor for converting surface flux (PARACODE)
FIFJOB(64)	Derivative of cumulative mass at local coordinate
FILE_CF0*40	Filename containing results for UCODE output
FILE_CNC*40	Filename containing concentration results for UCODE
FILE_EXT*40	Filename extension for UCODE interface
FILE_FLUXAREA*40	Filename for area fluxes
FILE_FLUXLINE*40	Filename for line fluxes
FILE_HED*40	Filename containing values of observed and simulated head (for UCODE)
FILE_NONE*20	Default filename for file that does not exist
FILE_PARA*40	PARACODE filename used with UCODE
FILE_PRE*40	.pre filename used with UCODE
FILE_RIV*40	River node filename used with UCODE
FILE_SUB*40	Parameter substitute filename used with UCODE
FILE_TPL*40	Template filename used with UCODE
FILENAME	Character string used for filename
FILEUCODE*40	.uni filename used with UCODE
FLUID_NODAL_ERRMIN	User-defined flow threshold errors
FLUXQLINE(200)	Total flux assigned to each line of flux
FLXDT	Delta time since the last flux record
FLXENE(NP_NSEX)	Elemental surface flux for fluid balance
FLXFLO(NBFPTC)	Nodal flux for fluid balance estimation
FLXMAS(NBFPTC)	Nodal flux for solute mass balance estimation
FLXQAREA(20)	Total flux assigned to each flux area
FLXTIM	Time at the end of the current time step
FMTPARA*60	Format for reading paracode (UCODE) file

Table 2.1 (contd)

Global Variable	Definition
FN*50	Character string used for filename
FNG(64)	Local x derivative at a given Gaussian point
FUNITS_AREA	Area units used for printing only
FUNITS_MASS	Mass units for .lp1 file
FUNITS_TIME	Time units for conductivity and flow
FUNITS_VOL	Test string for fluid volume output
GDVET(64)	Local coordinate eta derivative used in density dependent flow calculation
GDVSC(64)	Local coordinate sce derivative used in density dependent flow calculation
GIGJOB(64)	Integrated product of local y-derivative
GNG(64)	Local Y-derivative in local coordinates
GQPO(5)	Gaussian integration points
GQWT(5)	Gaussian integration weight
GRVITY	The factor used in estimating the hydraulic conductivity from intrinsic permeability and viscosity
H0	Reference head for density dependent fluid flow
H10_R(NP_NOD)	Head values used for interpolating observed head at current time step
H10_R_OLD(NP_NOD)	Head values used for interpolating observed head at previous time step
HALF_LIFE	Half-life for radioactive solutes in simulation units
HBINI(NP_LEAK)	Initial head at leakance nodes
HDEL(NP_NOD)	Delta head values at each internal node number
HEAD(NP_NOD)	Head values at each internal node number
HEAD_OLD(NP_NOD)	Head value when oscillation occurs in calculation of the phreatic surface
HEAD_SAV(NP_NOD)	Recent estimates of head
HEAD_STACK(100)	Head distribution in a vertical stack
HEAD_SIM(400000)	Simulated heads output for UCODE
HGEN(NP_HGR)	Head generation rate
HH_SAV(NP_OLD)	Saved head at old time step
HHEAD(NP_NOD)	Head values at previous time step
HIHJOB(64)	Local coordinate z derivative
HLDHFL(NP_HH)	Total cumulative fluid mass inflow and outflow associated with a held head node
HLDHMS(NP_HH)	Total cumulative solute mass inflow and outflow associated with a held head node
HLDMAS(NTBCC)	Total cumulative solute mass inflow and outflow associated with a held concentration node
HLEAK(NP_LEAK)	Specified head for leakance boundary condition
HNG(64)	Z derivative in local coordinates for 8 nodes and 8 integration points
HPIN	Hydraulic head at point of injection
HPOUT(NP_ESIN)	Pressure head of withdrawal.
IA(NP_UNK+100)	Starting index of each unknown in matrix for the flow equation
IBHELD(NP_HMX)	Starting index of each held node for matrix in the flow equation

Table 2.1 (contd)

Global Variable	Definition
ICONTINUE_LASTHEAD	Sets stopping criteria for defining phreatic surface
IDENT(LNPT)	Index that references the user surface node number.
IDENT(NP_NOD)	Starting index for internal surface nodes
IDRY_MASS	Integer switch to identify a given flux node as dry mass flux
IE	Starting element for which surface infiltration/evaporation is to be entered; also used as element counter
IEL_WT(NP_EUSER)	Identifier for confined or unconfined element
IELEM_DRYMASS(NP_DT,1000)	List of dry mass elements read from optional file
IELEM_FLXQAREA(500,20)	List of elements associated with area flux boundary conditions
IESURFQ_ZONE(100,1000)	Used in lp3_ucode for zonal surface Q assignment or estimation
IHIST	Option to save history of steady-state transport density related flows
IIEL_WT(NP_NUSER)	Switch identifying unconfined /confined element before start of new unconfined calculation
IIEND	A single element IE or the ending element number in a sequence of element numbers from IE to IIEND
IIPARM(12)	Inter parameter values of ITPACK
IIT	Initial time step or restart time step
ILAYERS(NP_NUSER)	Number of layers at each surface node
ILOG	Switch that identifies uniformly layered systems from variable layered system
IMASS_ERR	Option for checking error by substituting the solution and checking residual error
INCOL	Logical unit for system matrix (set to 11 for Head, 21=Temp, 31=Conc) based on solution loop
INDEX_ND(1000)	Internal node index for user surface node number
INITIAL	Switch that identifies constant or variable initial conditions
INODE_FLXQAREA(1500,20)	Internal node numbers for area fluxes through vertical faces
INTERM	Variable to control printing of debug statements for sub-time step; default is normally=0; >0 prints debug statements for the given total number of sub-time steps
INW_ITPACK	ITPACK integer work space =80*NP_UNK (number of unknowns)+100
IOPT	Switch that identifies variable time step subdivisions
IOPT_HEAD	Switch that sets the read for the head solution from a .b44 binary file.
IOPT_HSAVE	Switch that sets whether the head solution is saved to a .b44 binary file
IOPT_K	Unconfined system simulation option
IOPT_MASSFILE	Switch for saving solute mass data
IPARM(30)	Parameter values used in NSPCG solver
iparm_bcgs(25)	Values of IPARM for bcgs
iparm_lmin(25)	Values of IPARM for lmin
iparm_omin(25)	Values of IPARM for omin
IREC_B44	Counter for tracking record number in .b44 file

Table 2.1 (contd)

Global Variable	Definition
IRETARD	Switch that sets solute retardation
ISAVE_CONC	Switch used to save sub-timestep results
ISTEP	Counter used for tracking first loop through simulation
ISTREAM_OPT	Flag used for identifying read of stream boundary conditions
ISURE	Switch to include over and under burden calculations
IT	Time step number
IT_DRY	Counter for elemental dry mass injection defined in optional *dry_mass.elm* in current directory
ITER	Counter for number of iterations for density calculation
ITOTAL	Number of time steps
ITQIN	Switch identifying time constant or time dependent elemental sources
ITRANS	Switch used to identify aquifer types
ITRHO	Maximum number of iterations for density calculation
ITRIV	Time step for which new stage is specified when reading stream boundary conditions from file
ITSTOP	Stop time step for simulation
ITT	Start time step
IUNCONFINE(NP_ELM)	Unconfined elements using internal element number as index
IVERTDISP	Vertical dispersivity switch if optional *vertdisp.tab* is in the current work folder
IVIS	Switch for defining viscosity data set
IWKSP(INW_ITPACK)	Integer workspace for ITPACK
IWT_ITER	Integer counter for each water table convergence loop
IWT_PLAN	Maximum iterations for water table convergence
IWT_TIMES	Maximum iterations for elevation adjustment to head
JCOEF(NP_UNK*NP_STF*2)	Non-zero coefficients for each unknown head and concentration node
KKELEM(NP_ELM)	Options defining material properties associated with the material number
KTYPE	Switch defining hydraulic conductivity, intrinsic permeability, and viscosity
LAY(100)	Layer number at each node
LAYERS(LNPT)	User surface node number
LAYERS(NP_NOD)	Number of vertical layers at each node
LEAK(NP_LEAK)	Leakance nodes
LEAKND	Total number of leakance nodes
LIMITP	Number of elements to be printed to *lprog3.prn* file
LIMITT	Number of time steps to be printed to *lprog3.prn* file
LLMAX	Number of horizontal layers
LLTIME	Number of characters in TIME units descriptor FUNITS_TIME
LMASS	Number of characters in MASS units descriptor FUNITS_MASS
LNPT	The maximum user surface node number
LSUR	Total number of surface nodes

Table 2.1 (contd)

Global Variable	Definition
LTIME	Number of characters in TIME units descriptor FUNITS_TIME
LVOL	Number of characters in VOL units descriptor FUNITS_VOL
MATN	Number of materials
MAXLIM	Integer flag for identification of concentration exceeding maximum boundary condition or initial condition value
MAXMIN	Integer flag for identification of concentration below minimum boundary condition or initial condition value
MNTRY	Uniform materials option
MODULE	String that stores name of current program or subroutine
NANAME(NELEM)	User element number for internal element number
NBAND	Switch for tracing successful completion of *lband* execution
NBBPTC	Number of held nodes
NBFTPC	Number of nodes having nodal flux
NBOND	Value of NHBOND (head simulation loop) or NCBOND (concentration simulation loop)
NCBOND	Switch identifying held values for concentration boundary conditions
NCCYES	Flag for disabling transport calculations
NCROSS	Switch for defining dispersivity cross terms
NCV	The number of points used to define the viscosity-concentration relationship
NCYES	Switch that runs transport simulation
ND_PARA(6,100)	Matrix storing paracodes that are defined in *cf_ucode* and used in *lp3ucode*
NDIG	Number of digits used to report results
NDIGC	Number of digits used to report concentration output
NDIGF	Number of digits used to report flux output
NDIGH	Number of digits used to report flow output
NDIM	Switch that sets problem dimension
NDYES	Switch for salt dissolution
NELEM	Number of elements
NELQEX	Switch for elemental sink terms
NELQIN	Switch for elemental source terms
NEMAT(NP_ELM)	Material number for each element
NENAME(NP_ELM)	User element numbers
NENEXT(6)	Internal element numbers on six sides of element (top, bottom, left, right, front and back)
NENODT(NP_ELM)	Number of nodes in each element
NEW_HELD_BEGIN(100)	Switch for reading and translating starting time step from “on fly” for simulating release of large stage changes from a dam
NEW_HELD_STOP(100)	Switch for reading and translating ending time step from “on fly” for simulating release of large stage changes from a dam
NFLXQAREA_ZONES	Total number of area flux zones
NFLXQLINE_ZONE	Total number of line flux zones

Table 2.1 (contd)

NGAUSS	Number of integration points for each axis
NHBOND	Switch identifying held values for head boundary conditions
NHEAD_NEW	Switch for calculating new head
NHELD(NP_HMX*NP_STF)	Internal nodes associated in held matrix for flow through held nodes (positive index for unknown and negative index for held head nodes)
NHYES	Switch for writing flow data at each time step to .b07 binary file.
NLP3UCODE	Flag used to identify successful completion of *lp3ucode*
NNBASE(NP_ELM)	Flag used to identify elements at base of domain
NNCONC(NP_ELM)	Flag used to identify elements with held concentrations
NNDIS	Solution option for salt dissolution
NNENEXT(6,NP_ELM)	Internal element number for elements on each side of element (top, base, left, right, front, back)
NNHEAD(NP_ELM)	Flag to identify elements which have head held boundary conditions
NNHELD	Scalar flag to identify whether current element has held head boundary conditions
NNODBF(NP_NFLX)	Nodes with fluxes
NNODE(8)	Internal node number of an element
NNODE_CHEK(NP_NUSER)	Flag for unconfined surface nodes
NNODE_DRY(NP_NOD)	Flag for vertical stacks of dry nodes
NNPINCH(NP_ELM)	Flag for elements with pinched layers
NNQELEM(NP_ESOR)	Elements with source terms
NNQOUT(NP_NSEX)	Elements with sink terms
NNSEEP(NP_ELM)	Elements with seepage boundaries
NOD(8)	Internal node number of current element
NOD_FLXQAREA(500,20)	Matrix that stores area flux nodes
NOD_FLXQLINE(500,200)	Matrix that stores line flux nodes
NODALQ	Switch for nodal flux
NODB(NBBPTC)	The internal node number for held nodes
NODB(NP_HMX)	Held boundary conditions for current solution loop
NODBC(NTBCC)	The internal node number for held concentration node
NODBF(NBFPTC)	The internal node number for nodal fluxes
NODBH(NTBCH)	Internal node number for held head node
NODBH_R(NP_HH)	Held head boundary nodes read from .b44 file
NODBH_SAV(NP_NOD)	Held head boundary nodes for restoring during start of time loop
NODE(NP_UNK)	Unknown head and concentration nodes
NODE_DRY(NP_NOD)	Flag for surface node stack that is dry, indexed by user surface node number
NODE_ELEM(50,NP_NFLX)	Matrix that stores elements associated with nodal flux in transmissivity based flux distributions
NODE_FCWLND(100)	Nodal flux factor used in inverse parameter estimation
NODEC(NPTC)	Internal node number indexed by the internal concentration unknown
NODEH(NPTH)	Internal node number for head.
NODEH_SAV(NP_NOD)	Held head node numbers read from *.lp1* file

Table 2.1 (contd)

Global Variable	Definition
NODELM(NP_ELM*8)	Internal node numbers for each element
NODEPARA(6)	Stream and flux PARACODE values for inverse parameter estimation
NODER(NP_NOD)	Solution flag for head and concentration
NODERC(NP_NOD)	Solution flag for concentration
NODERH(NP_NOD)	Solution flag for head
NODHB_R_OLD(NP_HH_)	Held head nodes read from *.b44* files for previous time step
NODMAX	Maximum number of nodes in an element
NODT(NPT)	Internal node number
NONZER	Number of coefficients in sparse matrix
NP_DT	Maximum number of time steps
NP_ELM	Maximum number of element dimensions
NP_ESIN	Maximum number of elements with sinks
NP_ESOR	Maximum number of elements with sources
NP_EUSER	Maximum number of surface elements
NP_HC	Maximum number of held concentration nodes
NP_HH	Maximum number of held head nodes
NP_HMX	Maximum number of held head nodes
NP_LAYER	Maximum number of vertical layers
NP_LEAK	Maximum number of leakance nodes
NP_MAXWID	Maximum number of non-zero coefficients in system of equations
NP_NFLX	Maximum number of nodal flux nodes
NP_NOD	Maximum number of nodes
NP_NSEX	Maximum mass surface extractions at outflow nodes
NP_NUSER	Maximum number of surface nodes
NP_SELM	Maximum number of surface elements
NP_SNOD	Number of surface nodes
NP_STF	Maximum number of non-zero coefficients in a single row in the system of equations
NP_STRM	Maximum number of stream nodes
NP_UNK	Number of unknowns
NP_VSC	Number of points used in viscosity relationship with concentration
NPALL	Switch to print all primary variables at nodes
NPINCH_NODE(NP_NOD)	Flag for nodes with pinched layers
NPLAN	Switch that dictates storage to *.b07* file
NPMASS	Switch for printing mass balance information to output file
NPRES	Switch for defining initial pressure data
NPRINT	Switch that sets print options to *.prn* output files
NPRINT_CTL	Switch that controls printing of *cfest.ctl* file
NPRO3I	Flag for tracking successful completion of lprog3i
NPROG1	Flag for tracking successful completion of lprog1

Table 2.1 (contd)

Global Variable	Definition
NPROG3	Flag for tracking successful completion of lprog3
NPT	Total number of unknowns
NPTC	Number of concentration unknowns
NPTH	Number of head unknowns
NQ_NODE(NP_NFLX)	Node ID for transmissivity-based flux distribution
NQDOUT(NP_NOD)	Flag for outflow nodes with an internal node index
NQDS(NP_NFLX)	Flag for inclusion of diffusive flux at the outflow node
NQELEM(NP_ESOR)	User element numbers associated with source terms
NQNODE_ALL	Total number of nodes associated with nodal flux
NQOUT(NP_NFLX)	Outflow nodes used to estimate mass exiting system
NQSEX(NP_NSEX)	Nodes associated with surface extraction
NQTIME	Time step number where flux changes
NQTIME(NP_NFTM)	Time step indices for nodal flux rate changes
NRHO	Total number of nodes not converged in density calculation
NRIV_ACTIVE(NP_STRM)	Active stream nodes
NRIVER(NP_STRM)	Stream node numbers
NROHC	Total number of nodes exceeding convergence due to concentration changes
NS_NCOL	Maximum size of bandwidth between non-zero coefficients
NSEC	Switch used for terminating input decks
NSELEM	Maximum user element number assigned to a surface element
NSMAX	Maximum bandwidth in the system of equations
NSOLVE	Flag used for tracking simulation of head/pressure (NSOLVE = 1) concentration (NSOLVE = 3)
NSQELO(NP_ESIN)	User element numbers that have sink terms
NSQOUT	Total number of elements having element sink terms
NSQTOL	Total number of elements having sources
NSSAV	Maximum number of nonzeros in system of equations at previous time step
NSSS	Maximum number of nonzeros in system of equations at current time step
NSTDYC	Switch for steady-state or transient simulation of concentration
NSTDYH	Switch for steady-state or transient simulation of head
NSTDYT	Switch for steady-state or transient simulation of temperature
NSTEDY	Status of simulation for head or concentration
NSUBDT	Number of sub-divisions in a time step
NSUBP	Switch to print completion of key subroutines
NSUBTT(NP_DT)	Time step subdivisions
NSURFQ	Switch for surface infiltration
NSYS	Number of non-zero coefficients in system of equations in *lband*
NT_DRYMASS_ELEM(NP_DT)	Total number of dry mass elemental sources defined in an optional *dry_mass.elm*

Table 2.1 (contd)

Global Variable	Definition
NT_FLXQAREA(50)	Total number of flux areas in each file
NT_FLXQLINE(200)	Total number of flux lines in each file
NT_NDPARA(6)	Total number of node-based PARACODES
NTBCC	Number of held concentration nodes
NTBCH	Number of held head nodes
NTBCH_R	Number of held head nodes read from b44 file for current time step
NTBCH_R_OLD	Number of held head nodes read from b44 file for previous time step
NTIME	Time step counter
NTIMEQ	Number of time steps where there are changes in the total flux rate
NTINODE_FLXQAREA(20)	Total number of nodes in each flux area zone
NTOP	Total number of surface nodes
NTQOUT	Total number of outflow nodes
NTQSEX	Total number of surface extraction elements
NTRIV	Total number of stream nodes
NTSUBD	Number of subdivided time steps
NTSURFQ_ZONE(100)	Number of surface zones read in *cf_ucode*
NTT	Total number of nodes in the system
NTTYES	Solution option for temperature
NTV	Number of points in table defining viscosity-temperature relationship
NTYES	Switch that turns on temperature simulation
NUMBER(NP_NOD)	User node number indexed by internal node number
NZ_ITPACK	Total number of non zero coefficients
PARACODE*6	PARACODE variable size
PBWR	Pressure head for reference density
PPARACODE(76)	PARACODE variable used in inverse parameter estimation
PTHETA(NP_LAYER)	Pressure head at which THETA0 is defined
Q_CONC(NP_NFLX)	Nodal flux concentration
Q_MASS(NP_NFLX)	Nodal flow mass
QB(40)	Scalars associated with fluid balance
QELMIC	Cumulative elemental fluid mass from sources
QELMIT	Total elemental fluid mass from sources between previous and current time step
QELMOC	Cumulative elemental sink fluid mass
QELMOT	Total elemental sink fluid mass for a timestep
QELRI	Elemental source fluid mass rate
QELRI_HELD	Elemental source flow associated with held head
QELRO	Elemental sink fluid mass rate
QELRO_HELD	Elemental sink flow associated with held head
QEST(NP_NOD)	Estimated flow at held head nodes
QEST(NP_NOD)	Flux at the internal node number, or flow imbalance at user node number

Table 2.1 (contd)

Global Variable	Definition
QHNMIC	Cumulative nodal inflow fluid mass through held nodes
QHNMIT	Total nodal inflow fluid mass through held nodes between previous and current time step
QHNMOC	Cumulative nodal outflow fluid mass through held nodes
QHNMOT	Total nodal outflow fluid mass through held nodes between previous and current time step
QHNRI	Nodal inflow fluid mass rate through held nodes
QHNRO	Nodal outflow fluid mass rate through held nodes
QLAY_EXT(NP_DT,NP_LAYER)	Extraction for a layer at current time step
QLAY_INJ(NP_DT,NP_LAYER)	Injection for a layer at current time step
QLEAK(NP_LEAK)	Flux at leakance node
QLEAK_DT(NP_LEAK)	Total leakance flux at current time step
QMASS(NP_NOD)	Total mass flow at each node
QMND(NP_LAYER)	Total fluid mass in each element
QMNDC(NP_LAYER)	Total fluid mass in current time step
QMNI(NP_LAYER)	Initial fluid volume in layer
QMNN(NP_LAYER)	New estimated fluid volume in layer
QMNP(NP_LAYER)	Fluid volume in layer at current time step
QND_ASSIGN(NP_DT,1000)	Estimated transmissivity-based flux at each node
QNDMIC	Cumulative fluid mass inflow through nodal sources
QNDMIT	Total fluid mass inflow through nodal sources between previous and current time step
OC	Cumulative fluid mass outflow through nodal sinks
QNDMOT	Total fluid mass outflow through nodal sinks between previous and current time step
QNDRI	Fluid mass inflow rate through nodal sources
QNDRI_HELD	Total inflow fluid volume through held head nodes
QNDRO	Fluid mass outflow rate through nodal sinks
QNDRO_HELD	Total outflow fluid volume through held head nodes
QQLEAK	Total leakance flux at current time step
QQRIV(NP_STRM)	Total flow at stream node
QQRIVER	Total cumulative river flow at current time step
QRIV(NP_STRM)	Total flow at stream node
QSCONC(NP_SELM)	Concentration of infiltrating water
QSUAI	Surface infiltration area
QSUAO	Surface extraction area
QSUMIC	Cumulative surface fluid mass infiltration
QSUMIT	Total surface fluid mass infiltration between previous and current time step
QSUMOC	Cumulative surface fluid mass extraction
QSUMOT	Total surface fluid mass extraction between previous and current time step

Table 2.1 (contd)

Global Variable	Definition
QSURF	Infiltration rate per unit area per unit time
QSURF(NP_SELM)	Surface infiltration through each surface element at current time step
QSURF_FACT(100)	Multiplier on surface infiltration for inverse modeling (PARACODE)
QSURF_NEW(100)	New estimate of surface flow from inverse modeling (UCODE)
QSURF_RECH(NP_EUSER)	Surface flux for each surface element
QSURI	Surface fluid mass infiltration rate
QSURI_HELD	Surface fluid infiltration at held head nodes
QSURO	Surface fluid mass extraction rate
QSURO_HELD	Surface fluid outflow at held head nodes
QSYMDT	Change in fluid mass between previous and current time step
QSYMEC	Cumulative fluid mass balance error
QSYMET	Error in fluid mass balance between previous and current time step
QSYMI	Initial fluid mass in system
QSYMN	Fluid mass in system at current time step
QSYMNT	Fluid mass in system at end of current time step
QSYMP	Fluid mass in system at previous time step
QSYMPT	Fluid mass in system at end of previous time step
QSYMRC	Cumulative fluid mass balance error ratio
QSYMRT	Fluid mass balance error ratio between previous and current time step
RETARD	Retardation factor
RETARD_FACTOR(NP_LAYER)	Retardation factor by material type
RETARDED_SYSMASS_CUMM	Cumulative mass in solid phase
RETARDED_SYSMASS_DT	Cumulative mass in solid phase between previous and current time step
RHO(NP_NOD)	Density values indexed by internal node number
RHOC(NP_NOD)	Density component associated with concentration
RHOH(NP_NOD)	Density component associated with head
RLENGTH(NP_SNOD)	Stream node representative length
RPARM(30)	Parameter used in NSPCG solver
RPARM1_C	Iterative solver convergence factor for concentration
RPARM1_H	Iterative solver convergence factor for head
RRHO(NP_NOD)	Density at previous time step
RRHOS(NP_NOD)	Old density values indexed by internal node numbers
RRPARM(12)	Real parameters for ITPACK solver
RSYMIC	Cumulative mass in solid phase from inflow
RSYMIT	Total mass in solid phase from inflow between previous and current time step
RSYMOC	Cumulative mass in solid phase from outflow
RSYMOT	Total mass in solid phase from outflow between previous and current time step
SBBOTM(NP_STRM)	Streambed bottom elevation

Table 2.1 (contd)

Global Variable	Definition
SBOTME(NP_STRM)	Streambed top elevation
SC(NP_VSC)	Viscosity dependent concentrations
SCA(8)	Local coordinates of element
SCAZ(4)	Local coordinates of element (xz model)
SCOEF(NP_STRM)	Nodal conductivity coefficient for stream
SDEPTH(NP_STRM)	Stream depth
SEARCH_PARA*50	Character size length used for PARACODE search
SELEV(NP_STRM)	Stream elevation
SINFLO(NP_ESIN)	Cumulative outflow due to element sink indexed by user element number.
SINMAS(NP_ESIN)	Cumulative solute mass outflow due to element sink indexed by user element number
SNFFLO(NP_SNOD)	Cumulative fluid surface infiltration mass indexed by user element number
SNFMAS(NP_SNOD)	Cumulative solute surface infiltration mass indexed by user surface element
SOLUTE_NODAL_ERRMIN	User-defined mass threshold values for saving errors
SOUFLO(NP_ESOR)	Cumulative fluid mass inflow indexed by user element number
SOUMAS(NO_ESOR)	Cumulative solute mass inflow indexed by user element number
SPEC(NP_LAYER)	Specific storage coefficient
SPYIELD(NP_LAYER)	Specific yield coefficient
SQCONC(NP_ESOR)	Source concentration
SSUMOUTT(NP_ESIN)	Specified elemental fluid sink rate
SSUMQQ(NP_ESOR)	Specified elemental fluid source rate
STREAM_CONC(NP_STRM)	Concentration at stream node
STREAM_HEAD(NP_STRM)	Water level at stream node
STRMFLOW_SIM(100)	Simulated stream flow for each stream - used in lp3ucode
SUBDH	Sub-divided time step
TBEGIN	Initial time for simulation
TDONE	Number of completed time steps
TH(8)	Porosity at each node of element for current time step
TH_OLD(8)	Porosity at each node of element for previous time step
THETA(NP_LAYER)	Porosity by material type
THICK_ELEM(NP_ELM)	Element thickness
THICK_MIN	Minimum thickness for dry layer
TIME	Current simulation time
TRANS(500)	Estimated transmissivity-based flux at each node
TSALTM	Total salt mass added to fluid as a result of salt dissolution
TSALTQ	Total fluid mass added as a result of salt dissolution
TT(NP_DT)	Time steps
TTIME	Current simulation time when sub-time steps are implemented

Table 2.1 (contd)

Global Variable	Definition
TTIME_FACT(400000)	Factor for interpolating data between time steps to match observed time for UCODE
TTIME_R	Time read from *.b44* file
TTIME_R_OLD	Previous time read from *.b44* file
TTOTAL	Previous simulation time
U(NP_UNK+10)	Results from ITPACK and NSPCG
ULAST(NP_UNK+100)	Previous U vector values
UPSTRM	Upstream weighting factor used in transport
VARY_SURFQ(NP_NUSER)	Variable surface recharge from *varysurf.rec*
VCC(NP_VSC)	Tabular viscosity values for viscosity estimation
VERTDISP(NP_LAYER)	Vertical dispersivity by material type
VIS1	First regression parameter for viscosity data set
VIS2	Second regression parameter for viscosity data set
VIS3	Third regression parameter for viscosity data set
VISNOD(NP_NOD)	Nodal viscosity
VISR	Viscosity of resident fluid at reference temperature
VMAX	Maximum value of simulated variable (head or concentration)
VMIN	Minimum value of simulated variable (head or concentration)
WELL_ID*6	Well name used in inverse modeling
WK_NODEQ(NP_NFLX)	Work space for ITPACK solution
WKSP(NW_ITPACK)	Work space for ITPACK solution
X(NP_NOD)	X nodal coordinates
XBDLEM(8)	Base elemental area
XIDLEM(8)	Elemental surface area
XIJCOB(8)	Term used in integration; basis functions times volume
XIMN_MODEL	Minimum X coordinate of domain
XIXJOB(64)	Nodal volume matrix
XK(NP_LAYER)	Hydraulic conductivity/intrinsic permeability in X direction
XKE(8)	Intrinsic permeability and viscosity; used when KTYPE > 0
XKELEM(5,NP_ELM)	Array that stores the hydraulic conductivities in X, Y, and Z, porosity and specific storage for each element
XLA(1000)	Line length at node for line flux
XLEAK(NP_LEAK)	X coordinate of leakance node
XLENGTH(100)	Sum of all line fluxes
XMAX_MODEL	Maximum value of X coordinate
XN(8)	Basis function of each node at integration point
XNNG(64)	Basis function of each node at Gaussian point
XNODE(1000)	Local storage of X coordinates for length and area estimation
XQ(8)	X coordinate of each node in element; used in integration
XXDILEM(NP_ELM,8)	Element area
XYZK(3)	Hydraulic conductivity in X, Y and Z directions

Table 2.1 (contd)

Global Variable	Definition
Y(NP_NOD)	Y nodal coordinates
YK(NP_LAYER)	Hydraulic conductivity/intrinsic permeability in Y direction
YLEAK(NP_LEAK)	Y coordinates of leakance nodes
YMAX_MODEL	Maximum value of Y coordinate
YMIN_MODEL	Minimum value of Y coordinate
YNODE(1000)	Local storage Y coordinates for length and area estimation
YQ(8)	Y coordinate of each node in element; used in integration
Z(NP_NOD)	Z nodal coordinates
Z_SAV(NP_NOD)	Saved elevation values from previous head iterations
ZAV	Average elevation between two unconfined head estimation iterations
ZFILENAME	Binary file name character string
ZK(NP_LAYER)	Hydraulic conductivity/intrinsic permeability in Z direction
ZLAST(NP_NOD)	Estimated Z elevations for unconfined aquifer at previous time step
ZQ(8)	Z coordinate of each node in element; used in integration
ZTOP_SAV(NP_NUSER)	Z coordinates for surface nodes
ZY(8)	Element matrix right hand vector
ZZ_OLD(NP_OLD)	Elevation of lower interface of material at previous time step
ZZ_SAVE(NP_NOD)	Saved elevation of lower interface of material

3.0 Input Files

Multiple input files are used with the CFEST simulator, three of which are required for successful execution. These include the *.lp1* file, which serves as input to the *lprog1* module, the *.l3i* file, which contains input for the *lprog3i* preprocessor, and the *cfest.ctl* file, which contains control data for simulation execution. All other input files are optional and need only be in the current directory for CFEST to open the files for reading. These files include *solver.file*, which contains parameters for the NSPCG solver (Oppe et al. 1988) used in transport, and the *area_flx_file.lst* that lists filenames with area fluid mass flux data. During execution of the CFEST main module, *lprog3* checks for the existence of *solver.file* and *area_flx_file.lst*. If these filenames are not found in the current directory, CFEST execution proceeds without error. Hence, it is important to verify whether these files are necessary for proper execution of the model.

While all modules read inputs from the *cfest.ctl* file, only the *lprog1* preprocessor reads input from the *.lp1* file, whereas the *lprog3i* preprocessor reads input from the *.l3i* file. Because input filenames for both of these modules are given in the *cfest.ctl* file, it is not necessary that input filenames retain the *.lp1* and *.l3i* extensions. However, this is commonly practiced, and for ease of distinction, this convention is maintained in this text.

As previously stated, all modules read the *cfest.ctl* file. First, the *lprog1* module is executed, followed by *lband* and then *lprog3i* (see Section 4 for a description of program execution). These modules preprocess all input file data and write them to binary files for reading by *lprog3*, the CFEST main module. Because *lprog3* also reads the *cfest.ctl* file, any input options previously set can be overridden with execution of the main module. This can occur in two different ways. Flags can be changed in *cfest.ctl* after executing the three preprocessors; or, alternatively, the user can use flags in *cfest.ctl* that are specifically designed to overwrite input options set in the *.lp1* and *.l3i* files. For example, transport simulation flags are set in both the *.lp1* and *.l3i* files; however, solute transport can be disabled with a flag in the *cfest.ctl* file. Conversely, a uniform time step subdivision set in the *cfest.ctl* file is overwritten by variable time step subdivisions set in the *.l3i* files.

In general, the *.lp1* file provides data to the *lprog1* module on the finite-element grid and material and hydraulic properties of the conceptual model. The *.l3i* file provides data to the *lprog3* module on simulation time and time steps and solute and fluid fluxes. Both provide boundary condition data to CFEST: the *lprog1*module sets Dirichlet (fixed) and Cauchy (mixed) boundary conditions, whereas the *lprog3i* module sets Neumann (flux) boundary conditions.

In the sections that follow, input options are described for each CFEST input file. No units are used to describe inputs because it is the user's responsibility to use consistent input units. Input descriptions are tabularized with the required FORTRAN formats because all inputs are formatted read statements. Although the types of data are generally well defined within each input file, cross-referencing to related inputs in other input files is provided whenever appropriate.

3.1 The *lprog1* Input File

The *lprog1* module reads the *.lp1* file and stores the data in a binary format to be read by the main module, *lprog3*. The *lp1* input file contains data on the finite-element grid, material properties, and

solution switches. The finite-element grid is formed based on a Cartesian coordinate system and requires that each element contain eight nodes that set the corners of arbitrary hexahedral eight-node blocks. Only a surface grid is required as input, as CFEST is capable of forming the finite-element grid with depth given the material types. All material properties, such as hydraulic conductivity, porosity and permeability, are assigned to the elements in the grid.

The *lprog1* input is structured. With some compilers (e.g., Intel 8.x +), spaces will be read as zeros. Hence, if the data are not properly formatted in the input file, errors may occur. For example, if a node has a FORTRAN format of I3, and a node is input as "19 " (rather than " 19"), this may be interpreted by some compilers as node 190. Hence, it is very important to adhere to the FORTRAN formats provided in the tables below.

The *lprog1* module consists of four subroutines that perform the read of the *lp1* input file. In the sections that follow, data input types have been grouped by the subroutine that performs the read and then further subdivided based on data type or line number. A sample input file is presented in Figure 3.1 and is used to describe the *lprog1* input requirements on a line-by-line basis. No units are specified in the

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
Unit-element.lp1 Demo lp1 input file for one element stack with 3 layers
 0 0 0                                                  NTTYES,NCCYES,NNDIS
 0 0 0                                                  NDIM,KTYPE,NBURDN,NPRES
 0 0 0                                                  IHEAD,ITEMP,ICONC
   1.0000      1.0000      1.0000      1.0000          RCON,XYCONV,ZCONV,HCONV
   1.0000      1.0000      1.0000      1.0000           CONCPR,CONCPW,CONUKH,CONVIS
   1.0000      1.0000      1.0000                       CCONV,CONDIF,CONMOL
  0.00000     0.00000     0.00000     0.00000     0.00000    CW,CTW,CPW,CCMAS,CCC
  0.00000     0.00000                                        U0,T0CPW
   1000.0     0.00000     0.00000     0.00000               RHO0,H0,T0,C0
 0   1 Material #1
   1.0000      1.0000      1.0000                           XK,YK,ZK
 0.25000     0.00000     0.00000     0.00001          THETA0,PTHETA,CR,SPEC
  0.00000     0.00000     0.00000     0.00000          UKTX,UKTY,UKTZ,CPR
   100.00      10.000     0.00000                      ALPHAL,ALPHAT,DMOLE
 1                                                      END OF MAT. PROPERTIES
 1                                                      ILOG 1=CONSTANT LAYERS 0=VARIABLE
    3                                                   LMAX  # LAYERS-CONSTANT THICKNESS
    1   200.0    2   100.0    3   0.0                   (LL(J),ZZ(J),J=1,LMAX)
 1                                                      INITAL - HORIZTL INI.COND.SWITCH
  0.00000     0.00000     0.00000                       HI,TI,CI CONSTANT THROUGHOUT
 0        1   3  0.000       0.0        300.           NSEC,K,LMAX,X,Y,Z
 0        2   3  100.0       0.0        300.           NSEC,K,LMAX,X,Y,Z
 0        3   3  100.0       100.       300.           NSEC,K,LMAX,X,Y,Z
 0        4   3  0.000       100.       300.           NSEC,K,LMAX,X,Y,Z
 2                                                      END NODAL COORDS, START HELD BC
 0        1    300.                                     NSEC,NODB,BIV
 0        2    300.                                     NSEC,NODB,BIV
 0        3    300.                                     NSEC,NODB,BIV
 0        4    300.                                     NSEC,NODB,BIV
 0   300001   0.000                                     NSEC,NODB,BIV
 0   300002   0.000                                     NSEC,NODB,BIV
 0   300003   0.000                                     NSEC,NODB,BIV
 0   300004   0.000                                     NSEC,NODB,BIV
 6                                                      END OF HEAD B.C., START ELEMENT
 0        1 99 0    4        1        2        3        4
     3    1    1    1
 8                                                      END OF ELEMENT
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
```

Figure 3.1. Example *lp1* Input File Describing a One-Dimensional Simulation with One Surface Element and Three Elements with Depth, Each with a Different Material Type

discussion because it is the user's responsibility to maintain consistent length, time, and mass units for data in all of the input files. A schematic diagram of the grid described in the *.lp1* input file is shown in Figure 3.2. This figure also shows the node numbering scheme used by CFEST as it forms elements in the subsurface based on surface nodes and elements, as well as the vertical distribution of material types with depth.

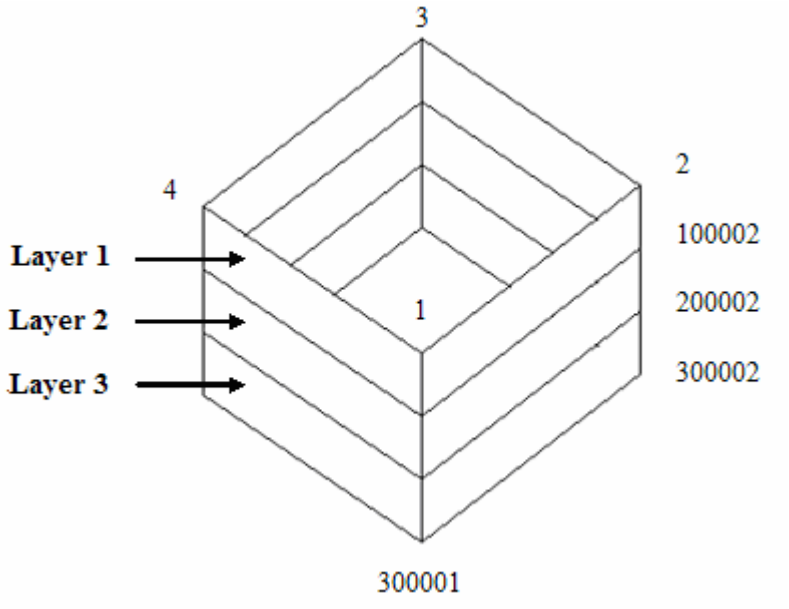

Figure 3.2. Schematic of a Unit Element Stack with Three Layers. The square is 100 × 100 ft.

3.1.1 Input File Structure

The *lprog1* module is the driver program for reading the problem title and calls the following four subroutines to read the associated groups of input data types (see Table 3.1):

1) Subroutine KXYZ to read in solution switches, fluid and porous media parameters, and material properties
2) Subroutine NODES to read in node locations and vertical distribution of materials (well log details), initial conditions, and over- and under-burden options
3) Subroutine BOUNDRY to read in boundary condition data
4) Subroutine ELEM to read in element properties

Only the first line of the.*lp1* input file is not read by the four subroutines. This first line is read in the main part of the *lprog1* module and is a character string (80 character maximum) that briefly describes the simulation.

Table 3.1. The Basic Structure of an *.lp1* Input File

Section	Description
Title	A suitable title (80 characters maximum)
Input for subroutine KXYZ	Solution options
	Dimensionality, transmissive parameter, over-under burden options
	Initial conditions options
	Conversion factors
	Aquifer fluid properties
	Viscosity data
	Material properties
	Salt dissolution option
	Heat generating materials options
	Over-burden–under-burden parameters
Input for subroutine NODES	Vertical temperature distribution
	Constant layer system option
	Constant layer system
	Constant initial condition option
	Constant values for initial conditions
	Node details
	Variable thickness over- under burden rock mass
Input for subroutine BOUNDRY	Held-head boundary conditions
	Held-temperature boundary conditions
	Held-concentration boundary conditions
	Stream boundary conditions
	Leakance boundary conditions
Input for subroutine ELEM	Element description
	Heterogeneous options

3.1.2 KXYZ Subroutine

The input parameters for KXYZ subroutine in the .lp1 file are shown in Table 3.2. The conditions for the presence of each parameter are shown in the 'condition' column of the table. More detailed descriptions for each input line can be found in the corresponding section listed in the last column of Table 3.2.

Table 3.2. Input Data Read from the *.lp1* File by Subroutine KXYZ

Parameters	Description	Condition	Section
NTTYES,NCCYES,NNDIS	Solution options	Required	3.1.2.1
NDIM,KTYPE, NBURDN,NPRES	Dimension flag, hydraulic conductivity type, over-under burden flag	Required	3.1.2.2
IHEAD,ITEMP,ICONC	Initial condition flag	Required	3.1.2.3
PERCON,XYCONV, ZCONV,HCONV	Conversion factors	Required	3.1.2.4
CONCPR,CONCPW, CONUKH, CONVIS		Required	
CCONV,CONDIF,CONMOL		Required	
CW,CTW,CPW,CCCMAS, CCC	Fluid properties	Required	3.1.2.5
U0,T0CPW		Required	
RHO0,H0,T0,C0		Required	
IVIS,GRVITY	Viscosity data	KTYPE > 0	3.1.2.6
NCV		KTYPE = 2	
SC,VCC			
NSEC,MAT,AMTER	Material properties	Required	3.1.2.7
XK,YK,ZK		Required	
THETA0,PTHETA,CR,SPEC		Required	
ALPHAL,ALPHAT,DMOLE		Required	
DKS,KFS,DCS	Salt dissolution option	NNDIS = 1	3.1.2.8

3.1.2.1 Solution Options

Solution switches are set on line 2 of the *.lp1* input file. These options, as well as their descriptions and input format requirements are shown in Table 3.3. Line 2 of the example input file in Figure 3.1 is shown below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
0 0 0                                                  NTTYES,NCCYES,NNDIS
```

The three zeros on the left side of the input line set the NTTYES, NCCYES, and NNDIS parameters, which represent the simulation switches for temperature, concentration, and salt dissolution, respectively. The first zero sets the NTTYES, temperature/heat parameter and indicates that temperature is not considered in this simulation. Because the current version of CFEST does not simulate heat transport, this switch should always be set to zero. The second zero indicates that transport is not simulated. The user should note that the *cfest.ctl* and *.l3i* files also set transport flags and that these flags need to be set consistently across all input files (see Sections 3.3.5 and 3.2.2). The third zero (NNDIS) means that salt dissolution will not be considered in this simulation.

Table 3.3. Solution Options NTTYES, NCCYES, NNDIS Description and Format

Parameters	Options	Description	Format
NTTYES	0	No solution for temperature to be performed.	3I2
	1	Solve for temperature in fluid.	
	2	Solve for heat conduction in a solid (no water and/or no concentration simulation).	
NCCYES	0	No solution for concentration to be performed.	
	1	Solve for concentration.	
NNDIS	0	No salt dissolution considered.	
	1	Consider salt dissolution.	

3.1.2.2 Dimensionality, Hydraulic Conductivity Type, and Over-Under Burden Options

Switches that set the problem dimensionality, hydraulic conductivity type, and over-underburden options are set on line 3 of the *.lp1* file and are described in Table 3.4. Line 3 of the example input file is shown below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
 0 0 0                                          NDIM,KTYPE,NBURDN,NPRES
```

The three zeros on the left side of the input line set the NDIM, KTYPE, and NBURDN parameters. These parameters represent the problem dimensionality, hydraulic conductivity type, and rock strata, respectively. The NPRES parameter, representing pressure data, is only input if KTYPE is greater than 0, and therefore is not needed in this example. In the input line shown here, the first zero sets the NDIM parameter to zero, which means that a three-dimensional simulation will be performed. This switch should always be set to zero because CFEST currently requires hexahedral elements for the grid. The second zero sets how the hydraulic conductivity will be read (KTYPE). In this example, a flag of zero indicates that the hydraulic conductivity will be entered by the user (see other options in Table 3.4). The KTYPE option equal to 1 is obsolete because heat flow is not simulated in the current version of CFEST. The KTYPE = 2 is still valid, though viscosity will not be defined as a function of temperature.

The third zero in this line sets the NBURDN parameter and indicates that confining rock strata for energy transport will not be considered in this case. This flag is still read in by *lprog1*, even though it is not used in the current implementation of CFEST. Because the KTYPE parameter was set to zero, the NPRES parameter does not need to be entered.

Table 3.4. Dimensionality, Transmissive Parameter, and Over-Underburden Options (NDIM, KTYPE, NBURDN, NPRES) Description and Format

Parameters	Options	Description	Condition	Format
NDIM	0	3D (x,y,z)		4I2
KTYPE	0	Hydraulic conductivity is entered by user.		
	1	Intrinsic permeability of medium and viscosity are a function of temperature only.		
	2	Same as 1, but viscosity is defined as a function of both temperature and concentration.		
NBURDN	0	No confining rock strata for energy transport.		
	1	Constant thickness overlying and underlying rock strata		
	2	Variable thickness at each node of over- and underlying rock strata.		
NPRES	0	Pressure data		
	1	Pressure estimates based on environmental density variation (normal). The entered head at the top surface is converted to pressure using resident fluid density and lower pressure integrated over the column.	KTYPE > 0	
	2	Pressure estimates based on reference density and head values specified. Used with only IHEAD=2 option for initial conditions.		

3.1.2.3 Initial Conditions Options

Initial condition switches are set in line 4 of the *.lp1* file and are described in Table 3.5. Line 4 of the example input file is shown below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
 0 0 0                                                  IHEAD,ITEMP,ICONC
```

Parameters IHEAD, ITEMP, and ICONC are set by the three zeros on the left side of the input line. Each of the parameters is set to option 0, which indicates that head, temperature, and concentration are of constant initial vertical distribution at each node. Setting these parameters to two requires that a vertical distribution of each primary variable be entered as an initial condition. The option for ITEMP should always be set to zero since heat flow is not currently supported.

Table 3.5. Initial Conditions Options (IHEAD, ITEMP, and ICONC) Description and Format

Parameter	Options	Description	Format
IHEAD	0	Constant initial vertical distribution of head at each node	3I2
	2	Explicitly enter the initial head at each node.	
ITEMP	0	Constant initial vertical distribution of temperature at each node	
	1	Initial vertical distribution of temperature to be defined as a function of elevation	
	2	Explicitly enter the initial temperature vertically at each node	
ICONC	0	Constant initial vertical distribution of concentration at each node	
	2	Explicitly enter the initial vertical distribution of concentration at each node	

3.1.2.4 Conversion Factors

There are three sets of parameters representing conversion factors. The conversion factor parameters are found on lines 5, 6, and 7 of the *.lp1* input file and shown below. The numbers on the left side of the lines set each of the parameters and serve as multipliers on input values in the *.lp1* file. These factors can be used to convert to different units, without having to manually scale inputs. The descriptions of each set of the parameters are shown in Tables 3.6–3.8. These tables describe multipliers on hydraulic conductivity, xyz coordinates, heat capacities, thermal conductivity, initial and boundary concentration data, dispersivity, and diffusion. Although still required as inputs in the *.lp1* file, the multipliers that operate on heat transport variables are not currently used in CFEST.

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
      1.0000      1.0000      1.0000      1.0000           RCON,XYCONV,ZCONV,HCONV
      1.0000      1.0000      1.0000      1.0000           CONCPR,CONCPW,CONUKH,CONVIS
      1.0000      1.0000      1.0000                  CCONV,CONDIF,CONMOL
```

Table 3.6. Conversion Factors PERCON, XYCONV, ZCONV, HCONV Description and Format

Parameter	Description	Format
PERCON	Intrinsic permeability/hydraulic conductivity values are multiplied by this factor.	G13.5, 3G12.5
XYCONV	For coordinate conversion, converts horizontal coordinates (x,y).	
ZCONV	For coordinate conversion, converts the vertical coordinates (z).	
HCONV	Converts both the initial and boundary condition of hydraulic head values read in by *lprog1*. Time-dependent held boundary conditions are read in *lprog3i*. Since HCONV is not saved for use in other programs, head values entered in *lprog3i* must be converted by the user in consistent units.	

Table 3.7. Conversion Factors CONCPR, CONCPW, CONUKH, CONVIS Description and Format

Parameter	Description	Format
CONCPR	Heat capacity conversion factor, used for the aquifer rock matrix and confining rock strata	G13.5, 3G12.5
CONCPW	Conversion factor for the heat capacity of water	
CONUKH	Conversion factor for all thermal conductivity values used in the aquifer rock matrix and the confining rock strata	
CONVIS	Conversion factor for viscosity data	

Table 3.8. Conversion Factors CCONV, CONDIF, CONMOL Description and Format

Parameter	Description	Format
CCONV	Conversion factor for the initial and held boundary condition concentration values	G13.5, 2G12.5
CONDIF	Conversion factor for dispersivity length	
CONMOL	Molecular diffusion conversion factor	

Lines 5, 6, and 7 set all of the conversion parameters to 1. This indicates that input values will remain the same and keep their initial units.

3.1.2.5 Aquifer Fluid Properties

Three sets of aquifer fluid parameters are set on lines 8, 9, and 10 of the *.lp1* file (shown below). The first set of parameters sets the fluid compressibility, the coefficient of thermal expansion, the heat capacity, and the concentration to mass and concentration to density relationships. The second set of parameters is temperature related. These inputs are required, but the values are not used in the current version of CFEST. The third set of parameters sets the fluid density and reference values for the primary variables. Detailed descriptions for each set of parameters are presented in Tables 3.9–3.11.

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
       0.0000     0.00000     0.00000     0.00000     0.00000    CW,CTW,CPW,CCMAS,CCC
       0.0000     0.00000                                        U0,T0CPW
       1000.0     0.00000     0.00000     0.00000                RHO0,H0,T0,C0
```

The simulation described by this input file does not consider solute transport and salt dissolution (NCCYES and NNDIS are both set to zero). Therefore, parameters CW, CTW, CPW, CCCMAS, and CCC of the 8th line and parameters U0 and T0CPW of the 9th line are also set to zero. The reference density parameter on line 10, RHO0, is set to 1000 kg/m^3 (the units are set in the *cfest.ctl* file) and the corresponding parameters, reference pressure head (H0), reference temperature (T0), and reference concentration (C0), are all set to 0.

Table 3.9. Aquifer Fluid Properties CW, CTW, CPW, CCCMAS, CCC Description and Format

Parameter	Description	Format
CW	Compressibility of the resident aquifer fluid (1/L). For transient flow.	G13.5,4G12.5
CTW	Coefficient of thermal expansion (1/T)	
CPW	Heat capacity of the resident aquifer fluid (E/M*T)	
CCCMAS	The concentration to mass relationship. CCCMAS is the factor by which the internal concentration in units of mass of contaminant per mass of fluid must be divided to obtain a concentration in the desired units. For example, for kilograms and meters with concentration in ppm, the CCCMAS = 10E-6.	
CCC	The concentration to density relationship. For concentration simulations this must be entered. CCC is the same factor as CCCMAS but it is entered as 0.0 unless you wish to consider density effects on fluid flow.	

Table 3.10. Aquifer Fluid Properties U0, T0CPW Description and Format

Parameter	Description	Format
U0	Fluid residual internal energy	G13.5,G12.5
T0CPW	Reference temperature for residual energy	

Table 3.11. Aquifer Fluid Properties Set RHO0, H0, T0, C0 Description and Format

Parameter	Description	Format
RHO0	Reference resident aquifer fluid density at the following pressure head, temperature, and concentration	G13.5, 3G12.5
H0	Reference pressure head of RHO0	
T0	Reference temperature of RHO0	
C0	Reference concentration of RHO0	

3.1.2.6 Viscosity Data Options

The viscosity data inputs are required only when KTYPE > 0. However, in the example input file shown in Figure 3.1, KTYPE has been set to 0. Therefore, no viscosity data are entered in this input file and viscosity parameter input lines are omitted. However, to acquaint the user with different scenarios, the example presented below is modified so that KTYPE > 0, and five options for viscosity parameters are considered. Each of the options is performed only when certain conditions are met. The descriptions of the five options are shown in Tables 3.12–3.13.

Table 3.12. Viscosity Data IVIS, GRVITY Description and Format for KTYPE > 0

Parameters	Options	Description	Format
IVIS	1	When KTYPE > 0, the relationships are described in terms of regression coefficients representing the temperature dependence of viscosity of resident fluid. This option is not supported in the current version of CFEST.	I6, G12.5
	2	When KTYPE = 1, tabular data is read in to describe viscosity-temperature relationships.	
		When KTYPE = 2, tabular data is read in to describe viscosity-concentration relationships.	
GRVITY		When KTYPE > 0, this factor is used in estimating the hydraulic conductivity from intrinsic permeability and viscosity. This term must be consistent in units of length and time with the other terms used in defining hydraulic conductivity (L/t^2)	

Table 3.13. Viscosity Data (NCV, SC, VCC) Description and Format; for KTYPE = 2, viscosity is considered to be a function of concentration)

Parameters	Description	Format
NCV	The number of points in the table defining the viscosity-concentration relationship	I6
SC	The matched arrays for concentration value, read in for NTV number of points	G13.5,3G12.5
VCC	The matched arrays for viscosity value, read in for NTV number of points (M/tL)	

Line 11 is the start of additional input for KTYPE > 0. In the first example, it is assumed that KTYPE = 2, and IVIS is set to 2. An example of an input line for this option is shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
     2       9.810                                          IVIS,GRVITY
```

In this option, tabular data are read in to describe viscosity-concentration relationships (KTYPE = 2). The temperature-viscosity relationship cannot be used until CFEST is updated to include heat transport.

The viscosity-concentration relationship is defined by parameters NCV, SC, and VCC which are shown in Table 3.13. There are two input lines for these three parameters. The first line contains the NCV parameter formatted as I6; the second line contains the SC and VCC parameters formatted as G13.5, 3G12.5. The two lines are shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----

     2                                                  NCV
          0.0       0.001        0.04      0.0013      SC(I),VCC(I),I=1,NCV
```

In the first line, the NCV parameter is set to 2, which indicates that there are two points defining the viscosity-concentration relationship. The two points are further defined in the second input line by parameters SC and VCC. When the concentration parameter SC is 0 mg/L, the viscosity parameter VCC is 0.001; when the concentration is 0.04 mg/L, the viscosity is 0.0013. In this example, a linear relationship between concentration and viscosity is observed.

3.1.2.7 Material Properties Options

Inputs for material properties can be classified into five groups. Several input options exist when NTTYES > 0 (line 2). However, because heat transport is not currently supported, the input parameter descriptions are presented only for NTTYES = 0 (no heat transport). Hence, a description of parameters NSEC, MAT, and AMATER, which describe material properties, is shown in Table 3.14.

The input line for the first set of material properties options is presented below, and is excerpted from line 11 in the example *.lp1* file presented in Figure 3.1.

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
 0   1  Material #1
```

The zero on the input line sets the NSEC parameter and indicates that a new material property is to be defined. The MAT parameter, which represents the material number, is set to 1. This indicates that the material properties of material one will be defined. The alpha-numeric description of the material or hydrologic layer, the AMATER parameter, is set to "Material #1."

Table 3.14. Material Properties NSEC, MAT and AMATER Description and Format

Parameters	Options	Description	Format
NSEC	0	Used as flag to start definition of new material property	I2, I4, 1x, A
	1	End of material property entries	
MAT	n	Identification number of a given hydrogeologic unit (the material number)	
AMATER	c	75-character alpha-numeric description of the material or hydrogeologic layer	

The second set of material properties options describes the hydraulic conductivity or intrinsic permeability, depending on the value of KTYPE set in line 3. The meaning of the XK, YK, and ZK parameters is defined in Table 3.15.

The input for line 12 is shown below. Because KTYPE has already been set to 0 (Section 3.1.2.2), the hydraulic conductivity property in x, y, and z directions is set to 1. Units are defined in the *cfest.ctl* file, but the user must ensure that all data are input with consistent units for all CFEST input files.

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
       1.0000      1.0000      1.0000                    XK,YK,ZK
```

Table 3.15. Material Properties of XK, YK, and ZK Description and Format

Parameters	Description	Format
XK	If KTYPE = 0, hydraulic conductivity in x direction	G13.5,2G12.5
	If KTYPE > 1, intrinsic permeability in x direction	
YK	If KTYPE = 0, hydraulic conductivity in y direction	
	If KTYPE > 1, intrinsic permeability in y direction	
ZK	If KTYPE = 0, hydraulic conductivity in z direction	
	If KTYPE > 1, intrinsic permeability in z direction	

Line 13 of the example input file sets the porosity, the pressure head at which porosity is defined, the compressibility, and the specific storage of the porous media (Table 3.16). The input line for material 1 is shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----X----X----
      0.25000     0.00000     0.00000     0.00001         THETA0,PTHETA,CR,SPEC
```

In this example, the porosity, THETA0, is set to 0.25 at reference pressure head parameter, PTHETA, which is set to 0. The media compressibility parameter, CR, is set to 0. The parameter value is approximated from the specific storage coefficient parameter, SPEC, set to 0.00001, and the compressibility of the resident aquifer fluid parameter, CW, which is set to 0 (see Section 3.1.2.5).

Table 3.16. Material Properties THETA0, PTHETA, CR, and SPEC Description and Format

Parameters	Description	Format
THETA0	Porosity at the reference pressure head, PTHETA	G13.5, 3G12.5
PTHETA	Pressure head at which THETA0 is defined	
CR	Compressibility of the media. If SPEC is given and CR = 0, it is approximated from SPEC and CW values	
SPEC	Specific storage coefficient	

The fourth input line describing material properties is used for heat flow, which is not currently supported. This input line, however, is still required to be present in the *.lp1* file, and, as such, the parameters are defined in Table 3.17. As shown below, the example input file has set all parameters to 0.

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----X----X----
      .00000     0.00000     0.00000     0.00000             UKTX,UKTY,UKTZ,CPR
```

Table 3.17. Material Properties UKTX, UKTY, UKTZ and CPR) Description and Format

Parameters	Description	Format
UKTX	Thermal conductivity of the combined media and resident fluid in X direction	G13.5, 3G12.5
UKTY	Thermal conductivity of the combined media and resident fluid in Y direction	
UKTZ	Thermal conductivity of the combined media and resident fluid in Z direction	
CPR	Heat capacity per unit volume of the rock matrix (E/TL^3)	

Line 15 of the *.lp1* file sets the fifth group of material properties, dispersivity, and diffusion (Table 3.18):

```
5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
       100.00      10.000     0.00000                    ALPHAL,ALPHAT,DMOLE
 1                                                       END OF MAT. PROPERTIES
```

In this example, the ALPHAL parameter is set to 100.00, which indicates that the longitudinal dispersivity length is 100 m (or units of length that are consistent with input). The ALPHAT parameter is set to 10, which sets the transverse dispersivity length to 10 m. The DMOLE parameter is set to 0, which turns off molecular diffusivity. The "1" on line 16 sets the NSEC parameter to 1 and terminates the loop that reads material properties options. More material types can be added on the following line, though the example input file contains only one material type.

Table 3.18. Material Properties Fifth Set (ALPHAL, ALPHAT, DMOLE) Description and Format

Parameters	Description	Format
ALPHAL	Longitudinal dispersivity length	G13.5,2G12.5
ALPHAT	Transverse dispersivity length	
DMOLE	Molecular diffusivity, including the effects of tortuosity	

3.1.2.8 Salt Dissolution

Inputs for salt dissolution are required only when NNDIS = 1 (line 2). Because parameter NNDIS has been previously set to 0 (see Section 3.1.2.1), the salt dissolution input line is not included in the example *.lp1* file. However, to acquaint the user with a different scenario, the example input line below assumes that NNDIS is set to 1 and describes the three salt dissolution parameters (Table 3.19). The input line for the salt dissolution options is presented below:

```
   5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
       0.200       0.1000     1.00000                              DKS,DFS,DCS
```

The input line indicates that the salt dissolution rate constant parameter, DKS, is set to 0.2. The fraction of material in the geohydrologic media parameter, DFS, is set to 0.1, and the maximum concentration that soluble material can contain, DCS, is set to 1.

Table 3.19. Salt Dissolution Option DKS, DFS and DCS Description and Format (required for NNDIS = 1)

Parameters	Description	Format
DKS	Salt dissolution rate constant (1/t)	G13.5, 2G12.5
DFS	Fraction of material in the geohydrologic media that is soluble	
DCS	Maximum concentration that soluble material can contain (C)	

3.1.3 NODES Subroutine

The NODES subroutine in the *lprog1* module reads in nodal coordinates, vertical distribution of materials (material contacts), initial conditions and over- and underburden options. A brief description of the variables read in the NODES subroutine is presented in Table 3.20, as are the switches necessary for the read to be invoked. More detailed descriptions for each input line can be found in the corresponding section listed in the last column of Table 3.20.

Table 3.20. Input Data for Subroutine NODES in *.lp1* File

Parameters	Description	Condition	Section
ILOG	Layer type flag (constant versus variable)	Required	3.1.3.1
LLMAX	Constant layer system	ILOG = 1	3.1.3.2
LL(J),ZZ(J)			
INITAL	Initial condition flag	IHEAD < 2 OR ICONC < 2	3.1.3.3
HEADI, TEMPI, CONCI	Uniform initial conditions	INITAL = 1	3.1.3.4
NSEC, K, LLMAX, XX, YY, ZTOP	Nodal input description	INITAL = 1	3.1.3.5
NSEC, K, LLMAX, XX, YY, ZTOP, HEADI, TEMPI, CONCI	Node input description with initial conditions	INITIAL = 0	3.1.3.6
LL(J), ZZ(J)	Variable material thickness	ILOG = 0	3.1.3.7
HEADD(J)	Vertically variable initial head conditions	IHEAD = 2	3.1.3.8
CONCEN(J)	Vertically variable initial concentration conditions	ICONC = 2	3.1.3.9

3.1.3.1 Layer Type Flag

The ILOG switch that appears on line 17 of the example input file determines how materials are layered in the domain. When ILOG is set to 0, a variable layered system is described; when ILOG = 1, materials are uniformly layered within the system. When layering is constant, the number of layers

(LMAX) is entered only once, and the material number and elevation of each layer must be entered in the next line. When material thicknesses vary, these parameters are entered in the section of the input file that describes node coordinates (see Section 3.1.3.5). Line 17 from the example input file appears below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
 1                              ILOG 1=CONSTANT LAYERS 0=VARIABLE
```

In this simulation the ILOG switch is set to 1, which describes a uniformly layered system , as shown in Table 3.21.

Table 3.21. Layer Type Flag (ILOG) Description and Format

Parameters	Options	Description	Format
ILOG	0	For variable layered systems. The variable number of layers or variable thickness with each node coordinate, LMAX, and thickness of each layer are entered in the node details description.	I2
	1	For uniformly layered systems. The LMAX and thickness of each layer is entered only once.	

3.1.3.2 Options for ILOG = 1

The input described in this section is required when ILOG = 1. This describes the total number of horizontal layers (LMAX), the material number (LL), and the elevation of the lowest contact between material types (ZZ) (Table 3.22). The example input file sets ILOG = 1, hence lines 18 and 19 of the *.lp1* file are shown below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    3                                   LLMAX .. # OF LAYERS -CONSTANT THICKNESS
   1   200.0    2   100.0    3     0.0                      (LL(J),ZZ(J),J=1,LMAX)
```

In the first input line parameter LMAX is set to 3, indicating that three horizontal layers are described. In the second input line, the elevation of the lower interface for each layer number (1, 2 and 3) is set to 200, 100, and 0. That is, the bottom elevation of the first layer is 200, the bottom elevation of the second layer is 100, and the bottom elevation of the third layer is 0.

Table 3.22. Options for ILOG =1 Parameter Description and Format

Parameters	Description	Format
LMAX	Total number of horizontal layers	I5
LL(J)	Material number for layer J	4(I5,F8.0)
ZZ(J)	Elevation of the lower interface for material layer	

3.1.3.3 Initial Condition Flag

The initial conditions flag, INITAL, sets how initial conditions are assigned for both head and concentration. These options are described in Table 3.23. If the INITAL flag is set to 1, then initial conditions are assigned uniformly for each node in the grid. An example of this condition from the example *.lp1* file is shown below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    1                                        INITAL - HORIZONTAL INI.COND.SWITCH
```

This flag sets the same initial condition for a vertical stack of nodes. If an initial condition is desired that varies with depth, then IHEAD must be set to 2 on line 4 of the *.lp1* file (see subsection 3.1.2.3).

Table 3.23. Constant Initial Condition Option (INITAL) Description and Format

Parameters	Options	Description	Format
INITAL	0	For variable initial conditions to be read with node coordinates	I2
	1	For constant initial head, temperature and concentration. One value of each variable is entered.	

3.1.3.4 Uniform Initial Conditions for INITAL = 1

The input for the uniform initial conditions is required only when parameter INITAL = 1. Because the parameter INITAL has been set to 1 in the example input file, an example of this condition is shown below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
      0.00000      0.00000      0.00000       HI,TI,CI CONSTANT THROUGHOUT
```

Parameters HEADI, TEMPI and CONCI are set to 0 in this simulation, and represent the starting values for head, temperature and concentration, respectively (Table 3.24). Although TEMPI is required input, its value is not utilized because heat transport is not supported in the current version of CFEST.

Table 3.24. The Uniform Initial Condition Description and Format for INITAL = 1

Parameters	Description	Format
HEADI	The uniform initial condition for head, unless overridden by IHEAD (refer to section 3.1.2.3)	1X,3F12.0
TEMPI	The uniform initial condition for temperature, unless overridden by ITEMP (refer to section 3.1.2.3)	
CONCI	The uniform initial condition for concentration, unless overridden by ICONC (refer to section 3.1.2.3)	

3.1.3.5 Nodal Input Description for INITAL = 1

The input data structure for describing the nodal coordinates is dependent on the value of INITAL. If INITAL = 1, initial condition data is not entered (see Table 3.25). This option is demonstrated by the example *.lp1* file shown for lines 22–25 below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
 0        1   3     0.000       0.0       300.              NSEC,K,LLMAX,XX,YY,ZZTOP
 0        2   3     100.0       0.0       300.              NSEC,K,LLMAX,XX,YY,ZZTOP
 0        3   3     100.0       100.      300.              NSEC,K,LLMAX,XX,YY,ZZTOP
 0        4   3     0.000       100.      300.              NSEC,K,LLMAX,XX,YY,ZZTOP
```

In all four input lines, parameter NSEC is set to 0, which indicates that nodal input follows. The user assigns a node number (K), followed by the number of material types at this location (LLMAX), the x and y coordinates, and the top elevation of the z coordinate. Hence, the surface coordinates for nodes 1, 2, 3, and 4 are (0,0,300), (100,0,300), (100,100,300) and (0,100,300), respectively. Refer to Figure 3.2 for a schematic representation of the grid formed by these nodal inputs.

Table 3.25. Nodal Input Description and Format for INITAL = 1

Parameters	Description	Format
NSEC	When NSEC = 0, continue reading nodal input; when NSEC > 0, end nodal input. More details on NSEC are shown in subsection 3.1.2.7.	1X,I1,I9,I4,3F 10.0
K	Node number	
LLMAX	The total number of horizontal layers at this node location	
XX	X coordinate for this node	
YY	Y coordinate for this node	
ZZTOP	The elevation of the surface of the aquifer system being modeled (top of the uppermost aquifer) at (x,y) node location	

3.1.3.6 Nodal Input Description for INITAL=0

If INITAL = 0, then initial condition data must be entered on the same line describing the coordinate data (see Table 3.26). Altering the example *.lp1* file, if INITAL = 0, then lines 22–25 would appear as

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
 0        1   3      0.00       0.00    300.00    300.00     25.00     0.00
 0        2   3     100.0       0.00    300.00    300.00     25.00     0.00
 0        3   3     100.0      100.0    300.00    300.00     25.00     0.00
 0        4   3      0.00      100.0    300.00    310.00     25.00     0.00
```

CONCI is set to zero because a flow simulation is assumed. Though TEMPI (set to 25 in this example) is a required input, its value is not used because heat transport is not supported in the current version of CFEST. This example assumes a uniform layering for the conceptual model (ILOG = 1). If ILOG = 0, additional data, described in subsection 3.1.3.7, are required for describing coordinate locations at depth.

Table 3.26. Nodal Input Description and Format for INITAL = 0

Parameters	Description	Format
NSEC	When NSEC = 0, continue reading nodal input; when NSEC > 0, end nodal input. More details on NSEC are shown in subsection 3.1.2.7.	1X,I1,I9,I4,6F10.0
K	Node number	
LLMAX	The total number of horizontal layers at this node location	
XX	X coordinate for this node	
YY	Y coordinate for this node	
ZZTOP	The elevation of the surface of the aquifer system being modeled (top of the uppermost aquifer) at this node location (x,y).	
HEADI	The vertically uniform initial condition for head at this node unless over-ridden by IHEAD = 2, in which case it is the value for the surface node. Values below the surface node will be entered on additional cards below.	
TEMPI	The same as HEADI except for temperature and switch is ITEMP	
CONCI	The same as HEADI except for concentration and switch is ICONC	

3.1.3.7 Nodal Input Description for ILOG = 0

When the non-uniform layering option is set (ILOG = 0), additional input is required in the coordinate description section of the *.lp1* file. These data include the material number (LL) and the bottom elevation of the material (ZZ). These parameters are described in Table 3.27. The example *.lp1* file has been modified below to show the required input structure for ILOG = 0:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----X----X----
 0        1   3      0.00      0.00    300.00    300.00     25.00     0.00
    1   200.0    2   150.0    3   0.000
 0        2   3     100.0      0.00    300.00    300.00     25.00     0.00
    1   200.0    2   100      3   50.00
 0        3   3     100.0     100.0    300.00    300.00     25.00     0.00
    1   200.0    2   100      3   10.00
 0        4   3      0.00     100.0    300.00    310.00     25.00     0.00
    1   200.0    2   150.0    3   0.000
```

On the 2nd, 4th, 6th, and 8th input lines above, parameter LL(J) sets the material number of each of the three layers, followed by the bottom elevation of the specified material number (ZZ(J)). For example, at the first node, the bottom elevation of the first layer is 200, the bottom elevation of the second layer 150, and the bottom elevation of the third layer is 0.

Table 3.27. Nodal Input Description for ILOG = 0 and INITAL = 1

Parameters	Description	Format
LL(J)	The material number for layer J at node K	30(I5,F8.0)
ZZ(J)	The elevation of the lower interface of that material layer (L) at node K	

3.1.3.8 Nodal Input Description for Initial Heads Varying with Depth (IHEAD = 2)

If IHEAD = 2, heads that vary with depth are required as input for the initial condition. This alters the nodal input structure, as shown in the modified example below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
0         1   3      0.00      0.00    300.00    300.00     25.00     0.00
    1   200.0    2   150.0    3   0.000
        250.0       170.0       35.0
 0        2   3     100.0      0.00    300.00    300.00     25.00     0.00
    1   200.0    2   100      3   50.00
        250.0       120.0       35.0
 0        3   3     100.0     100.0    300.00    300.00     25.00     0.00
    1   200.0    2   100      3   10.00
        250.0       120.0       25.0
 0        4   3      0.00     100.0    300.00    310.00     25.00     0.00
    1   200.0    2   150.0    3   0.000
        250.0       170.0       35.0
```

As described in Table 3.28, the initial head must be specified. This is demonstrated in lines 3, 6, 9, and 12 above. As outlined in the previous sections, surface coordinates and initial conditions are first input, followed by inputs describing the material contacts with depth. Then the initial head (HEADD) is read for each node in the vertical stack. For example, the initial head for node 1 at the bottom of the surface layer is 250, at the bottom of the second layer 170, and at the bottom of the domain, 35.

Table 3.28. Nodal Input Description for IHEAD = 2

Parameters	Description	Format
HEADD(J)	The initial head condition at node J below surface node	1X, 30G12.5

3.1.3.9 Nodal Input Description for Initial Concentrations Varying with Depth (ICONC = 2)

If ICONC = 2, then concentrations that vary with depth are required as input for the initial condition. This alters the nodal input structure as shown in the modified example below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
0         1   3      0.00      0.00    300.00    300.00     25.00     0.00
    1   200.0    2   150.0    3   0.000
        250.0       170.0       35.0
        1.000       0.500       0.10
 0        2   3     100.0      0.00    300.00    300.00     25.00     0.00
    1   200.0    2   100      3   50.00
        250.0       120.0       35.0
        1.000       0.400       0.10
 0        3   3     100.0     100.0    300.00    300.00     25.00     0.00
    1   200.0    2   100      3   10.00
        250.0       120.0       25.0
        1.000       0.400       0.10
 0        4   3      0.00     100.0    300.00    310.00     25.00     0.00
    1   200.0    2   150.0    3   0.000
        250.0       170.0       35.0
        1.000       0.400       0.10
```

As described in Table 3.29, the initial concentration must be specified. This is demonstrated in the example above on lines 4, 8, 12, and 16. As outlined in the previous sections, surface coordinates and initial conditions are first input, followed by inputs describing the material contacts with depth. The

Table 3.29. Nodal Input Description for ICONC = 2

Parameters	Description	Format
CONCEN(J)	The initial concentration condition at node J below surface node	1X, 30F12.0

above example also assumes that initial head is specified with depth. Then the initial concentration (CONCEN) is read for each node in the vertical stack. For example, the initial concentration for node 1 at the bottom of the surface layer is 1.0, at the bottom of the second layer 0.50, and at the bottom of the domain, 0.10.

3.1.3.10 Internal Node Numbering and Conversion of 2-D Topology to 3-D Mesh

CFEST uses an automatic procedure to generate the 3-D finite-element mesh from a 2-D quadrilateral surface grid. As outlined above, this is accomplished by providing a well log with the top and bottom elevations of each layer (or material type) at each node. Subsurface nodes are generated in the problem domain for each data point in the well log. Once nodes are generated with depth, hexahedral orthogonal curvilinear elements are formed. Hence, the vertical mesh size is based on the thickness of the material described in each of the well logs. To create thinner elements, additional data points are required in the well logs, although no changes may occur in the stratigraphy.

Node numbers beneath the surface are assigned by adding to the surface node, the product of the layer number times the maximum number of surface nodes (100,000). As CFEST converts the 2-D surface grid to a 3-D hexahedral finite-element grid, the additional subsurface nodes are numbered as

$$NN_{subsurface} = NN_{surface} + MN * 100000 \quad (3.1)$$

where $NN_{subsurface}$ is the node number for the additional subsurface node, $NN_{surface}$ is the surface node number, and MN is the material type or layer number defined by an integer in the well log. For example, material type number 3 beneath node 10 in the surface grid would be assigned a node number of 300,010 (i.e., 10 + 3*100,000 = 300,010). Subsurface element numbers are assigned in a similar manner.

3.1.4 BOUNDRY Subroutine

The BOUNDRY subroutine is used for reading fixed, stream and leakance boundary conditions (Table 3.30). Neumann boundary conditions are set in the *.l3i* file described in Section 3.2.9. There are three basic inputs that help to describe all boundary conditions: NSEC, MP, and BIV. The NSEC flag sets the boundary condition type, MP sets the node number that is assigned the boundary condition, and BIV is the value of the boundary condition for node MP.

Table 3.30. Input Data for Subroutine BOUNDRY in *.lp1* file

Parameters	Description	Condition	Subsection
NSEC, MP, BIV	Fixed boundary conditions (Dirichlet)	NSEC = 2	3.1.4.1
NSEC, ISNOD, SELE, SLENTH, SWIDTH, SBOTE, STHICK, SK, SDEP	Stream boundary conditions	NSEC = 5	3.1.4.2
NSEC, MP, BIV, XKL, XLL, XLA, HINI	General head boundary conditions (Cauchy)	NSEC = 7	3.1.4.3

The NSEC values used in the *lprog1* input file are shown in Table 3.31. NSEC is used as a flag to tell the *lprog1* module how to read the current or next input line. The values include NSEC = 0 and NSEC = 1 that were used to begin and end material property input (Section 3.1.2.7). When NSEC = 8, this signals the end of the *.lp1* input file. The NSEC variable is also used to begin element input reads (NSEC = 6) described in Section 3.1.5. As shown in Table 3.30, NSEC is also used as a flag for the boundary condition type. Note that NSEC = 3 sets a fixed temperature boundary, even though heat transport is not supported in the current version of CFEST.

Table 3.31. NSEC Options

Parameter	Options	Description	Format
NSEC	0	Continue reading input in present section	I2
	1	End the material property input	
	2	Begin reading held-head boundary conditions	
	3	Begin reading held-temperature boundary conditions	
	4	Begin reading held-concentration boundary conditions	
	5	Begin reading stream boundary conditions	
	6	Begin reading element details	
	7	Begin reading Cauchy (general head) boundary conditions	
	8	End of *lprog1* input file	

3.1.4.1 Fixed Boundary Conditions

Boundary conditions are necessary to define how the conceptual model interacts with the entire flow system. Dirichlet, or fixed boundaries, are used when head or concentration can be specified as a function of time or space. To set fixed boundary conditions, the boundary condition must be entered for each node in the domain, including the nodes below the surface. Lines 26–30 of the example *.lp1* file are shown below for a fixed head condition at the top and bottom of the vertical stack:

```
0        1    300.                          NSEC,NODB,BIV
0        2    300.                          NSEC,NODB,BIV
0        3    300.                          NSEC,NODB,BIV
0        4    300.                          NSEC,NODB,BIV
0   300001   0.000                          NSEC,NODB,BIV
0   300002   0.000                          NSEC,NODB,BIV
0   300003   0.000                          NSEC,NODB,BIV
0   300004   0.000                          NSEC,NODB,BIV
```

An example of input lines representing fixed concentration boundary conditions is presented below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
4                                                                 NSEC
0         3         2.0                                           NSEC,MP,BIV
0         4         2.0                                           NSEC,MP,BIV
6                                                                 NSEC
```

On the first input line, the NSEC parameter is set to 2, indicating that a fixed concentration condition is being set. On the next input line, the NSEC flag is set to 0, so that the *lprog1* module will continue reading input on that line. Nodes 3 and 4 are both assigned fixed concentrations of 2.0. On the last line of this example, parameter NSEC is set to 6, terminating boundary condition input.

The boundary conditions set in these examples are time invariant. Time-dependent, specified head or concentration boundaries can also be set, but the data must be present in a separate file in the current run directory. The filenames for time-dependent specified head and time-dependent specified concentration are found in the *cfest.ctl* file, and flags that set the time-dependent fixed boundary conditions are set in the *.l3i* file (see Section 3.2.4). The input format for time-variant boundary conditions is the same format described in Table 3.32, and boundary conditions must be present for each time step in the simulation.

Table 3.32. Fixed Boundary Conditions Description and Format

Parameters	Option	Description	Format
NSEC	2	Start fixed head boundary read	I2
	4	Start fixed concentration boundary read	
NSEC	0	Read boundary condition data on input line	1x,I1,I9, G12.5
MP	n	Node number to which the boundary condition is assigned	
BIV	n	Boundary condition value	

3.1.4.2 River/Stream Boundary Conditions

Rivers and streams are special forms of the head-dependent boundary condition. In a head-dependent boundary, the model computes the difference in head between the boundary and the model cell where the boundary is defined. The head difference is then multiplied by a conductance term to get the amount of water flowing into or out of the aquifer. River boundaries should be used where a surface water feature partially penetrates a layer and can both remove water from the aquifer and infiltrate water into the aquifer. River boundaries should not be used if the flow in the river can go to zero under losing conditions.

In CFEST, the NSEC flag for a river boundary condition is 5. Since a river boundary condition is not used in the example *.lp1* file, an example river boundary input for a single node is given below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
5                                                                 Start river
0         1    100.0    500.0     10.0     90.0     10.0   0.0010   100.0
```

The example shows a river boundary condition set at node 1, where the length of the stream represented by node 1 is 500.0, the average stream width is 10.0, the stream bed elevation is 90.0, the sediment thickness is 10.0, the stream bed hydraulic conductivity is 0.001, and the minimum depth of stream flow above the bottom of the stream bed is 100.0. The required input is described in Table 3.33.

Table 3.33. River Boundary Parameters and Description

Parameters	Options	Description	Format
NSEC	5	Start stream boundary condition read	I2
NSEC	0	Read boundary condition data on input line	I2,I9,7F9.0
ISNOD	n	Stream node number	
SELE	n	Dummy variable (will be used as time constant stream head)	
SLENTH	n	Length of stream represented by ISNOD	
SWIDTH	n	Average stream width	
SBOTE	n	Stream bed elevation	
STHICK	n	Sediment thickness	
SK	n	Stream bed hydraulic conductivity	
SDEP	n	Minimum depth of stream flow above bottom of stream bed	

3.1.4.3 Cauchy Boundary Conditions

A Cauchy boundary, also referred to as a general head boundary, is a generic form of the head-dependent flow boundary, where flux is dependent on the hydraulic head. Cauchy boundaries are normally used along the edge of the model to allow groundwater to flow into or out of the model under a regional gradient. Cauchy boundaries allow flow to take place across the boundary, but the amount of flow is regulated by the water level that is set for a point or boundary at some distance outside the model area and by the conductance that is set for the area between the actual model boundary and the distant point or boundary.

In CFEST, the Cauchy boundary condition flag is NSEC = 7. Because a river boundary condition is not used in the example *.lp1* file, an example river boundary input for a single node is given below:

```
  5  10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
START General-Head Condition
 0      1    350.00 .1600E-02    150.0     22570.    -100.0 nsec,k,hd,xkk,xl,xarea,hb
```

This example shows a general-head boundary condition set for node 1. The flux is 350.00, the hydraulic conductivity in the x-coordinate direction is 0.0016, the distance between node 1 and the domain boundary is 150, the cross-sectional area of the cell is 22570, and the initial head at node 1 is -100.00. The required input is described in Table 3.34.

Table 3.34. River Boundary Parameters and Description

Parameters	Options	Description	Format
NSEC	7	Start Cauchy (general head) boundary condition read	I2
NSEC	0	Read boundary condition data on input line	I2, I7, 5F10.0
MP	n	Cauchy boundary node number	
BI	n	Flux at Cauchy boundary node	
XKL	n	Hydraulic conductivity in x direction	
XLL	n	Distance between node and domain boundary	
XLA	n	Cross-sectional area of cell	
HINI	n	Initial head	

3.1.5 Input Data for the ELEM Subroutine

The input data for the ELEM subroutine includes information on the element structure and the hydraulic properties assigned to each of the elements. Because CFEST builds a three-dimensional finite-element grid from surface data and depth of material contacts, only a two-dimensional surface element description is required. Because CFEST elements are hexahedrons, the four nodes that make up the corners of each element at the surface must be specified. The nodes of the element are specified in a counter-clockwise direction. Three-dimensional elements are then generated from the vertical distribution of materials given in the nodal description section of the input file (Section 3.1.3).

Table 3.35. Input Data for Subroutine ELEM in *.lp1* File

Parameters	Description	Condition	Subsection
K, NMAT,KELEM, NN, NND(J)	User element data	Required	3.1.5.1
LAYERT, NZMAT	Elemental material type	NMAT = 99	3.1.5.2
XYZK(1), XYZK(2), XYZK(3), STORI, PORI	Elemental hydraulic properties assigned by layer	KELEM = 1	3.1.5.3
XYZK(1), XYZK(2), XYZK(3), STORI, PORI	Elemental hydraulic properties assigned by element	KELEM = 2	3.1.5.4

3.1.5.1 First Input Line for ELEM subroutine

The ELEM subroutine reads three sets of input data. The first set specifies the user element number, the user node numbers that make up the four corners of each surface element, and how hydraulic properties (hydraulic conductivities, porosity and specific storage) are assigned. Input options for NMAT and KELEM specified in this section of the *.lp1* file dictate how subsequent input lines are read. The input descriptions are shown in Table 3.36. Lines 35, 36, 37 and 38 of the example *.lp1* file are shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
6                                                                 START ELEMENT
0        1 99 0    4        1        2        3        4
    3    1    1    1
8
```

Table 3.36. Element Descriptions and Options

Parameters	Options	Description	Format
NSEC	6	Start element read	1X, I1
NSEC	8	End element read and *lprog1* input file	
NSEC	0	Continue element read	1X, I1, I9, I3, I2, I5, 4I9
K	User-specified element number	The user surface element number. Each generated subsurface element is increased by 100,000. The NPRINT = 5 option set in the *cfest.ctl* file (Section 3.3.3) can print all vertical element details.	
NMAT	n	Material number for each element below the surface element is assigned according to the material contact data (well log) specified in the nodal input description (Section 3.1.3)	
	99	Material numbers will be specified at the surface and with depth rather than assigning material numbers based on those specified in the nodal material contact (well log) description. These values are entered in subsequent input line(s) (see subsection 3.1.5.2).	
KELEM	0	Properties associated with the material number specified in the material properties (subsection 3.1.2.7) are used.	
	1	Hydraulic properties - hydraulic conductivity (KX, KY, KZ), specific storage (SPEC), and porosity (THETA0) found on subsequent input line are assigned to the vertical stack of elements (see subsection 3.1.5.3).	
	2	Hydraulic properties hydraulic conductivity (KX, KY, KZ), specific storage (SPEC), and porosity (THETA0), are to be entered for each element below the surface. These values are entered on subsequent input line(s) (see subsection 3.1.5.4).	
NN	4	The number of nodes defining the surface element. This value should always be set to 4.	
NDD(J)	n	User node numbers, listed in a counter-clockwise direction, that compose user element number K	

First, parameter NSEC is set to 6 to indicate that element data will be read on the next input line. On the following input line (36), NSEC = 0 to indicate that element information follows on the same input line. K is set to 1, which assigns the surface element number to 1. NMAT is set to 99, which means that material numbers in the entire vertical stack of elements will be user specified (see Table 3.37). As specified in Table 3.36, NN must be set to 4 so that the nodes that mark the four corners of a quadrilateral surface element are properly assigned. The four integers that follow set NDD(J) to 1, 2, 3, and 4, which assign nodes 1, 2, 3 and 4 to surface element number 1. The nodes must be specified in a counter-clockwise direction. Line 37 indicates that there are three layers and all of these three layers are assigned properties from material number 1. On the last line (38), parameter NSEC is set to 8, which signals the end of the *.lp1* file.

Table 3.37. Material Assignment for NMAT = 99

Parameters	Description	Format
LAYERT	Number of layers below this surface element	1X, 16I5
NZMAT	New material number to be assigned to the elements at this location. For example, NZMAT(1) is the new material number for uppermost element at this location, NZMAT(LAYERT) is the new material number for the bottom element at this location, etc.	

3.1.5.2 Material Assignment for NMAT = 99

When NMAT is set to 99, material properties are assigned by element, rather than using the material contact data provided in the nodal input section (see Section 3.1.3). When materials are assigned by element, LAYERT and NZMAT are read on the next input line (Table 3.37), as shown in the modified example *.lp1* file below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
6                                                                  START ELEMENT
0        1 99 0    4        1        2        3        4
    3    1    2    2                                          LAYERT,NZMAT
8                                                              NSEC
```

The first two input lines were explained in the previous section and are repeated here for clarity. The third line specifies three layers (LAYERT) at this surface element. At the surface, material type 1 is assigned to the element, and material number 2 is assigned to the two subsurface elements.

3.1.5.3 Hydraulic Property Assignment for KELEM = 1

When KELEM = 1, the hydraulic conductivity, specific storage, and porosity are specified for the entire vertical stack of elements at this location. When materials are assigned uniformly with depth, the hydraulic properties are assigned on the next input line as shown in the modified example *.lp1* file below:

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
6                                                                  START ELEMENT
0        1 99 1    4        1        2        3        4
    3    1    2    2
       1.000      1.0000      0.1000       0.001        0.10
8
```

For all of the elements in the vertical stack (1, 200001, 300001), the hydraulic conductivity in both the x and y coordinate directions is 1.0. The hydraulic conductivity in the z coordinate direction is 0.1. The specific storage parameter is set to 0.001 and the porosity is specified to be 0.1.

3.1.5.4 Hydraulic Property Assignment for KELEM = 2

When KELEM = 2, the hydraulic conductivity, specific storage, and porosity are specified for *each* element in the vertical stack. The input parameter description and format are described in the previous section (Table 3.38). When materials are not uniformly assigned with depth, the hydraulic properties input line is repeated for each element as shown in the modified example *.lp1* file below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
6                                                                    START ELEMENT
0        1 99 2    4        1        2        3        4
    3    1    2    2
       1.000      1.0000      0.1000       0.001        0.10
       1.000      1.0000      0.1000       0.010        0.10
       1.000      1.0000      0.0100       0.001        0.01
8
```

Three sets of hydraulic properties are listed for each element in the vertical stack. The first set of properties corresponds to element 1, the second to element 200001, and the third to 300001. This allows for non-uniform distribution of hydraulic properties with depth.

Table 3.38. Hydraulic Properties for KELEM = 1

Parameters	Description	Format
XYZK(1)	Hydraulic conductivity in x direction (KX) to be used in all elements at this location	1X, 5G12.5
XYZK(2)	Hydraulic conductivity in y direction (KY) to be used in all elements at this location	
XYZK(3)	Hydraulic conductivity in z direction (KZ) to be used in all elements at this location	
STORI	Specific storage (SPEC) to be used in all elements at this location	
PORI	Porosity (THETA0) to be used in all elements at this location	

3.2 The *lprog3i* Input File

The *lprog3i* module reads the *.l3i* file and stores the data in a binary format to be read by the main module, *lprog3*. The *.l3i* input file contains data on solution switches, as well as flux (Neumann) boundary conditions. Total simulation time, time steps, and subdivision of time steps are also set in the *.l3i* file. Output times need not be specified because simulation output is written to binary files at each specified elapsed time.

Like *lprog1, lprog3i* input is structured. With some compilers (e.g., Intel 8.x +), spaces will be read as zeros. Hence, if the data are not properly formatted in the input file, errors may occur. For example, if a node has a FORTRAN format of I3 and a node is input as “19 ” rather than “ 19”, this may be interpreted by some compilers as node 190. Hence it is very important to adhere to the FORTRAN formats provided in the tables below and ensure that values are right justified within the format specificaton.

Unlike *lprog1*, the *lprog3i* main program controls the input file read. Hence, the input file description for *lprog3i* is not subdivided by subroutines; rather, it is grouped by data type. A sample input file is presented in Figure 3.3 and is used to describe the *lprog3i* input requirements on a line-by-line basis. No units are specified in the discussion because it is the user's responsibility to maintain consistent length, time, and mass units for data in all of the input files. The model domain is the same as the example used for the *lprog1* description and is shown in Figure 3.2.

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
Unit-element.l3i Demo l3i input file for one element stack with 3 layers
    1    0    0                                                  NSTDYH,NSTDYT,NSTDYC
    5    0                                                       ITOTAL, NTEXP
        0.00                                                     TBEGIN
       10.00       20.00       50.00      100.00      200.00    TIME(I), I=1,ITOTAL
    0    0    0                                                  NHBOND,NTBOND,NCBOND
    0                                                           MNTRY
    0                                                           NSURFQ
    0                                                           NELQIN - ELEM. SOURCE
    0                                                           NELQEX - ELEM. SINK
    0                                                           NODALQ
    0
    0                                                           IOPT=0,NO SUBDIVISION
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
```

Figure 3.3. Example *.l3i* Input File That Is Used to Prescribe Time-Dependent Data

3.2.1 Input File Structure

The input file can be divided into groups based on data type. The primary sections include:

- Secondary title
- Transient or steady-state options
- Time step control options
- Time-dependent held boundary condition switches
- Elemental sources and sinks
- Nodal sources and sinks
- Time step subdivisions for transport.

An outline of the inputs required for the *.l3i* file is shown in Table 3.39. Detailed descriptions for each input can be found in the corresponding sections listed in the fourth column of Table 3.39. The first line of the input file is a character string (80 character maximum) that briefly describes the simulation.

3.2.2 Steady-State or Transient Simulation Options

In the first line of the *.l3i* file, the steady-state/transient switches are set. Three switches, NSTDYH, NSTDYT, and NSTDYC, are used for defining steady-state or transient simulation of head, temperature, and concentration, respectively. Although the temperature switch is a required input, the current version of CFEST does not support heat transport.

Table 3.39. The Basic Structure of the LPROG3I Input File

Parameters	Description	Condition	Section
	Secondary title	Required	3.2.1
NSTDYH,NSTDYT,NSTDYC	Transient/steady-state switches	Required	3.2.2
ITOTAL,NTEXP, TBEGIN,TIME	Time step control options	Required	3.2.3
NHBOND,NTBONT,NCBOND	Time-dependent held boundary conditions	Required	3.2.4
MNTRY	Uniform materials debug option	Required	3.2.5
NSURFQ,QSURF, QSTEMP,QSCONC	Surface infiltration	Required	3.2.6
NELQIN,NQEL, SQQ,SQTTMP, SQCTMP,HPIN,ELEV	Elemental sources	Required	3.2.7
NELQEX,NSQE, SUMTMP,HPOTMP	Elemental sinks	Required	3.2.8
NODALQ,NP,BIVF, BIVFT,BIVFC	Nodal sink and source options	Required	3.2.9
IOPT,NSUBTT	Variable time subdivisions	Required	3.2.10

Fluid flow can be steady-state or transient. Two options exist for a transient flow simulation, transient flow only and steady-state head initially to obtain initial conditions and then transient flow. Though switches for steady-state concentration are provided, transport is primarily simulated as transient. Options for the simulation switches are described in Table 3.40. Line 2 from the example *.l3i* file is shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    1    0    0                                             NSTDYH,NSTDYT,NSTDTC
```

The three numbers on the left side set the NSTDYH, NSTDYT, and NSTDYC parameters. The 1 indicates that the simulation solves a transient flow solution. The second zero is required input, but the steady-state temperature flag is not used. The third zero sets the NSTDYC parameter for steady-state concentration. In the absence of any concentration data in any of the input files, this switch effectively turns off transport.

Table 3.40. Simulation Switches Description and Format

Parameters	Options	Description	Format
NSTDYH	0	Solve for steady-state flow	3I5
	1	Solve for transient flow	
	2	Solve for steady-state flow initially to obtain initial conditions and then begin transient simulation	
NSTDYT	0	Solve for steady-state temperature	
	1	Solve for transient temperature	
NSTDYC	0	Solve for steady-state concentration	
	1	Solve for transient concentration	

3.2.3 Time Step Control Options

Numerical simulations are usually carried out to predict results for specific elapsed periods of time. In CFEST, instead of reading the size of each time increment, elapsed times are read. All data are saved to binary files at each specified elapsed time.

On the third line of the *.l3i* input file, the number of time steps (ITOTAL) is specified, followed by a switch (NTEXP) that specifies whether time steps are entered by the user (NTEXP = 0) or generated internally (NTEXP = 1) (see Table 3.41). The input expected on line 4 of the *.l3i* file is dependent on the value of NTEXP. When parameter NTEXP = 0, the (elapsed) simulation time steps are entered by the user. In this case two parameters, TBEGIN and TIME(I), are used in the next input lines, as shown:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    5    0                                                  ITOTAL, NTEXP
         0.0                                  TBEGIN
        10.00       20.00       50.00      100.00      200.00  TIME(I), I=1,ITOTAL
```

Parameter ITOTAL is set to 5, indicating that 5 time steps are considered in this simulation. Parameter NTEXP is set to 0, so elapsed times are entered by the user. The time steps, i.e., the difference between elapsed times, are calculated internally. Parameter TBEGIN sets the initial simulation time to 0. Parameter TIME(I) sets the rest of the elapsed times to 10, 20, 50, 100 and 200, with time steps of 10, 10, 30, 70, and 100. A maximum of 5 elapsed times are entered on each row of the input file (see Table 3.42).

There are two ways to set up a steady-state simulation. If an input file has already been created for a transient simulation, a steady-state flow simulation will still be performed as long as NSTDYH = 0 in the second line of the *.l3i* file (see Section 3.2.2). Alternatively, a steady-state simulation can be set up by not specifying any time-step data, and by setting both ITOTAL and NTEXP to zero, as shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    0    0                                                  ITOTAL, NTEXP
```

In this case, NSTDYH, NSTDYT, and NSTDYC must all be set to zero on the second line of the *.l3i* file, and NHBOND, NTBOND, and NCBOND must also be set to zero on line 6 of the *.l3i* file (Section 3.2.4).

Table 3.41. Time Step Control Input Description and Format

Parameters	Options	Description	Format
ITOTAL	N/A	Number of time-steps	2I5
NTEXP	0	For entering specific times from the start of simulation	
	1	Generated by LPROG3I	

Table 3.42. Initial Time and Elapsed Time Description and Format for NTEXP = 0

Parameters	Description	Format
TBEGIN	Initial time for simulation (t)	F12.0
Time (I), I =1, ITOTAL	Specific time values are entered for the points in time at which results are to be saved. These must be in increasing order and TIME(1) > TBEGIN. The program checks to ensure that delta time > 0, where delta time = TIME(I) – TIME(I-1) for I = 2 to ITOTAL, and the initial delta time is TIME(I) – TBEGIN	5F12.0

When parameter NTEXP = 1, time steps are generated by the *lprog3i* module, resulting in a change in the input format structure. In place of elapsed times, four parameters TIME(1), FACT, DELMAX, and TBEGIN, are set on the next line, as shown in the modified *.l3i* example below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    5    1                                                  ITOTAL, NTEXP
       10.00         2.0      100.00         0.0           TIME(1),FACT,DELMAX,TBEGIN
```

In this example the total number of time steps is the same (ITOTAL = 5), and the initial simulation time is the same (TBEGIN = 0.0). Only the first value of the time array (TIME) is set to 10. The other two parameters, FACT and DELMAX, are used to calculate time steps as shown in the following example:

First time step: $\Delta t(1) = TIME(i) - TBEGIN$

Subsequent time steps: $\Delta t(i+1) = \Delta t(i) \times FACT$

$\Delta t(1) = 10 - 0 = 10$

$\Delta t(2) = 10 \times 2 = 20$

$\Delta t(3) = 20 \times 2 = 40$

$\Delta t(4) = 40 \times 2 = 80$

$\Delta t(5) = 80 \times 2 = 160 \ > DELMAX \therefore \Delta t(5) = 100$

The elapsed simulation steps are 10, 10+20=30, 30+40=70, 70+80=150, and 150+100=250. The maximum change in the time step was exceeded on the fifth calculation and was reduced to the user specified limit (DELMAX).

Table 3.43. Initial Time and Elapsed Time Description and Format for NTEXP = 1

Parameters	Description	Format
TIME(1)	First time step is entered by specifying the time after TBEGIN when the first time step is complete. Subsequent times are generated by *lprog3*.	4F12.0
FACT	Delta time(i+1) = delta time(i) * FACT. (when FACT=1, constant time steps are generated)	
DELMAX	Maximum time step, required if FACT > 1.0	
TBEGIN	Initial time for simulation (t)	

3.2.4 Time-Dependent Held Boundary Conditions

The *lprog1* module is used to set fixed boundary conditions that are constant in time. If held boundary conditions are time-dependent, then two input requirements must be met. First, the time-dependent held boundary condition data must be stored in file(s) that are named in the *cfest.ctl* file (see Section 3.3.2). Second, switches in the *.l3i* file must be set so that the main module reads the fixed boundary condition data at each time step.

The time-dependent held boundary conditions can be stored in binary or ascii format. Head and concentration data should be stored in separate files. Files should contain all the held values for each time step, arranged in the same sequential order and format as the held nodes specified in the *.lp1* file.

Three parameters, NHBOND, NTBOND, and NCBOND, are used in the input line for this section (see Table 3.44). Although the temperature flag (NTBOND) is required input, its value is not used since the current version of CFEST does not support heat transport. The line from the example *.l3i* file is shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    0    0    0                                                 NHBOND,NTBOND,NCBOND
```

In this input line, parameters NHBOND, NTBOND, and NCBOND are set to zero. Therefore, boundary conditions for head, temperature, and concentration are time-constant held values and no additional data are read.

Table 3.44. Time-Dependent Fixed Boundary Condition Switches Description and Format

Parameters	Options	Description	Format
NHBOND	0	Time-constant held values for head boundary conditions, no new data read in	3I5
	1	Time-dependent held values for head boundary conditions; the time for each step is assumed to be the same as the input in Section 3.2.3. No interpolation is carried out.	
NTBOND	0	Time-constant held values for temperature boundary conditions, no new data read in	
	1	Time-dependent held values for temperature boundary conditions; the time for each step is assumed to be the same as the input in Section 3.2.3. No interpolation is carried out.	
NCBOND	0	Time-constant held values for concentration boundary conditions, no new data read in	
	1	Time-dependent held values for concentration boundary conditions; the time for each step is assumed to be the same as the input in Section 3.2.3. No interpolation is carried out.	

3.2.5 Uniform Materials Debug Switch

If highly variable properties (e.g., hydraulic and transport properties) are assigned, round-off errors may cause unreasonable results. For such cases, the MNTRY switch was created to easily assign the same material number to all elements in the domain. The MNTRY parameter is strictly a debug switch. After running simulations with a variety of MNTRY values, such problems may become isolated.

The MNTRY line from the example *.l3i* file shows that the debug option has been turned off:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
0                                                       MNTRY
```

Table 3.45. Uniform Materials Description and Format

Parameters	Options	Description	Format
MNTRY	0	Normal	I5
	N	N is the material number assigned uniformly to all elements. It is used for debugging.	

3.2.6 Surface Infiltration

The surface infiltration switch NSURFQ determines whether elemental recharge data are entered in this section. The options for NSURFQ are shown in Table 3.46. Depending on the value assigned to NSURFQ, either parameters shown in Table 3.47 or Table 3.48 are entered on the next input line in the *.l3i* file, or no input is entered at all. The input line from the example *.l3i* file is shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    0                                                       NSURFQ
```

In this example, NSURFQ is set to 0, and no surface infiltration occurs in this simulation. No further input on surface infiltration is required. However, to acquaint the user with the different input structures required by each of the NSURFQ options, examples are presented below.

Table 3.46. Surface Infiltration Option Description and Format

Parameters	Options	Description	Format
NSURFQ	0	No surface infiltration	I5
	1	Constant rate of surface infiltration/evaporation for the entire region and for all time steps	
	2	Time-constant but spatially-variable surface infiltration/evaporation for specific elements	
	3	Same as 2, but rates for each time step are entered	

When NSURFQ = 1, three parameters, QSURF, QSTEMP, and QSCONC, which represent the rate of infiltration, its temperature, and its concentration, respectively, are entered in the next input line. Though heat transport is not now supported, the temperature of the infiltrating water is still a required input. The description and format of these parameters are shown in Table 3.47 and an example input line below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    1                                                             NSURFQ
        10.0         25.0         1.0                           QSURF,QSTEMP,QSCONC
```

Parameter NSURFQ is set to 1, indicating that constant surface flux is prescribed for all elements and all time. Parameter QSURF is set to 10, which sets the infiltration rate per unit area per unit time to 10. Parameter QSTEMP is set to 25, indicating that the temperature of the infiltrating water is 25°C. Parameter QSCONC is set to 1.0; thus, the concentration of infiltrating water is 1.0.

Table 3.47. Surface Infiltration Parameters Description and Format for NSURF = 1

Parameters	Description	Format
QSURF	Infiltration rate per unit area per unit time	3F12.0
QSTEMP	Temperature of infiltrating water	
QSCONC	Concentration of infiltrating water	

When NSURFQ = 2 or NSURFQ = 3 for spatially variable surface infiltration/evaporation, five parameters, IE, QS, QST, QSC, and IIEND, are required as input (Table 3.48). The infiltration rate for those elements remaining constant in time need only be specified for the first time step. Thus, only those elements for which surface infiltration varies with time need to be repeated for each time step.

Table 3.48. Surface Infiltration Parameters Description and Format for NSURF = 2 and NSURF = 3

Parameters	Options	Description	Format
IE	n	Starting element for which surface infiltration/evaporation is to be entered. A blank card terminates a data set (when NSURFQ=2) or a time step (when NSURFQ = 3)	I9, 3E12.0, I9
QS	n	Infiltration rate per unit area per unit time	
QST	n	Temperature of infiltrating water	
QSC	n	Concentration of infiltrating water	
IIEND	0	Signifies that the starting and ending element are the same (single element assigned surface infiltration)	
	n	The ending element number in a sequence of element numbers from IE to IIEND which all have the same infiltration rate	

The input format allows sequences of elements whose infiltration is the same to be entered in blocks as shown in the example below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    2                                                             NSURFQ
        1        10.0         25.0         1.0         5         IE,QS,QST,QSC,IIEND
```

Parameter NSURFQ is set to 2 indicating that time-constant but spatially-variable surface infiltration/ evaporation is entered from element 1 (IE) to element 5 (IIEND). Parameter QS is set to 10.0, which sets the infiltration rate per unit area per unit time to 10. The temperature parameter QST is set to 25, but this value is not used since the current version of CFEST does not support heat transport. Parameter QSC is set to 1.0, which assigns the infiltrating water a concentration of 1.0. The description and format of these parameters are shown in Table 3.48.

3.2.7 Element Based Source Options

Element sources and sinks are entered sequentially, but in different sections of the input file. The NELQIN switch determines whether any source terms are to be considered. The options for NELQIN are shown in Table 3.49. Subsequent input lines are dependent on the value assigned to NELQIN, as shown in Table 3.50. The input line from the example *.I3i* file is shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    0                                                        NELQIN - ELEM. SOURCE
```

NELQIN is set to 0. Therefore, elemental source terms are not considered in this simulation and no further input is entered for this section. However, an example of NELQIN = 2 is also provided to show how the input structure differs when elemental sources are present.

Table 3.49. Element Based Source Switch Description and Format

Parameters	Options	Description	Format
NELQIN	0/1	No elemental source terms	I5
	2	Time-constant elemental source terms	
	3	Time-dependent elemental source terms	

When NELQIN = 2 or NELQEX = 3, six parameters, NQEL, SQQ, SQTTMP, SQCTMP, HPIN, and ELEV, are entered in the next input lines. These inputs describe the user element number, the volumetric rate, temperature, and concentration, as well as the hydraulic head and elevation at the point of injection (Table 3.50). Although the temperature value is not used, it is still a required input. Example input for NELQIN = 2 appears below:

Table 3.50. Element Based Source Parameter Description and Format for NELQIN = 2 or NELQIN = 3

Parameters	Description	Format
NQEL	Element number having source term. A 0 terminates a data set (when NELQIN=2) or a time step (when NELQIN = 3)	I9, 5E12.0
SQQ	Source volumetric rate	
SQTTMP	Temperature	
SQCTMP	Concentration	
HPIN	Hydraulic head at point of injection	
ELEV	Elevation at point of injection	

```
       5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    2                                   NELQIN
        1        10.0        25.0        1.0        10.0         9.0
        2        12.0        25.0        1.0        10.0         9.0
        3        15.0        25.0        1.0        11.0        10.0
    0
```

In this example, elemental source terms are time-constant. Each line (for elements 1, 2 and 3) represents a single element, followed by its source terms. For example, element 1 has a source volumetric rate of 10 at a temperature of 25°C, a concentration of 1.0, a hydraulic head at the point of injection of 10, and an elevation of 9. The 0 terminates the elemental source input.

The source values for those elements remaining constant in time need only be specified for the first time step they become active. Thus, only those elements for which the source rate varies with time need to be repeated. For example, if a source only changes at two points in time, then only two entries at these time points need to be entered. However, inputs are required for each time step in the simulation when NELQIN = 3.

3.2.8 Element Based Sink Options

Elemental sinks are entered after elemental sources. The NELQEX switch determines whether any sink terms are to be considered. The options for NELQEX are shown in Table 3.51. Similar to the elemental source section in the *.l3i* file, subsequent input lines are dependent on the value assigned to NELQEX, as shown in Table 3.51. The input line in the example *.l3i* file is shown below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    0                                                  NELQEX - ELEM. SINK
```

Parameter NELQEX is set to 0. Therefore, elemental sinks are not considered in this simulation and no further input is required in this section of the *.l3i* file. However, an example of NELQEX = 2 is also provided to show how the input structure differs when elemental sinks are present.

Table 3.51. Element Based Source Switch Description and Format

Parameters	Options	Description	Format
NELQEX	0/1	No elemental sink terms	I5
	2	Time-constant elemental sink terms	
	3	Time-dependent elemental sink terms	

When NELQIN = 2 or NELQEX = 3, three parameters, NSQE, SUMTMP, and HPOTMP, are entered in the next input lines. These inputs describe the user element number, the sink volumetric rate, and the pressure head of withdrawal (see Table 3.52). Example input for NELQEX = 2 appears below:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    2                                   NELQEX
        4        10.0        1.0                                  NSQE,SUMTMP,HPOTMP
        5        12.0        2.0                                  NSQE,SUMTMP,HPOTMP
    0
```

In this example, elemental sink terms are time constant. Each line (for elements 4 and 5) represents a single element, followed by its sink terms. For example, Element 4 has a sink volumetric rate of 10 and a pressure head withdrawal of 1. The 0 terminates the elemental sink input.

The sink values for those elements remaining time constant need only be specified for the first time step they become active. Thus, only those elements for which the source rate varies with time need to be repeated. For example, if a source only changes at two points in time, then only two entries at these time points need to be entered. However, inputs are required for each time step in the simulation when NELQEX = 3.

Table 3.52. Element Based Source Parameter Description and Format for NELQEX = 2 or NELQEX = 3

Parameters	Description	Format
NSQE	Element number having sink term. A 0 terminates a data set (when NELQEX=2) or a time step (when NELQEX = 3)	I9, 2E12.0
SUMTMP	Sink volumetric rate	
HPOTMP	Pressure head of withdrawal	

3.2.9 Nodal Sink and Source Options

The NODALQ switch determines whether any nodal sources or sinks are considered in the simulation. Similar to the elemental sink/source sections, the input structure changes when nodal sources and/or sinks are present. Options for NODALQ are shown in Table 3.53. If no nodal sources are present, then no further input is required. Options for NODALQ = 2 are shown in Table 3.54, and options for NODALQ = 3 are shown in Table 3.55. The example below shows the input structure for NODALQ = 0:

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    0                                                       NODALQ
```

In this example, parameter NODALQ is set to 0, and no further input is required in this section of the *.l3i* file.

Table 3.53. Nodal Sink and Source Switch Description and Format

Parameters	Options	Description	Format
NODALQ	0/1	No nodal flux	I5
	2	Time-constant nodal flux	
	3	Time-dependent nodal flux	

When NODALQ = 2, four parameters, NP, BIVF, BIVFT, and BIVFC, are entered in subsequent input lines (see Table 3.54). These parameters represent the node number, and the fluid, temperature, and concentration fluxes, respectively. Although the temperature input is required, the current version of CFEST does not support heat transport, and its value is not used. An example of NODALQ = 2 is shown below:

Table 3.54. Nodal Sink and Source Input Parameter Description and Format for NODALQ = 2

Parameter	Options	Description	Format
NP	N/A	Node number having nodal flux (L^3/t); a 0 terminates a data set.	I8, 3F12.0
BIVF	N	Integrated flow volume. If the nodal flux rates are negative (production/withdrawal), the temperature and concentration are specified as 0. These become a function of simulation results.	
	1.0E-30	The node NP must be a held head node whose temperature and/or concentration is specified by BIVFC	
	-1.0E-30	The node NP is a dry mass injection node. The rate of dry mass injection is specified by BIVFC in units of contaminant mass per unit time (M/t).	
BIVFT	N/A	Temperature of injection fluid	
BIVFC	N/A	Concentration of injection fluid (C) or concentration rate (M/t) if BIVF = -1.0E-30	

```
      5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
   2                                                                NODALQ
      1   0.125E-01       0.000      1.000                          NP,BIVF,BIVFT,BIVFC
      2   0.125E-01       0.000      1.000                    NP,BIVF,BIVFT,BIVFC
   0
```

Parameter NODALQ is set to 2; therefore, the nodal flux is time-constant. Each line of input corresponds to a distinct nodal source. For example, parameter NP is set to 1 and 2, indicating that the time-constant nodal flux occurs at nodes 1 and 2. For each node the integrated flow volume is set to 0.12500E-01, the temperature parameter is set to 0, and the concentration of the fluid is set to 1.0. Because heat flow is not currently supported, the temperature value is not used, but its input is still required. The 0 on the last input line indicates that input for nodal fluxes has terminated. It is also possible to inject a dry mass of a contaminant that then mixes with the existing volume of water. When a dry mass is injected, the fluid volume must be flagged with –1.0E-30, as shown in the following example:

```
     5  10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
   2                                                      NODALQ
      1    -1.0E-30      0.0000      1.0000                NQ,BIVF,BIVFT,BIVFC
      2    -1.0E-30      0.0000      1.0000                 NQ,BIVF,BIVFT,BIVFC
   0
```

In this case, the values of the parameters remain the same as in the first example except that parameter BIVF is set to -1.0E-30. Thus, the mass injection rate is 1 M/t (Mass/time step) since parameter BIVFC is set to 1. The 0 on the last input line indicates that input from this data set has terminated.

When NODALQ is set to 3, two more parameters are entered in addition to those in Table 3.54. These parameters occur on the input line *before* the nodal source data in Table 3.54 are entered and provide information on the timing of the fluxes. These parameters, shown in Table 3.55, are the total number of time steps at which changes in the nodal fluxes occur (NTIMEQ), and a list of the time step numbers at which the fluxes change (NQTIME(I)). Example input lines for NODALQ = 3 (modified from the example *.l3i* file) are shown below:

Table 3.55. Nodal Sink and Source Input Parameter Description and Format for NODALQ = 3

Parameter	Description	Format
NTIMEQ	Number of time steps at which there are changes in the nodal flux rates	10I5
NQTIE(I), I=1,NTMIEQ	The time step numbers where the flux changes	

```
    5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    3                              NODALQ
    2    1    3                      NTIMEQ, NQTIME(I), I=1,NTIMEQ
       1   0.125E-01      0.0000      1.0000                NQ,BIVF,BIVFT,BIVFC
       2   0.125E-01      0.0000      1.0000                NQ,BIVF,BIVFT,BIVFC
    0
       1         0.0      0.0000      0.0000                 NQ,BIVF,BIVFT,BIVFC
       2         0.0      0.0000      0.0000              NQ,BIVF,BIVFT,BIVFC
    0
```

When NODALQ = 3, nodal fluxes are time-dependent. Parameter NTIMEQ is set to 2, indicating there are two time steps at which changes in the nodal flux rates occur. Parameter NQTIME is set to 1 and 3, indicating that flux changes occur at time steps 1 and 3. From time step 1 to time step 3, the integrated flow volume is 0.0125, and concentration is 1 at nodes 1 and 2. After time step 3, the integrated flow volume is terminated. The 0 on the last line indicates that input from this data set is terminated.

For NODALQ = 3, once a node is included in the flux list, it must be present for every time step at which nodal flux rate changes occur. For example, if nodal flux changes occur 100 times and the flux at node 4 only occurs at the first time step, it must still be listed 99 times in the input file with the fluid and concentration flux set to 0.

3.2.10 Variable Time-Step Subdivisions Option

Numerical simulations are usually carried out to predict results for specific elapsed periods of time. For numerical reasons, it may be necessary to reduce the transport time steps to a finer resolution than those used for fluid flow. In this case, one may define the time steps (ITOTAL, Section 3.2.3) to be points in time when the solution will be saved. However, smaller time steps can be taken to simulate transport without saving the solution. The IOPT option in the *.l3i* file allows the user to specify the number of time step subdivisions for each time step in the simulation.

When IOPT = 0, subdivision of the time steps does not occur (see Table 3.56). This option is shown in the example *.l3i* file below:

```
        5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    0                                                    IOPT =0 NO VARI SUBDIVISION
```

Table 3.56. Subdivision Time Step Switch Parameter Description and Format

Parameter	Options	Description	Format
IOPT	0	No variable time step subdivisions	I5
	1	Variable time step subdivisions	

When IOPT = 1, time step subdivisions (NSUBTT (J), J=1,ITOTAL) are specified for each time step on subsequent input lines given the format in Table 3.57. If no subdivision is to occur for a specific time step, a 1 should be entered. An example of input lines for this case (IOPT = 1 and ITOTAL = 5) is shown below.

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
----X----X----X----X----X----X----X----X----X----X----X----X----X----X----x----x----
    1                                                    IOPT =1  VARI SUBDIVISION
    5   10   10    10   1
```

Table 3.57. Subdivided Time Step Data Description and Format for IOPT = 1

Parameters	Description	Format
NSUBTT(J), J=1,ITOTAL	Number of subdivisions into which time step J is to be subdivided	10I5

For each time step in the simulation, time step subdivisions take place. At the first time step, the subdivision is 5; for the second to fourth time steps, the subdivision is 10; and for the fifth time step, there in no subdivision since NSUBTT is assigned a value of 1.

Note that time step subdivisions may also be set on line 8 of the *cfest.ctl* file (Section 3.3.4). The uniform time step subdivision parameter (NSUBD) read in the *cfest.ctl* file is overridden by the time step subdivisions set in the *.l3i* file.

3.3 CFEST Control File

The CFEST control file (*cfest.ctl*) is read by all CFEST modules and contains several different data types. Its inputs have evolved with more recent code developments, which has allowed the structure of the *.lp1* and *.l3i* files to remain the same (and backwards compatible with previous simulations). The *cfest.ctl* file contains data on input filenames, output control switches, and solver parameters, as well as several simulation switches. In many instances, the *cfest.ctl* file *overwrites* input data in the *.lp1* and *.l3i* files. This is useful when debugging a simulation but may also lead to undesirable results. Hence, the user must use care in setting simulation switches in the *cfest.ctl* file.

Unlike the *.lp1* and *.l3i* files, the input data in the *cfest.ctl* file is unstructured and comma delimited. However, input is only unstructured with respect to FORTRAN formatted statements. Inputs are ordered in the *cfest.ctl* file, with each line assigned a specific input. Comments are permitted within the file and must be preceded with an asterisk (*). Blank lines are also permitted, but the order of the data lines must always be maintained.

A sample input file is presented in Figure 3.4 and will be used to describe the *cfest.ctl* input requirements on a line-by-line basis. The example *cfest.ctl* file can be used in conjunction with the *.lp1* and *.l3i* files presented in Sections 3.1 and 3.2; however, since none of the inputs contribute to the description of the conceptual model, it can also be used in other simulations as long as input filenames are updated.

```
*************** CFEST run control file ******************
'cfest_001',                    Important this version number should match code version
0,                              nprint_ctl
'test005.qa',                   IMPORTANT ***     This filename should be in single quotes
'tmp/test005',                  Line 1 - path and name for binary & ascii output files
'test005.lp1',                  Line 2 - LPROG1 input file name
'test005.l3i',                  Line 3 - LPROG3I input filename
'headbc.hld',                   Line 4 - time dependent head B.C. file name
'concbc.dat',                   Line 5 - time dependent conc B.C. file name
FPS                             Line 6 - Model units (valid options are "FPS" or "MKS")
 0, 0, 0, 0, 0, 0, 0, 0,        Line 7 - print options: lprog1,lband,lprog3i,lprog3 and
limitp,limitt,nsubp,npmass
 0,100,                         Line 8 - NTSUBD,INTERM Time step sub-division
 0,0,                           Line 9 - ITT (initial or restart time step),ITSTOP(0=ALL)
 1, 1, 0, 0,                    Line 10 - NHEAD_NEW (1=y 0=old H),IOPT_HEAD(save H 0=n
1=y),NCYES,NDYES(1=y,0=n)
 4,4,                           Line 11 - NDIGH,NDIGC ** digits in head & conc output
 1.00,                          Line 12 - Implicit (=1.0) CRANK-NICOLSON (0.5-1.0)
* Following parameters used for transport runs only for flow analysis only specify defaults
1.00,                           Line 13 - UPSTRM (0.0 to 1.0) Upstream factor
1.00,0.000,                     Line 14 - RETARD,HALF-LIFE in simulation units
 3,                             Line 15 - NCROSS (0=norm,1=enhanced,2=same as 1 but exit, 3=cross
deriv. to 0
5,0.001,                        Line 16 - ITRHO (Max. rho iter.), RHO convergence (0.01-.0001)
1.E-06,1.E-06                   Line 17 - RPARM1_H, RPARM1_C iterative solver convergence factors
1,                              Line 18 - ITRANS (0=confined, 1=unconfined, 2=combination of both)
* Following used if ITRANS > 0
10,0.1,0.1,                     Line 19 - IWT_TIMES,ERR_TRANS,THICK_MIN
0,                              Line 20 - IOPT_K (0=normal,1=unsat_k.tab)
* Options to check nodal fluid and mass balance
0,0,0.100000E-00,0.100000E-00, Line 21 -
IOPT_MASSFILE,IMASS_ERR,FLUID_NODAL_ERRMIN,SOLUTE_NODAL_ERRMIN
1.0,                            Line 22 - FACT_VOL factor for printing fluid volume
cu. m                           Line 23 - FUNITS_VOL text string used for printing fluid volume
kg,                             Line 24 - Mass units in density term used for printing
seconds,                        Line 25 - time units(e.g. sec, days)
1.0,                            Line 26 - FACT_AREA used for printing only
sq,m.                           Line 27 - Area Units used for printing only
0,5,                            Line 28 -IACCEL(NSPCG Accelerator),Itmax_fact
```

Figure 3.4. Example *cfest.ctl* Input File Used to Identify Input Filenames, Output Binary Directory, Simulation Switches, and Solver Parameters

3.3.1 Version, *cfest.ctl* Print Control and Log Filename

Although line numbering in the control file only appears in comments, line numbering usually begins on the fourth line of the *cfest.ctl* file. The first "unnumbered" three lines are used to identify the CFEST version, print control option for *cfest.ctl*, and a log file as shown below:

```
'cfest_005',                Important this version number should match code version
0,                          nprint_ctl
'test005.qa',                   IMPORTANT ***     This filename should be in single quotes
```

The version number in this example is cfest_005 and must match the code version. With the print level set to 0, only the *.lp1* and *.l3i* file names will be printed to the screen. The ".qa" file is used to log the execution of each of the modules. If a module is executed multiple times, this information will be logged to this file. Text strings must be bracketed with single quotation marks. These parameters and their description are summarized in Table 3.58.

Table 3.58. Version, *cfest.ctl* Print Control and Log Filename Description

Parameter	Options	Description
VERSION	n	CFEST version; e.g., 'cfest_005'
NPRINT_CTL	0	Prints to screen only the *.lp1* and *.l3i* file names listed in *.cfest.ctl* file
	1	Prints to screen uncommented lines in the *cfest.ctl* file
	2	Prints to screen complete echo of *cfest.ctl* file, including comment lines
FILE_LOG	c	Prints status of completed modules by appending to named file

3.3.2 Lines 1–6: Input File Names and Binary Output Directory

The first six lines with line numbers in the *cfest.ctl* file list the directory in which binary file names are to be stored, the input file names, and model units. Example input lines appear below:

```
'tmp/test005',                    Line 1 - path and name for binary & ascii output files
'test005.lp1',                    Line 2 - LPROG1 input file name
'test005.l3i',                    Line 3 - LPROG3I input filename
'headbc.hld',                     Line 4 - time dependent head B.C. file name
'concbc.dat',                     Line 5 - time dependent conc B.C. file name
FPS                          Line 6 - Model units (valid options are "FPS" or "MKS")
```

In this example, binary output files will be stored in the *tmp* subdirectory of the current directory with a filename prefix of test005 (see Section 4.2). The next two lines name the *.lp1* and *.l3i* file, and lines 4 and 5 name head and concentration filenames that contain the time-dependent fixed head and concentration data. Data in these files follow the same input format and node order as the fixed boundary conditions listed in the *.lp1* file (see subsection 3.1.4.1). These data must be provided for each time step. In addition, the NHBOND and NCBOND flags in the *.l3i* file must also be set to 1 (Section 3.2.4). If these flags are not set in the *.l3i* file, the files named in the *cfest.ctl* file are not read and no error will occur.

Line 6 of the *cfest.ctl* file names model units. Two alternatives exist, "FPS" and "MKS," but these units are only used for printing and have no effect on internal computations. Unit consistency is not maintained internally in CFEST, and it is the user's responsibility to ensure that consistent units are used with all data inputs.

3.3.3 Line 7: Print Control Options for .prn Output Files

Print control (NPRINT) is specified on line 7 of the *cfest.ctl* file. Parameters that control output for *lprog1, lband, lprog3i* and *lprog3* are specified on this line. Additional debug output can be requested by also setting the LIMITP, LIMITT, NSUBP and NPMASS parameters. Print levels, however, do not yield the same level of output for each of the modules. For example, NPRINT = 3 generates different data to the output (*.prn*) files for each of the modules. As a result, the different print levels are not outlined in this section of the manual but in the Output File Description in Section 5.

Because of the large amounts of data that can be generated, in general, NPRINT = 0 is the recommended level of printing because it prints the least amount of data to the output files. Increasingly higher levels can be set for identifying problems in a simulation. An example input line is found below:

```
0, 0, 0, 0, 0, 0, 0, 0,              Line 7 - print options: lprog1,lband,lprog3i,lprog3 and
limitp,limitt,nsubp,npmass
```

In this example, minimum printing to the *.prn* files will occur.

3.3.4 Lines 8–9: Time Step Controls

For numerical reasons, it may be necessary to take smaller time steps for transport than those used for fluid flow. Lines 8 and 9 of the *cfest.ctl* file provide data on time steps. In line 8, NTSUBD sets the number of subdivisions for each elapsed time interval in the *.l3i* file for transport. Unlike the IOPT option in the *.l3i* file, the subdivisions set with the *cfest.ctl* file are applied uniformly to each time step. This time step subdivision set in the *cfest.ctl* file is overwritten by any subdivisions set in the *.l3i* file. The INTERM parameter, which also appears on line 8, controls information sent to the screen for the subdivided time steps. Screen output normally captures time step number, time step size, and simulation time, as well as the maximum and minimum values for head and concentration for the elapsed times listed in the *.l3i* file. The parameter INTERM specifies the time step number from the *.l3i* file at which this information is printed to the screen for the subdivided time steps.

Line 9 is used for restart simulations (see Program Execution, Section 6). For ITT > 1, the *lprog3* module will restart the simulation at the time step number specified by ITT. The ITSTOP parameter, which also appears on the same line, can be used to limit the number of time steps that a simulation will run. For example, if 50 time steps are listed in the *.l3i* file, the user could set ITT = 25, and the simulation would terminate normally after completion of time step 25. If the simulation is to run to completion, then ITSTOP should be set to 0 so that all time steps in the *.l3i* file are simulated. Example input lines for time step controls appear below:

```
2,100,                               Line 8 - NTSUBD,INTERM Time step sub-division
 3,0,                                Line 9 - ITT (initial or restart time step),ITSTOP(0=ALL)
```

In this example, each transport time step is subdivided by 2, and unless the simulation runs at least 100 time steps, no screen output will be printed for the subdivided time steps. This simulation will restart on time step 3 and run the number of time steps listed in the *.l3i* file. These parameters and their description are summarized in Table 3.59.

Table 3.59. Time Step Control Description

Parameter	Options	Description
NTSUBD	0/1	No time step subdivisions
	n	Each transport time step subdivided by n
INTERM	n	Time step number at which screen output is to begin being generated for subdivided time steps
ITT	n	Restart simulation at nth time step
ITSTOP	n	Terminate simulation at nth time step

3.3.5 Line 10: Simulation Switches

Line 10 of the *cfest.ctl* file sets several simulation switches for flow and transport. The NHEAD_NEW switch is used to read a previously generated transient head solution that can be used for transport runs. This approach has the advantage of solving the flow equation only once and then using it on multiple transport runs. There are three options for the NHEAD_NEW switch. When NHEAD_NEW = 0, the head solution is read from the *.b44* binary file. If the binary solution does not exist, this switch results in a runtime error. When NHEAD_NEW = 1, the head solution is calculated for each time step. A third option, NHEAD_NEW = -1 also exists. This setting is similar to the NHEAD_NEW = 0 option but uses the last saved record as a steady-state head solution for all transport time steps that exceed the ending time of the head solution saved to the binary file.

The second simulation switch that appears on line 10 is also related to the "decoupling" of the head and transport solutions. For IOPT_HEAD = 1, the transient head solution is saved. IOPT_HEAD = 0 turns off this option, and the head solution is not saved to the *.b44* binary file. A flow simulation with IOPT_HEAD = 1 must be completed, and the binaries must be present in the directory specified in line 1 before the NHEAD_NEW = 0 or NHEAD_NEW = -1 options can be used for a transport simulation.

The third simulation switch is similar to the NCCYES switch that appears in the *.lp1* input file (subsection 3.1.2.1). When NCCYES = 1, transport is simulated, whereas NCCYES = 0 turns off transport. This switch provides the flexibility to develop *.lp1* and *.l3i* input files for both flow and transport but execute only flow in an initial, exploratory run. The fourth simulation switch, NDYES, is used in a similar manner but for salt dissolution and also overrides the NNDIS switch that appears in the *.lp1* file. These parameters and their descriptions are summarized in Table 3.60. An example input line appears below:

```
1, 1, 0, 0,        Line 10 - NHEAD_NEW (1=y 0=old H),IOPT_HEAD(save H 0=n 1=y),NCYES,NDYES(1=y,0=n)
```

In this example, the head solution is being calculated and saved to the *.b44* file. No transport or salt dissolution is considered.

Table 3.60. Simulation Switch Description

Parameter	Options	Description
NHEAD_NEW	-1	Do not calculate new head solution; Read saved head solution from previous run and use the last saved record as a steady-state solution for transport time steps that extend beyond the last time step of the flow solution.
	0	Do not calculate new head solution; read saved head solution from previous run
	1	Calculate new head solution
IOPT_HEAD	0	Do not save head solution
	1	Save head solution for future transport runs
NCYES	0	Do not simulate transport
	1	Execute transport simulation
NDYES	0	Do not simulate salt dissolution
	1	Simulate salt dissolution

3.3.6 Lines 11–12: Screen Print Control and Crank-Nicholson Parameter

Line 11 of the *cfest.ctl* file allows the user to set the number of significant digits for head and concentration values printed to the .prn file (Table 3.61). In the example shown below, four significant digits will be used to print the maximum and minimum values of both head and concentration:

```
4,4,                    Line 11 - NDIGH,NDIGC ** digits in head & conc output
 1.00,                  Line 12 - Implicit (=1.0) CRANK-NICOLSON (0.5-1.0)
```

As shown in this example, the user may also specify the value of the weighting parameter (CRANK). For CRANK = 1, the spatial derivatives are approximated only for the current time step, and the differencing scheme is said to be fully implicit. When CRANK = 0, the values of the spatial derivatives are approximated by the previous time step. For CRANK = 0.5, the spatial derivatives are best approximated halfway between the current and previous time levels, and second-order accuracy in time can be achieved. Although the fully implicit method attains only first order accuracy in time, CRANK = 1 is generally recommended for stability. However, if greater accuracy in time is desired, CRANK = 0.5 may be used as long as oscillations in the solution do not occur (see Table 3.61).

Table 3.61. Screen Print Control and Crank-Nicholson Parameter Description

Parameter	Options	Description
NDIGH	n	Prints n digits of head solution to *.prn* output files
NDIGC	n	Prints n digits of concentration solution to *.prn* output files
CRANK	0.0	Explicit solution
	0.5	Crank-Nicholson solution
	1.0	Fully Implicit solution

3.3.7 Lines 13–16: Transport Parameters

Lines 13–16 contain data related to transport: an upstream weighting factor, solute retardation and half-life, cross-terms for dispersive-diffusive fluxes, and convergence criteria for density (see Table 3.62). An example of these inputs is shown below:

```
1.00,          Line 13 - UPSTRM (0.0 to 1.0) Upstream factor
1.00,0.000,    Line 14 - RETARD,HALF-LIFE in simulation units
 3,            Line 15 - NCROSS (0=norm,1=enhanced,2=same as 1 but exit, 3=cross deriv. to 0
5,0.001,       Line 16 - ITRHO (Max. rho iter.), RHO convergence (0.01-.0001)
```

Although the user may specify the value of the upstream weighting parameter (UPSTRM), it is recommended that this factor be set to 1.0. Higher-order solutions for advective transport can be achieved with central differencing (i.e., UPSTRM = 0.5), but oscillations in the transport solution may result.

The solute retardation factor is set on line 14 of the *cfest.ctl* file. The retardation factor is calculated by assuming that a linear isotherm and a constant distribution coefficient, (K_d) [mLg^{-1}], represents sorption-desorption on the sediments:

$$R_f = 1 + \frac{K_d \rho_b}{\theta} \tag{3.1}$$

where R_f is the retardation factor, θ is the porosity, and ρ_b is the bulk density of the porous media. When a solute is unretarded, then the RETARD parameter is set to 1.

The first-order radioactive decay constant for both the free and sorbed solute, λ [s^{-1}], is also set on this line of the *cfest.ctl* file and is defined as

$$\lambda = \frac{\ln(2)}{t_{1/2}} \tag{3.2}$$

where $t_{1/2}$ is the solute half-life, the time for the chemical to decay to half its original concentration. Radioactive decay is not considered when HALF_LIFE = 0.0.

Numerical dispersion, oscillatory solutions and non-positive concentrations can be problematic for transport simulations. Specifically, negative values of concentration may be generated and can yield several problems, especially when coupling flow with transport. The use of cross-dispersion terms can cause these oscillations and negative concentrations. Hence, line 15 of the *cfest.ctl* file allows the user to select whether cross-terms (NCROSS) are used in the transport solution (Table 3.62). For NCROSS = 3, no cross terms are used. For NCROSS = 0, only dxy, dxz and dyz cross-terms are used to calculate dispersion. For NCROSS = 1 and NCROSS = 2, all cross-terms are used but set to zero for NCROSS = 2 on an outflow boundary. It is recommended that NCROSS = 1 or NCROSS = 3 be used for most applications.

Line 16 specifies two parameters that are used when calculating density in a coupled flow and transport simulation. ITRHO specifies the maximum number of iterations to use when calculating fluid density, whereas DENMAX sets the convergence tolerance. A recommended tolerance is $0.0001 \leq$ DENMAX ≤ 0.01.

Table 3.62. Transport Parameter Description

Parameter	Options	Description
UPSTRM	$0 \leq n \leq 1$	Upstream weighting factor for advective transport
RETARD	n	Retardation factor for sorbed solutes
HALF_LIFE	n	Half-life for radioactive solutes
NCROSS	0	Cross terms dxy, dxz and dyz used for dispersion
	1	All cross terms used
	2	All cross terms used, but no cross-dispersion terms used at exit boundary
	3	No cross-dispersion terms used
ITRHO	n	Maximum number of iterations used to calculate fluid density
DENMAX	n	Convergence tolerance for calculating fluid density

3.3.8 Line 17: Relative Convergence Criteria

All iterative solvers require stopping criteria. In CFEST, the relative convergence criteria for both head and concentration (see Table 3.63) can be set for the ITPACK and the nonsymmetric preconditioned conjugate gradient (NSPCG) software packages, as shown below:

```
1.E-06,1.E-06          Line 17 - RPARM1_H, RPARM1_C iterative solver convergence factors
```

Generally, a relative convergence criterion of 1.0E-06 is recommended as a minimum for both head and concentration solutions.

Table 3.63. Relative Convergence Criteria for Solver

Parameters	Options	Description
RPARM1_H	n	Sets relative convergence criterion for head; used in the ITPACK solver
RPARM1_C	n	Sets relative convergence criterion for concentration; used in NSPCG solver

3.3.9 Lines 18–20: Unconfined Aquifer Parameters

Because the governing equation for ground water flow differs for confined and unconfined aquifers, the user must enter on line 18 of the *cfest.ctl* file ITRANS = 0 for confined aquifers, ITRANS = 1 for unconfined, and ITRANS = 2 for combined. For ITRANS = 2, a file that contains a list of elements that are confined (confine.lst), and a file that contains a list of elements that are unconfined (unconfin.lst) must also be present in the current run directory. The element lists may be listed in free format.

Phreatic (unconfined) solutions are computed through an iterative technique that adjusts the saturated thickness so that the calculated head is at the top of the system. In the *cfest.ctl* file, the user specifies the maximum number of iterations for elevation adjustment to head. The user also specifies the maximum head difference between iterations (ERR_TRANS).

If there is an increase in the elevation of the water table, the elements in the model domain are stretched vertically. Conversely, if the water table drops, the elements are compressed vertically into a pinched layer. If the mesh contracts through layer boundaries, the initial hydrostratigraphy is preserved by collapsing the hexagonal elements so that the vertical distance between the two z planes is reduced by a minimal user-defined thickness, THICK_MIN, if the IOPT_K parameter on line 20 is set to 0. For IOPT_K = 1, instead of compressing the elements, the hydraulic conductivity of "dry" elements is altered using data in the *unsat.dat* file in the current run directory. This option, however, has not yet been tested; hence, no further documentation is provided.

The collapsed layers temporarily adopt the same material properties of the current saturated layer. Because the elements are not eliminated from the domain, if the water table rises again, the collapsed elements are expanded so that the hydraulic head matches the water table elevation. Example input for lines 18–20 is shown below:

```
1,                     Line 18 - ITRANS (0=confined, 1=unconfined, 2=combination of both)
* Following used if ITRANS > 0
10,0.1,0.1,            Line 19 - IWT_TIMES,ERR_TRANS,THICK_MIN
0,                     Line 20 - IOPT_K (0=normal,1=unsat_k.tab)
```

In this example, the aquifer is unconfined. The maximum number of iterations used in calculating the phreatic surface is 10, the maximum difference between head iterations is 0.1, and the minimum element thickness is 0.1. These parameters are described in Table 3.64.

Table 3.64. Unconfined Aquifer Parameters

Parameter	Options	Description
ITRANS	0	Confined aquifer
	1	Unconfined aquifer
	2	Both confined and unconfined; user must provide list of elements to be used for each condition
IWT_TIMES	n	Sets maximum number of iterations for defining phreatic surface; 3–5 iterations is recommended
ERR_TRANS	n	Maximum head difference between unconfined aquifer iterations; recommended tolerance is $0.01 \leq$ ERR_TRANS ≤ 0.1
THICK_MIN	n	Minimum thickness to maintain for "collapsed" elements if IOPT_K = 0
IOPT_K	0	Collapse element using minimum thickness specified by THICK_MIN
	1	Hydraulic conductivity of "dry" elements is altered using data in *unsat.dat* file in current directory

3.3.10 Line 21: Mass Balance Checks

Mass balance checks can be performed by substituting the solution and checking for residual error. Unless debugging is required, the additional computation resulting from these operations is not recommended. Line 21 from the example input file is shown below:

```
0,0,0.100000E-00,0.100000E-00,    Line 21 -
IOPT_MASSSFILE,IMASS_ERR,FLUID_NODAL_ERRMIN,SOLUTE_NODAL_ERRMIN
```

In this example, IOPT_MASSFILE = 0, which means no additional output files are generated. For IOPT_MASSFILE = 1, mass balance information is written to *cf_tmpbinary/*.zzz*.

Also shown in line 21 of the input file is the IMASS_ERR flag, in this example, set to 0. In this case, no mass balance checks are performed. However, for IMASS_ERR = 1, mass balance is checked by resubstituting the solution and checking the residual error. Results from this mass balance check are saved to *cf_tmpascii/fluidbal.err* for flow and *cf_tmpascii/massbal.err* for solute using the user-defined minimum values (FLUID_NODAL_ERRMIN and SOLUTE_NODAL_ERRMIN) for saving results to the output files (see Table 3.65).

Table 3.65. Mass Balance Check Parameters and Description

Parameter	Options	Description
IOPT_MASSFILE	0	No additional output files are generated
	1	Writes mass balance information to cf_tmpbinary/*.zzz
IMASS_ERR	0	No mass balance checks performed and no additional output files are generated
	1	Performs mass balance checks by resubstituting the solution and checking the residual error. Writes mass balance information to *cf_tmpascii/fluidbal.err* and *cf_tmpascii/massbal.err*. Also writes matrix and right-hand side vector to c*f_tmpbinary/scratch.bin*
FLUID_NODAL_ERRMIN	n	User-defined minimum value for saving errors for flow
SOLUTE_NODAL_ERRMIN	n	User-defined minimum value for saving errors for solute mass

3.3.11 Lines 22–27: Model Units Used for Printing

Lines 22–27 provide CFEST with a collection of conversion factors (FACT_VOL and FACT_AREA) and text strings (FUNITS_VOL, FUNITS_MASS, FUNITS_TIME and FUNITS_AREA) used for printing (see Table 3.66). None of these parameters affect computations within the code, and are *only* used for printing. Example input lines appear below:

```
1.0,                    Line 22 - FACT_VOL factor for printing fluid volume
cu. m                   Line 23 - FUNITS_VOL text string used for printing fluid volume
kg,                     Line 24 - Mass units in density term used for printing
seconds,                Line 25 - time units(e.g. sec, days)
1.0,                    Line 26 - FACT_AREA used for printing only
sq,m.                   Line 27 - Area Units used for printing only
```

In this example, text strings for volume, mass, time and area are set to m^3, kg, seconds and m^2. No output conversions on volume and area are made since both of these factors are set to 1.

Table 3.66. Model Units Used for Printing

Parameter	Options	Description
FACT_VOL	n	Factor used to convert volume in computational units to units desired for printing
FUNITS_VOL	n	Text string used for printing fluid volume
FUNITS_MASS	n	Text string used for printing mass units for density in the *.lp1* file
FUNITS_TIME	n	Text string used for printing time units
FACT_AREA	n	Surface area conversion factor used for printing
FUNITS_AREA	n	Text string used for printing area units

3.3.12 Line 28: Solver Parameters

The last line of the *cfest.ctl* file sets the solver, or in NSPCG parlance the accelerator, to be used in the transport solution. The NSPCG software is publicly available and was developed at the Center for Numerical Analysis at The University of Texas at Austin for solving large sparse systems of linear algebraic equations (Young and Kincaid 1981). The NSPCG user's guide (Oppe et al. 1988) can be found at http://rene.ma.utexas.edu/CNA/NSPCG/manuals/usernsp/.

The parameters in the example input line below set IACCEL and ITMAX_FACT (Table 3.67):

```
0,5,                          Line 28 -IACCEL(NSPCG Accelerator),Itmax_fact
```

In this example, the BCGS solver is selected as the base method (IACCEL_C = 0) for solving transport equations. Other solvers are implemented if the *solver.file* file is in the current directory (Section 3.4.2) with IRETRY specified as greater than 0 in the *solver.file* file. Otherwise, the selected method is the only solver that will be implemented. The maximum number of iterations is set by multiplying the number of unknowns by ITMAX_FACT.

Table 3.67. Solver Parameters and Description

Parameter	Options	Description
IACCEL_C	0	BCGS
	1	LANMIN
	2	OMIN
ITMAX_FACT	n	n × (number of nodes) is the maximum number of iterations allowed in the concentration solution

3.4 Optional Input Files

In addition to the three primary input files (*.lp1*, *.l3i* and *cfest.ctl*), other input files may also be used to execute a CFEST simulation; these input files are optional but may be required if an option has been selected with a file requirement. For example, for IOPT_K = 0, data containing unsaturated hydraulic properties must be present in a file named *unsat.dat*. The CFEST main module, *lprog3*, only reads the other optional input files such as the *area_flx_file.lst* and *solver.file* files if present in the current run directory. If these files are not present, no error will result even though the inputs may be necessary to accurately describe the conceptual model (Section 3.4.1) or maintain numerical stability (Section 3.4.2).

3.4.1 Area Flux Files (area_flx.file.lst)

Fluxes that are distributed throughout a vertical stack of nodes can also be simulated in CFEST. This boundary flux is useful for describing nonuniform fluxes across planes. Given a series of surface nodes, CFEST identifies the vertical node distribution and the area associated with each node, and then distributes the total flux by the transmissivity of the vertical column.

To use this option, the user must provide a file that contains the total number of surface nodes over which the flux is distributed, the total flux, and a node list (see Table 3.68). The data are entered as free format and comma-delimited and must contain the total number of nodes and total flux on the first line and the node list on the second line:

```
3, 1207.00,
878,881,883,
```

The user specifies the filename in which these data are saved. However, an *area_flx.file.lst* file that names the user-specified file must exist within the current run directory. For example, if the above data were saved in a file called *east.flx*, the contents of the *area_flx.file.lst* would be

```
1,
east.flx
```

Multiple files may be named in the *area_flux.file.lst* file. The total number of files to be read must be specified on the first line followed by a comma, and each filename must be listed on a separate line:

```
3,
east.flx
n-east.flx
south.lst
```

Table 3.68. Area Flux File Parameters and Description

Parameters	Description
FILENAME	Filename containing surface node numbers and flux volume
NT_FLXQAREA(IFILE)	Number of surface nodes to be assigned a specified flux volume
FLXQAREA(IFILE)	Flux volume to be distributed across vertical column(s) of nodes
NOD_FLXQAREA(J,IFILE)	Surface nodes that define vertical location(s) where flux is to be distributed

3.4.2 Solver File (solver.file)

CFEST uses the NSPCG computer package (Oppe et al. 1988) to solve large sparse systems of linear equations by iterative methods. Using NSPCG requires selecting a storage mode for the matrix that limits the choice of preconditioners and accelerators (solvers). Solver choice is based on the characteristics of the matrix (e.g., symmetric positive definite, mildly nonsymmetric). In addition, the choice of preconditioner, accelerator, and storage mode are linked. CFEST uses a nonsymmetric, compressed storage format for sparse matrices, which limits the preconditioners to Richardson's, Jacobi, Least Squares Polynomial, and Neumann methods (Table 3.69). Testing has determined that the Jacobi method is the most robust; hence, this is the only preconditioner used for solving the transport equation. Testing of the solvers (accelerators) has shown that the BCGS, LANMIN and OMIN approaches perform best. The user can specify in the *cfest.ctl* file the base solver (IACCEL_C = 0, 1 or 2) to be used in the simulation.

Testing has determined that BCGS is the most robust solver, but it is extremely expensive computationally. It requires a large number of (expensive) iterations to attain convergence; at least five times the number of unknowns. OMIN, however, is not only fast on a per iteration basis but nonconvergence occurs quickly. This difference in solver performance is so large that CFEST has implemented additional controls on setting solver parameters. This is accomplished with the use of an optional input file named

solver.file. This file not only allows the user to specify solver parameters, but also specify parameters that control the logic used when convergence is not obtained. An example *solver.file* file appears below:

```
# Lines starting with # are ignored and can be used to make comments and document choices.
# Input format is free format so comments can be placed after the last input.
1 ITERPRINT (if 1 print iteration and timing information to standard out, if 0 do not print)
1 IRETRY (if 1 retry with OMIN to speed up solution, if 0 do not try)
2 IRETRYINTERVAL (no. time/sub-timesteps to use the base accelerator before retrying with OMIN)
600 IRETRYMAXITER (the maximum iteration count for OMIN when using it for speedup tries)
1.0 CONVRG_1_RATIO
1.0 CONVRG_2_RATIO
10 13580 0 88 0 1 1 1 30 1000 40 5 1 0 1 0 0 2 -1 -1 0 1 1 0 1,1 OMIN's IPARM Values and ULAST2
10 13580 0 88 0 1 1 1 5 100000 0 5 1 0 1 0 0 2 -1 -1 0 1 1 0 1,1 BCGS's IPARM Values and ULAST0
10 13580 0 88 0 1 1 1 5 100000 0 5 1 0 1 0 0 2 -1 -1 0 1 1 0 1,1 LANMIN's IPARM Values and ULAST1
```

The OMIN accelerator is always implemented first for IRETRY = 1, using the maximum iteration count specified by IRETRYMAXITER. If OMIN fails, CFEST uses the next base solver (e.g., BCGS or LANMIN) that was specified by IACCEL_C in the *cfest.ctl* file. The OMIN solver is implemented again once the base solver has successfully converged on IRETRYINTERVAL time steps. If the selected base method (IACCEL_C) also fails, the third solver is used. Hence, for IRETRY=1, convergence failure only occurs when all three methods have failed to converge in the same time step. For ITERPRINT = 1, the timing and the number of iterations is printed to the screen. These parameters are listed in Table 3.70.

Two parameters set in *solver.file* use the ratio of the achieved convergence to the relative convergence criterion (RPARM1_C) (see Section 3.3.8) when a solver fails to converge to a solution. If this ratio is less than CONVRG_1_RATIO, the last solution attained is used as the starting value in the next solver routine. When this same quotient is greater than CONVRG_2_RATIO, the solution is accepted as being converged, and CFEST proceeds to the next time step. Table 3.70 describes the parameters.

In the final part of the *solver.file* file, the 25 IPARM values are defined on a single line for each solver (OMIN, BCGS and LANMIN). IPARM is an integer vector of length 25 containing various integer parameters including stopping test criteria, preconditioner and acceleration flags, and output controls. These parameters are described in Table 3.71. The 26th entry on each of the solver lines is a flag that sets the initial guess for the new solver. For IULAST = 0, the initial guess is set to 0. When IULAST = 1, the initial guess is the final solution from the nonconverged solver routine. Hence, if ULAST = 1, the final solution will always be used as an initial guess, whereas the CONVRG_1_RATIO parameter performs the same function only if the ratio of the achieved convergence to the relative convergence criterion is less than CONVRG_1_RATIO.

Table 3.69. Permitted Preconditioner and Accelerator Combinations[a]

Accelerators (solvers)	**Preconditioners**																		
	RICHi	JACi	LJACi	LJACXi	SORi	SSORi	ICi	MICi	LSPi	NEUi	LSORi	LLSPi	LNEUi	BICi	BICXi	BICXi	MBICi	MBICXi	RSi
OMIN	×	×	×	×		×	×	×	×	×	×	×	×	×	×	×	×	×	×
LANMIN	×	×	×	×		×	×	×	×	×	×	×	×	×	×	×	×	×	×
BCGS	×	×	×	×		×	×	×	×	×	×	×	×	×	×	×	×	×	×

(a) Shading indicates that the preconditioner-accelerator combination can be used with compressed storage format for sparse matrices.

Table 3.70. Solver Parameters and Description

Parameter	Options	Description
ITERPRINT	0	No solver printout to screen
	1	Prints to screen the number of iterations and time spent in the solver
IRETRY	0	Use base solver specified by IACCEL_C in *cfest.ctl* file
	1	Use base solver only when the one specified by IACCEL_C in *cfest.ctl* file fails. Retry OMIN once the base solver has successfully converged for IRETRYINTERVAL time steps. Attempt third solver if other two solvers have failed to converge. Hence, convergence failure only occurs if all three solvers have failed to converge on the same time step.
IRETRYINTERVAL	n	Number of time steps to solve with base solver before using OMIN solver
IRETRYMAXITER	n	Maximum number of iterations for OMIN for IRETRY = 1
CONVRG_1_RATIO	n	Threshold value *n* used to compare the ratio of the achieved convergence to the relative convergence criterion (RPARM1_C, set in the *cfest.ctl* file). If ratio is less than *n*, then the last solution is used as the starting value in the next solver method.
CONVRG_2_RATIO	n	If the ratio of the achieved convergence to the relative convergence criterion (RPARM1_C, set in the *cfest.ctl* file) is greater than *n,* accept the solution as converged.

Table 3.71. IPARM Parameters in the *cfest.ctl* file

Parameter	Options	Description	Condition
IPARM(1) or NTEST	1-10	Stopping test number for mathematical descriptions) (http://rene.ma.utexas.edu/CNA/NSPCG/manuals/usernsp/node12.html#stop (NTEST = 2 is default)	
	6	The SOR accelerator uses a specialized stopping test, so this parameter is ignored when SOR is called unless NTEST = 6, when the exact stopping test is used.	
IPARM(2) or ITMAX	100	Maximum number of iterations allowed on input to solver (100 is the default value). On output, ITMAX is the number of iterations performed.	
IPARM(3) or LEVEL	<0	No output.	
	0	Print fatal error message only (LEVEL = 0 is default).	
	1	Print warning messages and minimum output.	
	2	Print reasonable summary.	
	3	Print parameter values and comments.	
	4	Print approximate solution after every iteration.	
IPARM(4) or NOUT	n	Fortran unit number for output (NOUT = 6 is default).	
IPARM(5) or IDGTS	<0	IDGTS < 0, skip error analysis of the final computed solution.	
	0	Perform error analysis of the final computed solution to determine accuracy. Compute DIGIT1 (measure of approximate number of digits of accuracy of the final solution) and DIGIT2 (approximate number of digits of accuracy using estimated relative residual with the final approximate solution) and store in vector.	If LEVEL <1, no printing occurs

Table 3.71 (contd)

Parameter	Options	Description	Condition
IPARM(5) or IDGTS (contd)	1	Perform error analysis and print DIGIT1 and DIGIT2.	If LEVEL <1, no printing occurs
	2	Perform error analysis and print final approximate solution vector.	
	3	Perform error analysis and print final approximate residual vector.	
	4	Perform error analysis and print final approximate solution and residual vectors.	
IPARM(6) or MAXADP	0	No adaptive procedures are performed for EMAX, the maximum eigenvalue estimate of $Q^{-1}A$.	
	1	Perform adaptive procedures on EMAX, the maximum eigenvalue estimate of $Q^{-1}A$ (MAXADP = 1 is default).	
IPARM(7) or MINADP	0	No adaptive procedures are performed for EMIN, the minimum eigenvalue estimate of $Q^{-1}A$.	
	1	Perform adaptive procedures on EMIN, the minimum eigenvalue estimate of $Q^{-1}A$ (MINADP = 1 is default).	
IPARM(8) or IOMGAD	0	Perform no adaptive procedures on OMEGA, the over-relaxation parameter for the SOR and SSOR methods.	
	1	Perform adaptive procedures on OMEGA, over-relaxation parameter for SOR and SSOR methods (IOMGAD = 1 is default).	
IPARM(9) or NS1	n	Number of old vectors to be saved for the truncated acceleration methods (NSI = 5 is default).	
IPARM(10) or NS2	n	Frequency of restarting for the restarted acceleration methods. By default, NS2 is set to a large value to ensure that restarting is not performed (NS2 = 100000 is default).	
IPARM(11) or NS3	n	Used only in ORTHOMIN; Denotes the size of the largest Hessenberg matrix to be used to estimate the eigenvalues. Recommended value is NS3 = 40, but NS3 = 0 is default.	
IPARM(12) or NSTORE	1	Primary format storage mode is used.	
	2	Symmetric diagonal format storage mode is used (NSTORE = 2 is default).	
	3	Nonsymmetric diagonal format storage mode is used.	
	4	Symmetric coordinate format storage mode is used.	
	5	Nonsymmetric coordinate format storage mode is used.	
IPARM(13) or ISCALE	0	No scaling of matrix is performed before iterating (ISCALE = 0 is default).	
	1	Matrix is scaled to have a unit diagonal, and solution vector and vector of unknowns are scaled accordingly. If NTEST = 6, the approximate vector of unknowns is also scaled.	
IPARM(14) or IPERM	0	No permutations on matrix are performed (IPERM = 0 is default).	
	1	Permutations on matrix are performed.	Vector P must contain a coloring vector

Table 3.71 (contd)

Parameter	Options	Description	Condition
PARM(15) or IFACT	0	No factorization for current call (previous factorization used).	
	1	Matrix *A* is factored for current call (IFACT = 1 is default).	
IPARM(16) or LVFILL	0	No fill is allowed beyond the original matrix nonzero pattern (LVFILL = 0 is default).	
	1	Fill is allowed by the original nonzero pattern, but no further fill is allowed by the just filled-in elements.	
	2	Fill is allowed if it is due to the original pattern or LVFILL = 1 filled-in elements.	
IPARM(17) or LTRUNC	n	Truncation bandwidth to be used when approximating the inverses of matrices with dense banded matrices. In general, an increase in LTRUNC means more accurate factorization at the expense of increased storage (LTRUNC = 0 is default).	
IPARM(18) or IPROPA	0	Matrix does not have point or block property associated with the selected preconditioner.	
	1	Matrix has the point or block property associated with the selected preconditioner.	
	2	Not known if matrix has the point or block property associated with the selected preconditioner. Compute if needed (IPROPA = 2 is default).	IPARM(16) = 0
IPARM(19) or KBLSZ	n	1-D block size used in line preconditioners. It is the largest integer such that, if matrix *A* is considered as a block matrix, diagonal blocks have dense bands (KBLSZ = -1 is default).	
IPARM(20) or NBL2D	n	2-D block size used only for CGCR acceleration applied to 3-D problems on a box domain (NBL2D = -1 is default).	
IPARM(21) or IFCTV	0	Use scalar routine for the incomplete factorization of matrix stored in symmetric or nonsymmetric diagonal storage mode.	
	1	Use vectorized routine for the incomplete factorization of matrix stored in symmetric or nonsymmetric diagonal storage mode (IFCTV = 1 is default). Vectorized routine performs better for matrix factorization patterns with property associated with selected preconditioner.	IPARM(18) = 1
IPARM(22) or IQLR	0	No basic preconditioner is used.	
	1	Left preconditioning is used (IQLR = 1 is default).	
	2	Right preconditioning is used.	
	3	Split preconditioning is used.	
IPARM(23) or ISYMM	0	Matrix is symmetric	
	1	Matrix is nonsymmetric.	
	2	Matrix symmetry not known and NSPCG should make this determination (ISYMM = 2 is default).	IPARM(12) = 1
IPARM(24) or IELIM	0	Do not remove rows and columns when the diagonal entry is extremely large compared to the nonzero off-diagonal entries in that row (IELIM = 0 is default).	
	1	Remove rows and columns when diagonal entry is extremely large compared to nonzero off-diagonal entries in that row.	
IPARM(25) or NDEG	n	Specifies degree of polynomial preconditioner (NDEG = 1 is default).	

3.4.3 Other Optional Input Files

Depending on options selected in the required input files (*.lp1, .l3i* and *cfest.ctl*) files, additional input files may be required in the current run directory. These input files include data on confined and unconfined elements, dry mass injection rates, time-dependent stream data, or unsaturated hydraulic conductivities. These input files are briefly described in Table 3.72, as well as the conditions required for their presence. These input files are free-format and hence no input file formatting is described.

Table 3.72. Optional Input File Description

Input Filename	Condition	Description	Content
unsat.dat	IOPT_K = 1	Used to simulate unsaturated flow using the van Genuchten (1980) equation for NPT elements that may dry out (unsaturated hydraulic conductivity, air entry pressure, saturated hydraulic conductivity, porosity, and residual saturation)	NPT, UNSAT, AIRINT, SATK, THETA_SAT, THETA_R
confine.lst	ITRANS = 2	List of NT elements that are confined	NT_IEL, IEL_DUM(I)
unconfin.lst	ITRANS = 2	List of NT elements that are unconfined	NT_IEL, IEL_DUM(I)
dry_mass.elm	Read if present in current run directory	List containing element number, dry mass injection rate, and flag for surface or subsurface element (IFLAG = 0 = surface element)	IIE, QM, IFLAG
streambc.out	Read if present in current run directory	Stream head and concentration data for NTRIV nodes and data format	FMT, NTRIV NRIV(I),STAGE(I), RIV_CONC(I)

4.0 Program Execution

The CFEST simulator and associated utilities are a collection of FORTRAN77 subroutines that have been concatenated to form a single source code file for each program. These source code files can be compiled and executed on either a Windows or UNIX operating system. For non-PNNL users, CFEST and CFEST utilities are distributed as executables. For PNNL users and for those with access to the source code, executables are created by compiling and linking the source code with an ANSI FORTRAN compiler. Because the source code is written in FORTRAN77, CFEST must be compiled statically. Array sizes are set within the source code, and any changes in array dimensions need to be performed within the source so that the proper amount of memory is statically allocated during compilation.

4.1 Compilation

Because CFEST programs are concatenated into a single source file, a makefile is not generally used to compile and link sources into executables. Compile and link options can be specified on the command line in conjunction with the name of the executable. For example, on a UNIX operating system and using the GNU FORTRAN compiler, the user can issue the following command to compile the *lprog1_005* preprocessor:

```
g77 lprog1_005.f –o lprog1_005
```

Other FORTRAN compilers can also be used. Presently, the Intel compiler (version 9.0) is used on a Redhat workstation Versions 3 and 4 at PNNL to compile and link CFEST source using the following command line options:

```
ifort -i_dynamic -Vaxlib -w -vms -zero -tpp7 -save -scalar_rep -rcd -O3 lprog1_005.f -o lprog1_005
```

If solver libraries are required (e.g., *lprog3_005* and *lp3ucode*), the following command line can be used to link the libraries to the CFEST modules:

```
ifort -i_dynamic -Vaxlib -w -vms -zero -tpp7 -save -scalar_rep -rcd -O3 lprog3_005.f ndsrc2c.for nspcg.for -o lprog3_005
```

The CFEST package includes support for performing both inverse (lp3ucode) and forward simulations (lprog3). The source used to perform inverse modeling interfaces with the universal inverse modeling program, UCODE (Poeter and Hill 1998), by substituting UCODE-provided parameter estimates into the simulation runs and providing results to UCODE for comparison with observations. These capabilities are described in more detail in Section 6.10. For the inverse source, a makefile is used because it also links to the PETSc solver libraries (Balay et al. 2004). By executing the *make* command, the executable is created by compiling and linking all the source object coding. To create an executable for the *lp3ucode* version *lp3ucode_pet_r3*, the following command is issued at the command line prompt:

```
make –f Makefile.petsc lp3ucode_pet_r3
```

This command produces object files with an *.o* extension for every source listed in the *makefile* for the *lp3ucode_pet_r3* executable. These object files, appearing in the current directory, are not necessary for code execution but are used for forming the executable. In this *makefile*, the PETSc and MPI libraries are also linked to the executable. PETSc requires the MPI libraries, even though CFEST is implemented in serial.

These libraries must be installed and the appropriate environmental variables set before executing the *make* command.

4.2 Execution

To execute a CFEST run, three preprocessing programs must first be executed before executing a flow or flow-and-transport simulation. Each of these preprocessing programs must be executed in the designated order, and the directory named in the *cfest.ctl* file for binary storage must be created before their execution (see Table 4.1). After executing the first preprocessor (*lprog1_005*), two directories, *cf_tmpascii* and *cf_tmpbinary*, are created. The *cf_tmpascii* directory stores ascii data files (*.prn*) for each program run within the current directory. The *cf_tmpbinary* directory stores binary data that the CFEST simulator reads.

Table 4.1. CFEST Input and Output Files

Program	Input files	Output Files			
		tmp/	***cf_tmpbinary/***	***cf_tmpascii/***	***./***
lprog1_005	*cfest.ctl.* *lp1*	*.b01* *.ear*		*lprog1.prn*	*nnelem.lis* *.qa* *cfest_runs.out*
lband_005	*cfest.ctl*	*.b01*	*ini*	*lband.prn*	*.qa* *cfest_runs.out*
lprog3i_005	*cfest.ctl* *.l3i*	*.b08*		*lprog3i.prn*	*.qa* *cfest_runs.out*
lprog3_005	*cfest.ctl* *area_flux_file.lst* *solver.file*	*.b07* *.b44 and .b45* *(archived flow)*	*.bin*	*lprog3.prn*	*.qa* *cfest_runs.out*

There are three primary input files: *cfest.ctl*, *.lp1*, and the *.l3i* file. The *cfest.ctl* file is read by all CFEST programs and utilities and must reside in the current directory. Flags that track the successful completion of each of the preprocessing programs are written to an output file called *cf_runs.out*. Concentration maximum and minimum data are also written to this file. These data are written to both the *cf_runs.out* file and the *.b01* file after each program execution. Each program also logs summary information to a *.qa* file (e.g., program name, simulation title, and the time the run is executed) each time a program is executed. This file is always appended, so multiple executions of a single program will be logged to this file.

The *lprog1_005* preprocessor is the first program to be executed. It reads data from the *.lp1* file and writes node and element data, material properties, and fixed boundary condition information to two binary files, the *.b01* and *.ear* files stored in the user-specified directory named in the *cfest.ctl* file. For this example, the directory has been named *tmp*. In addition, ascii data files logging input from the *.lp1* file are created in the *lprog1.prn* file stored in the *cf_tmpascii* directory. If the level of debug printing is set to 5 in the *cfest.ctl* control file, a list of nodes and the elements containing those nodes are written to an ascii file in the current directory (*nnelem.lis*).

Second, the *lband_005* preprocessor, which reads the data stored in the *.b01* file and sets up the sparse matrix compressed row format vectors for use in the solver, is executed. These data are written to the *.ini* files stored in the *cf_tmpbinary* directory. If the simulation only involves head, then only one *.ini* file is created (*hmatrix.ini*). If transport is also considered, a *cmatrix.ini* file is also created. Ascii data from the *lband_005* run are stored in the *cf_tmpascii/lband.prn* file.

The third preprocessor to be executed is the *lprog3i_005* program, which reads time step information and flux boundary condition data from the *.l3i* file and stores it in the *.b08* file. Ascii data files logging input from the *.l3i* file are created in the *lprog3i.prn* file stored in the *cf_tmpascii* directory.

After execution of the three preprocessing programs, the main program that performs the flow and transport calculations, *lprog3_005*, is executed. This program reads the data stored in the binary files found in the *tmp* and *cf_tmpbinary* directories. It can optionally read an *area_flux_file.lst* file, any files named in that file list, and a *solver.file* file if those files reside in the current directory (see Sections 3.4.1 and 3.4.2). If the flow and transport simulator is executed for the first time, *lprog3_005* reads data from the *.b01, .b08* and *.ear* files. If executing a restart simulation (see Section 4.3), data stored in the *.b07* file are read. If executing a simulation that reads a stored, transient head solution (see Section 4.4), data are read from the *.b44* and *.b45* binaries stored in the *tmp* directory.

Binary data for each time step are written to the *.b07* file in the *tmp* directory. If a head solution is stored for subsequent transport runs, these data are written to the *.b44* file. Output from the flow and transport calculations is stored in the *lprog3.prn* file located in the *cf_tmpascii* directory.

4.3 Restart Simulations

There are two methods for performing restart simulations with *lprog3_005*. The first method is used when restarting an unfinished simulation (e.g., due to a power failure). The second method is applied when a simulation has successfully completed, but the simulation time needs to be extended.

To restart a simulation that has terminated prematurely, the starting time step, *ITT*, must be reset on line 9 of the *cfest.ctl* file. For a normal (non-restart) simulation, this parameter is set to 0. However, in a restart simulation, this number will reflect the desired starting time step number. Once this number is changed, *lprog3_005* can be re-executed within the current directory.

To extend the simulation time for a successfully completed run, a multistep procedure is required. The first step is to create a copy of the current run directory for executing the restart simulation. Within this directory, the *.l3i* file should be modified to reflect the new time step information for extending the simulation run. Existing binary files in the *tmp* and *cf_tmpbinary* directories are also deleted within the restart directory.

The next step is to execute *lprog1_005*, followed by the *lband_005* preprocessor. At this point, the CFEST utility *linitial_005* (Section 6.4), is executed within the restart directory to extract the head and concentration distributions from the successfully completed run. This program prompts the user for the location of the *.b07* file to be read, and the time step number desired to set the initial condition for the restart run. Once a new *.b01* file is written to the restart directory, *lprog3i_005* is executed, followed by *lprog3_005*.

4.4 Stored Head Solution

When multiple runs of solute transport are required and the flow and transport calculations are decoupled, the user can archive a transient flow solution. The advantage of this approach is the time saved by avoiding having to calculate the same flow solution for multiple runs. Since the transport is decoupled from flow, the transport simulation simply reads the archived flow data.

The archiving of the transient flow solution is controlled in the *cfest.ctl* file. On line 10, the first parameter, NHEAD_NEW, controls whether or not a new flow solution is calculated. To archive the flow solution, this parameter should be set to 1, so that flow is calculated. The second parameter on line 10, IOPT_HEAD, should also be set to 1 so that the solution is saved to CFEST binaries (*.b44* and *.b45*) stored in the *tmp* directory.

Once the solution is archived, the *.b44* and *.b45* binaries must be present in the *tmp* directory of the current transport run. These can be copied to the current directory, or, if running in UNIX, symbolic links can be set to point to the archived flow solution. Line 10 of the *cfest.ctl* file also needs to be modified. In this case, the NHEAD_NEW parameter is set to 0 so a new flow solution is not calculated. Because the flow solution is already archived, the IOPT_HEAD parameter should have a value of 0.

CFEST also provides an option for running the transport simulation beyond the number of time steps archived in the flow solution. For example, a flow solution may approximate a steady-state solution after 1,000 years of simulation, but transport calculations are required out to 10,000 years. In this case, if the NHEAD_NEW flag on line 10 of the *cfest.ctl* file is set to -1, then the *lprog3_005* module of the CFEST simulator uses the final archived flow solution for every time step beyond the flow simulation time. Hence, in this example, the flow solution only needs to be run for 1,000 years, at which time the flow solution is approximated by the steady-state solution. Transport calculations use the final flow solution generated at 1,000 years until reaching the end of the simulation (10,000 years).

5.0 Output File Descriptions

As described in the Section 4, CFEST writes output to both binary and ascii files. Unlike the data written to the binaries, the user can control the level of output written to the ascii files stored in *cf_tmpascii*. For the four modules required for a CFEST run, the output files are named by program name: *lprog1.prn*, *lband.prn*, *lprog3i.prn*, and *lprog3.prn*. These output files are primarily used for checking binary output, and identifying problems in either the head or concentration solutions. Because output is written at every time step, the user can control the amount of data written to these files. These options and the output they generate are examined in further detail in this section.

The NPRINT parameter, which is set on line 7 of the *cfest.ctl* file, controls the flow of data written to the output files. A switch is defined for each of the three preprocessors, as well as the main module. In general, higher values of NPRINT generate higher levels of output because the higher values maintain the previous print level options, but also provide additional output to the *.prn* file. One exception to this rule is with NPRINT levels that follow the "echo input file" option. For example, in *lprog1*, NPRINT = 4 echoes the *.lp1* file to the *.prn* file. However, when NPRINT = 5, the input file echo does not occur.

Descriptions of NPRINT options are summarized in Tables 5.1–5.4, and are described in more detail in the sections that follow. Example output from *.prn* files is also shown, though due to space considerations, only partial output is presented. There are also four additional parameters, primarily used for debugging purposes, that affect output written to the *lprog3.prn* file. These options are also set in the *cfest.ctl* file on line 7, following the NPRINT options for *lprog1, lband, lprog3i* and *lprog3*. These four parameters (LIMITP, LIMITT, NSUBP, and NPMASS) are described in Table 5.5.

5.1 Output File *lprog1.prn*

Output from the *lprog1* program is written to the *lprog1.prn* file stored in the *cf_tmpascii* directory. Output generated from *lprog1* contains information on nodal coordinates, boundary conditions, stream/river properties, material and fluid properties, and the nodes and elements of the grid. The values that may be assigned to NPRINT range from 0 to 5. The following sections examine each of the values of NPRINT, and are summarized in Table 5.1.

Table 5.1. Description of NPRINT Options for *lprog1*. Higher values of NPRINT maintain the previous level's output control, except for NPRINT = 5, which does not echo the input file to output.

NPRINT	Description
0	Prints simulation units only.
1	Error messaging enabled. Prints summary grid and material property data.
2	Prints simulation switches, fluid properties, detailed surface grid information.
3	Prints detailed subsurface coordinate data and stream boundary conditions.
4	Echoes input file.
5	Prints initial conditions for surface and subsurface nodes.

5.1.1 NPRINT = 0

When parameter NPRINT is set to 0, the only output written to the *lprog1.prn* file is the simulation units read from the *cfest.ctl* file (e.g., `MKS = Simulation Units`). Essentially, when NPRINT = 0, no output from the *.lp1* file is written to the *lprog1.prn* file.

5.1.2 NPRINT = 1

For NPRINT > 0, error messaging is enabled. This is the minimum level recommended for CFEST simulations. In addition to printing NPRINT = 0 information and enabling error messages, NPRINT = 1 also prints information on

- Title, CFEST version, and run time
- Solution option for flow
- Total number of nodes, elements and material types
- Total number of prescribed boundary conditions
- Total number of unknowns for each primary variable

An example of *lprog1.prn* output file for NPRINT = 1 is shown in Figure 5.1

```
COATS_1964_DIRICHLET_BC.LL1 Q=.001, ALPHA=16M  DELY=50M  DELX VARIES  POR=0.1
CFEST Code - Version number-001-
                                                Wed Feb  8 14:22:15 2006
-------------------------------------------------------------------------------------------------------------------------------------------
 ----HEAD-PRESSURE OPTION----
   KTYPE = 0 ** FLUID SIMULATIONS WILL BE DONE IN HEAD.
   USER SHOULD CHECK THAT:
    (1) PARAMETERS SUCH AS CW, PBWR, POROSITY (CRR,PTHETA) SHOULD
        BE IN TERMS OF UNIT CHANGE IN HEAD.
   ***NOTE*** DENSITY EFFECTS ON FLOW WILL NOT BE CONSIDERED.
    MKS       = Simulation Units
-------------------------------------------------------------------------------------------------------------------------------------------
              MAXIMUM NUMBER ASSIGNED TO SURFACE NODES    (LNPT) =             26
              TOTAL NODES IN THE SYSTEM                   (NTT)  =             52

              THE FOLLOWING PARAMETERS ARE RELATED TO B.C
              TOTAL HELD POTENTIAL B.C. PRESCRIBED        (NTBCH)=              4
              TOTAL HELD TEMPERATURE B.C. PRESCRIBED      (NTBCT)=              0
              TOTAL HELD CONCENTRATION B.C. PRESCRIBED    (NTBCC)=              4

              THE FOLLOWING TWO PARAMETERS ARE RELATED TO THE STIFFNESS MATRIX
              TOTAL NUMBER OF UNKNOWN HEAD(PRESSURE) NODES(NPTH) =             48
              TOTAL NUMBER OF UNKNOWN TEMPERATURE NODES   (NPTT) =             52
              TOTAL NUMBER OF UNKNOWN CONCENTRATION NODES (NPTC) =             48
-------------------------------------------------------------------------------------------------------------------------------------------
             THE FOLLOWING PARAMETERS ARE ASSOCIATED WITH ELEMENTS
              MAXIMUM NUMBER ASSIGNED TO SURFACE ELEMENTS (NSELEM)=            12
              TOTAL NUMBER OF ELEMENTS                    (NELEM)=             12
              MAXIMUM NUMBER OF NODES IN ANY ELEMENT      (NODMAX)=             8
              MAXIMUM  NON-ZERO NODES IN ANY ELEMENTS     (NONZER)=             8

              THE FOLLOWING IS RELATED TO MATERIAL NUMBER ASSIGNED
              MAXIMUM NUMBER ASSIGNED TO MATERIAL         (MATN) =              1
-------------------------------------------------------------------------------------------------------------------------------------------
The LPROG1     CFEST module completed (no fatal errors).
Check the output for warning messages (if any).
```

Figure 5.1. Example *lprog1.prn* output for NPRINT = 1

5.1.3 NPRINT = 2

When NPRINT = 2, in addition to information generated by NPRINT = 0 and NPRINT = 1, the following is also written to the *lprog1.prn* output file:

- Simulation flags and switches for both flow and transport
- Conversion factors
- Aquifer fluid properties
- Material properties data
- Coordinates and initial condition data for surface nodes (subsurface node data requires NPRINT = 3)
- The minimum and maximum time steps and initial concentrations
- Prescribed boundary conditions by node
- Total number of unknowns for each primary variable
- Stream boundary condition data without column headers.

The stream boundary condition data is also printed for NPRINT = 3, but in that case, column headers are provided (Section 5.2.4). A truncated example of an *lprog1.prn* output file for NPRINT = 2 is shown in Figure 5.2.

5.1.4 NPRINT = 3

For NPRINT = 3, in addition to generating output for NPRINT = 2, the following output will also be written to the *lprog1.prn* file:

Coordinates and initial condition data for subsurface nodes
Stream boundary condition data with column headers

An example of the additional information written to the *lprog1.prn* file is shown in Figure 5.3.

5.1.5 NPRINT = 4

The only difference between the output produced by NPRINT = 3 and NPRINT = 4 is that the *.lp1* input file read by *lprog1* is echoed to the *lprog1.prn* output file. This feature allows the user to easily verify that the input file is being read correctly. An example of selected echoed input lines in an *lprog1.prn* output file is presented in Figure 5.4.

```
 ----FLAG OPTIONS----
       NTTYES = 0         NCCYES = 1         NNDIS  = 0
       NDIM   = 0         KTYPE  = 0         NBURDN = 0        NPRES  = 0
       IHEAD  = 0         ITEMP  = 0         ICONC  = 0
-------------------------------------------------------------------------------------------------------------------------------------------------------
 ----CONVERSIONS FACTORS----
       PERCON =  1.00     XYCONV =  1.00     ZCONV  =  1.00     HCONV  =  1.00
       CONCPR =  1.00     CONCPW =  1.00     CONUKH =  1.00     CONVIS =  1.00
       CCONV  =  1.00     CONDIF =  1.00     CONMOL =  1.00
-------------------------------------------------------------------------------------------------------------------------------------------------------
 ----CONCENTRATION SIMULATION OPTION (NCCYES=1)----
.
- ----AQUIFER FLUID PROPERTIES----
       CW    = 0.00     CTW   = 0.00     CPW   = 1.00     CCCMAS = 0.1000E-04
       U0    = 0.00     T0CPW = 0.00                 CCC   = 0.000
.
.
-------------------------------------------------------------------------------------------------------------------------------------------------------
 Material Number   1  ---  AQUIFER MATERIAL
 X-K            Y-K            Z-K      ROCK COMPRES.          THETA      THETA-REF. PRES. SPECIFIC STORAGE
10.0000         10.0000         10.0000        0.100000E-14     0.100000         0.00000         0.100000E-15
.
.
-------------------------------------------------------------------------------------------------------------------------------------------------------
         COORDINATES AND OTHER DETAILS FOR EACH  NODE
-------------------------------------------------------------------------------------------------------------------------------------------------------
    INTERNAL   USER  X-COORD.    Y-COORD.    Z-COORD     INITIAL     TEMP       CONCEN       RHO
    NODE NO.   NODE                                      HEAD        DEG.         M/L**3
                 NO.
       1        1   0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      1000.0
       3        2   0.0000      50.000      0.0000      0.0000      0.0000      0.0000      1000.0
.
.
-------------------------------------------------------------------------------------------------------------------------------------------------------
 ----SUMMARY NODE INFORMATION----
 TOTAL NODES=       52    SURFACE NODES=    26 MAXIMUM NUMBER ASSIGNED TO A SURFACE NODE=      26
 TMIN=   0.00000000000000D+000   TMAX=   0.00000000000000D+000
 CMIN=   0.00000000000000D+000   CMAX=   0.00000000000000D+000
.
.
-------------------------------------------------------------------------------------------------------------------------------------------------------
    TOTAL NUMBER OF NODES WITH PRESCRIBED HEAD B.C. ARE=         4
-------------------------------------------------------------------------------------------------------------------------------------------------------
 NODE #=      25 HEAD=   0.000     NODE #=      26 HEAD=   0.000     NODE #=   100025 HEAD=   0.000
 NODE #=   100026 HEAD=   0.000     NODE #=

-------------------------------------------------------------------------------------------------------------------------------------------------------
    TOTAL NUMBER OF NODES WITH PRESCRIBED CONC B.C.=          4
-------------------------------------------------------------------------------------------------------------------------------------------------------
 NODE#       1 CONC= 1.000     NODE#       2 CONC= 1.000     NODE#   100001 CONC= 1.000
 NODE#   100002 CONC= 1.000     NODE#
.
.
RivNode=         1 Elev=    100.00 L=    500.00 W=     10.00 ElvRBedBase=     90.00 Sdep=   100.000 BedThk=    10.000 BedK= 0.1000000E-02
RivNode=        11 Elev=    100.00 L=   1000.00 W=     10.00 ElvRBedBase=     90.00 Sdep=   100.000 BedThk=    10.000 BedK= 0.1000000E-02
.
.
                                ELEMENT DETAILS
 Element number:     1  Material number:  1  Total nodes:: 8  Element nodes:     1     3     4     2 100001 100003 100004 100002
 Element number:     2  Material number:  1  Total nodes:: 8  Element nodes:     3     5     6     4 100003 100005 100006 100004
 Element number:     3  Material number:  1  Total nodes:: 8  Element nodes:     5     7     8     6 100005 100007 100008 100006
 Element number:     4  Material number:  1  Total nodes:: 8  Element nodes:     7     9    10     8 100007 100009 100010 100008
.
.
-------------------------------------------------------------------------------------------------------------------------------------------------------
    TOTAL UNKNOWN HEAD NODES ARE=       48  THESE NODES ARE AS GIVEN BELOW
-------------------------------------------------------------------------------------------------------------------------------------------------------
     1 100001     2 100002     3 100003     4 100004     5 100005     6 100006     7 100007     8
 100008     9 100009     10 100010     11 100011     12 100012     13 100013     14 100014     15 100015
    16 100016     17 100017     18 100018     19 100019     20 100020     21 100021     22 100022     23
 100023     24 100024
```

Figure 5.2. Truncated Example *lprog1.prn* Output for NPRINT = 2. Vertical dots mark where full output is not shown due to space considerations.

```
-------------------------------------------------------------------------------------------------------------------------
      COORDINATES AND OTHER DETAILS FOR EACH  NODE
-------------------------------------------------------------------------------------------------------------------------
  INTERNAL  USER X-COORD.   Y-COORD.   Z-COORD    INITIAL    TEMP      CONCEN      RHO
  NODE NO.  NODE                                   HEAD      DEG.       M/L**3
              NO.
    1      1  0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    1000.0
    2  100001  0.0000    0.0000   -1.0000    0.0000    0.0000    0.0000    1000.0
    3      2  0.0000   50.000    0.0000    0.0000    0.0000    0.0000    1000.0
    4  100002  0.0000   50.000   -1.0000    0.0000    0.0000    0.0000    1000.0
    5      3  5.0000    0.0000    0.0000    0.0000    0.0000    0.0000    1000.0
    6  100003  5.0000    0.0000   -1.0000    0.0000    0.0000    0.0000    1000.0
    .
    .
    .
   47     24  55.000   50.000    0.0000    0.0000    0.0000    0.0000    1000.0
   48  100024  55.000   50.000   -1.0000    0.0000    0.0000    0.0000    1000.0
   49     25  60.000    0.0000    0.0000    0.0000    0.0000    0.0000    1000.0
   50  100025  60.000    0.0000   -1.0000    0.0000    0.0000    0.0000    1000.0
   51     26  60.000   50.000    0.0000    0.0000    0.0000    0.0000    1000.0
   52  100026  60.000   50.000   -1.0000    0.0000    0.0000    0.0000    1000.0
.
.
             ***STREAM NODE BOUNDARY CONDITION DATA***
 ---------------------------------------------------------------------------------------------
                                                  STREAM      MINIMUM   STREAM
STREAM   STREAM       STREAM     STREAM   BOTTOM       STREAM     BED       STREAM BED
NODE  ELEVATION     LENGTH      WIDTH  ELEVATION      DEPTH  THICKNESS   CONDUCTIVITY
 ---------------------------------------------------------------------------------------------
RivNode=        1 Elev=    100.00 L=    500.00 W=     10.00 ElvRBedBase=     90.00 Sdep=   100.000 BedThk=    10.000 BedK= 0.1000000E-02
RivNode=       11 Elev=    100.00 L=   1000.00 W=     10.00 ElvRBedBase=     90.00 Sdep=   100.000 BedThk=    10.000 BedK= 0.1000000E-02
RivNode=       21 Elev=    100.00 L=   1000.00 W=     10.00 ElvRBedBase=     90.00 Sdep=   100.000 BedThk=    10.000 BedK= 0.1000000E-02
RivNode=       31 Elev=    100.00 L=   1000.00 W=     10.00 ElvRBedBase=     90.00 Sdep=   100.000 BedThk=    10.000 BedK= 0.1000000E-02
RivNode=       41 Elev=    100.00 L=   1000.00 W=     10.00 ElvRBedBase=     90.00 Sdep=   100.000 BedThk=    10.000 BedK= 0.1000000E-02
-----------------------------------------------------------------------------------------
```

Figure 5.3. Truncated example *lprog1.prn* output for NPRINT = 3. Vertical dots mark where full output is not shown due to space considerations.

5.1.6 NPRINT = 5

For NPRINT = 5, the input file echo for NPRINT = 4 is disabled. However, all of the output written to the *lprog1.prn* file for NPRINT = 3 is also written to the *lprog1.prn* output file. In addition to these data, the following are also written to the output file:

- Initial conditions for surface and subsurface nodes for each primary variable
- Number of layers in the surface elements

Figure 5.5 is an example of the additional output written to the *lprog1.prn* file when NPRINT = 5.

```
TITLE ECHO--COATS_1964_DIRICHLET_BC.LL1 Q=.001, ALPHA=16M  DELY=50M  DELX VARIES  POR=0.1

    COATS_1964_DIRICHLET_BC.LL1 Q=.001, ALPHA=16M  DELY=50M  DELX VARIES  POR=0.1
 CFEST Code - Version number-001-
                                                    Wed Feb  8 14:23:50 2006
 NTTYES      LINE ECHO-- 0 1 0
 NDIM         LINE ECHO-- 0 0 0 0
 IHEAD        LINE ECHO-- 0 0 0
 PERCON       LINE ECHO--   1.0000     1.0000     1.0000     1.0000
 CONCPR       LINE ECHO--   1.0000     1.0000     1.0000     1.0000
 CCONV        LINE ECHO--   1.0000     1.0000     1.0000
.
 CW           LINE ECHO--   0.0000     0.0000     1.0000    0.10000E-04  0.0000
 U0           LINE ECHO--   0.0000     0.0000
 RHO0         LINE ECHO--   1000.0     0.0000     0.0000
.
I,AMATER     LINE ECHO-- 0   1 QUIFER MATERIAL
 XK,YK,ZK     LINE ECHO--   10.000      10.000      10.000
 THETA(I)     LINE ECHO--  0.10000     0.0000    0.10000E-14  0.0000
 UKTX(I)      LINE ECHO--   0.0000     0.0000     0.0000      500.00
 ALPHAL(I)    LINE ECHO--   16.000     0.0000     0.0000
-------------------------------------------------------------------------------------------------------------------------------------------
 Material Number   1  ---  QUIFER MATERIAL
         X-K             Y-K             Z-K      ROCK COMPRES.           THETA       THETA-REF. PRES. SPECIFIC STORAGE
      10.0000          10.0000          10.0000         0.100000E-14      0.100000          0.00000          0.100000E-15
 HEAT K-X       HEAT K-Y         HEAT K-Z      ROCK HEAT-CP          ALPHA-L          ALPHA-T       MOLECULAR DIFF.
      0.00000          0.00000          0.00000             500.000                    16.0000             0.00000          0.00000
 I,AMATER     LINE ECHO-- 1   0                                      END OF MAT. PROPERT

 ILOG        LINE ECHO--    1
-------------------------------------------------------------------------------------------------------------------------------------------
  ----CONSTANT LAYERING OPTION----
     ILOG= 1
 LLMAX       LINE ECHO--    1
     NUMBER OF LAYERS SELECTED (LLMAX)=    1
 LL(J),ZZ(J) LINE ECHO--    1    -1.0
     CONSTANT LAYERING DETAILS (DEPTH ZZ(I) REPRESENTSELEVATION OF BASE OF MATERIAL LL(I)
    MAT.    1 DEPTH -1.0000     MAT.
 INITAL      LINE ECHO-- 1

HEADI        LINE ECHO--   0.0000     0.0000     0.0000
-------------------------------------------------------------------------------------------------------------------------------------------
         COORDINATES AND OTHER DETAILS FOR EACH  NODE
-------------------------------------------------------------------------------------------------------------------------------------------
    INTERNAL   USER  X-COORD.     Y-COORD.     Z-COORD      INITIAL      TEMP         CONCEN        RHO
    NODE NO.   NODE                                        HEAD      DEG.                    M/L**3
               NO.
 NSEC,K,LLDUM,XX,YY,ZTOP                  LINE ECHO--
 COORDINATE DATA CARD 0        1   2   0.0       0.0       0.0
.
HEAD B.C.    LINE ECHO--
 COPY OF HEAD B.C. CARD 0       25      0.00
 HEAD B.C.    LINE ECHO--
 COPY OF HEAD B.C. CARD 0       26      0.00
```

Figure 5.4. Truncated example *lprog1.prn* output for NPRINT = 4 (echo input). Vertical dots mark where full output is not shown due to space considerations.

```
-------------------------------------------------------------------------------------------------------------------------------
      INITIAL CONDITIONS HAVE BEEN UPDATED ACCORDING TO PRESCRIBED BOUNDARY CONDITIONS
-------------------------------------------------------------------------------------------------------------------------------

      INITIAL CONDITIONS AFTER NECESSARY MODIFICATION DUE TO CHANGE IN BOUNDARY CONDITIONS
-------------------------------------------------------------------------------------------------------------------------------
  I   NODE      HEAD     TEMP      CONC       RHO      I      NODE      HEAD     TEMP      CONC       RHO

  1      1 0.00000E+00    0.00      1.00    1000.0      2     100001 0.00000E+00    0.00      1.00    1000.0
  3      2 0.00000E+00    0.00      1.00    1000.0      4     100002 0.00000E+00    0.00      1.00    1000.0

-------------------------------------------------------------------------------------------------------------------------------
                                    ELEMENT DETAILS
-------------------------------------------------------------------------------------------------------------------------------
 ----SURFACE ELEMENT NUMBER=      1 -Only a single layer exists at this location.-
Element number:      1  Material number:  1  Total nodes:: 8  Element nodes:     1     3     4     2 100001 100003 100004 100002
-------------------------------------------------------------------------------------------------------------------------------
 ----SURFACE ELEMENT NUMBER=      2 -Only a single layer exists at this location.-
Element number:      2  Material number:  1  Total nodes:: 8  Element nodes:     3     5     6     4 100003 100005 100006 100004
-------------------------------------------------------------------------------------------------------------------------------
```

Figure 5.5. Truncated example *lprog1.prn* output for NPRINT = 5. Vertical dots mark where full output is not shown due to space considerations.

5.2 Output File *lband.prn*

Output from the *lband* preprocessor is written to the *lband.prn* file stored in the *cf_tmpascii* directory. The *lband* program calculates the vector and matrix structures for the linear system of equations for head and concentration using the compressed row and column storage format for sparse matrices, which only stores non-zero elements. The *lband.prn* file contains information on the structure of these arrays. NPRINT for the *lband* preprocessor can range from 0 to 3. In the sections that follow, the resulting output for the values of NPRINT summarized in Table 5.2 are presented in more detail.

Table 5.2. Description of NPRINT Options for *lband*. Higher values of NPRINT maintain the previous level's output control.

NPRINT	Description
0	Prints total number of non-zero matrix coefficients.
1	Error messaging enabled; prints summary grid information and number of unknowns.
2	Prints node and element numbers, vector and matrix indices.
3	Prints detailed information on the vectors and matrices used in solver.

5.2.1 NPRINT = 0

For NPRINT = 0, only the total number of nonzero coefficients, MAXNZ, is written to *lband.prn* file (e.g., MAXNZ 544).

5.2.2 NPRINT = 1

For NPRINT > 0, error messaging is enabled. This is the minimum level recommended for CFEST simulations. In addition to printing NPRINT = 0 information and enabling error messages, NPRINT = 1 also prints information on:

- Title, CFEST version, and run time
- Total number of nodes and surface nodes and elements
- Total number of unknowns for each primary variable
- Total number of Dirichlet boundary conditions

An example *lband.prn* output file for NPRINT = 1 is shown in Figure 5.6.

```
------------------------------------------------------------------------------
>>>>>>>>>>------------ The LBAND     Module of the: ------------<<<<<<<<<<
------------------------------------------------------------------------------
**** ****** CFEST-001 (Coupled Fluid Energy and Solute Transport) code **********
------------------------------------------------------------------------------
********** This CFEST code module was run on Tue Jan 24 11:28:58 2006  <<<<<<<<<<
------------------------------------------------------------------------------

*******************************************************************************************************************
            TOTAL NUMBER OF SURFACE NODES (LNPT)  =                    26
            TOTAL NODES IN THE SYSTEM     (NTT)   =                   52
            TOTAL NUMBER OF HEAD UNKNOWN  (NPTH)  =                    48
            TOTAL NUMBER OF TEMP UNKNOWN  (NPTT)  =                    52
            TOTAL NUMBER OF CONC UNKNOWN  (NPTC)  =                    48
            TOTAL NUMBER OF TOP NODES     (NTOP)  =                    0
            TOTAL NUMBER OF ELEMENTS      (NELEM) =                    12
*******************************************************************************************************************
MAXNZ       544
HELD NODE =     1 IBHELD =  8 HELD NODE =     2 IBHELD =  8 HELD NODE =100001 IBHELD =  8
HELD NODE =100002 IBHELD =  8 HELD NODE =
HELD NODE =       1  IB =      1  IEND =      8
NHELD'S =      1  100001     2  100002     3  100003     4  100004
HELD NODE =       2  IB =      9  IEND =     16
NHELD'S =      1  100001     2  100002     3  100003     4  100004
HELD NODE =  100001  IB =     17  IEND =     24
NHELD'S =      1  100001     2  100002     3  100003     4  100004
HELD NODE =  100002  IB =     25  IEND =     32
NHELD'S =      1  100001     2  100002     3  100003     4  100004

The LBAND     CFEST module completed (no fatal errors).
Check the output for warning messages (if any).
```

Figure 5.6. Example *lband.prn* Output for NPRINT = 1

5.2.3 NPRINT = 2

When NPRINT = 2, in addition to information generated by NPRINT = 0 and NPRINT = 1, the following is also written to the *lband.prn* output file:

- Element numbers and corresponding node numbers
- The index of matrix rows (ROW)
- The number of matrix entries (COL)
- The position of matrix entries (NCOL).

A truncated example of an *lband.prn* output file for NPRINT = 2 is shown in Figure 5.7. This information is primarily used for debugging, but should the user require more information on the compressed row and column formats for sparse matrices, the user is referred to documentation of the ITPACK (Young and Kincaid 1981) and NSPCG (Oppe et al. 1988) solvers.

```
--------------------------------------------------------------------------------
>>>>>>>>>>---------- The LBAND     Module of the: ----------<<<<<<<<<<
--------------------------------------------------------------------------------
********** CFEST-001 (Coupled Fluid Energy and Solute Transport) code **********
--------------------------------------------------------------------------------
********** This CFEST code module was run on Wed Feb  8 14:22:58 2006  <<<<<<<<<<
--------------------------------------------------------------------------------

*****************************************************************************************************
            TOTAL NUMBER OF SURFACE NODES (LNPT)  =                  26
            TOTAL NODES IN THE SYSTEM     (NTT)   =                 52
            TOTAL NUMBER OF HEAD UNKNOWN  (NPTH)  =                  48
            TOTAL NUMBER OF TEMP UNKNOWN  (NPTT)  =                  52
            TOTAL NUMBER OF CONC UNKNOWN  (NPTC)  =                  48
            TOTAL NUMBER OF TOP NODES     (NTOP)  =                  0
            TOTAL NUMBER OF ELEMENTS      (NELEM) =                 12
*****************************************************************************************************
NODES =     1  100001     2  100002     3  100003     4  100004     5  100005     6  100006     7  100007     8  100008     9  100009    10  100010    11  100011       12
100012    13  100013    14  100014    15  100015    16  100016    17  100017    18  100018    19  100019    20  100020    21  100021    22  100022       23  100023
24  100024    25  100025    26  100026
IE =   1  NE =   1  NODES =     1     3     4     2  100001  100003  100004  100002
 IE =   2  NE =   2  NODES =     3     5     6     4  100003  100005  100006  100004
.
.
 ROW =   1 BAND =   8 NCOL =   1   2   3   4   5   6   7   8
 ROW =   2 BAND =   8 NCOL =   1   2   3   4   5   6   7   8
 .
.
 MAXNZ       544
 ROW =   1 BAND =   8 NHELD=    1
               2   3   4   5   6   7   8
 ROW =   2 BAND =   8 NHELD=    1
               2   3   4   5   6   7   8
 ROW =   3 BAND =   8 NHELD=    1
               2   3   4   5   6   7   8
 ROW =   4 BAND =   8 NHELD=    1
               2   3   4   5   6   7   8
 HELD NODE =     1 IBHELD =  8 HELD NODE =     2 IBHELD =  8 HELD NODE =100001 IBHELD =  8
 HELD NODE =100002 IBHELD =  8 HELD NODE =
 HELD NODE =       1  IB =      1  IEND =      8
 NHELD'S =      1  100001     2  100002     3  100003     4  100004
 HELD NODE =       2  IB =      9  IEND =     16
 NHELD'S =      1  100001     2  100002     3  100003     4  100004
 HELD NODE =  100001  IB =     17  IEND =     24
 NHELD'S =      1  100001     2  100002     3  100003     4  100004
 HELD NODE =  100002  IB =     25  IEND =     32
 NHELD'S =      1  100001     2  100002     3  100003     4  100004
```

Figure 5.7. Truncated Example *lband.prn* Output for NPRINT = 2. Vertical dots mark where full output is not shown due to space considerations.

5.2.4 NPRINT = 3

For NPRINT = 3, in addition to generating output for NPRINT = 2, the following output will also be written to the *lband.prn* file:

- The number of entries in each row of the NCOL vector.
- The indices used to identify the beginning of each row in the coefficient matrix (NSTART) and the indexes that identify the end of each row (NSTOP) in the coefficient matrix.
- The bandwidth of the matrix (IBANDW).

An example of the additional information written to the *lprog1.prn* file is shown in Figure 5.8.

.

```
.
READ NUMBER KK =   1 NCOL VALUES =     0     0     0     0     0     0     0     0     0     0
WRITE NUMBER KK =   1 NCOL VALUES =     1     2     3     4     5     6     7     8     0     0
READ NUMBER KK =   5 NCOL VALUES =     0     0     0     0     0     0     0     0     0     0
WRITE NUMBER KK =   5 NCOL VALUES =     1     2     3     4     5     6     7     8     0     0
READ NUMBER KK =   7 NCOL VALUES =     0     0     0     0     0     0     0     0     0     0
WRITE NUMBER KK =   7 NCOL VALUES =     1     2     3     4     5     6     7     8     0     0

MAXNZ       544
NSTART=
  1=  49   2=  56   3=  63   4=  70   5=  77   6=  88   7=  99   8= 110   9= 121  10= 132
 11= 143  12= 154  13= 165  14= 176  15= 187  16= 198  17= 209  18= 220  19= 231  20= 242
 21= 253  22= 264  23= 275  24= 286  25= 297  26= 308  27= 319  28= 330  29= 341  30= 352
 31= 363  32= 374  33= 385  34= 396  35= 407  36= 418  37= 429  38= 440  39= 451  40= 462
 41= 473  42= 484  43= 495  44= 506  45= 517  46= 524  47= 531  48= 538
NSTOP=
  1=  55   2=  62   3=  69   4=  76   5=  87   6=  98   7= 109   8= 120   9= 131  10= 142
 11= 153  12= 164  13= 175  14= 186  15= 197  16= 208  17= 219  18= 230  19= 241  20= 252
 21= 263  22= 274  23= 285  24= 296  25= 307  26= 318  27= 329  28= 340  29= 351  30= 362
 31= 373  32= 384  33= 395  34= 406  35= 417  36= 428  37= 439  38= 450  39= 461  40= 472
 41= 483  42= 494  43= 505  44= 516  45= 523  46= 530  47= 537  48= 544
IBANDW       48       0       0       0       0       0
       0       0       0       0       0       0
       0       0       0       0       0       0
       0       0       0       0       0       0
       0       0       0       0       0       0
       0       0       0       0       0       0
       0       0       0       0       0       0
       0       0       0       0       0       0
       0
  1=   1   1:   2=   2   2:   3=   3   3:   4=   4   4:   5=   5   5:   6=   6   6:   7=   7   7:   8=   8   8:
  9=   9   9:  10=  10  10:  11=  11  11:  12=  12  12:  13=  13  13:  14=  14  14:  15=  15  15:  16=  16  16:
 17=  17  17:  18=  18  18:  19=  19  19:  20=  20  20:  21=  21  21:  22=  22  22:  23=  23  23:  24=  24  24:
 25=  25  25:  26=  26  26:  27=  27  27:  28=  28  28:  29=  29  29:  30=  30  30:  31=  31  31:  32=  32  32:
 33=  33  33:  34=  34  34:  35=  35  35:  36=  36  36:  37=  37  37:  38=  38  38:  39=  39  39:  40=  40  40:
 41=  41  41:  42=  42  42:  43=  43  43:  44=  44  44:  45=  45  45:  46=  46  46:  47=  47  47:  48=  48  48:
```

Figure 5.8. Truncated Example *lband.prn* Output for NPRINT = 3. Vertical dots mark where full output is not shown due to space considerations.

5.3 Output file *lprog3i.prn*

Output from the *lprog3i* preprocessor is written to the *lprog3i.prn* file stored in the *cf_tmpascii* directory. The *lprog3i* program preprocesses the data on steady-state and transient options, time steps, elemental sources and sinks, and nodal injection/extraction data, as well as information on fluid properties. NPRINT for the *lprog3i* preprocessor can range from 0 to 3. In the sections that follow, the resulting output for the values of NPRINT summarized in Table 5.3 are presented in more detail.

Table 5.3. Description of NPRINT Options for *lprog3i*. Higher values of NPRINT maintain the previous level's output control.

NPRINT	Description
0	No output.
1	Error messaging enabled. Prints maximum and minimum values for primary variables and time step data.
2	Prints Dirichlet boundary conditions and surface infiltration (recharge) data.
3	Echoes input file.

5.3.1 NPRINT = 0

For NPRINT = 0, no output is generated to the *lprog3i.prn* file. When NPRINT = 0, no output read in from the *.l3i* is written to the *lprog3i.prn* file.

5.3.2 NPRINT = 1

For NPRINT > 0, error messaging is enabled. This is the minimum level recommended for CFEST simulations. In addition to printing NPRINT = 0 information and enabling error messages, NPRINT = 1 also prints information on

- Minimum and maximum values for primary variables extracted from the initial condition
- Transient and steady-state options for primary variables
- Total number of time steps and time step length
- Time-dependent Dirichlet boundary condition data
- Flux (Neumann) boundary condition data.

An example of *lprog3i.prn* file for NPRINT = 1 is shown in Figure 5.9.

5.3.3 NPRINT = 2

When NPRINT = 2, in addition to information generated by NPRINT = 0 and NPRINT = 1, the following is also written to the *lprog3i.prn* output file:

- Surface infiltration rate data
- Primary variable values for nodes affected by elemental surface infiltration data

For this example problem, there is no difference between the output of NPRINT = 1 and NPRINT = 2 because there is no surface recharge (NSURFQ).

5.3.4 NPRINT = 3

The only difference between the output produced by NPRINT = 2 and NPRINT = 3 is that the *.l3i* input file read by *lprog3i* is echoed to the *lprog3i.prn* output file. This feature allows the user to easily verify that the input file is being read correctly. An example of selected echoed input lines in the *lprog3i.prn* output file is presented in Figure 5.10.

```
>>>>>>>>>>-------------- The LPROG3I    Module of the: --------------<<<<<<<<<<
 ------------------------------------------------------------------------------
********** CFEST-001 (Coupled Fluid Energy and Solute Transport) code **********
 ------------------------------------------------------------------------------
********** This CFEST code module was run on Wed Feb  8 14:22:58 2006  <<<<<<<<<<
 ------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------------------------------------------
 COA
  TMIN=  0.00   TMAX=  0.00      CMIN=       0.00      CMAX=       1.00
NSTDYH=     0 NSTDYT=     0 NSTDYC=     1
    HEAD  STEADY STATE SOLUTION PRESCRIBED
    CONC  TRANSIENT SOLUTION PRESCRIBED
ITOTAL=    11     NTEXP=     0
    TOTAL NUMBER OF TIME STEPS (ITOTAL)=        11
    NTEXP (0=EXPLICIT  1=GENERATE)     =        0
        STEP#    1     ELAPSED TIME      10.00       DELTAT     10.00
        STEP#    2     ELAPSED TIME      20.00       DELTAT     10.00
        STEP#    3     ELAPSED TIME      50.00       DELTAT     30.00
        STEP#    4     ELAPSED TIME      100.0       DELTAT     50.00
        STEP#    5     ELAPSED TIME      200.0       DELTAT     100.0
        STEP#    6     ELAPSED TIME      300.0       DELTAT     100.0
        STEP#    7     ELAPSED TIME      400.0       DELTAT     100.0
        STEP#    8     ELAPSED TIME      500.0       DELTAT     100.0
        STEP#    9     ELAPSED TIME      600.0       DELTAT     100.0
        STEP#   10     ELAPSED TIME      700.0       DELTAT     100.0
        STEP#   11     ELAPSED TIME      800.0       DELTAT     100.0
    HEAD HELD BOUNDARY IS TIME INVARIANT
    TEMP HELD BOUNDARY IS TIME INVARIANT
    CONC HELD BOUNDARY IS TIME INVARIANT
NSURFQ=     0
    NO SURFACE INFILTRATION PRESCRIBED
NELQIN=     0
NO SOURCES PRESCRIBED FOR ELEMENTS
NELQEX     0
NO SINKS PRESCRIBED FOR ELEMENTS
NODALQ     2
-------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------
    TOTAL NUMBER OF NODES WITH FLUX B.C.=        4   TIME STEP NUMBER=    1
-------------------------------------------------------------------------------------------------------------------------------
NODE#       1 FLUX=  0.1250E-01 TEMP   0.000    CONCENTRATION   1.000    CONC_STOP   0.000
NODE#       2 FLUX=  0.1250E-01 TEMP   0.000    CONCENTRATION   1.000    CONC_STOP   0.000
NODE#  100001 FLUX=  0.1250E-01 TEMP   0.000    CONCENTRATION   1.000    CONC_STOP   0.000
NODE#  100002 FLUX=  0.1250E-01 TEMP   0.000    CONCENTRATION   1.000    CONC_STOP   0.000
NODE#
-------------------------------------------------------------------------------------------------------------------------------
TMIN=  0.00 TMAX=  0.00     CMIN=       0.00     CMAX=       1.00

The LPROG3I    CFEST module completed (no fatal errors).
Check the output for warning messages (if any).
```

Figure 5.9. Example *lprog3i.prn* Output for NPRINT = 1

```
-------------------------------------------------------------------------
DETAIL              INPUT ECHO-- COATS_1964_DIRICHLET_BC.L3I
NSTDYH,NSTDYT,NSTDYC INPUT ECHO--    0    0    1
ITOTAL,NTEXP        INPUT ECHO--   11    0
TBEGIN              INPUT ECHO-- 0.000
(TT(I),I=1,ITOTAL)   INPUT ECHO--   10.00      20.00      50.00      100.0      200.0
(TT(I),I=1,ITOTAL)   INPUT ECHO--   300.0      400.0      500.0      600.0      700.0
(TT(I),I=1,ITOTAL)   INPUT ECHO--   800.0
NHBOND,NTBOND,NCBOND INPUT ECHO--    0    0    0
MNTRY               INPUT ECHO--    0
NSURFQ              INPUT ECHO--    0
NELQIN              INPUT ECHO--    0
NELQEX              INPUT ECHO--    0
NODALQ              INPUT ECHO--    2
NPIN,BIVF-T-C,C_STOP INPUT ECHO--      1    0.1250E-01     0.000        1.000        0.000
NPIN,BIVF-T-C,C_STOP INPUT ECHO--      2    0.1250E-01     0.000        1.000        0.000
NPIN,BIVF-T-C,C_STOP INPUT ECHO--  100001    0.1250E-01     0.000        1.000        0.000
NPIN,BIVF-T-C,C_STOP INPUT ECHO--  100002    0.1250E-01     0.000        1.000        0.000
IOPT                INPUT ECHO--    0).
```

Figure 5.10. Example *lprog3i.prn* Output for NPRINT = 3 (echo input)

5.4 Output File lprog3.prn

Output from the *lprog3* main program is written to the *lprog3.prn* file stored in the *cf_tmpascii* directory. The *lprog3* module is the main part of the CFEST simulator that performs the flow and transport calculations. Hence, the *lprog3.prn* file contains information on the flow and transport solutions, as well as mass balance information (for solute mass balance, NPMASS > 0, see Table 5.5). NPRINT for the *lprog3* program can range from 0 to 5. In the sections that follow, the resulting output for the values of NPRINT summarized in Table 5.4 are presented in more detail.

Table 5.4. Description of NPRINT Options for *lprog3*. Higher values of NPRINT maintain the previous level's output control, with the exception of NPRINT = 2, which does not print the primary variable output at every node.

NPRINT	Description
0	Error messaging enabled. Prints simulation switches and mass balance.
1	At each time step, prints maximum and minimum values for primary variables and mass balance. Also prints at each time step the pressure head, elevation, and concentration at each node.
2	Activates LIMITT and LIMITP parameters (Table 5.5); prints transient Dirichlet boundary conditions and fluxes for range of time steps and nodes specified by LIMITT and LIMITP; excludes primary variable output at every node (NPRINT = 1).
3	Prints boundary conditions, total number of unknowns for primary variables, and primary variables before and after solve.
4	Prints solver flags and time dependent boundary conditions.
5	If LIMITT and LIMITP = 0, prints values used to construct flow and transport vectors and matrices. If LIMITT and LIMITP > 0, also prints element properties.

5.4.1 NPRINT = 0

Unlike the CFEST preprocessing programs which turn off nearly all output written to the *.prn* file, the NPRINT = 0 option enables error messaging for *lprog3* and the following information is written to the *lprog3.prn* file:

- Simulation flags
- Mass balance information

An example *lprog3.prn* output file for NPRINT = 0 is shown in Figure 5.11.

```
NSTDYH= 0 NODALQ= 2 NSURFQ= 0 NELQIN= 0 NELQEX= 0 NTIMEQ= 1
 IOPT_K= 0 IOPT_Q= 0 IOPT_STOP= 0 IOPT_HSAVE= 0 IOPT_MASSFILE= 0
============================================================
 NCYES= 1 RETARDATION FACTOR=       1.000 HALF_LIFE=   0.000000E+00
============================================================
------------------------------------------------------------------------------
 ITRANS=-1 (0=CONFINED,1=UNCONFINED,2=CONFINED+UNCONFINED)
 NSTREAM =   0 (0=NO RIVER, > 0 TOTAL RIVER NODES)
 NTSUBD  =  1 (SUBDIVISION OF TIME STEPS)
------------------------------------------------------------------------------
 ------------------------------------------------------------------------------------------------------------
 FLUID BALANCE (cu. meters/day)**  FOR STEADY STATE
 ------------------------------------------------------------------------------------------------------------
 FLUID BALANCE (cu. meters)
 TIME-STEP  =  1 SUB-TIME-STEP  1 TIME=  0.100000E+02(day)
 ------------------------------------------------------------------------------------------------------------
       :    NODAL     :    HELD NODES: ELEMENTAL    :INFILTRATION  :   INFIL. AREA: ****TOTAL****:
       : (cu. meters)  :( cu. meters) :( cu. meters ): (cu. meters ):( sq. meters ): (cu. meters )
 IN(VOL.):   0.500000E-01:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.500000E-01:
 OUT(VOL):   0.000000E+00:  0.500000E-01:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.500000E-01:
 ------------------------------------------------------------------------------------------------------------
 FOLLOWING ARE IN cu. meters UNITS
       : CURR. STORAGE :PREV. STORAGE :STORAGE CHANGE:  IN-OUT FLOW :       ERROR :   ERROR RATIO:
 SYSTEM= :   0.300000E+03:  0.300000E+03:  0.000000E+00: -0.130035E-13:  0.130035E-13:  0.100000E+01:
 ------------------------------------------------------------------------------------------------------------

 End Time=Wed Feb  8 14:16:39 2006
```

Figure 5.11. Example *lprog3.prn* Output for NPRINT = 0

5.4.2 NPRINT = 1

When NPRINT = 1, in addition to information generated by NPRINT = 0, the following is also written to the *lprog3.prn* output file:

- Title and run time
- Index of time steps and corresponding elapsed time
- The maximum and minimum primary variables for all time steps at each of the nodes
- The depth and primary variable data for all nodes at each time step
- The fluid mass balance information at each time step
- The solute mass balance information for each time step (NPMASS =1) and sub-time step (NPMASS = 2) (see Section 5.6, Table 5.5)

A truncated example of a *lprog3.prn* output file for NPRINT = 1 is shown in Figure 5.12.

5.4.3 NPRINT = 2

When NPRINT = 2, the parameters *LIMIT* and *LIMITP* are activated. These parameters (see Table 5.5) limit the output to the specified number of time steps and elements. In addition to information generated by NPRINT = 0 and NPRINT = 1, the following is also written to the *lprog3.prn* output file:

- Time-dependent Dirichlet boundary condition data
- Flux information
- Primary variable data for nodes numbered from 1 to LIMITP and for time steps numbered from 1 to LIMITT.

A truncated example of a *lprog3.prn* output file for NPRINT = 2 is shown in Figure 5.13.

```
 1 DELTAT=   10.00     TIME= 10.0000    HEAD MAX. 0.600000E-02 NODE# 100002  MIN. 0.500000E-03 NODE# 100023 ETime0.894070E-09
 1 DELTAT=   10.00     TIME= 10.0000    CONC MAX. 0.696104E-01 NODE# 100004  MIN. 0.000000E+00 NODE#     15 ETime0.894070E-09
      STEADY STATE SOLUTION FOR ----- HEAD
.
 MAXIMUM HEAD VALUE=  0.6000000E-02  MINIMUM HEAD VALUE=   0.000000
.
NODE   DEPTH      HEAD   NODE   DEPTH      HEAD      NODE   DEPTH      HEAD     NODE   DEPTH      HEAD     NODE   DEPTH      HEAD
-------------------------|------------------------ ----|-------------------------|-------------------------|---------- ----------|
     1    0.00   0.0060|       2    0.00   0.0060|       3    0.00   0.0055|       4    0.00   0.0055|       5    0.00   0.0050|
100001 -1.00   0.0060| 100002 -1.00   0.0060|100003  -1.00   0.0055|100004  -1.00   0.0055|100005 -1.00   0.0050|
-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
     6    0.00   0.0050|       7    0.00   0.0045|       8    0.00   0.0045|       9    0.00   0.0040|      10    0.00   0.0040|
100006 -1.00   0.0050|100007  -1.00   0.0045|100008  -1.00   0.0045|100009 -1.00   0.0040| 100010  -1.00   0.0040|
-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
.
 CONC TIME STEP   1 DELTAT=  10.0     TIME SUB. DIV.=  1 ELAPSED TIME 10.00
 MAXIMUM CONC VALUE=   1.000000      MINIMUM CONC VALUE=   0.000000
-------------------------------------------------------------------------------------------------------------------------------------------
     NODE   DEPTH      CONC   NODE   DEPTH      CONC   NODE   DEPTH      CONC   NODE   DEPTH      CONC   NODE   DEPTH      CONC
-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
     1    0.00   1.0000|       2    0.00   1.0000|       3    0.00   0.0696|       4    0.00   0.0696|       5    0.00   0.0024|
100001   -1.00   1.0000|  100002   -1.00   1.0000|  100003   -1.00   0.0696|  100004   -1.00   0.0696|  100005   -1.00   0.0024|
-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
     6    0.00   0.0024|       7    0.00   0.0001|       8    0.00   0.0001|       9    0.00   0.0000|      10    0.00   0.0000|
100006   -1.00   0.0024|  100007   -1.00   0.0001|  100008   -1.00   0.0001|  100009   -1.00   0.0000|  100010   -1.00   0.0000|
.
 ---------------------------------------------------------------------------------------------------------------
 FLUID BALANCE (cu. meters/day)**  FOR STEADY STATE
 ---------------------------------------------------------------------------------------------------------------
 FLUID BALANCE (cu. meters)
 TIME-STEP  =  1 SUB-TIME-STEP  1 TIME=  0.100000E+02(day)
 ---------------------------------------------------------------------------------------------------------------
       :   NODAL     :    HELD NODES: ELEMENTAL    :INFILTRATION  :   INFIL. AREA: ****TOTAL****:
       : (cu. meters)  :( cu. meters) :( cu. meters ): (cu. meters ):( sq. meters ): (cu. meters )
 IN(VOL.):   0.500000E-01:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.500000E-01:
 OUT(VOL):   0.000000E+00:  0.500000E-01:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.500000E-01:
 ---------------------------------------------------------------------------------------------------------------
 FOLLOWING ARE IN cu. meters UNITS
       : CURR. STORAGE :PREV. STORAGE :STORAGE CHANGE:  IN-OUT FLOW :       ERROR :   ERROR RATIO:
 SYSTEM= :   0.300000E+03:  0.300000E+03:  0.000000E+00: -0.130035E-13:  0.130035E-13:  0.100000E+01:
 ---------------------------------------------------------------------------------------------------------------
 ---------------------------------------------------------------------------------------------------------------------------------------
    SOLUTE (Kg) BALANCE FOR TIME STEP ** DELTAT=  0.100000E+02(day)
 ---------------------------------------------------------------------------------------------------------------------------------------
 SOLUTE MASS (Kg) BALANCE**
 TIME-STEP  =  1 SUB-TIME-STEP  1 TIME=  0.100000E+02(day)
 ---------------------------------------------------------------------------------------------------------------------------------------
       :   NODAL     :    HELD NODES: ELEMENTAL    :DIRECT INJECT.:       DECAY :INFILTRATION  :   INFIL. AREA: ****TOTAL****
 INFLOW= :   0.000000E+00:  0.180301E-01:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.180301E-01:
 OUTFLOW=:   0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:
 ---------------------------------------------------------------------------------------------------------------------------------------
       :  CURR. SYSTEM : PREV. SYSTEM : SYSTEM CHANGE:  IN-OUT FLOW :       ERROR :   ERROR RATIO:
 SYSTEM= :   0.134015E+00:  0.125000E+00:  0.901507E-02:  0.180301E-01: -0.901507E-02:  0.936971E+00:
 ---------------------------------------------------------------------------------------------------------------------------------------
```

Figure 5.12. Truncated Example *lprog3.prn* Output for NPRINT = 1. Vertical dots mark where full output is not shown due to space considerations.

5.4.4 NPRINT = 3

For NPRINT = 3, in addition to the data generated from NPRINT = 2, the following is also written to the *lprog3.prn* output file:

- Number of unknowns for primary variables
- Solver flags for time-dependent boundary conditions
- Primary variable data before and after the solve.

A truncated example of a *lprog3.prn* output file for NPRINT = 3 is shown in Figure 5.14.

```
 1 DELTAT=   10.00    TIME= 10.0000    HEAD MAX. 0.600000E-02 NODE# 100002  MIN. 0.500000E-03 NODE# 100023 ETime0.894070E-09
  1 NODBF          1    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
  2 NODBF          2    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
  3 NODBF   100001    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
  4 NODBF   100002    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
 1 DELTAT=   10.00    TIME= 10.0000    CONC MAX. 0.696104E-01 NODE# 100004  MIN. 0.000000E+00 NODE#    15 ETime0.894070E-09
-------------------------------------------------------------------------------------------------------------------------------------------
   SOLUTE (Kg) BALANCE FOR TIME STEP ** DELTAT=  0.100000E+03(day)
-------------------------------------------------------------------------------------------------------------------------------------------
SOLUTE MASS (Kg) BALANCE**
TIME-STEP  = 11 SUB-TIME-STEP  1 TIME=  0.800000E+03(day)
-------------------------------------------------------------------------------------------------------------------------------------------
      :  NODAL     :    HELD NODES: ELEMENTAL    :DIRECT INJECT.:       DECAY :INFILTRATION  :   INFIL. AREA: ****TOTAL****
INFLOW= :   0.000000E+00:  0.703089E-01:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.703089E-01:
OUTFLOW=:   0.125623E-03:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.000000E+00:  0.125623E-03:
-------------------------------------------------------------------------------------------------------------------------------------------
      :  CURR. SYSTEM : PREV. SYSTEM : SYSTEM CHANGE:  IN-OUT FLOW :       ERROR :   ERROR RATIO:
SYSTEM= :   0.836459E+00:  0.764945E+00:  0.715141E-01:  0.701833E-01:  0.133076E-02:  0.100159E+01:
-------------------------------------------------------------------------------------------------------------------------------------------
```

Figure 5.13. Truncated Example *lprog3.prn* Output for NPRINT = 2. Vertical dots mark where full output is not shown due to space considerations.

```
 Number of Unknown are       48 for HEAD
     1  100001      2  100002      3  100003      4  100004      5  100005      6  100006
     7  100007      8  100008      9  100009     10  100010     11  100011     12  100012
    13  100013     14  100014     15  100015     16  100016     17  100017     18  100018
    19  100019     20  100020     21  100021     22  100022     23  100023     24  100024

  held b.c.         4 for HEAD
    25  0.00        26  0.00      100025  0.00      100026  0.00
  1 NODBF          1    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
  2 NODBF          2    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
  3 NODBF   100001    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
  4 NODBF   100002    FLUX  0.1250000E-01    TEMP   0.000000       CONC   1.000000
 B VALUES AFTER FLUX CONDITIONS CONSIDERED
     1  0.125000E-01  100001  0.125000E-01      2  0.125000E-01  100002  0.125000E-01      3   0.00000     100003   0.00000
     4   0.00000     100004   0.00000          5   0.00000     100005   0.00000          6   0.00000     100006   0.00000
     7   0.00000     100007   0.00000          8   0.00000     100008   0.00000          9   0.00000     100009   0.00000
    10   0.00000     100010   0.00000         11   0.00000     100011   0.00000         12   0.00000     100012   0.00000
    13   0.00000     100013   0.00000         14   0.00000     100014   0.00000         15   0.00000     100015   0.00000
    16   0.00000     100016   0.00000         17   0.00000     100017   0.00000         18   0.00000     100018   0.00000
    19   0.00000     100019   0.00000         20   0.00000     100020   0.00000         21   0.00000     100021   0.00000
    22   0.00000     100022   0.00000         23   0.00000     100023   0.00000         24   0.00000     100024   0.00000

START OF LBACKSU FOR HEAD
   B IN BACK SUB AFTER SOLVER
  1  0.6000000E-02100001  0.6000000E-02    2  0.6000000E-02100002  0.6000000E-02    3  0.5500000E-02100003  0.5500000E-02
  4  0.5500000E-02100004  0.5500000E-02    5  0.5000000E-02100005  0.5000000E-02    6  0.5000000E-02100006  0.5000000E-02
  7  0.4500000E-02100007  0.4500000E-02    8  0.4500000E-02100008  0.4500000E-02    9  0.4000000E-02100009  0.4000000E-02
 10  0.4000000E-02100010  0.4000000E-02   11  0.3500000E-02100011  0.3500000E-02   12  0.3500000E-02100012  0.3500000E-02
 13  0.3000000E-02100013  0.3000000E-02   14  0.3000000E-02100014  0.3000000E-02   15  0.2500000E-02100015  0.2500000E-02
 16  0.2500000E-02100016  0.2500000E-02   17  0.2000000E-02100017  0.2000000E-02   18  0.2000000E-02100018  0.2000000E-02
 19  0.1500000E-02100019  0.1500000E-02   20  0.1500000E-02100020  0.1500000E-02   21  0.1000000E-02100021  0.1000000E-02
 22  0.1000000E-02100022  0.1000000E-02   23  0.5000000E-03100023  0.5000000E-03   24  0.5000000E-03100024  0.5000000E-03

  1 DELTAT=   10.00    TIME= 10.0000    HEAD MAX. 0.600000E-02 NODE# 100002  MIN. 0.500000E-03 NODE# 100023 ETime0.894070E-09
 BACK SUB COMPLETED FOR HEAD IT=    1
 TSTEP NSTDYH IN LUNITS      10.00       0
 END OF LUNITS
 START OF LBACKSU FOR CONC
   B IN BACK SUB AFTER SOLVER
 3  0.6961035E-01100003  0.6961035E-01    4  0.6961035E-01100004  0.6961035E-01    5  0.2422801E-02100005  0.2422801E-02
 6  0.2422801E-02100006  0.2422801E-02    7  0.8432594E-04100007  0.8432594E-04    8  0.8432594E-04100008  0.8432594E-04
 9  0.2934977E-05100009  0.2934977E-05   10  0.2934977E-05100010  0.2934977E-05   11  0.1020622E-06100011  0.1020622E-06
12  0.1020622E-06100012  0.1020622E-06   13  0.3549154E-08100013  0.3549154E-08   14  0.3549154E-08100014  0.3549154E-08
15   0.000000   100015   0.000000      16   0.000000   100016   0.000000      17   0.000000   100017   0.000000
18   0.000000   100018   0.000000      19   0.000000   100019   0.000000      20   0.000000   100020   0.000000
21   0.000000   100021   0.000000      22   0.000000   100022   0.000000      23   0.000000   100023   0.000000
24   0.000000   100024   0.000000      25   0.000000   100025   0.000000      26   0.000000   100026   0.000000

  1 DELTAT=   10.00    TIME= 10.0000    CONC MAX. 0.696104E-01 NODE# 100004  MIN. 0.000000E+00 NODE#    15 ETime0.894070E-09
 BACK SUB COMPLETED FOR CONC IT=    1
```

Figure 5.14. Truncated example *lprog3.prn* output for NPRINT = 3. Vertical dots mark where full output is not shown due to space considerations.

5.4.5 NPRINT = 4

For NPRINT = 4, in addition to the data generated from NPRINT = 3, solver flags for primary variables from the LUNITS insert in *lprog3* are also written to the *lprog3.prn* output file. A truncated example of a *lprog3.prn* output file for NPRINT = 4 is shown in Figure 5.15.

```
.
START OF LUNITS
 IN LUNITS IM   1 NSOLVE=  1 NBBPTC=   4 NODB AND BIV (CHANGE/TIME) ARE AS FOLLOWS

     25  0.00         26  0.00      100025  0.00      100026  0.00
 IN BACKSU HEAD ARE AS BELOW
    1 0.600E-02100001 0.600E-02    2 0.600E-02100002 0.600E-02    3 0.550E-02100003 0.550E-02    4 0.550E-02100004 0.550E-02
    5 0.500E-02100005 0.500E-02    6 0.500E-02100006 0.500E-02    7 0.450E-02100007 0.450E-02    8 0.450E-02100008 0.450E-02
    9 0.400E-02100009 0.400E-02   10 0.400E-02100010 0.400E-02   11 0.350E-02100011 0.350E-02   12 0.350E-02100012 0.350E-02
   13 0.300E-02100013 0.300E-02   14 0.300E-02100014 0.300E-02   15 0.250E-02100015 0.250E-02   16 0.250E-02100016 0.250E-02
   17 0.200E-02100017 0.200E-02   18 0.200E-02100018 0.200E-02   19 0.150E-02100019 0.150E-02   20 0.150E-02100020 0.150E-02
   21 0.100E-02100021 0.100E-02   22 0.100E-02100022 0.100E-02   23 0.500E-03100023 0.500E-03   24 0.500E-03100024 0.500E-03
   25  0.00    100025  0.00       26  0.00    100026  0.00
 START OF LUNITS
 IN LUNITS IM   3 NSOLVE=  3 NBBPTC=   4 NODB AND BIV (CHANGE/TIME) ARE AS FOLLOWS

      1  0.00          2  0.00      100001  0.00      100002  0.00
```

Figure 5.15. Truncated Example *lprog3.prn* Output for NPRINT = 4. Vertical dots mark where full output is not shown due to space considerations.

5.4.6 NPRINT = 5

For NPRINT = 5 and when LIMITT and LIMITP are zero, then in addition to the data generated from NPRINT = 3, the following is also written to the *lprog3.prn* output file:

- Head information for each node in subroutine ELCONC
- Held nodes, including subsurface nodes, in subroutine ELCONC
- System matrix in subroutine ELHEAD if applicable.

In the ELHEAD and ELCONC subroutines, the matrices and vectors are assembled for the solver. A truncated example of an *lprog3.prn* output file for NPRINT = 5 and LIMITT = LIMITP = *0* is shown in Figure 5.16. For LIMITT and LIMITP > 0, example output is shown in Figure 5.17. With this option, element data is also written to the *lprog3.prn* file.

```
HEAD VALUES IN ELCONC
      1 0.600E-02   100001 0.600E-02      2 0.600E-02   100002 0.600E-02
      3 0.550E-02   100003 0.550E-02      4 0.550E-02   100004 0.550E-02
      5 0.500E-02   100005 0.500E-02      6 0.500E-02   100006 0.500E-02
      7 0.450E-02   100007 0.450E-02      8 0.450E-02   100008 0.450E-02
.
 NNHELD=   32 (IBHELD(I),I=1,NTBCC)
    8   16   24   32
(NHELD(I),AHELD(I),I=1,NNHELD)
      1   0.32278      100001   0.51389E-02      2   0.51389E-02   100002   0.25694E-02      3  -0.75000E-02
  100003  -0.37500E-02      4  -0.37500E-02   100004  -0.56250E-02      1   0.51389E-02   100001   0.25694E-02
      2   0.32278      100002   0.51389E-02      3  -0.37500E-02   100003  -0.18750E-02      4  -0.75000E-02
  100004  -0.37500E-02      1   0.51389E-02   100001   0.32278         2   0.25694E-02   100002   0.51389E-02
      3  -0.37500E-02   100003  -0.75000E-02      4  -0.18750E-02   100004  -0.75000E-02      1   0.25694E-02
  100001   0.51389E-02      2   0.51389E-02   100002   0.32278         3  -0.18750E-02   100003  -0.37500E-02
      4  -0.37500E-02   100004   0.0000
(number(nheld(i)),BHELD(I),I=1,NTBCC)
      1  -0.46250E-01
  100001  -0.46250E-01
      2  -0.46250E-01
  100002  -0.46250E-01

 1 DELTAT=   10.00     TIME= 10.0000    CONC MAX. 0.696104E-01 NODE# 100004  MIN. 0.000000E+00 NODE#     15 ETime0.670552E-09
 NHELD IN BACKSU NSOLVE          3
  1  0.3227778E+00   2  0.5138889E-02   3  0.5138889E-02   4  0.2569444E-02
  5 -0.7500000E-02   6 -0.3750000E-02   7 -0.3750000E-02   8 -0.5625000E-02
  1  0.5138889E-02   2  0.2569444E-02   3  0.3227778E+00   4  0.5138888E-02
```

Figure 5.16. Truncated Example *lprog3.prn* Output for NPRINT = 5 and *LIMIT* = *LIMITP* = 0. Vertical dots mark where full output is not shown due to space considerations.

```
    IX  1 NE      1 MN  1 NN  8 NOD      1      3      4      2   100001   100003   100004   100002
    CROCK 1.000     THETA0.100
    XIJCOB  31.250     31.250     31.250     31.250     31.250     31.250     31.250     31.250
    IX  2 NE      2 MN  1 NN  8 NOD      3      5      6      4   100003   100005   100006   100004
    CROCK 1.000     THETA0.100
    XIJCOB  31.250     31.250     31.250     31.250     31.250     31.250     31.250     31.250
    IX  3 NE      3 MN  1 NN  8 NOD      5      7      8      6   100005   100007   100008   100006
```

Figure 5.17. Truncated Example *lprog3.prn* Output for NPRINT = 5 and *LIMIT* and *LIMITP* > 0. Vertical dots mark where full output is not shown due to space considerations.

5.5 Other Print Control Parameters for *lprog3*

In addition to the NPRINT control option, the *lprog3* module of the CFEST simulator provides four more parameters for controlling output written to the *lprog3.prn* file. These parameters, which work in conjunction with the NPRINT parameter, were discussed previously. For example, NPRINT must be greater than or equal to 2 for the LIMITP and LIMITT parameters to have an effect on output written to the *lprog3.prn* file, whereas NPRINT ≥ 1 for the NPMASS print control parameter. These minimum NPRINT levels, as well as a description of the parameters and their options, appear in Table 5.5.

5.6 Additional Output Files

Table 4.1 lists the standard output files that are written by each of the CFEST modules. When running the CFEST main module, *lprog3*, additional output may also be generated. Table 5.6 lists the output files and a brief description of their contents, as well as conditions that cause them to be generated.

Table 5.5. Description of LIMITP, LIMITT, NSUBP, NPMASS Print Control Parameters

Parameter	Minimum NPRINT Level	Options	Description
LIMITP	2	0	No additional printout
		n	When lprog3 NPRINT > =2, generate printout for only *n* elements
LIMITT	2	0	No additional printout
		n	When lprog3 NPRINT > =2, generate printout for only *n* time steps (including sub-steps)
NSUBP	0	0	No screen output
		1	Print to screen the completion of each key subroutine in *lprog3*
NPMASS	1	0	No additional printout
		1	Print fluid and contaminant mass balance after each time step
		2	Print fluid and contaminant mass balance after each time step and sub-divided time step

Table 5.6. Optional Output File Descriptions

Output Filename	Condition	Description
initial.prn	NPRINT > 1	Initial condition information
nodalq_bal.prn	NSTDYH > 0	Held and flux node boundary condition data
	NSTDYH > 0 LEAKND > 0	Leakance boundary condition data
	NSTDYH > 0 NSTREAM > 0	Stream boundary condition data
nodalm_bal.prn	IOPT_MASSFILE = 1 NCYES = 1 NTBCH > 0	Fluid flux and mass flux at each held head node at each time step
	KTYPE > 0 NCYES = 1 NTBCC > 0	Mass flux at each held concentration node at each time step
	LEAKND > 0 NCYES = 1	Mass flux at each leakance node at each time step
dry_nodes.prn	If elements dry for a free surface	Data for nodes connected to drying elements during free surface calculation
	IOPT_K = 0 and elements dry for a free surface	Data for nodes connected to drying elements during free surface calculation
restartz.prn	ITRANS > 0	Initial restart elevations for unconfined aquifers
stream.prn	NSTREAM > 0	Stream data at each time step
scratch.bin	IMASS_ERR > 0	Data on residual error check for mass transport
.zzz	IOPT_MASSFILE = 1	Mass balance data
fluidbal.err	fluid mass balance error > FLUID_NODAL_ERRMIN	Fluid mass balance data for errors exceeding user-specified tolerance
massbal.err	solute mass balance error > SOLUTE_NODAL_ERRMIN	Fluid mass balance data for errors exceeding user-specified tolerance

6.0 CFEST Utilities

Several CFEST-based utilities can also be used to pre- and post- process data. Because CFEST writes all data in binary form, post-processors are required for data extraction. Most of the utilities described in this section are used for this purpose. Only brief descriptions of the pre-processor used for inverse simulation (*cf_ucode*), as well as the main module used to perform the inverse (*lp3ucode_005*), are provided. The use of these codes is documented in Cole et al. (2001).

6.1 Ascii Data Extraction for Tecplot: *tec_util_005*

The *tec_util_005* program is a post-processor used to extract *lprog3_005* binary data to Tecplot[(a)] input format. The post-processor must be executed in the current run directory since it reads the binary files located in the *cf_tmpbinary* directory. Head and concentration data, hydraulic conductivities, and nodes and elements, as well as velocities, are written to a *t-allhc.dat* file in the *tec_dir* directory, which is created upon execution of *tec_util_005*. Note that all hydraulic properties are written out as nodal properties. If elemental properties are desired, then the *cf_tecplt3d_005* program should be used (see Section 6.2).

Inputs are required at run time. Table 6.1 lists the required inputs and their descriptions. If an *lp3ucode.ctl* file is present in the run directory, an extra input (LP3UCODE) is required, which determines whether or not the parameter multipliers found in the *.sub* file are to be used in the data extraction. This *.sub* file is used for inverse simulation (see Section 6.10). If an *lp3ucode.ctl* file is not present, the LP3UCODE input is not required.

The *tec_util_005* program first prompts the user for head and concentration data extraction, fence diagram creation, or exit code. If head and concentration data extraction is selected (IOPT_TEC = 1), then a *tec_util.ctl* file is required. Run time inputs are described in Table 6.1. (In Table 6.1, the IOPTT parameter is used as a flag three times.) Example run time inputs are given by the following:

```
1       Extract “sub” properties.
1       Extract head and concentration all layers
5       Start Time Step
32      End Time Step
1       Time step Increment
1       Create 3d composite file of whole system (0=NO 1=Y)
0       Generate individual Layers Separately (0=NO 1=Y)
0       Generate NODELM files (0=NO 1=Y
99      End current session
```

In this example, the multipliers in the *.sub* file are used, and head and concentration data are extracted for all layers in the domain. The starting time step is 5 and the ending time step is 32, and all time planes between these two endpoints are extracted. In this example, and for most applications, the node (*.nod*) and element (*.elm*) files are not generated. Execution of *tec_util_005* is terminated with the termination flag, *99*.

For IOPT_TEC = 1, a *tec_util.ctl* file must be present in the current run directory. A description of the required input parameters for the control file is found in Table 6.2. The input format for the control file is free format. An example *tec_util.ctl* file is shown below:

(a) Tecplot RS, Tecplot © 1988 – 2006 Amtec Engineering, Inc., Bellevue, Washington.

```
0,                Line 1 Ifactorxy (1=convert modeling units to meters 0=no)
0,                Line 2 IoptHSU (0=none 1=merge model layers to HSU GROUP)
'NONE'            Line 3 File name having two columns (1) Model Layer (2) HSU #
1,                Line 4 Extract Head data IOPT_H? (0=no 1=yes)
1,                Line 5 Extract Conc data IOPT_C? (0=no 1=yes)
1944.0            Line 6 Simulation start year
0                 Line 7 IDIG_YEAR (0=integer >0 number of digit after "." for YEAR display
365.25            Line 8 FACTOR_TIME (number of time units/year e.g. 365.25 (days)
2,2,              Line 9 IX_YEAR, IY_YEAR **Plot Co-ordinates in % to write year
'F10.4'           Line 10 Output Format for Head data(F8.2,F12.4,E15.6 ..)
'E10.4'           Line 11 Output Format for Conc. data(F8.2,F12.4,E15.6 ..)
0,0,0,0,0,        Line 12 Vector plot, injection/pumping, held BC, max/min data
```

In this example, head and concentration data are extracted (lines 4 and 5). No additional data (e.g., vector, source/sink, held boundary conditions, max/min data) will be extracted, and no grouping of hydrostatic units will occur (lines 2 and 3). This option should be used if refinement of a single unit occurred in the simulation, but for display purposes these subdivided units are to be merged into a single hydrostatic unit. Other input lines in the control file set format descriptors for the head and concentration data (lines 10, 11), X and Y starting locations for printing the year (line 9), as well as the format for the year (line 7). The simulation time units were days; thus the conversion factor (line 8) to years is 365.25.

Table 6.1. Required Run Time Inputs for *tec_util_005*

Parameter	Options	Description
LP3UCODE	0	Extract default data; this parameter occurs only when lp3ucode is performed.
	1	Extract "sub" properties
IOPT_TEC	1	Head and Concentration all layers
	2	Fence diagram (*tecfence.ctl* required)
	99	End current session
ITSTART	> 0	Start time step
ITEND	0	End time step
ITINC	1	All time step increments
	n	Every n^{th} time step
IOPTT	0	Do not create 3D composite file of whole system
	1	Create 3D composite file of whole system
IOPTT	0	Do not generate individual layer separately
	1	Generate individual layer separately
IPT3D2D	0	Only occurs when generating individual layer separately; generate 3D data file for each layer
	1	Generate 2d data file for each layer
IOPTT_VECT	0	No flow vectors generated; only occurs when generating individual layer separately
	1	Generates flow vector
IOPTT	0	No generation of node and element files
	1	Generates node and element files
IOPT_TEC	1	Head and Concentration all layers
	2	Fence diagram (*tecfence.ctl* required)
	3	Local scale processing
	99	End current session

Table 6.2. Input Parameters for the *tec_util.ctl* Control File (IOPT_TEC = 1)

Line #	Parameter	Options	Description
1	IFACTOR_XY	0	No units conversion
		1	Convert units to meters
2	IOPT_HSU	0	Use model layers
		1	Merge model layers to HSU Group
3	FILE_HSU	filename	File name that lists model layers and the corresponding hydrostatic unit
		none	Hydrostatic grouping not used/no filename
4	IOPT_H	0	No head data extraction
		1	Extract head data
5	IOPT_C	0	No concentration data extraction
		1	Extract concentration data
6	YYEAR	n	Simulation start year
7	IDIG_YEAR	0	Integer
		n	Number of digits after integer year
8	FACTOR_TIME	n	Time conversion factor for year (e.g., 365.25 days)
9	IX_YEAR	n	Starting X position to print year on plot (in % plot coordinates)
	IY_YEAR	n	Starting Y position to print year on plot (in % plot coordinates)
10	FMTHEAD		FORTRAN output format for head data (e.g., F8.2)
11	FMTCONC		FORTRAN output format for concentration data (e.g., F8.2)
12	IOPT_VECT	0	No extraction of vector plot data
		1	Extraction of vector plot data
	IOPT_QDATA	0	No extraction of injection/pumping data
		1	Extraction of injection/pumping data
	IOPT_HELD	0	No extraction of held boundary condition data
		1	Extraction of held boundary condition data
	IOPT_MNMX	0	No extraction of minimum/maximum data
		1	Extraction of minimum/maximum data
	IOPT_GRDBND	0	No extraction of external boundary data
		1	Extraction of external boundary data

If a fence diagram is desired (IOPT_TEC = 2), then a *tecfence.ctl* file is required (Table 6.3). File names listed in the *tecfence.ctl* file should use a file extension of *.ver* for vertical sections and a file extension of *.hor* for horizontal sections. The first line in the *.ver* file contains the total number of nodes listed in the file, followed by a list of surface nodes that are space delimited on the second line in the file. The *tec_util_005* utility automatically identifies subsurface nodes that correspond to the surface nodes. For the *.hor* files, the layer number and the total number of elements are listed on the first line. On the next line of the *.hor* file, space delimited surface elements are listed. The *tec_util_005* utility automatically identifies nodes associated with both the unit and the X and Y locations of the elements specified in the *.hor* file.

Table 6.3. Input Parameters for the *tecfence.ctl* Control File (IOPT_TEC = 2)

Line #	Parameters	Options	Description
1	NFILES	n	Total number of filenames
2	FILENAME	filename	Filenames containing vertical (*.ver*) and horizontal (*.hor*) section data

6.2 Binary Data Extraction for Tecplot: *cf_tecplt3d_005*

The *cf_tecplt3d_005* utility is an upgrade of the *tec_util_005* post-processor. Although both extract data in Tecplot format, unlike *tec_util_005, cf_tecplot3d_005* extracts data in binary format. In addition, *cf_tecplt3d_005* extracts elemental properties for velocity, hydraulic conductivity and specific storage, rather than the node-based properties extracted with *tec_util_005*. Only *tec_util_005*, however, has the ability to extract properties using multipliers from a *.sub* file that resulted from an inverse simulation.

Unlike *tec_util_005*, no control file is needed to execute *cf_tecplt3d_005*. Run time required inputs include the simulation start year, time conversion factor, starting and ending time steps, and the time increment for data extraction (see Table 6.4). Example run time inputs are shown below:

```
1                IOPT_VEC3DTXT; Generate elemental velocities
2000             START_YEAR; simulation start year
1                IOPT_TIME; convert simulation time to years
1                IT_START; starting time step;
200              IT_END; ending time step
1                IT_INC; time increment for data extraction
0                IOPT_FILENAME; generate ./tec_dir/3d.plt
```

Table 6.4. Run Time Inputs for *cf_tecplt3d_005*

Parameter	Options	Description	Condition
IOPT_VEC3DTXT	0	No velocity data extracted	
	1	Extract elemental velocities to files *vec_time****.txt;* one file for each time plane	
START_YEAR	0	No change in simulation start year	
	n	Simulation start year	
IOPT_TIME	0	No time conversion	
	1	Convert simulation time from days to years	
IT_START	n	Starting time step	
IT_END	n	Ending time step	
IT_INC	n	Time increment at which data will be extracted from IT_START to IT_END	
IOPT_FILENAME	0	Generate default binary file *./tec_dir/3d.plt* for Tecplot	
	1	User-specified *.plt* filename	
FILENAME	filename	Prints to specified file	IOPT_FILENAME = 1

In this example, velocity data is written to a *vec_time****.txt* file in the *./tmp_vec3d* subdirectory created at run time for each time step at which data is extracted. The simulation start year is set at 2000, simulation units are converted from days to years, and the starting and ending time steps are 1 and 200, respectively. Data are extracted for each time step between 1 and 200 and is written to *./tec_dir/3d.plt*. Note that for the 200 time steps, the *tmp_vec3d* directory contains 200 *vec_time****.txt* files from *vec_time0001.txt* to *vec_time_0200.txt*.

Three different output files are generated, one binary file formatted for the Tecplot plotting program, and two ascii files. The first ascii file contains time series data for head and concentration, the coordinates at the center of each element, and elemental velocities in X, Y and Z (for IOPT_VEC3DTXT = 1; see Table 6.5). A second ascii file is generated that contains data on dry nodes and elements.

Table 6.5. Output File Description for *cf_tecplt3d_005*

Output files	Description
*./tmp_vec3d/vec_time****.txt*	Velocity output files, where **** is the time step number (e.g., *vec_time0001.txt* for the velocities at the first time step). These files contain data on the current time step, elapsed time, the coordinates of the element centroid, and elemental velocities.
./cf_tmpascii/cf_tecplot3d.dbg	Dry node and element data
./tec_dir/3d.plt (or user-defined file name with path)	Binary output file in Tecplot format. Contains X, Y and Z coordinates, node number, elevation, head, concentration, element number, material number, conductivity in X, Y and Z, specific storage, specific yield, elemental velocities in X, Y and Z, and dry node and element data.

6.3 Identification of Peak Concentrations: *compliance_005*

The *compliance_005* is a post-processing program that identifies the location of peak concentrations and arrival times for a user-defined boundary. It reads a surface boundary node list from a user-specified file (Table 6.6) and extracts concentration data at the specified nodes. One surface node is listed on each line in the input file.

There are three output files whose filenames are specified by the user (Table 6.6). The first output file stores the maximum concentration for each surface node in the user specified list. A second output file stores time series concentrations for each surface node and its corresponding subsurface node locations. The third output file stores the maximum concentration for each time step and identifies the nodal location. Example inputs required at run time are shown below:

```
1                          Multiplier on concentration (for unit conversion and print adj)
node.lst                   Input filename containing list of surface nodes
maxsurface.dat             Output filename with maximum concentrations at each node
timedependent.dat          Output filename with time series concentrations at each node
2000                       Simulation start year
365.25                     Number of days in a year
maxconc.dat                Output filename with maximum concentration at each node
4                          Number of digits to be printed after the decimal for concentration
```

Table 6.6. Run Time Input Requirements for *compliance*

Parameter	Options	Description
CONCADJ	1	Concentration adjustment factor
FILENAME	filename	Surface boundary node list file name
FILENAME	filename	Output filename that stores the maximum concentrations at each surface node
FILENAME	filename	Output filename that stores time series concentration data for each surface node and corresponding subsurface node
YEARSTART	n	Simulation start year
YEARADAYS	n	Number of days in a year
FILENAME	filename	Output filename that stores maximum concentrations and node (surface or subsurface) at which the maximum occurred for each time step
NDIG	n	The number of digits to be printed after the decimal. The format is E8.NDIG and F12.NDIG

In this example, the concentration adjustment factor is 1. Note that even though no unit conversion may be necessary, an adjustment factor is desirable if concentrations are very small. For example, the time series concentration data is written with an F12.NDIG format. If NDIG = 4, as in the example above, and concentrations are below 0.0001, the print format of F12.4 will not capture small concentrations. Using a multiplier that increases the concentration allows for the print format to capture small maximum concentrations. The example run time inputs show that the surface boundary node list is read from the *node.lst* file, and the three output files are named *maxsurface.dat, timedependent.dat* and *maxconc.dat*. The simulation start year is 2000, and the number of days in one year is 365.25.

6.4 Initial Conditions: *linitial_005*

The *linitial_005* utility is a program that is used to establish a new set of initial conditions from a previous run. For example, a time plane, *n*, is selected from a previous result file (i.e., a *.b07* file) and the record is transferred to a *.b01* file where initial conditions are stored. The source time plane can originate from the current run directory or from a previous run. The inputs and descriptions are listed in Table 6.7.

The *linitial_005* utility is usually used to extend a simulation that has already successfully terminated. This type of restart simulation requires that new time step information be entered in the *.l3i* file. If boundary conditions have changed, then those data can also be modified in both the *.l3i* and *.lp1* input files. To use the initial conditions from the successfully completed run, the *lprog1_005* program is first executed, followed by *lband_005* and *lprog3i_005*. The *linitial_005* program is then executed. Example run time inputs are shown below:

```
1                                   IOPTFILE
../run1/cf_tmpbinary/phase1.b07     Filename
10                                  Time plane used for extraction
1                                   Initial conditions write flag (0=no,1=yes)
1                                   Update head flag (0=no,1=yes)
1                                   Update concentration flag (0=no,1=yes)
```

In the above example, *linitial_005* is executed from the current restart directory. Hence, IOPTFILE is set to 1 so that it reads a *.b07* file in another directory. Since the file is not read from the current directory, the next run time input requires that a *.b07* filename be specified. Other run time inputs determine if head and concentration data are written as the initial conditions, and if the initial conditions are written to the *cf_tmpascii/linitial.prn* file.

Once *linitial_005* has successfully terminated, the initial conditions have been written to the current run directory *.b07* file. The *lprog3_005* module can then be executed with the new initial conditions.

Table 6.7. Run Time Input Requirements for *linitial_005*

Parameter	Options	Description
IOPTFILE	0	Extract information from *.b07* file located in current run directory
	1	Extract information from *.b07* file located elsewhere
FILENAME	filename	The filename and path of *.b07* file. (if IOPTFILE = 1)
ICOPT	n	The number of the time plane used as the source for initial conditions
ICRPIN	0	Do not print new initial conditions
	1	Print new initial conditions to *cf_tmpascii/linitial.prn*
IOPTH	0	Do not update head
	1	Update head
IOPTC	0	Do not update concentration
	1	Update concentration

6.5 Initial Concentration Conditions: *lconc-inout_005*

Like the *linitial_005* program, the *lconc-inout_005* utility can also be used to set initial conditions. Although it has functionality similar to *linitial_005*, it only sets concentration initial conditions. It also has an additional capability for writing initial conditions to a grid that has a different number of nodes than the grid containing the source data. This is useful if nodes are either deleted or added to a grid to simulate changes in the stratigraphy. Whereas the *linitial_005* program only writes to the initial conditions file (*.b01*), the *lconc-inout_005* program can also write concentration data for a restart simulation to a new time plane (*.b07* file) as well.

The *lconc-inout_005* program is usually executed in two steps. In the first step, concentration data are extracted from a previous simulation, and in the second step, the concentration data are written to the binaries for a current simulation. When *lconc-inout_005* is executed, the user is prompted for a value for the IOPTIO parameter (Table 6.8). For IOPTIO = 1, concentration data is exported from the current run directory. The utility reads concentration data from the *cf_tmpbinary/.b07* file for a user-specified time plane (ICOPT = *n*), and writes the data to a user-specified ascii file (AFILENAME). Example run time inputs are shown below:

```
1               IOPTIO; ASCII IN OR OUT? (0=Import (in), 1=Export (out))
conc.dat        AFILENAME – Output filename
5               ICOPT; Time plane number
```

In the above example, concentration data is extracted from time plane number 5 of the current working directory, and written to a file named *conc.dat*.

Table 6.8. Run Time Input Requirements for *lconc-inout_005* (IOPTIO = 1)

Parameter	Options	Description
IOPTIO	1	Export concentration data to an ASCII file
AFILENAME	filename	ASCII output filename for storing concentration data
ICOPT	n	Time plane number to be used as the source for concentration data extraction

In the second step, IOPTIO = 0, concentration data are written to the *.b01* or *.b07* binary files stored in the *cf_tmpbinary* directory (Table 6.9). This directory, presumably, is different than the directory from which the concentration data were extracted when IOPTIO = 1. Example run time inputs for IOPTIO = 1 are shown below:

```
0                   IOPTIO; ASCII IN OR OUT? (0=Import (in), 1=Export (out))
../phase1/conc.dat  AFILENAME – Output filename
0                   ICOPT; Time plane number
0
```

In this example, concentration data are read from the *conc.dat* file located in the *./phase1/* directory and written to the initial conditions (*.b01*) file. No data are written to the *.b07* file. The *lconc_inout_005* program for IOPTIO = 0 is run after executing *lprog1_005*, *lband_005* and *lprog3i_005* and before execution of *lprog3_005*.

Table 6.9. Run Time Input Requirements for *lconc-inout_005* (IOPTIO = 0)

Parameter	Options	Description
IOPTIO	0	Import concentration data from ASCII file for use as initial conditions (*.b01*) or to rewrite timeplane for a restart (*.b07*) simulation
AFILENAME	filename	ASCII input filename with concentration data
ICOPT	0	Override the initial concentrations
IRWR	0	Do not rewrite concentrations in *.b07* file
	1	Rewrite concentrations in *.b07* file
FILENAME	filename	Name and path of *.b07* file.
INWR	n	Time plane to rewrite in *.b07* file

6.6 Time Dependent Data Extraction: *lresult_005*

The *lresult_005* utility is a post-processing program used to extract time series data at a user-specified location. Head, concentration, and density data may be extracted for a single point over time or at all nodes for a single time plane. Several different options exist for extracting data, and run time input requirements differ depending on the first run time input requirement INDEX and on switches set for the simulation (Table 6.10). For INDEX = -1, for example, the parameter NPRES, which sets the print format for head, is a required run time input. If NPRES = 3, then the user is prompted for a value of NREF, the reference time plane number; otherwise NREF is not used. For INDEX = 6, IFRAME and NDIG inputs are not used. Similarly, when IFRAME = 0 (extract data at a single node), an input value for NODEE is required (the node at which the data are to be extracted). For IFRAME = 1, NODEE is not required because data are extracted at every node in the domain. An example of *lresult_005* run time input is shown below:

Table 6.10. Run Time Input Requirements for *lresult_005*

Parameter	Options	Description	Condition
INDEX	1	Extract head or pressure	
	-1	Extract head or pressure from a reference time plane	
	3	Extract concentration	
	-3	Extract concentration change from the initial condition	
	4	Extract density	
	-4	Extract density change from the initial condition	
	5	Extract fluxes for held flow boundaries (QEST)	
	6	Extract head, concentration, density and fluxes for each node at every time plane	
NPRES	1	Print head as sum of elevation and pressure heads using reference density	ABS(INDEX) = 1 KTYPE > 0
	2	Print head as sum of elevation and pressure heads using density at node	
	3	Print head change from the initial condition	
ITTO	n	Start time step	
ITTT	n	Final time step (0 = retrieve all time steps)	
IFRAME	0	Print data from a single node	
	1	Print data for all nodes at each time plane	
	2	Print X, Y, Z, head and concentration	NPRES condition is ignored; prints head only
FACT	n	X and Y coordinate conversion factor	IFRAME = 2
NDIG	n	Number of digits to be printed after the decimal (F8.<NDIG>)	
NPRINT	0	No screen output	IFRAME = 0
	1	Echo output data to screen	
IFILENEW	0	Write all output to a single output file	IFRAME = 0
	1	Write output for each node to user-specified FILENAME	
FILENAME	c	Output file name(s). For IFILENEW=0, only one file name is entered; for IFILENEW=1, multiple file names are entered, one for each node requested for data extraction	
NODEE	n	Surface node number for data extraction	IFRAME = 0
	-n	Surface node number and corresponding subsurface nodes	
	0	Exit	
	Ctrl^z		
	Ctrl^y		

```
3           INDEX, retrieve concentration
1           ITT0, start time step
0           ITTT, final time step
0           IFRAME
6           NDIG F8.<NDIG>
1           NPRINT, echo nodal result
0           IFILENEW
conc.dat    FILENAME, output file name
5           NODEE
-10         NODEE
0           NODEE
```

In this example, *lresult_005* retrieves concentration data from all time steps (ITT0 = 1 and ITTT = 0). For IFRAME = 0, nodal concentration data extraction occurs with an output format of F8.6. Concentration data are printed to the *conc.dat* output file and echoed to the screen (NPRINT = 1). As shown above, *lresult_005* prompts for multiple node numbers until an exit status (e.g., NODEE = 0) (Table 6.10) is received. Data are extracted for surface nodes 5 and 10, as well as subsurface nodes beneath node 10. Note, however, that in order to extract data for subsurface nodes (NODEE = -n), IFILENEW must be set to 0.

6.7 Data Recovery: *ltrace_005*

The *ltrace_005* utility is primarily used when a simulation terminates abnormally (e.g., due to a loss of power). The utility allows the user to change simulation flags that indicate that a main module has terminated successfully, as well as the number of time steps that have been successfully completed (Table 6.11). For example, if the loss of power caused an *lprog3_005* simulation to crash, the data at the current time step may be corrupted. Should this occur, it is desirable to reset the number of time steps completed to one unaffected by the data corruption, as shown in the example below:

```
1           IWISH (Change flags?)
1,1,1,1,    NPROG1, NBAND, NPROG3I, NPROG3
10          IT
```

In this example, the simulation switches are set to successful completion, and the number of completed time steps reset to 10.

Table 6.11. Run Time Input Requirements for *ltrace_005*

Parameter	Options	Description	Condition
IWISH	0	Do not change simulation flags and number of completed time steps	
	1	Change simulation flags and number of completed time steps	
NPROG1	0	*lprog1_005* not yet executed	IWISH = 1
	1	*lprog1_005* successfully terminated	
NBAND	0	*lband_005* not yet executed	
	1	*lband_005* successfully terminated	
NPROG3I	0	*lprog3i_005* not yet executed	
	1	*lprog3i_005* successfully terminated	
NPROG3	0	*lprog3_005* not yet executed	
	1	*lprog3_005* successfully terminated	
IT	n	Number of time steps completed	

6.8 Fluid and Mass Fluxes at Held Head Boundaries: *lgethheld_005*

The *lgethheld_005* utility is a post-processor used to extract fluid and mass fluxes at nodes where held head boundary conditions are imposed. Unlike other CFEST post-processors, no run time inputs are required. Instead, a control file (*lgethheld.ctl*) containing required inputs must be present in the current run directory (Table 6.12). An example *lgethheld.ctl* file is shown below:

```
1,            ICYES
50,           IMAX
'F12.3',      FMTQ
'E15.6',      FMTM
```

In this example ICYES = 1, which means that both fluid and mass flux data are written to the output files. For fluid fluxes output is always written *to ./hheldq/hheldq001.tab*, whereas mass flux data are always written to *./hheldm/hheldm001.tab*. The user specifies IMAX, which sets the number of nodes at which flux data are reported. If IMAX is less than the total number of held head boundary nodes, some of the flux data will not be reported and is determined sequentially by user node number. The format is F12.3 (FMTQ) for fluid flux output and E15.6 (FMTM) for mass flux output. Even when ICYES = 0, the FMTM parameter is still a required input.

Flux data are written to the output files for each time step. A *TotalQandMass.dbg* file is written to the *./cf_tmpascii* subdirectory, which allows the user to compare total fluxes to those written to the *./cf_tmpascii/lprog3.prn* output file (see Section 4.2).

Table 6.12. Input Requirements for the *lgehtheld_005* Control File (*lgethheld.ctl*)

Line #	Parameter	Options	Description
1	ICYES	0	Extract fluid fluxes only
		1	Extract fluid and mass fluxes
2	IMAX	n	Maximum number of nodes in one file (e.g., 50,100,…)
3	FMTQ		Fluid flux output format (must be enclosed in single quotes)
4	FMTM		Mass flux output format (must be enclosed in single quotes)

6.9 Data Extraction for ArcInfo: *rectgrid_out_005*

6.9.1 Control File

The *rectgrid_out_005* utility is a post-processing program used to extract head, concentration, surface recharge, horizontal conductivity, vertical conductivity, specific storage, and specific yield data stored in CFEST binary files to ArcInfo[(a)] grid format. No inputs are required at run time, but a control file (*rectgrid_out.ctl*) must be present in the current run directory. The *rectgrid_out.ctl* file sets options for extracting data to be used within the ARC/Info Geographic Information System. An example *rectgrid_out.ctl* file appears below:

(a) ArcInfo © 1995 - 2006 Environmental Systems Research Institute, Inc. (ESRI), Redlands, California..

```
0,1,                        Nprint,NprintCtl(0=normal,1=print non_comment Lines,2=Echo all CTL
0,                          Irun_Lp1L3i(0=n,1=y).Lp1Lp3i updates kelem property binary files
'%cfest%\lp1lp3i_001.exe', Used if Irun_Lp1L3i=1
0,                          nref
0,                          iopt_hsu (0=extract by layer 1=by hsu);
'xlate.hanford',            Two Column table of material type and hsu no.
1,'(13F10.3)',              iopt_h,format; Extract Head
0,'(13F10.3)',              iopt_c,format; Extract Conc
0,'(12E15.6)',              iopt_q,format;Surface Recharge
0,'(12E15.6)',              iopt_Kx,format; K horizontal
0,'(12E15.6)',              iopt_kz,format; K vertical
0,'(12E15.6)',              iopt_ss,format; Specific Storage
0,'(12E15.6)',              iopt_sy,format; Specific Yield  (same as porosity)
1167,1311,50.0,,,           NX,NY,DXY (NX & NY should be <=1500)
100,100                     xmin,ymin,
1.0,                        factorXY conversion factor from ft to m (0.3048)
5,                          # points in distance calc. in Arc grid replaced w/nearest active value
```

The first line of the control file sets print options NPRINT and NPRINT_CTL. For NPRINT = 0 no screen output occurs, and for NPRINT_CTL = 1 non-commented lines in the *rectgrid_out.ctl* file are printed to the screen (Table 6.13). The second line of the control file sets the IRUN_LP1L3I flag for updating the hydraulic properties of the elements stored in the CFEST binary files. The third line of the control file names the program used in the update, but since IRUN_LP1L3I = 0 in the example above, the content of LP1LP3I_EXE is not used, but is still a required input.

Line 4 sets the time plane at which the data extraction occurs (NREF). For NREF = 0, data representing the initial condition are extracted. For IOPT_HSU = 0 (line 5) data are not extracted by hydrostatic unit, and the filename listed on line 6 of the control file is not used (but is still a required input).

On lines 7–13, flags are set for each data type that is extracted. For IOPT_* = 1, data are extracted for the data type. If IOPT_* = 0, the data are not extracted. Following the data extraction flag is the FORTRAN format that is used for writing to the output files. In the example above, only head is extracted (IOPT_H = 1) and is written with a format of 13F10.3. No other data are extracted.

Line 14 sets the number of data points in X (NX = 1167) and Y (NY = 1311), as well as the cell size or vertical and horizontal distance between data points (DXY = 50.0). The coordinates of the lower left corner are set at 100 in X and Y (XMIN = 100 and YMIN = 100), and no conversion of X and Y coordinates occurs (FACTORXY = 1). The number of points used to calculate the distance from an inactive data point in the ArcInfo X-Y grid to an active cell is 5. This active data point is then used at the inactive data point location.

6.9.2 Output Files

Several output files are generated by the *rectgrid_out_005* utility for archiving head, concentration, surface recharge, horizontal conductivity, vertical conductivity, specific storage, and specific yield data in ArcInfo format. For each data type requested, an ascii gridded file is generated in the *rect_out* directory (Table 6.14). Apart from the gridded data files, a *rectgrid_out.dxy* file is generated that contains the coordinates at the lower left corner (XMIN, YMIN), those at the upper right corner (XMAX,YMAX), the number of grid points in X and Y, and the distance between grid points (NX,NY,DXY).

Table 6.13. Input Requirements for the *rectgrid_out_005* Control File (*rectgrid_out.ctl*)

Line #	Parameters	Options	Description
1	NPRINT	0	No screen output
		1	Screen output for program process and error reporting
	NPRINTCTL	0	No echo to screen of *rectgrid_out.ctl*
		1	Echo input lines in *rectgrid_out.ctl* to screen except comment lines
		2	Echo all input lines in *rectgrid_out.ctl* to screen
2	IRUN_LP1LP3I	0	No update of properties stored in binary files
		1	Execute lp1lp3i_exe to update binary property files
3	LP1LP3I_EXE	filename	Path and executable program used to update property binary files for IRUN_LP1LP3I = 1
4	NREF	n	Time plane at which data extraction occurs
5	IOPT_HSU	0	Extract data by material type
		1	Extract data by user-defined hydrostatic unit
6	FILENAME	filename	Filename containing two-column table with material type and hydrostatic unit number for IOPT_HSU = 1
7	IOPT_H	0	No data extraction for head
		1	Extract head using user-defined format
	FORMAT		FORTRAN format for head
8	IOPT_C	0	No data extraction for concentration
		1	Extract concentration using user-defined format
	FORMAT		FORTRAN format for concentration
9	IOPT_Q	0	No data extraction for surface recharge
		1	Extract surface recharge using user-defined format
	FORMAT		FORTRAN format for surface recharge
10	IOPT_KX	0	No data extraction for horizontal hydraulic conductivity
		1	Extract horizontal hydraulic conductivity using user-defined format
	FORMAT		FORTRAN format for horizontal hydraulic conductivity
11	IOPT_KZ	0	No data extraction for vertical hydraulic conductivity
		1	Extract vertical hydraulic conductivity using user-defined format
	FORMAT		FORTRAN format for vertical hydraulic conductivity
12	IOPT_SS	0	No data extraction for specific storage
		1	Extract specific storage using user-defined format
	FORMAT		FORTRAN format for specific storage
13	IOPT_SY	0	No data extraction for specific yield
		1	Extract specific yield using user-defined format
	FORMAT		FORTRAN format for specific yield
14	NX	n ≤ 1500	Number of grid points in the X coordinate direction
	NY	n ≤ 1500	Number of grid points in the Y coordinate direction
	DXY	n	Horizontal and vertical distance between data points
15	XMIN,	n	X coordinate of lower left corner of domain
	YMIN	n	Y coordinate of lower left corner of domain
16	FACTORYXY	n	Conversion factor for X and Y coordinate data
17	NXY_OUTER	n	Number of data points in the X-Y ArcInfo grid used to calculate the distance between an active and inactive data point

Table 6.14. Output Files Generated by *rectgrid_out_005*

Output file	Description
./rectgrid_out/gridedhh.l##	Extracted head data at layer number
./rectgrid_out/gridedcc.l##	Extracted concentration data at layer number
./rectgrid_out/gridedqq.top	Extracted surface recharge data
./rectgrid_out/gridedkx.l##	Extracted horizontal conductivity data at layer number
./rectgrid_out/gridedkz.l##	Extracted vertical conductivity data at layer number
./rectgrid_out/gridedss.l##	Extracted specific storage data at layer number
./rectgrid_out/gridedsy.l##	Extracted specific yield data at layer number
./rectgrid_out/rectgrid_out.dxy	Coordinates at the lower left corner (XMIN, YMIN), coordinates at the upper right corner (XMAX,YMAX), and the number of grid points in X and Y, and the distance between grid points (NX,NY,DXY).
./cf_tmpascii/sca_grid.lis	Local X coordinate of element corresponding to each grid point in the gridded data file
./cf_tmpascii/eta_grid.lis	Local Y coordinate of element corresponding to each grid point in the gridded data file
./cf_tmpascii/elm_grid.lis	User element number corresponding to each grid point in the gridded data file

In addition to these files, *rectgrid_out_005* generates *.lis* files in the *.cf_tmpascii* subdirectory containing data on elements used to write to the gridded data files. The local X coordinate data at each point in the gridded files are written to *sca_grid.lis*, while the local Y coordinate data are written to *eta_grid.lis,* and the element corresponding to the data extracted is written to the *elm_grid.lis* file (Table 6.14).

6.10 Inverse Simulations with UCODE

CFEST has additional capabilities for performing inverse simulations with UCODE, a computer code for Universal Inverse Modeling (Poeter and Hill 1998). UCODE provides an efficient means for parameter optimization and quantitative uncertainty analysis. UCODE performs inverse modeling, posed as a parameter-estimation problem, by calculating parameter values that minimize a weighted least-squares objective function using nonlinear regression. UCODE minimizes objective functions using a modified Gauss-Newton method.

To implement CFEST with UCODE, a second module, *lp3ucode_005*, was developed based on *lprog3_005*. The *lp3ucode_005* module has the same capabilities as the *lprog3_005* module but calculates differences between observed and simulated data for UCODE input and implements new parameter estimates based on UCODE output. The UCODE program is not integrated into *lp3ucode_005* and thus the program is needed for successful execution of an inverse simulation. An additional utility, the *cf_ucode_005* preprocessor, was developed for automated generation of UCODE input files and for labeling temporal and spatial well data.

Only a brief description of program functionality is presented here. Both the *cf_ucode_005* and *lp3ucode_005* modules are described in detail in Cole et al. (2001). The UCODE functionality is also described in Cole et al. (2001), as well as in the UCODE User Guide (Poeter and Hill 1998).

6.10.1 UCODE Preprocessor: *cf_ucode_005*

The *cf_ucode_005* utility was developed to provide a set of streamlined procedures and tools to simplify the use of UCODE. The *cf_ucode_005* utility reads UCODE parameters from a control file (*cf_ucode.ctl*) and generates *.uni*, *.pre*, *.sub*, *.tpl*, *.hed* and other interface files. The control file also includes the names of the files containing the observation data and the filename containing the paracodes (i.e., parameter codes) that simplify the parameterization. The *cf_ucode_005* preprocessor also generates a control file (*lp3ucode.ctl*) that the main module, *lp3ucode_005*, uses to identify input filenames (Cole et al. 2001).

The paracodes are grouped into the following broad categories: 1) hydraulic properties (i.e., hydraulic conductivities in the x, y, and z direction; specific yield; specific storage; porosity), 2) surface recharge, 3) stream and river properties, 4) boundary flux, and 5) well flow rates. They can be defined based on zones or on material types. These codes do not yet include transport-related parameters (e.g., dispersivity, retardation, and porosity).

Before executing the *cf_ucode_005* the three preprocessors, *lprog1_005*, *lband_005* and *lprog3i_005*, must first be executed (see Section 4). Once *cf_ucode_005* has successfully completed, the main module designed to interface with UCODE, *lp3ucode_005,* is executed.

6.10.2 Main Module UCODE Interface: *lp3ucode_005*

One of the required inputs to UCODE is the name and location of the application used in the inverse simulation. Once UCODE is executed, it reads all required UCODE input generated by the c*f_ucode_005* utility, including the *lp3ucode_005* application named in the *.uni* file. UCODE executes the *lp3ucode_005* module, which then reads filenames identified in the *lp3ucode.ctl* control file. The *.sub* file listed in the *lp3ucode.ctl* file is read, and then hydraulic properties and flux rates are updated.

For each parameter change, *lp3ucode_005* generates debug output in *./cf_tmpascii/paracode.dbg*, which is overwritten for each new execution of *lp3ucode_005*. The *lp3ucode_005* module also extracts simulation results for the observation data points and updates *.cfo*, *.hed*, and *.riv* files.

Once an inverse simulation has been completed, the *.sub* file includes optimized values of parameters. Final results can be viewed in *.hed* and *.riv* files. Users are advised to re-run *lp3ucode_005* after successful completion of UCODE because the final parameter estimates do not necessarily correspond to the most recent simulation but to an earlier iteration. If *lp3ucode_005* is executed in parallel, it is even less probable that the final simulation corresponds to the best estimates of the parameters (Cole et al. 2001).

7.0 Literature Cited

Balay S, K Buschelman, V Eijkhout, WD Gropp, D Kaushik, MG Knepley, LC McInnes, BF Smith, and H Zhang. 2004. *PETSc Users Manual.* ANL-95/11 Rev. 2.1.5, Argonne National Laboratory, Argonne, Illinois.

Cole CR, MP Bergeron, SK Wurstner, PD Thorne, S Orr and MI McKinley. 2001. *Transient Inverse Calibration of Hanford Site-Wide Groundwater Model to Hanford Operational Impact: 1944–1966.* PNNL-13447, Pacific Northwest National Laboratory, Richland, Washington.

Freedman VL, Y Chen, and SK Gupta. 2005. *CFEST Coupled Flow, Energy & Solute Transport, Version CFEST005, Theory Guide*. PNNL-15452, Pacific Northwest National Laboratory, Richland, Washington.

Gupta SK, CR Cole, CT Kincaid, and AM Monti. 1987. *Coupled Fluid, Energy, and Solute Transport (CFEST) Model: Formulation and User's Manual.* BMI/ONWI-660, Battelle Memorial Institute, Columbus, Ohio.

Gupta SK. 1997. *Draft User's Manual, CFEST-96 Flow and Solute Transport, Constant/Variable Density, Computationally Efficient, and Low Disk PC/Unix Version*. Environmental System Technologies, Irvine, California.

Oppe TC, WD Joubert, and DR Kincaid. 1988. *NSPCG User's Guide, Version 1.0*. Center for Numerical Analysis, The University of Texas at Austin. Available at: http://rene.ma.utexas.edu/CNA/NSPCG/manuals/usernsp/.

Poeter EP and MC Hill. 1998. *Documentation of UCODE: A Computer Code for Universal Inverse Modeling*. U.S. Geological Survey Water-Resources Investigations Report 98-4080, USGS, U.S. Army Corp of Engineers Waterways Experiment Station, and International Ground Water Modeling Center of the Colorado School of Mines, Golden, Colorado.

Young D and D Kincaid. 1981. "The ITPACK Package for Large Sparse Linear Systems." *Elliptic Problem Solvers,* M Schultz (ed.). Academic Press, New York, pp. 163–185.

Distribution

No. of Copies

ONSITE

2 CH2MHill Hanford Group, Inc.

F.J. Anderson	E6-35
F.M. Mann	E6-35

5 Fluor Hanford Corporation

J.V. Borghese	E6-35
M.E. Byrnes	E6-35
T.W. Fogwell	E6-35
B.J. Ford	E6-35
D. Miller	

5 U.S. Department of Energy

B.L. Charboneau	A6-38
D.R. Hildebrand	A6-38
J.G. Morse	A6-38
M. Thompson	A6-38
A.C. Tortoso	A6-38

2 DOE Office of River Protection

R.W. Lober	H6-60
S.A. Wiegman	H6-60

1 EPA

D.A. Faulk	B1-46

22 Pacific Northwest National Laboratory

R. W. Bryce	K9-35
M.P. Bergeron	K9-36
Y. Chen	K9-36
C. R. Cole	K9-36
P.W. Eslinger	K6-52
V. L. Freedman (2)	K9-36
E.J. Freeman	K9-36
M.D. Freshley	K9-33
A. Gilca	K6-52
G.E. Hammond	K9-36

C.T. Kincaid	K9-33
C.J. Murray	K6-81
W.E. Nichols	K9-36
P.D. Thorne	K9-33
M.B. Triplett	K6-52
V.R. Vermeul	K6-96
D.L. Ward	K9-36
M.D. Williams	K6-96
S.K. Wurstner	K9-36
Z.F. Zhang	K9-36
Information Release (2)	P8-55

1 Washington Department of Ecology

D. Goswami H0-57

OFFSITE

1 Argonne National Laboratory

Argonne National Laboratory
Geosciences and Information Technology
9700 South Cass Avenue
Argonne, IL 60439
ATTN: D. Miller

1 USGS – Tacoma

934 Broadway, Suite 300
Tacoma, WA 98402
ATTN: R. Dinicola