MODFLOW/SEAWAT-2000WMD
Report – Version 4

Jorge Restrepo

Angela Montoya

Jorge Restrepo

Angela Montoya

September 2008

Prepared by the Department of Geosciences,
Florida Atlantic University.
For the
South Florida Water Management District,
Water Supply Department.

Blank page

Acknowledgements

Blank page

Table of Contents

List of Tables

Blank page

List of Figures

Blank page

MODIFICATIONS TO MODFLOW/SEAWAT-2000WMD SOURCE CODE

INTRODUCTION

The purpose of this research is to modify the SEAWAT-2000 (Langevin et al., 2003) software to make it compatible with the customized hydrologic packages used by the South Florida Water Management District (DISTRICT). The updated source code is called herein SEAWAT-2000WMD. The Resource Evaluation and Subregional Modeling Division(RESM) has been using various source codes of MODFLOW and SEAWAT for modeling projects of its own, but so far there is no single source code capable of covering all the required functionality entirely. Thus, the updated code will increase the current capabilities of the modeling software used by the RESM Division by allowing just one program to handle all the tasks that previously were performed by different source codes.

RESM applies models using both SEAWAT-2000 and MODFLOW-96 (Harbaugh and McDonald, 1996). A problem arises from the fact that these codes do not include many of the custom packages that RESM applies (e.g., Wetland, Diversion, and GEN packages). The Division would like to combine the hydrologic functionality of the custom packages with density-dependent software in order to properly maintain one source code version. Constant density models have been developed in MODFLOW-96 while the newer codes, such as SEAWAT-2000 (which is built around the more recent MODFLOW-2000 (Harbaugh et al., 2000)), offer both constant and variable density applications with solute transport capabilities.

Improving the source code will keep modeling applications current with the needs of the DISTRICT. The Diversion and Wetland packages will be generalized to closely represent the process for the physical system under south Florida conditions (e.g., time-dependent operational rules in the Diversion, solving the diffusion equation using the Wetland package for surface water in any cell presenting flooding conditions).

SEAWAT-2000 (Langevin et al., 2003) is a combined version of MODFLOW (Harbaugh et al., 2000) and MT3DMS (Zheng and Wang, 1999; Zheng, 2006) designed to simulate three-dimensional, variable-density ground-water flow and multi-species transport. The Variable-Density Flow (VDF) Process in SEAWAT was based on the constant-density Ground-Water Flow (GWF) Process of MODFLOW. The VDF Process uses the MODFLOW and MT3DMS methodologies to solve the variable-density ground-water flow and the solute transport equations, respectively. The MT3DMS part of SEAWAT is referred to as the Integrated MT3DMS Transport (IMT) Process.

Background material is available in the published user guides for SEAWAT, SEAWAT-2000, MODFLOW-2000, and MT3DMS.

The version of SEAWAT used herein (Langevin et al., 2007) is capable of simulating simultaneous solute and heat transport and the effect of fluid viscosity on resistance to ground-water flow. SEAWAT simulates heat transport using the intrinsic capabilities of MT3DMS. With this approach, the effect of temperature is calculated as one of the species being computed by MT3DMS. Other features have also been implemented by the USGS to this version of SEAWAT to improve overall functionality. The new main features are:

- An option to update the flow field if fluid densities have changed under a user-specified value. This option may substantially reduce runtimes for some simulations where a new flow solution is not needed for each transport time-step.

- Auxiliary variables can now be used with the Time-Variant Constant-Head (CHD) package to designate the reference density associated with the prescribed head. This option allows for more control of boundary heads specified with the CHD package (e.g., sea water).

The performance of the updated code program for SEAWAT-2000WMD has been tested in several applications including the Lower East Coast Subregional Model and Lower West Coast Floridan Aquifer Model. Future applications, however, might reveal errors that were not detected in the test simulations. Users are requested to notify FAU of any errors found in the computer program. For convenience, the AMG solver code (Wilson and Naff, 2004) is distributed with SEAWAT-2000. The AMG solver is, however, produced by Institute for Algorithms and Scientific Computing (SCAI), Sankt Augustin, Germany, and thus, though freely distributed, subject to restrictions as defined by SCAI.

Introduction Specific to MODFLOW-2000

MODFLOW is a computer program that simulates three-dimensional ground-water flow through a porous medium by using a finite-difference method (McDonald and Harbaugh, 1988). MODFLOW was designed to have a modular structure that facilitates two objectives: ease of understanding and ease of enhancing (Harbaugh et al., 2000). Input instructions for MODFLOW-2000 are divided into four major sections: Global Process, Ground-Water Flow Process, Array Utility Modules, and List Utility Module. The Global Process reads several files, and each file is described herein. The Ground-Water Flow Process reads at least one file for each package.

The Global Process (GLO) controls overall program operation and sets up data structures that can be used by all processes. Most of the initialization parts of MODFLOW have been put into the Global Process. The Global Process has been included in MODFLOW-2000 to control overall program flow, open files, and read global data such as space and time discretization. The horizontal grid spacing (DELR and

DELC), layer elevation (TOP and BOT variables in MODFLOW-96), and transient/steady-state data is included in the Global Process discretization file. Unlike other processes, the Global Process does not actually solve an equation.

To provide users with the option to retain only the output of greatest interest and limit the total amount of said output, MODFLOW-2000 can produce either one or two listing output files. When the user specifies that one output file is to be generated, all listed output from the model is written to that file. When the user specifies two output files are to be generated, the output that applies to the overall model run is written to a file called the GLOBAL file, but most of the output from Ground-Water Flow is written to a file called the LIST file.

The GLOBAL file contains information that applies to the model run as a whole. Output from the Define (DF), Allocate (AL), and Read and Prepare (RP) procedures is written to the GLOBAL file. The output includes the names and types of all files that are opened. Input for the Global Process is echoed to the GLOBAL file; this information includes a description of the spatial and temporal discretizations to be used.

The LIST file contains output from the procedures of the Ground-Water Flow (GWF) Process. This output includes allocation information; values used by the GWF Process for layers and individual cells in solving the flow equation, and calculated model results.

The GWF Process includes all aspects of solving the flow equation, including the formulation of the finite-difference equations, data input, the solving of resulting simultaneous equations, and data output. The GWF Process in current versions of MODFLOW-2000 contains ten previously existing packages and other packages added to MODFLOW-2000. The fifteen packages include:
- Ten packages documented in McDonald and Harbaugh (1988),

- Solvers PCG2 (Hill, 1990) and DE4 (Harbaugh, 1995),

- Horizontal Flow Barrier (HFB) package (Hsieh and Freckleton, 1993),

- Time-Variant Specified Head package (CHD; Leake and Prudic, 1991), and

- Layer Property Flow package (LPF).

The LPF package is an internal flow package that calculates the inter-block conductance coefficients and the parts of the flow equation that deal with ground-water storage. Since the BCF6 package performs the same function, LPF is therefore an alternative to the BCF6 package. That is, both packages should not be used simultaneously. The Block Center Flow package in MODFLOW-2000 has been modified; the package no longer reads any discretization data. Therefore, an existing MODFLOW-96 BCF file must be modified in order to work with MODFLOW-2000. Furthermore, the LPF package can be used to represent horizontal anisotropy that varies in magnitude within a model layer.

The BAS6 package data file in MODFLOW-2000 has been modified (when compared to MODFLOW-96) to remove parts incorporated into the Global Process discretization file.

Other input files for MODFLOW-96 will still work in MODFLOW-2000. All of the output is written to the LIST file. The head-dependent packages are:

- RIV—the riverbed conductance can be defined using parameters.

- DRN—the drain conductance can be defined using parameters.

- GHB—the boundary conductance can be defined using parameters.

- EVT—the maximum evapotranspiration flux can be defined using parameters.

Head-independent packages are:

- WEL—the well recharge rate (negative sign indicates discharge) can be defined using parameters.

- RCH—the recharge flux can be defined using parameters.

MODFLOW-2000 contains the Time-Variant Specified-Head (CHD) package (Leake and Prudic, 1991), and this has been modified to allow parameters to define the specified head. Input files for MODFLOW-96 will still work in MODFLOW-2000. All of the output is written to the LIST file.

The main four solvers are included in MODFLOW-2000 are documented:

- SIP—strongly implicit procedure (McDonald and Harbaugh, 1988),

- SOR—slice successive over-relaxation (McDonald and Harbaugh, 1988),

- PCG—pre-conditioned Conjugate Gradient (Hill, 1990), and

- DE4—direct solution based on alternating diagonal ordering (Harbaugh, 1995).

There is no fundamental change in the solvers for MODFLOW-2000. From the perspective of the solvers, the information that defines the simultaneous equations that must be solved at each cell is the same as that in earlier versions of MODFLOW.

Introduction Specific to MT3DMS

The MT3DMS packages (Zheng and Wang, 1999) that form the Integrated MT3DMS transport (IMT) process are the Basic Transport (BTN), Advection (ADV), Dispersion (DSP), Source/Sink Mixing (SSM), Reaction (RCT), and Generalized Conjugate Gradient (GCG) packages. Input for these packages are read from files types in the name file. Packages for the Integrated MT3DMS Transport Process, IMT, are identical to their standard formats for MT3DMS. The only requirement, which is also a requirement for the standard version of MT3DMS, is that the input data must correspond with the MODFLOW-2000 input data. For instance, the input file for the BTN package

requires the number of columns, rows, and layers. This same information is required in the DIS file and must be consistent with MT3DMS input data set, or the program will produce unexpected results. Users should refer to Zheng and Wang (1999) for standard input instructions and recommendations for the MT3DMS packages.

Introduction Specific to SEAWAT-2000

The input instructions for the Variable-Density Flow (VDF) are associated with a specific input file required by the VDF Process. The VDF Process is activated by including VDF as a file type in the name file; otherwise, the simulation uses the GWF Process independently of density.

The VDF Process is coupled with the IMT Process if the value for MTDNCONC is set to any value greater than zero. The variable MTDNCONC is defined as the MT3DMS species number that will be used in the equation of state to compute fluid density. For this coupled simulation, the MODFLOW-2000 flow time-steps are further divided into transport time-steps. Input parameters for controlling the lengths of the transport time-steps are included in the BTN and ADV packages. The flow and transport equations can be iteratively coupled only if the finite-difference solver is used (MIXELM set to zero in the ADV package) and the GCG package is active. The variable MIXELM is defined as an integer flag for the advection solution option (e.g., if the flag equals zero, the standard finite-difference method with upstream- or central-in-space weighting is used).

Although the IMT Process was added primarily to work with the VDF Process, this process also has been designed to work with the GWF Process (Langevin et al., 2003). This option provides users with the ability to perform constant-density (and variable-density) flow and transport simulations with a single program. The flow equation is solved for every transport time-step rather than for every flow time-step. This means that constant-density simulations of flow and transport with SEAWAT-2000 may take longer than simulations with standard versions of MODFLOW-2000 and MT3DMS. The results, however, should be quite similar.

Running MODFLOW-2000 and SEAWAT-2000

When MODFLOW/SEAWAT-2000 is invoked, it reads the names of name files from modflow.bf file. Each line of the modflow.bf file contains the name of a single name file, and each name can be up to 80 characters long. Before starting a simulation, the name of the name file is written to the report file. If the name file exists, the simulation is executed; when done, another message is written to the report file. If a name file does not exist, a message is written to the report file and processing continues with the next name file in modflow.bf. When batch mode is used to run MODFLOW-2000, two additional file units are used—97 and 98. Running the program with a batch file, or in batch mode, can usually save time. Units 97 to 103 are not recommended to be used in any of the SEAWAT simulations.

Note: It is recommended that any unformatted input files be saved in the comma-separated values format (with the file extension CSV) and formatted input files should be saved in the field order values format if the records are not too long(with the file extension TXT).

SEAWAT-2000WMD uses the data declaration and sharing using a FORTRAN module. Most data in MODFLOW-2000 is to subroutines as arguments, making clear which data are used by the subroutines. Shared data in modules are passed to subroutines using the Fortran USE statement. The "ONLY" specifier to USE allows the specific variables being used to be identified in each subroutine. FORTRAN modules eliminate the need for most subroutine arguments. This modification was done to make more portable and clear source code for MULTIBUD (Rodberg, K., 1999). This make possible to use pointers as simple variables. A few steps are needed to USE a variable: declaring the pointers in the module, then a non-array pointer must be allocated much as the array pointer is allocated and finally release the memory when the information is not needed any more. Allocating a scalar pointer causes memory to be created for that scalar. FORTRAN has a function for releasing allocated memory, which is called deallocation. Memory is used throughout a simulation in SEAWAT, so deallocation is not needed until the end. However, explicit memory deallocation generally is not required in SEAWAT.

SCOPE OF WORK FOR THE SOURCE MIGRATION FROM MODFLOW-96 TO SEAWAT-2000WMD

This scope of work pertains to the migration of the current software packages in MODFLOW-96 into SEAWAT-2000. FAU moved the packages to SEAWAT with some improvements to the code in some cases. The following packages shall be migrated to SEAWAT-2000WMD:

- UGEN – Utility Generation package,

- WTL – Wetland package,

- SIP - Strongly Implicit Procedure enhancements,

- RDF – Reinjection Drainflow package,

- TRG – Trigger package with multi well types,

- WELT – Multiple Wells package,

- MBUD – Zone Water Budget utility,

- MDIV – Multi-Op Diversion package and,

FAU conducted the source code changes in a Windows computer environment. Three examples were used to test, and verify the modified SEAWAT code. FAU together with the DISTRICT carried out a careful quality analysis and control of the software in the two cases. FAU is providing the updated version of the SEAWAT-2000 source code to the DISTRICT.

IMPLEMENTATION OF THE SOURCE MIGRATION

Introduction

Modifications to the SEAWAT-2000 code, for the Utility Generation, Wetland, Strongly Implicit Procedure Enhancements, Diversion (including the Multi-Op Diversion), Trigger (including Generated Trigger), Zone Water Budget Utility (MULTIBUD), Multiple Wells packages is specially documented in the source code using:

    C**FAU --

It is used in the code at the beginning and the end of any change or addition to SEAWAT-2000WMD. For example, a portion of the code is depicted in Figure 1. The packages are also documented in narrative form in MS Word in the MODFLOW/SEAWAT-2000 USER MANUAL and in this report with examples whenever they are needed.

General Procedure for Memory Allocation

A few steps are needed to declare a variable array. The array must be declared using the type of variable and the ALLOCATABLE FORTRAN command as follows:

    C--SEAWAT: SEAWAT2000 DECLARATION
    C**FAU --
        REAL UX
        CHARACTER*21 xch16
        CHARACTER*3 xch3
        ALLOCATABLE UX(:), xch16(:), xch3(:)

Then, after using the allocation subroutine for each new package and defining the size of the array needed, the main SEAWAT-2000WMD program will actually allocate the memory for the variable as demonstrated below:

    C------DYNAMICALLY ALLOCATE ux,xch16,xch3
        LENUX = ISUMUX - 1
        IF(LENUX.LT.1) LENUX=1
        lenxch16 = isumch16 - 1
        IF(lenxch16.LT.1) lenxch16=1
        lenxch3 = isumch3 - 1
        IF(lenxch3.LT.1) lenxch3=1
        ALLOCATE (UX(LENUX),xch16(lenxch16),xch3(lenxch3))
    Finally release the memory when the information is not needed any more:
    C**FAU --
    C**FAU: DEALLOCATE FAU ARRAYS

> IF(IUNIT(85).GT.0)DEALLOCATE (UX,xch16,xch3)
> C**FAU ---

INTRODUCTION OF THE MIGRATED PACKAGES TO THE SEAWAT 2000WMD

The following packages were migrated to SEAWAT-2000 by the Center for Hydrology and Water Resources at Florida Atlantic University. These are existing packages that the DISTRICT applies, which are in the DISTRICT MODFLOW-96 source code. The current functionality of the packages is kept as outlined below.

Utility Generation (UGEN) Package

The Utility Generation package (UGEN) was developed by the Center for Hydrological and Water Resources at Florida Atlantic University, in cooperation with the District, as a tool to generate time-related MODFLOW input during model execution based on time-dependent quantities such as flows and stages (Restrepo et al., 2003). The implementation of UGEN increases the efficiency of the packages used with it by generally reading the static information only once. Traditional MODFLOW input files must be built prior to execution for each package and for the entire simulation period. UGEN creates the input files "on the fly" by linking the static input parameters, such as physical location, with the dynamic temporal data, much like a relational database (Restrepo et al., 2003). The static and dynamic data are linked by "location name" identifiers. The UGEN package can be used to generate several MODFLOW-2000 packages: Well (WEL), River (RIV), Drain (DRN), General Head Boundary (GHB), Re-injection Drain Flow (RDF), Time-Variant Specified-Head (CHD), and Diversion (DIV). It can also be used to correct heads due to the presence of a saltwater interface and to calculate the hydraulic conductance for River, General Head Boundary, and Drain packages using the properties of the layers and grid system.

Generating the input with UGEN increases the functionality of MODFLOW by reducing execution time and storage requirements. UGEN is compatible with the modular structure and programming language (FORTRAN) of MODFLOW. The placement of the static and time-series data into separate files also reduces model setup time.

```fortran
C       DEFINE (DF) PROCEDURE
        CALL GLO1BAS6DF(INUNIT,IUNIT,CUNIT,IREWND,NIUNIT,IOUTG,IOUT,
     &                  VERSION,NCOL,NROW,NLAY,NPER,ITMUNI,ISUMGX,
     &                  MXPER,ISUMIG,ISUMGZ,INBAS,LENUNI,ISUMX,ISUMZ,
     &                  ISUMIX,LAYHDT,24,IFREFM,INAMLOC,IPRTIM,IBDT,
     &                  SHOWPROG,NOTICECOUNT)

C**FAU -----------------------------------------------------------------
C**FAU neww echos and time subroutines for control purposes------------
        WRITE(iout,490)' Using NAME file: ',FNAME
        WRITE(IOUTG,490)' Using NAME file: ',FNAME
        call utl2ctime(0,0,mdd,msec_diff,msec_new,msec_old,
     1                             mlsec,t_acc_run,0,iout)
        call utl2ctime(0,0,mdd,msec_diff,msec_new,msec_old,
     1                             mlsec,t_acc_run,0,IOUTG)

C**FAU -----------------------------------------------------------------
        IF(IUNIT(12).GT.0)THEN
          INOC=IUNIT(12)
          CALL URDCOM(INOC,IOUT,LINE)
          LLOC=1
          CALL URWORD(LINE,LLOC,ISTART,ISTOP,2,IHHH,R,IOUT,INOC)
          CALL URWORD(LINE,LLOC,ISTART,ISTOP,2,IDDD,R,IOUT,INOC)
          CALL URWORD(LINE,LLOC,ISTART,ISTOP,2,IHEE,R,IOUT,INOC)
          CALL URWORD(LINE,LLOC,ISTART,ISTOP,2,IDDD,R,IOUT,INOC)
          CALL URWORD(LINE,LLOC,ISTART,ISTOP,2,NUMBUD,R,IOUT,INOC)
          REWIND INOC
        ENDIF
C**FAU -----------------------------------------------------------------
C**FAU:Module is implemented to convert pointers.
C**FAU:IF(IUNIT(84)>0 package is used, if IUNIT(84)==0 default is used.
        CALL  GWF1FLG1AR(IUNIT(84),CUNIT,IUNIT,NIUNIT,NCOL,NROW,NLAY,
     &                  IFLG_TYP,NUMBUD,IOUT)
C       ****************************************************************
C       Allocate and Read for GWF FLAG Package
C       ****************************************************************

C**FAU -----------------------------------------------------------------
C**FAU: MULTIBUD ADDED END
        CALL READCFG(IUNIT(97),NROW,NCOL,NCOL,NROW,NLAY,NUMBUD)
C**FAU -----------------------------------------------------------------
C
        CALL OBS1BAS6DF(IOBS,IOSTAR,IOWTQ,IOWTQDR,IOWTQGB,
     &                  IOWTQRV,IOWTQST,IQ1,IUNIT(27),JT,LCCOFF,LCHFB,
     &                  LCIPLO,LCIPLP,LCIQOB,LCNDER,LCNQOB,LCOBADV,
     &                  LCOBDRN,LCOBGHB,LCOBBAS,LCOBRIV,LCOBSE,LCOBSTR,
```

Figure 1 Example of a portion of the source code is illustrated.

UGEN General Files

When MODFLOW is run with UGEN, three input files are needed. The first, the UGEN file, defines the number of stations for stages/flows and any flags (i.e., flags that activate UGEN for any MODFLOW package with dynamic input data). The second type of input file needed is the observation file. UGEN observation files contain stage/flow values, respectively, for each stress period, for all stations. Stress periods are essentially represented by rows, with the stations being listed across each row as distinct columns. The UGEN includes an option to "skip" records in the observation file time series, i.e. if the current time series has daily records and the user wants to run a weekly simulation there is no need to change the original time series. There is also a flag in the UGEN that tells the program to skip a given amount of records so the program will run on a weekly basis. Nevertheless, the user has to be aware of matching the new number of stress periods and stress period-lengths in the DIS and BTN files. Stage data is not averaged just skipped and the last value is used. Flow data is summed and averaged, taking into account the skipped values.

The GEN allows the user to have flexibility with the data set up. However, if you want to use the CHD package, the CHD input requires the heads at the beginning and end of the stress period. In the new modification of the CHD, the program, depicted in Figure 2, uses the head at the end of the previous stress period as the initial head for the current stress period and the user must provide the value at the end of the stress period in the GEN package stage file. As shown in the CHD source code, the head is computed by linear interpolation based on the proportion of the total stress period length compared to the time step length. However, this is a minor issue because most of the models at the DISTRICT use daily or weekly stress periods. For the River, Drain and General Head packages, this assumption (i.e., using the head at the end of the stress period) could be an issue and could be solved by creating an additional array to keep the initial value and interpolate as in the CHD package for each stress period. Alternatively, the user still has the option in GEN to not use the skip variable and provide the average value; however, in this case, it would not be reasonable to use the CHD with the average option. For calibration, you must use the head at the end of the stress period because it is integrating the value to the point at the end of the time step or stress period. Therefore, it is ideal to provide the value at the end of the stress period to coincide with the head solution. Additionally, if there is a high value between the stress periods, the model should not make the decision to compute the average. The user must do the QA/QC and make that decision. An solution may be to reduce the stress period length for that period of the simulation.

The third type of UGEN input file is the modified MODFLOW input package file to be used with UGEN (e.g. RIV, DRN, WEL, GHB, DIV, RDF, CHD, or HDXY). This input file will provide the same information that MODFLOW requires. However, instead of having stage/flow values, it will have the location name identifier—said location (station) name must be provided with a length of no more than 21 characters, defined by:

CHARACTER*21 xch16.

This represents the "static" portion of the UGEN input files that the utility links with the "dynamic" information provided in the observation files. These modified MODFLOW input files may be kept constant during all simulations, or may be changed if necessary.

All procedures are executed by independent subroutines called by the main SEAWAT-2000 program. Creating independent subroutines preserves the modular structure that is characteristic of MODFLOW. If UGEN is not activated, MODFLOW is equally capable of running the input files as constructed in the original MODFLOW form.

```
C
C6-------INITIALIZE HNEW TO 0 AT SPECIFIED-HEAD CELLS.
      DO 50 L=1,NCHDS
      IL=CHDS(1,L)
      IR=CHDS(2,L)
      IC=CHDS(3,L)
      HNEW(IC,IR,IL)=DZERO
   50 CONTINUE
C
C2-------COMPUTE PROPORTION OF STRESS PERIOD TO CENTER OF THIS TIME STEP
      IF (PERLEN.EQ.0.0) THEN
        FRAC=1.0
      ELSE
        FRAC=PERTIM/PERLEN
      ENDIF
C
C2-------PROCESS EACH ENTRY IN THE SPECIFIED-HEAD CELL LIST (CHDS)
      DO 100 L=1,NCHDS
C
C3-------GET COLUMN, ROW AND LAYER OF CELL CONTAINING BOUNDARY
      IL=CHDS(1,L)
      IR=CHDS(2,L)
      IC=CHDS(3,L)
C
      IF (PERLEN.EQ.0.0 .AND. CHDS(4,L).NE.CHDS(5,L)) THEN
        WRITE(IOUT,200)IL,IR,IC
  200   FORMAT(/,' ***WARNING***  FOR CHD CELL (',I3,',',I5,',',I5,
     &')  START HEAD AND END HEAD DIFFER',/,
     &' FOR A STRESS PERIOD OF ZERO LENGTH --',/,
     &' USING ENDING HEAD AS CONSTANT HEAD',
     &' (GWF1CHD6AD)',/)
      ENDIF
C5-------COMPUTE HEAD AT CELL BY LINEAR INTERPOLATION.
      HB=CHDS(4,L)+(CHDS(5,L)-CHDS(4,L))*FRAC
C
C6-------UPDATE THE APPROPRIATE HNEW VALUE
      HNEW(IC,IR,IL)=HNEW(IC,IR,IL)+HB
      HOLD(IC,IR,IL)=HNEW(IC,IR,IL)
  100 CONTINUE
C
```

Figure 2 CHD, the program showing the interpolation procedure.

SEAWAT package input files using the UGEN

Input files contain one record for each cell/reach. The record defines cell location, station name, and associated parameters. Unlike the original version of MODFLOW, stage/flow values are not specified within those files. The station name points to the stage/flow values for every stress period in the observation file.

The modified MODFLOW package files include stress period-related MODFLOW input files, such as RIV, DRN, WEL, GHB, MDIV, CHD, and RDF. These files contain the same information that MODFLOW requires. However, instead of having an input field for stage or flow values, the associated location (station) name must be provided with a length of no more than 21 characters. The MODFLOW input packages may be constant during all simulations or may be updated if the static information changed.

During a MODFLOW simulation, UGEN begins each stress period by reading observation values (i.e., stages and/or flows). The stage/flow station name in the MODFLOW package input file using the UGEN is used to replace (in memory) the numerical value associated with the station name. The program "points out" the same station name in the observation file and assigns the observation value to each station name. Then, MODFLOW finds the solution for the stress period. Then, a new stress period is started. The example application is in MODFLOW/SEAWAT-2000WMD USER MANUAL.

Wetland (WTL) Package

The Wetland (WTL) package was developed by the DISTRICT and the Center for Hydrology and Water Resources at Florida Atlantic University (FAU) (Restrepo et al., 1998). The current version has been revised and improved by FAU in association with DISTRICT as the package has been applied to subregional models in south Florida. The Wetland package was incorporated into SEAWAT and enables the top layer of the grid system to contain overland or groundwater flow. This package can account for vegetation characteristics, simulation of sheet flow, sloughs, levees, and barriers, and evapotranspiration.

This package is suitable for modeling the wetting and drying of the wetlands by combining vegetation and soils as part of model layer 1. This approach allows the flow equations to remain valid when the water surface falls below the soil surface (i.e., Darcian flow). When the model is applied to a water ponding case, the effect of the value of soil transmissivity is insignificant relative to the corresponding value for the surface water, which is governed by the Kadlec equation. However, the soil layer plays an important role in the vertical direction, as it is the case with the vertical hydraulic conductivity.

The Wetland package simulates preferential flow paths and vertical impermeable barriers to both horizontal overland and groundwater flow. However, barriers such as levees and sloughs were not used in the Wetland package for this model. These features will either increase or decrease the flow rate. Levees and sloughs are indirectly defined as a hydraulic characteristic by using the cell faces for the inter-block transmissivity to designate preferential flow orientations. In the case of fully or partially penetrating barriers, the package can completely restrict flow through the cell face or allow seepage through the face. The establishment of barriers to flow requires the user to specify which cell faces in the modeled feature are perpendicular to the cell flow in order to block the flow. Specifying preferential flow paths requires the user to select the cell faces in the modeled feature that will follow the primary spatial extent of the slough to allow water to move freely down a path. This methodology works best when the principal direction corresponds to the grid orientation.

The main changes carried out to the Wetland package are the use of the TOPO surface of layer 1 in the Discretization package. There is no need to read the TOPO

surface (i.e., ZBOTT) in the Wetland package. The code was cleaned removing variables that were not used through the code or input. One error in the interblock conductivity from previous MODFLOW-96 version was corrected in the code. In this new version, the Wetland package only overwrites the interblock transmissivity within the wetland areas. The values outside of the wetland areas use the information coming from the Block-Centered Flow package.

Strongly Implicit Procedure (SIP) Package Enhancements

Two alternative enhancements were developed by the DISTRICT in 1998 for the Strongly Implicit Procedure (SIP) package in order to improve or maintain model stability. Both alternatives have added optional variables. If the variables are not used, the SIP package will function normally.

Three optional variables have been added to the SIP package by the DISTRICT. These are MNITER on the first data line and HCLOSEMAX, and NOSTOP on the second data line. MNITER specifies the minimum forced number of iterations. The closure criteria, HCLOSE and HCLOSEMAX serve as a lower bound and an upper bound, respectively. If you choose not to use the aforementioned options there is no need to change the SIP input file.

If HCLOSEMAX is specified, SIP will use this input to determine whether or not to stop a simulation when the maximum number of iterations has been reached. If SIP reaches the maximum number of iterations and the maximum head change in a cell is less than HCLOSEMAX, SIP will move on to the next time step instead of stopping the simulation. A new implemented variable MNITER allows the model to continue iterating if the HCLOSEMAX criterion has been violated. This option provides a means of implementing a tight closure criterion for 99 percent of the simulation while allowing it to be relaxed for a couple of problem days.

When NOSTOP is included and set equal to one, the program will not terminate if the HCLOSEMAX criterion has been violated. Instead, the heads in the cells, which violated HCLOSEMAX will be reset to their values at the end of the last previous time, step and a warning message will be written to the output file. NOSTOP is particularly useful when working to improve a model having stability problems. If this option is not used, there is no need to change the SIP input file. However, if this enhancement is used, the input data file should be changed.

There was an issue in the SIP package from the DISTRICT's MODFLOW-96 source code when the model reaches 100 iterations and the error is greater than HCLOSE (e.g. 0.05) and that error is greater than HCLOSEMAX (e.g. 0.5) (see below for the input data and source file). In this case, the head for that cell is re-set equal to STRTN, the initial conditions for the whole simulation. Then the program accepts the solution for that stress period. One should mean to re-set the head for that cell equal to initial conditions for that stress period. For example, If one starts the simulation on January where the

heads are low, and there is a conversion problem for the solution when the heads are high , then, one would made a wrong assumption to initialize high heads with low heads. The new SIP version for SEAWAT is corrected to initialize the head with the initial conditions for the specific stress period.

Reinjection Drainflow (RDF) Package

The Reinjection Drainflow (RDF) package was created in 1999 for the DISTRICT by McDonald Morrissey Associates, Inc. to simulate a drain from which flow could be directed into another cell rather than being permanently removed from the groundwater flow system, as is the case in the original Drain package (Jones, 1999). The reinjection option is invoked by adding the destination (layer, row, and column) of the cell to which the drain flow is directed. The user specifies both a source destination and a sink destination. The destination in the RDF is similar to the source and sink concept from the Diversion package (Restrepo et al., 1998); however, the source and sink destinations in the RDF have a 1:1 relationship while the Diversion package source and sink sets have a many to many relationship. If no destination cell is specified or if the destination cell is inactive, the RDF package functions exactly like the Drain package. The reinjection rate is calculated as the product of the drain conductance and the elevation of the head in the cell about the elevation of the drain.

Enhancements for the RDF package were made by FAU (Restrepo, 2003) to incorporate a second level for the sink destination cells. The purpose of this second level is to restrict water moving into the sink cells by setting a maximum level. If the heads in the sink cells are below the maximum level, the discharge will be allowed only if the source cells are above the minimum. The benefit of using this approach is that the user can vary the operations on a daily basis, as in an operational schedule, by varying the control levels. This package has the ability to change the schedule each stress period. Therefore, if the schedule changes halfway through the calibration period, the changes can be implemented. The stages for the RDF package are provided in the UGEN package.

Multi-Operation Diversion (MDIV) Package

The Diversion (DIV) package (Restrepo et al., 1998) was developed by the DISTRICT and Florida Atlantic University (FAU) as a tool to simulate the effects of man-made structures such as pumping stations or drains on groundwater and surface water levels. The current version has been revised and improved by FAU in association with the DISTRICT as the package has been applied to sub-regional models in south Florida.

The Diversion package routes water according to operational rules or physical constraints of the system by conserving mass balance (i.e., keeping the water in the system). The water is supplied by a group of source cells, and is distributed to a group of sink cells associated with each structure. Sources and sinks can be external or internal cells or areas outside of the model domain. External cells are located outside of the active

model domain; internal cells are always located inside the active model area. Sources may be located outside the active domain (e.g., external surface storage, which makes deliveries within the active area). Sinks may be located inside the active domain (e.g., a swale that receives water from a source).

The flow rate values are supplied by the user for each stress period and represent the movement of water from source to sink (if positive flow) or from sink to source (if negative flow). The on/off source and sink levels specified in the input file function as operational rules for the structure. In the case of positive flow, the term on level source refers to the lowest water level limit in a source at which water can be released to the sink. The term off level sink is the highest water level limit in the sink cells at which they can receive water from the source. The target for the hydraulic head at source and sink cells are respectively, the on level source limits and the off level sink limit. Additional targets include off level source that is the highest water level at which source cells at the structure can be released to the sink, and on level sink that is the lowest water level limit in the sink cells at which they can receive water from the source. In the case of negative flow, the corresponding targets applied as before but the movement of water is from sink to source.

The first task completed was the migration of the DISTRICT MODFLOW-96 DIV package to SEAWAT-2000. Some of the variables that were defined in the input data set but were not used in the code were removed from the code. An explanation of applicable variables used in the current input file is given in this report.

The second task completed was to apply UGEN to the current DIV package to be able to change operational rules in time, since the package was limited to fixed operational rules over time. This functionality is extremely important when modeling south Florida hydrology as operational schedules change over time. The package now has the option to have the rules fixed in time (existing) or input into the UGEN stage observation file, which allows rules to vary in time according to the stress period (new option). The UGEN storage requirements will still be small with the daily variation; however, in the future, it would be valuable to have a third option to implement rule curves and schedules.

The new changes in the Diversion package allow the user to obtain, for every source and sink, the time series of the target stage level during the entire simulation. The input format remains almost identical to that used for the LECsR model via MODFLOW-96. The user now can read the 'target stage' of each source and sink for every stress period from the UGEN stage file. The link is done by the UGEN package through a name specified in the MDIV package.

The deviations from the target stage (off level sink and off level sink) can be read in the same place where the actual stages are currently located. In case that only the actual and constant stage values for off level sink and off level sink are desired for a particular sink or source, its target stage may be read from the main input file (e.g. MDIV

file). In this case, the user may read zero values for the 'target stage' of each source and sink for every stress period from the UGEN stage file (e.g., station ZERO).

Figure 3 shows the time series for the target stage level for a sink cell. In this example, the value of off level sink (e.g., lv_sk_of) is 1 ft., while the value of lv_sk_on is -1 ft. The "target stage" specified in the UGEN stage file is specified through a name specified in the MDIV package (e.g., stress period 4, target stage=4.5 ft). Note that these variables change accordingly with the target stage level throughout the simulation. The package is used in a wider range of situations such as modeling reservoirs and impoundments, spreader swales and flow ways. Current enhancements allow routing water as a series of "cascading" sub-basins and representation of several rules at the same location. Flows can be routed from upstream to downstream basins. The example application is in MODFLOW/SEAWAT-2000WMD USER MANUAL.
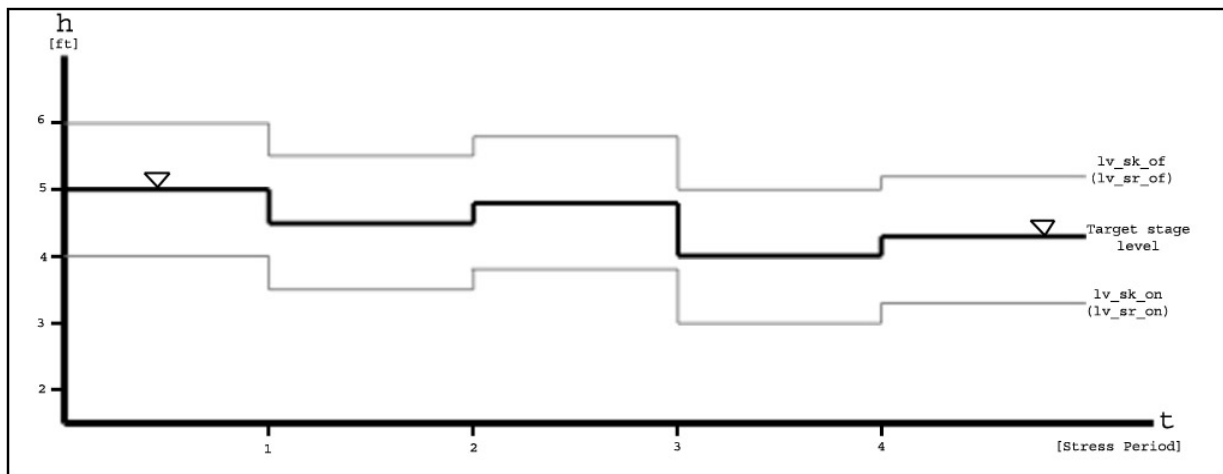

Figure 3. Time series for the target stage level for a sink cell.

Trigger Package (TRG) with Multi Well Types

The Trigger (TRG) package was developed in 1992 by the DISTRICT and Water Resources Management, Inc. in order to initiate cutbacks in well pumpage in response to head declines in specified "trigger cells". It was originally created for use in the South Florida Water Management Model and was modified later in 1999 to work with MODFLOW-96 (Randall, 1992).

The Trigger package employs two different methods to reduce pumpage: the cell mechanism and the time series mechanism. The cell mechanism triggers cutbacks when water levels fall below user-specified limits at selected cells, while the time series mechanism triggers cutbacks at user-specified time periods, regardless of water levels. Trigger cells are located in areas of concern, and are associated with larger, rectangular zones, which are user-specified rectangles, within the model. In each zone, there are trigger cells. If the head (or drawdown) in a trigger cell at the end of a stress period is

lower than a standard, which is user-specified, the program looks for each well in the zone and reduces the pumpage for the next stress period by a fraction specified in the input data. A combination of both the cell and the time series mechanisms determines which zones will be subjected to cutbacks.

In the cell mechanism, a violation is signaled when the head in a trigger cell falls below the specified trigger value. Cutbacks in pumpage are calculated and applied in phases based upon the severity of the violation. The user may require that the violation must occur for a certain number of consecutive stress periods (called the "delay period") before a cutback is applied in order to avoid overreacting to short-term dips in head.

The time series mechanism applies cutbacks according to specified time periods such as the dry season, or periods when low Lake Okeechobee levels are planned. These cutbacks remain in effect until the end of the dry season, or until a more severe cutback is triggered. Violations occur in phases, with Phase 1 being the least severe by convention. Dry season cutbacks without associated low trigger cell levels are always Phase 1 level cutbacks. If there are trigger cells in a zone having different levels of violation, the cell with the most severe violation determines the cutback phase for that zone.

Pumping reductions are applied to different wells according to their usage type. Some examples of usage types are urban, agricultural, golf, and "zero" type, which means that the well is exempt from cutbacks. When pumpage reductions are triggered, the program outputs two files: one showing where the triggers were set and what phase was assigned, and another showing, which wells, will have reduced pumpage, and by what amount. The cut back amount is determined in the water use permit.

In 2005, source code modifications to the Trigger package were requested by the Water Use Division at the DISTRICT to better reflect existing DISTRICT water shortage policies during drought periods. It was necessary to modify the source code by adding a constraint to the logic of the cell mechanism.

Due to the large number of consumptive use wells, the Trigger and Well packages' input can be large and difficult to do quality assurance and quality control. The Trigger package is standardized in such a way that the Utility Generation (UGEN) package can be used for the different well types that are currently available. The UGEN package is one that helps conserve computer resources, such as processing power and storage.

Currently, the UGEN package can be used with the WEL package, but there is no capability to use the Trigger package with the WEL and UGEN package combination. The logic of the Trigger package cell mechanism will not change; however, the Trigger well type (e.g., PWS, Golf, Nursery, etc.) needs to be linked to the WEL package static information (i.e., layer, row, and column) in the UGEN package. This link will allow the program to handle the input on the fly. Having the WEL pumping information in the UGEN flow observation file is very important for quality assurance and quality control of

the information. It is more straightforward to analyze a timeseries of pumping rather than pumping data in the SEAWAT format.

Multiple Budget (MBUD) Package

MULTIBUD is a post-processing utility program developed by the DISTRICT (Rodberg, K. ,1999) for computing subregional water budgets for MODFLOW. The user designates the subregions by specifying cells in the region. A separate budget is computed for each region. The budget for a region includes a component of flow between each adjacent region. MULTIBUD reads the SEAWAT-2000WMD output of flows, which contains components for each budget term per cell (e.g., ET and recharge flow rates). MULTIBUD then accumulates each flow component for all the cells in each specified zone.

MULTIBUD was recently modified by the DISTRICT and Ecology & Environment, Inc (2005) with the intention of generating water budgets in a more precise time step. As in the other packages, a character unit –in this case, "BUD"- is added to the namefile to activate the package.

MULTIBUD (BUD) was migrated into the SEAWAT-2000WMD source code in order to have the same budget functionality as in the MODFLOW-96 source code. This functionality is extremely important when analyzing water budgets.

As explained before, SEAWAT-2000WMD uses the data declaration and sharing using a FORTRAN module. Shared data in modules are passed to subroutines using the Fortran USE statement. FORTRAN modules eliminate the need to change the subroutine arguments in this package. Furthermore, this modification was done to make more portable and clear source code for MULTIBUD. This make possible to use pointers as simple variables. In this case makes the program a bit faster in the calculation procedure.

Input for the Multiple Budget package is specified in the configuration namefile. The configuration file was designed to recognize input based on the tag located on the first ten (10) characters of a line. This tag consists of a keyword in uppercase. It is important to include these keywords in the input file to ensure the proper function of the package. If a keyword consists of less than ten characters, the remaining characters must be blank spaces. The example application is in MODFLOW/SEAWAT-2000WMD USER MANUAL.

Interpolation Head Boundary (HBXY) Package

The Interpolation Head Boundary package (HBXY) was developed by the Center for Hydrology and Water Resources at Florida Atlantic University as a tool to interpolate time-related SEAWAT-2000WMD input during model execution. Although this package has not been fully implemented yet, one of its components is currently used in the Lower West Coast Floridan Model. This component is only provided by FAU in an executable

form. A full implementation of this package is only available upon contract establishment. HBXY interpolates the heads in the input files "on the fly" by linking the static input parameters, the physical location (X, Y), with the dynamic temporal data, much like a relational database. The static and dynamic data are linked by "location name" identifiers.

The HBXY package can be used to generate the model initial conditions and to generate the piezometric heads for the following MODFLOW-2000 packages: General Head Boundary and Constant Head Boundary. It can also be used to calculate the vertical hydraulic conductance for the General Head Boundary package. The HBXY package must be used with the UGEN package.

Generating the input with HBXY increases the functionality of MODFLOW/SEAWAT by reducing the pre-processing. HBXY is compatible with the modular structure and FORTRAN language in MODFLOW-2000. The HBXY also includes an option to "skip" records in the UGEN time series, i.e. if the current time series has daily records and the user wants to run a weekly simulation there is no need to change the original time series. There is a flag in the HBXY that tells the program to skip a given amount of records so the program will run on a weekly basis. Nevertheless, the user has to be aware of matching the new number of stress periods and stress period-lengths in the DIS and BTN files. This option is quite handy for calibration purposes, wherein the model must run many times; therefore, running time is an issue to take into account. This user guide describes the required input data files.

All procedures are executed by independent subroutines called by the MODFLOW main program. Creating independent subroutines preserves the modular structure that is characteristic of MODFLOW. If HBXY is not activated, MODFLOW is equally capable of running the input files constructed in the original form.

SEAWAT-2000WMD MODEL VERIFICATION

During the update of SEAWAT-2000WMD, a simple problem was designed to test each additional package annexed to the code. In addition to testing SEAWAT-2000WMD under this problem, the code was also verified using the SEAWAT Lower West Coast Floridan Aquifer System (LWCFAS), model, and MODFLOW-96 Lower East Coast Subregional model (LECsR). At the same time, testing SEAWAT-2000 with MODFLOW-96 was accomplished by converting the input data sets for (LECsR) into the MODFLOW-2000 format.

Simulation results for SEAWAT-2000WMD are near identical to the results produced for the previous version of SEAWAT. The final mass-balance results for each run of the LWCFAS are presented herein. Those results are presented in Appendix A.

Simulation results for SEAWAT-2000WMD are identical to the results produced for the current DISTRICT's MODFLOW-96. The final output with mass-balance results for each run of the LWCFAS and LECsR are presented herein in Appendices A and B.

UTILITY WELL CREATION PACKAGE IN MODFLOW (PRE-WEL/WELT)

Input for the Well Creation package (pre-WEL/WELT) is documented in this section. Free format is used for reading all values and the order of specification of structure name does not affect results. This utility creates both the regular input for the well package in SEAWAT-2000WMD, the well package in MODFLOW using the GEN, and the multiple-well package for SEAWAT. Both input types should not be used simultaneously.

The pre-WEL/WELT utility uses several files associated with permit data allocation such as yearly well-field pumpage, monthly pumpage value distributions year-by-year, and static information associated with well location, well type and assigned percentage for that permit. Files used in the pre-processor utility are described in Table 1.

Table 1 Files used in the pre-processor utility

VARIABLE	Files/Input	Description
NTYEASRS	16	Number of years
well_info	'lecsrmf_8601.csv'	Model Information
outfile3(1)	'outfile3.1986'	Permit allocation 1st year
outfile3(2)	'outfile3.1987'	Permit allocation 2nd year
outfile3(3)	'outfile3.1988'	..
outfile3(4)	'outfile3.1989'	..
outfile3(5)	'outfile3.1990'	..
outfile3(6)	'outfile3.1991'	..
outfile3(7)	'outfile3.1991'	..
outfile3(8)	'outfile3.1991'	..
outfile3(9)	'outfile3.1991'	..
outfile3(10)	'outfile3.1995'	..
outfile3(11)	'outfile3.1995'	..
outfile3(12)	'outfile3.1997'	..
outfile3(13)	'outfile3.1998'	..
outfile3(14)	'outfile3.1999'	..
outfile3(15)	'outfile3.2000'	..
outfile3(16)	'outfile3.2000'	Permit allocation last year

Model Information

The "well_info" file contains the static information associated with well cell-location, well type, permit name and percentage allocated for that permit. The percentage

value is specified to each year for all their respective well names. The first field of each record contains the permit name, well name, well name id, use type, layer, row, and col. The following fields contain the percentage values of each year for all wells during the whole simulation. For a permit, you can have several types of wells.

The field "well name id" is the name or reference in the UGEN package assigned to a particular location (permit). The name id must not be longer than 21 characters. This variable is associated with a set of data in the UGEN observation file(s).

The model information could be constant during all simulation or may be updated if the year allocation for specific wells with an associated permit is changed. The example shown in Table 8 illustrates the model information for four permits in an arbitrary simulation.

Permit allocation

The year-by-year associated files are the permit allocation files. They contain annual pumpage values and the weighting factor for each month, for all their respective permit names. The first record of each file contains the permit name. The following records contain the annual pumpage values in MGY followed by the assigned monthly factor. The sum of monthly factors should always add up to one. The example shown in Table 9 illustrates the permit allocation file for 1986 for some arbitrary permits.

The field "permit name" is the name of a permit that is being used in that particular year. The permit name in the "well info" file is used to "point out" pumpage values for each year from the permit allocation file. If the permit name in the "well info" file is not specified in the allocation file, a value of zero is immediately assigned to that well.

Blank page

REFERENCES

Guo, Weixing, and Langevin, C.D., 2002. User's guide to SEAWAT: A computer program for simulation of three-dimensional variable-density ground-water flow. U.S. Geological Survey Techniques of Water-Resources Investigations, book 6, chap. A7, 77p.

Harbaugh, A.W., 1995, Direct solution package based on alternating diagonal ordering for the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 95-288, 46p.

Harbaugh, A.W., Banta, E.R., Hill, M.C., and McDonald, M.G., 2000. MODFLOW-2000, the U.S. Geological Survey modular ground-water model – User guide to modularization concepts and the ground-water flow process. U.S. Geological Survey Open-File Report 00-92, 121 p.

Harbaugh, A.W., and McDonald, M.G., 1996, User's documentation for MODFLOW-96, an update to the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 96-485, 56 p.

Harbaugh, A.W., 1995, Direct solution package based on alternating diagonal ordering for the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 95-288, 46 p.

Hill, M.C., 1990, Preconditioned Conjugate-Gradient 2 (PCG2), A computer program for solving ground-water flow equations: U.S. Geological Survey Water-Resources Investigations Report 90-4048, 43 p.

Hsieh, P.A., and Freckleton, J.R., 1993, Documentation of a computer program to simulate horizontal-flow barriers using the U.S. Geological Survey modular three-dimensional finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 92-477, 32 p. (Note: the input instructions in this documentation are outdated. Use the input instructions in Harbaugh et al. (2000) instead.)

Jones, M. 1999. A Reinjection Drainflow Package for MODFLOW. Albuquerque, NM, 33 p.

Langevin, C.D., Shoemaker, W.B., and Guo, Weixing. 2003. SEAWAT-2000, the U.S. Geological Survey Modular Ground-Water Model–Documentation of the SEAWAT-2000 Version with the Variable-Density Flow Process (VDF) and the Integrated MT3DMS Transport Process (IMT). U.S. Geological Survey Open-File Report 03-426, 43 p.

Langevin, C.D., Thorne, D.T., Jr., Dausman, A.M., Sukop, M.C., and Guo, Weixing, 2007, SEAWAT Version 4: A Computer Program for Simulation of Multi-

Species Solute and Heat Transport: U.S. Geological Survey Techniques and Methods Book 6, Chapter A22, 39 p.

Leak, S.A. and Prudic, D.E., 1991, Documentation of a computer program to simulate aquifer-system compaction using the modular finite-difference ground-water flow model: U.S. Geological Survey Techniques of Water-Resources Investigations, book 6, chap. A2, 68 p.

Leake, S.A., and Lilly, M.R., 1997, Documentation of a computer program (FHB1) for assignment of transient specified-flow and specified-head boundaries in applications of the modular finite-difference ground-water flow model (MODFLOW): U.S. Geological Survey Open-File Report 97-571, 50 p.

McDonald, M.G., A.W. Harbaugh. 1988. A Modular, Three-Dimensional Finite-Difference Groundwater Flow Model. Techniques of Water Resources Investigations of the United States Geological Survey, Book 6, U.S. Government Printing Office, Washington, D.C.

McDonald, M.G., Harbaugh, A.W., Orr, B.R., and Ackerman, D.J., 1992, A method of converting no-flow cells to variable head cells for the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 91-536, 99 p.

Mehl, S.E. and Hill, M.C., MODFLOW-2000, the U.S. Geological Survey modular ground-water model -- User guide to the LINK-AMG (LMG) Package for solving matrix equations using an algebraic multigrid solver: U.S. Geological Survey Open-File Report 01-177, 33 p.

Survey Modular Ground-Water Model–Documentation of the SEAWAT-2000 Version with the Variable-Density Flow Process (VDF) and the Integrated MT3DMS Transport Process (IMT). U.S. Geological Survey Open-File Report 03-426, 43 p.

Restrepo, J.I. and A.M Montoya. 1997. MODFLOW Wetland Module: Final Report. Department of Geography and Geology, Florida Atlantic University, Boca Raton, FL, 86 pp.

Restrepo, J.I., A.M. Montoya, and J. Obeysekera. 1998. A Wetland Simulation Module for the MODFLOW Ground Water Model. Ground Water, 36(5):764-770.

Restrepo, J.I., A.M. Montoya, and D. Garces. 2001. The utility generation package (UGEN) for the groundwater flow model, MODFLOW. Submitted to the conference MODFLOW 2001. Golden, Colorado September 11-14, 2001.

Restrepo, J.I., A.M. Montoya and D. Garces. 2003. The Updated Utility Generation Package (USGEN) for the Groundwater Flow Model, MODFLOW. Developed by the Hydrological Research Center at Florida Atlantic University, Boca

Raton, FL in cooperation with the South Florida Water Management District, West Palm Beach, FL.

South Florida Water Management District. Diversion Systems: Structures and Pump Stations. Prepared by Florida Atlantic University for the South Florida Water Management District. [Online] Available http://iweb/iwebB501/omod/hsm/modflow/div_system.htm, December 2005.

Wilson, J.D. and Naff, R.L., 2004, The U.S. Geological Survey modular ground-water model -- GMG linear equation solver package documentation: U.S. Geological Survey Open-File Report 2004-1261, 47 p.

Zheng, Chunmiao, and Wang, P.P. 1999. MT3DMS—A modular three-dimensional multispecies transport model for simulation of advection, dispersion and chemical reactions of contaminants in ground-water systems; documentation and user's guide. Jacksonville, Fla., U.S. Army Corps of Engineers Contract Report SERDP-99-1.

Zheng, Chunmiao, 2006, MT3DMS v5.2 Supplemental User's Guide, Technical Report to the U.S. Army Engineer Research and Development Center, Department of Geological Sciences, University of Alabama, 24 p.

Zheng, Chunmiao, Hill, M.C., and Hsieh, P.A., 2001, MODFLOW-2000, the U.S. Geological Survey modular ground-water model - User guide to the LMT6 package, the linkage with MT3DMS for multi- species mass transport modeling: U.S. Geological Survey Open-File Report 01-82, 43 p.

Blank page

APPENDIX A

LECsR main name-file and output for the first 100 stress periods for both MODFLOW-96 and SEAWAT(v4) 2000 WMD

See attached CD

Blank page

APPENDIX B
LWCFL main name-file and output for 250 stress periods for both SEAWAT(v3) 2000 WMD and SEAWAT(v4) 2000 WMD

See attached CD

Blank page

APPENDIX C

USGS case 7 "7_swtv4_ex" main name-file and output for both USGS SEAWAT(v4) and SEAWAT(v4) 2000 WMD

See attached CD

MODFLOW/SEAWAT 2000WMD
USER MANUAL – Version 4

Jorge Restrepo

Angela Montoya

September 2008

**Prepared by the Department of Geosciences,
Florida Atlantic University.
For the
South Florida Water Management District,
Water Supply Department.**

Blank page

Acknowledgements

Blank page

Disclaimer:

The input documentation of standard MODFLOW-2000 (Harbaugh et al., 2000)**, MT3DMS** (Zheng and Wang, 1999; Zheng, 2006)**, and SEAWAT-2000** (Langevin et al., 2003, 2007) **is referenced exactly from the U.S. Geological Survey and MT3D documentation, with only minor changes**– due to inclusion of the District's packages**. However, the new package additions are now documented, as well as any other modifications that were required to the source code. The new material has been created by FAU. This documentation is not for publication purposes, but rather, for SFWMD-user convenience.**

User should contact FAU Dr. Jorge Restrepo at restrepo@fau.edu **or SFWMD Angela Montoya at** amontoy@sfwmd.gov **if any errors.**

Blank page

Table of Contents

Blank page

List of Tables

Blank page

INPUT INSTRUCTIONS FOR MODFLOW/SEAWAT-2000WMD

INTRODUCTION

The purpose of this research is to modify the SEAWAT-2000 (Langevin et al., 2003) software to make it compatible with the customized hydrologic packages used by the South Florida Water Management District (DISTRICT or SFWMD). The updated source code (program) is called herein SEAWAT-2000WMD. The Resource Evaluation and Subregional Modeling Division (RESM) has been using various versions of MODFLOW and SEAWAT for modeling projects of its own, but so far there is no single source code capable of covering all the required functionality entirely. Thus, the updated code will increase the current capabilities of the modeling software used by RESM D by allowing just one program to handle all the tasks that were performed previously by different source codes.

SEAWAT-2000 (Langevin et al., 2003) is a combined version of MODFLOW (Harbaugh et al., 2000) and MT3DMS (Zheng and Wang, 1999; Zheng, 2006) designed to simulate three-dimensional, variable-density ground-water flow and multi-species transport. The Variable-Density Flow (VDF) process in SEAWAT was based on the constant-density Ground-Water Flow (GWF) process of MODFLOW. The VDF process uses the MODFLOW and MT3DMS methodologies to solve the variable-density ground-water flow and the solute transport equations, respectively. The MT3DMS part of SEAWAT is referred to as the Integrated MT3DMS Transport (IMT) process. Background material is available in the published user guides for SEAWAT, SEAWAT-2000, MODFLOW-2000, and MT3DMS.

The version of SEAWAT used herein (Langevin et al., 2007) is capable of simulating simultaneous solute and heat transport and the effect of fluid viscosity on resistance to ground-water flow. SEAWAT simulates heat transport using the intrinsic capabilities of MT3DMS. With this approach, the effect of temperature is calculated as one of the species being computed by MT3DMS. Other features have also been implemented by the USGS to this version of SEAWAT to improve overall functionality. The new main features are:

> An option to update the flow field if fluid densities have changed under a user-specified value. This option may substantially reduce runtimes for some simulations where a new flow solution is not needed for each transport time-step.

> Auxiliary variables can now be used with the Time-Variant Constant-Head (CHD) package to designate the reference density associated with the prescribed head. This option allows for more control of boundary heads specified with the CHD package (e.g., sea water).

The performance of the updated source code for SEAWAT-2000WMD has been tested in several applications including the Lower East Coast Subregional Model and Lower West Coast Floridan Aquifer Model. Future applications, however, might reveal errors that were not detected in the test simulations. Users are requested to notify to FAU of any errors found in the computer program.

GLOBAL PROCESS INPUT INSTRUCTIONS

Name File

The name file contains the names of most input and output files used in a model simulation and controls the parts of the model program that are active. ("OPEN/CLOSE" files, described in the Input Instructions for Array Reading Utility Modules section, are not included in the name file.) The name file is read on unit 99. File units from 99 to 103 should not be used in SEAWAT. The name file is constructed as follows.

1. Ftype Nunit Fname

The name file contains one of the above records (item 1) for each file. All variables are free format. The length of each record must be 199 characters or less. The records can be in any order except for records where Ftype (file type) is "GLOBAL" or "LIST" as described below.

Comment records are indicated by the # character in column one and can be located anywhere in the file. Any text characters can follow the # character. Comment records have no effect on the simulation.

EXPLANATION OF VARIABLES IN THE NAME FILE

Ftype—is the file type, which must be one of the following character values. Ftype may be entered in all uppercase, all lowercase, or any combination.

GLOBAL for the global listing file—if this type is not present, the LIST file is used for the global listing file as well as for the forward run listing file.

LIST for the forward run listing file—if this type is not present, the GLOBAL file is used for the forward run listing file as well as for the global listing file.

The name file must always include a record that specifies GLOBAL or LIST for Ftype. Both records can be included, and if so, the GLOBAL record must be the first non-comment record and the LIST record must be the second non-comment record. If only one of the records is included, it must be the first non-comment record.

- DIS for the discretization file
- MULT for the multiplier array file
- ZONE for the zone array
- BAS6 for the Ground-Water Flow process Basic package
- OC for the Ground-Water Flow process Output Control Option
- MBUD for SFWMD Multiple Budget file
- FLG for SFWMD Global Flag file
- BCF6 for the Ground-Water Flow process Block-Centered Flow package
- LPF for the Ground-Water Flow process Layer Property Flow package

- HFB6 for the Ground-Water Flow process Horizontal Flow Barrier package
- RCH for the Ground-Water Flow process Recharge package
- RIV for the Ground-Water Flow process River package
- WEL for the Ground-Water Flow process Well package
- DRN for the Ground-Water Flow process Drain package
- GHB for the Ground-Water Flow process General-Head Boundary package
- EVT for the Ground-Water Flow process Evapotranspiration package
- CHD for the Ground-Water Flow process Constant-Head package
- UGEN for the Utility Generation package
- HBXY for the Interpolation Head Boundary package
- WELT for the Well Trigger package/ Generated Multiple Wells package
- WTL for the Surface-Ground-Water Flow process Wetland package
- DIV for the Diversion package
- TRG for the Trigger package
- RDF for the Reinjected Drain Flow package
- OBSG for the Observation package
- VDF for the Variable Density Flow, VDF
- VSC for the Variable Viscosity
- BTN for the Solute Transport process Basic Transport
- ADV for the Solute Transport process Advection
- DSP for the Solute Transport process Dispersion
- SSM for the Solute Transport process Source/Sink Mixing
- RCT for the Solute Transport process Reaction
- SIP for the Strongly Implicit Procedure package Enhancement
- SOR for the Slice-Successive Over-Relaxation package
- PCG for the Preconditioned Conjugate-Gradient package
- DE4 for the Direct Solution package
- LMG4 for the Multigrid solver package
- GCG for the Solute Transport process Generalized Conjugate Gradient packages

DATA(BINARY): for binary (unformatted) files, such as those used to save cell-by-cell budget data and binary (unformatted) head and drawdown data

DATA for formatted (text) files, such as those used to save formatted head and drawdown and for input of data from files that are separate from the primary package input files

DATAGLO(BINARY) for binary (unformatted) files, such as those used to save cell-by-cell budget data and binary (unformatted) head and drawdown data.

DATAGLO for formatted (text) files, such as those used to save formatted head and drawdown and for input of data from files that are separate from the primary package input files.

Nunit—is the FORTRAN unit to be used when reading from or writing to the file. Any legal unit number on the computer being used can be specified except units 96-99. Unit 99 is used for the name file and for reading multivalued variables using the OPEN/CLOSE option of the utility modules. Units 97 and 98 are used for batch files as explained below. Unit 96 is used for a file to which error messages may be written. The unit number for each file must be unique.

Fname—is the name of the file, which is a character value. Pathnames may be specified as part of Fname.

An example of the name file is shown in **Table** 1. This example reads the names of files being used by the Lower West Coast Floridan Aquifer Model, one of the District models. It includes several of the add-on District packages and utilities, such as the BUD, FLG and UGEN file, the MODFLOW package input files using UGEN, and the associated observation stage and flow file. Input for these packages is described in the following sections.

Table 1. Name-file Input Data of LWCFAS Model

Type File	Unit File
GLOBAL	6 lwcfas_12K.glo
LIST	7 lwcfas_12K.lst
BAS6	1 lwcfas_12K.bas
DIS	29 lwcfas_12K.dis
LPF	11 lwcfas_12K.lpf
ZONE	40 lwcfas_12K.zone
CHD	33 lwcfas_12K.chd
WEL	12 lwcfas_12K.wel
GHB	17 lwcfas_12K.ghb
OC	22 lwcfas_12K.oc
PCG	19 lwcfas_12K.pcg
UGEN	85 lwcfas_12K_CTRL.gen
HBXY	88 lwcfas_12K.hbxy
OBSG	89 lwcfas_12K.obsg
FLG	23 LWC_flg.txt
MBUD d	105 lwcfas_12K.bud
VDF	80 lwcfas_12K.vdf
BTN	81 lwcfas_12K.btn
ADV	82 lwcfas_12K.adv
DSP	83 lwcfas_12K.dsp
SSM	84 lwcfas_12K.ssm
GCG	86 lwcfas_12K.gcg

SFWMD Global Flag File

Input to define global flags for different packages (Restrepo and Montoya, 2008) in SEAWAT-2000WMD is read from the file that is specified with "FLG" as the file type. A global flag is one that will apply a change to the whole model.

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. NFLAGS

2. IVET_FLG(1), IVET_FLG(2), IVET_FLG(3), IVET_FLG(4)

EXPLANATION OF VARIABLES READ FROM THE GLOBAL FLAG FILE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

IVET_FLG(1)— is the flag for written values with a given precision value for output binary files.
- If IVET_FLG(1)=0, file is written by default in REAL*4
- If IVET_FLG(1)=1, file is written in REAL*8

IVET_FLG(2)—is the flag for multibud.
- If IVET_FLG(2)=0, do not use multibud.
- If IVET_FLG(2)=1, use multibud.

IVET_FLG(3)— is the flag for accumulated fluxes that can be applied in the Output Control package.
- If IVET_FLG(3)=0, MODFLOW summation utility is not active
- If IVET_FLG(3)>1, MODFLOW summation utility is active for a given number of packages.

Note: the flag value is the number of packages that the summation utility is applied to. The summation package has been implemented only through the formatted output control file. To use the summation package, the user needs to read the output control instructions. If you are using the summation feature, this integer must be at least as large as the number of packages, which write out cell-by-cell flows. This flag value should not be too large in order to reduce wasting memory. On the other hand, the program will issue an error message and abort the program if too small of a value is specified.

IVET_FLG(4)— is the flag for inputting the Wells package with rows and columns, or xy coordinates. (Since this option is not completed yet, zero should be input).
- If IVET_FLG(4)=0, well package reads rows and columns for well location
- If IVET_FLG(4)=1, well package reads x and y coordinates for well location

IVET_FLG(5)—is the flag for the PCG2 package for the minimum forced number of iterations in one time step. (Since this option is not completed yet, zero should be input).

- If IVET_FLG(5)=0, this option is not used
- If IVET_FLG(5)=1, this option is used

IVET_FLG(6)— is the flag to create MODFLOW input files without UGEN. (Since this option is not completed yet, zero should be input).
- If IVET_FLG(6)=0, Regular MODFLOW input files are not created
- If IVET_FLG(6)=1, Regular MODFLOW input files are created

Discretization File

Discretization information is read from the file that is specified by "DIS" as the file type (Harbaughet al., 2000).

FOR EACH SIMULATION

0. [#Text]

(Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times).

1. NLAY NROW NCOL NPER ITMUNI LENUNI

2. LAYCBD(NLAY)

3. DELR(NCOL) - U1DREL

4. DELC(NROW) - U1DREL

5. TOP(NCOL,NROW) - U2DREL

6. BOTM(NCOL,NROW) - U2DREL

(Item 6 is repeated for each model layer and Quasi-3D confining bed in the grid. These layer variables are read in sequence going down from the top of the system. Thus, the number of BOTM arrays must be NLAY plus the number of Quasi-3D confining beds.)

FOR EACH STRESS PERIOD

7. PERLEN NSTP TSMULT Ss/tr

EXPLANATION OF VARIABLES READ FROM THE DISCRETIZATION FILE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NLAY—is the number of layers in the model grid.

NROW—is the number of rows in the model grid.

NCOL—is the number of columns in the model grid.

NPER—is the number of stress periods in the simulation.

ITMUNI—indicates the time unit of model data, which must be consistent for all data values that involve time. Be sure to use consistent units for all input data even when ITMUNI indicates an undefined time unit. When the time unit is defined, MODFLOW uses it to print a table of elapsed simulation time:

    0. Undefined
    1. Seconds
    2. Minutes
    3. Hours
    4. Days
    5. Years

LENUNI—indicates the length unit of model data, which must be consistent for all data values that involve length. Note that the program will still run even if "undefined" length units are specified because the fundamental equations used in MODFLOW do not require that the length unit be identified. But be sure to use consistent units for all input data even when LENUNI indicates an undefined length unit:

    0. Undefined
    1. Feet
    2. Meters
    3. Centimeters

LAYCBD—is a flag, with one value for each model layer that indicates whether or not a layer has a Quasi-3D confining bed below it. Zero (0) indicates no confining bed Non-zero indicates a confining bed. The LAYCBD value for the bottom layer must be 0.

DELR—is the cell width along rows. Read one value for each of the NCOL columns. This is a multi-value one-dimensional variable with one value for each column.

DELC—is the cell width along columns. Read one value for each of the NROW rows. This is a multi-value one-dimensional variable with one value for each row.

Top—is the top elevation of layer 1. For the common situation in which the top layer represents a water-table aquifer, it may be reasonable to set Top equal to land-surface elevation.

BOTM—is the bottom elevation of a model layer or a Quasi-3d confining bed.

PERLEN—is the length of a stress period.

NSTP—is the number of time steps in a stress period.

TSMULT—is the multiplier for the length of successive time steps. The length of a time step is calculated by multiplying the length of the previous time step by TSMULT. The length of the first time step, $\Delta t1$, is related to PERLEN, NSTP, and TSMULT by the relation

$$\Delta t_1 = PERLEN \left(\frac{TSMULT - 1}{TSMULT^{NSTP} - 1} \right)$$

Ss/tr—is a character variable that indicates if the stress period is transient or steady state. The only allowed options are "SS" and "TR", but these are case insensitive.

Multiplier File

Input to define multiplier arrays is read from the file that is specified with "MULT" as the file type. Multiplier arrays can be used to calculate layer variables from parameter values (Harbaugh et al., 2000).

FOR EACH SIMULATION

0. [#Text]

(Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times).

1. NML

2. MLTNAM [FUNCTION]

(If Item 2 does not contain the optional FUNCTION keyword, read Item 3).

3. [RMLT(NCOL,NROW)] - U2DREL

(Otherwise, if Item 2 contains the optional FUNCTION keyword, read Item 4)

4. [MLTNAM1 [op1 MLTNAM2] [op2 MLTNAM3] [op3 MLTNAM4]...]

Repeat items 2-4 for each of the NML multiplier arrays.

EXPLANATION OF VARIABLES READ FROM THE MULTIPLIER FILE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Text is printed when the file is read.

NML—is the number of multiplier arrays to be defined.

MLTNAM—is the name of a multiplier array. This name can consist of 1 to 10 characters and is not case sensitive. The name "NONE" should not be used because it is a reserved word. The optional keyword, FUNCTION, indicates that the multiplier array will be constructed from other multiplier arrays that have already been defined. Construction is by arithmetic combinations of the multipliers; see the explanation below for variable 'op1, op2, op3 ...'

RMLT—is a multiplier array.

MLTNAM1, MLTNAM2, MLTNAM3, —are names of multiplier arrays that have already been defined.

'op1, op2, op3, …'—are arithmetic operators used to define a multiplier array based on other multiplier arrays. Each operator can be either "+", "-", "*", or "/". Operations are applied from left to right to each array element. The operators must be separated from the multiplier array names by at least one space.

Zone File

Input to define zone arrays is read from the file that is specified with "ZONE" as the file type. Zone arrays can be used to specify the cells in a layer variable that are associated with a parameter (Harbaugh et al., 2000).

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. NZN

2. ZONNAM

3. IZON(NCOL,NROW) - U2DINT

Items 2-3 are repeated for each of the NZN zone arrays.

EXPLANATION OF VARIABLES READ FROM THE ZONE FILE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NZN—is the number of zone arrays to be defined.

ZONNAM—is the name of a zone array. This name can consist of 1 to 10 characters and is not case sensitive; that is, any combinations of the same characters with different case are equivalent. The name "ALL" should not be used for a zone array because it is a reserved word.

IZON—is a zone array.

SFWMD Multiple Budget File

Input to define budget zones for different packages in SEAWAT-2000WMD is read from the configuration file that is specified with "MBUD" as the file type. The configuration file was designed to recognize input based on the tag located on the first ten (10) characters of a line. This tag consists of a keyword in uppercase. It is important to include these keywords in the input file to ensure the proper function of the file. If a keyword consists of less than ten characters, the remaining characters must be blank spaces. A list of the keywords used by the MBUD package is shown in Table 2. **Note: When using SEAWAT-2000WMD with Multibud, the user must use the SFWMD Global Flag File (see input instructions page 20).**

Table 2. Description of the keywords used in the Multiple Budget package.

KEYWORD	DESCRIPTION
COMMENT	Comment lines are used to identify the structure of the file. Comments labeling arrays separated by a blank space are oriented vertically; e.g. the array REPORT, containing the flags for inflow, outflow, net flow and report, respectively, would appear like this: COMMENT | I | O | N | R COMMENT | N | U | E | E COMMENT | | T | T | P REPORT | 0 | 0 | 1 | 0 Therefore, the flag for inflow (IN) equals 0, the flag for outflow (OUT) equals 0, the flag for net flow (NET) equals 1, and the flag for report (REP) equals 0.
BASINP	Used to specify the name and extension of the BAS6 file.
CELLXCELL	Used to specify the name and extension of the file that contains data for the cell-by-cell flows.
TERMS	Used to identify the array containing the flags for the following terms (in respective order): Storage term Constant head term East (right) cell face flow term West (left) cell face flow term North (frontal) cell face flow term South (posterior) cell face flow term Upper cell face flow term Lower cell face flow term Wells term Drains term Evapotranspiration term Leakage term Recharge term General Head term Sources and sinks term RDF term. Lakes term Operation term The array TERMS is binary, which means it only accepts a value of 0 for an inactive term and a value of 1 for an active term.
PER	Used to tag the line containing the number of stress periods for the simulation.
REPORT	Used to identify the array containing the flags for the following parameters to be included in the report file (output): IN (Inflows) OUT (Outflows) NET (Inflows – Outflows)

KEYWORD	DESCRIPTION
	REP (Report)
MULTIPLIER	Used to recognize the multiplier constant that modifies inflows/outflows.
BEGIN	Used to recognize the location in the file where a new zone definition begins. Every occurrence of BEGIN must be paired with the keyword END. In the line immediately after BEGIN a name must be assigned to the zone, no keyword required.
END	Used to mark the location in the file of the end of a new zone definition.
LAYERS	Located after the line that contains the zone name, this keyword defines the array with the top and bottom layers contained in the zone.

Zone Definition

The Multiple Budget (MBUD) package (Rodberg, K., 1999) calculates the water budget for a given set of zones. These zones must be defined in the configuration file. A new zone definition must start with a new line with the keyword BEGIN. The next line contains the name designated to the zone (ten characters maximum), no keyword is required.

The third line is used to define the active layers in the new zone. It must start with the keyword LAYERS followed by four blank spaces (keywords must be ten characters long) plus one additional blank space that separates the keyword from the entry. In the same line, immediately after the five blank spaces, the top and bottom layer numbers are separated by a single blank space.

The next lines define the area of the zone. Each line is made of three entries representing the starting row, starting column, and ending column numbers, respectively. For example, take a model composed of three layers, ten rows, and ten columns (see Table 3, left side). The zone to be defined, named NEWZONE, is represented by the gray cells. The new zone definition in the configuration file contains the lines shown in Table 3 (right side).

Table 3. Representation of an area of a zone and lines in the configuration budget file for this zone.

	1	2	3	4	5	6	7	8	9	10
1					▓	▓				
2				▓	▓	▓	▓			
3					▓	▓	▓	▓		
4			▓	▓	▓	▓	▓	▓		
5				▓	▓	▓	▓			
6			▓	▓	▓	▓	▓	▓		
7		▓	▓	▓	▓					
8		▓	▓	▓			▓	▓	▓	
9					▓	▓	▓	▓		
10							▓	▓	▓	

```
BEGIN
NEWZONE
LAYERS   1 3
   1   5   6
   2   4   7
   3   5   8
   4   3   8
   5   4   8
   6   3   9
   7   2   6
   8   3   5
   8   8   10
   9   5   9
  10   7   10
END
```

Input Configuration File

Note: Keywords (below in italic) must be exactly 10 characters long. Any remaining characters must be filled with blank spaces. Also, there must be one blank space separating the 10-character-long keyword and the variable it tags.

[*COMMENT*   Text]

Item 0 is optional. This item can be repeated many times. For convenience, the comment lines labeling arrays separated by a single blank space should be oriented vertically, using one character per line.

BASINP   (BASFILE)

CELLXCELL (CELLFILE)

TERMS   (TERMFLG)

PER   (NUMPER)

REPORT   (REPFLG)

MULTIPLIER (MULT)

FOR EACH REGION:

BEGIN

(REGNAM)

LAYERS   (TOPLAY) (BOTLAY)

(STROW)   (STCOL)   (ENDCOL)

Item 10 must be repeated as many times as necessary to define the extension of the zone. See the New Zone Definition section above for more details.

END

EXPLANATION OF THE VARIABLES READ FROM THE BUD FILE.

BASFILE—The name and extension of the BAS6 file.

CELLFILE—The name and extension of the file containing the cell-by-cell flow information.

TERMFLG—The array that contains the flags to activate the calculation of the following terms:
1.   Storage term
2.   Constant head term
3.   East (right) cell face flow term
4.   West (left) cell face flow term
5.   North (frontal) cell face flow term
6.   South (posterior) cell face flow term
7.   Upper cell face flow term
8.   Lower cell face flow term
9.   Wells term
10.   Drains term
11.   Evapotranspiration term
12.   Leakage term
13.   Recharge term
14.   General head boundary term
15.   Sources and sinks term
16.   RDF term.
17.   Lakes term
18.   Operation term

TERMFLG is a binary array,. This means that it only recognizes values of 1 (for active terms) or 0 (for inactive terms).

NUMPER—Represents the number of stress periods.

REPFLG—The array that contains the flags for the parameters to be included in the report file (output). The parameters are in the following order:

1. Inflow
2. Outflow
3. Net flow (Inflow – Outflow)
4. Report

MULT—The multiplier constant that modifies the flows in the simulation.

REGNAM—The name of the region. It must be 10 characters long maximum.

TOPLAY—The number of the top layer of the region.

BOTLAY—The number of the bottom layer of the region.

STROW—The row used in a given line to define the extension of the zone.

STCOL—The column where consecutive active cells begin in STROW.

ENDCOL—The column where consecutive active cells end in STROW.

Disclaimer:

This section of documentation is taken mainly from MODFLOW-2000 (Harbaugh et al., 2000) with minor editing. However, several new additions are now documented. The new material has been created by FAU. This documentation is not for publication purposes, but rather, for SFWMD-user convenience

GROUND-WATER FLOW PROCESS INPUT INSTRUCTIONS

Basic (BAS) Package

Input for the Ground-Water Flow process Basic (BAS) package except for output control input is read from the file that is specified with "BAS6" as the file type. The file type has been changed from that used in MODFLOW-96, which was "BAS", because the input data are not compatible (Harbaugh et al., 2000).

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. Options (199 text characters).

If there are no options to specify, then a blank record must be included for Item 1.

2. IBOUND(NCOL,NROW) or (NCOL,NLAY) -- U2DINT

If not a cross section, a layer variable is read for each layer in the grid. If a cross section, NLAY rows of NCOL values are read.

3. HNOFLO (10-character field unless Item 1 contains '   FREE'.)

4. STRT(NCOL,NROW) or (NCOL,NLAY) -- U2DREL

If not a cross section, a layer variable is read for each layer in the grid. If a cross section, NLAY rows of NCOL values are read.

EXPLANATION OF VARIABLES READ BY THE BAS PACKAGE

Text—is a character variable that starts in column 2. The first two comment lines will become variable HEADNG, which is used as a printout title throughout the program. (If there are no comment lines, then HEADNG will be blank.) HEADNG is limited to 80 columns, but subsequent Text lines can be up to 199 columns. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file.

Options—is a character variable that is scanned for words (separated by one or more spaces) that specify program options. Three options are currently recognized. Unrecognized words are ignored, and a word may be specified in either uppercase or lowercase. A blank record is acceptable and indicates no options.

XSECTION indicates that the model is a 1-row cross section for which STRT and IBOUND should each be read as single two-dimensional variables with dimensions of NCOL and NLAY.

CHTOCH indicates that flow between adjacent constant-head cells should be calculated.

FREE indicates that free format is used for input variables throughout the Basic package and other packages as indicated in their input instructions. Be sure that all variables read using free format have a non-blank value and that a comma or at least one blank separates all adjacent values.

IBOUND—is the boundary variable. One value is read for every model cell. Usually, these values are read a layer at a time; however.
- If IBOUND(J,I,K) < 0, cell J,I,K has a constant head.
- If IBOUND(J,I,K) = 0, cell J,I,K is inactive.
- If IBOUND(J,I,K) > 0, cell J,I,K is active.

HNOFLO—is the value of head to be assigned to all inactive (no flow) cells (IBOUND = 0) throughout the simulation. This value does not affect model results but serves to identify inactive cells when head is printed. This value is also used as drawdown at inactive cells if the drawdown option is used.

STRT—is initial (starting) head—i.e., head at the beginning of the simulation. STRT must be specified for all simulations, including steady-state simulations. One value is read for every model cell. Usually, these values are read a layer at a time. For simulations in which the first stress period is steady state, the values used for STRT generally do not affect the simulation (exceptions may occur if cells go dry and(or) rewet).

Output Control (OC) Option

Input to the Output Control Option of the Ground-Water Flow process is read from the file that is specified as type "OC" in the name file. If no "OC" file is specified, default output control is used. Under the default, head and overall budget are written to the listing file (printed) at the end of every stress period. The default printout format for head and drawdown is 10G11.4.

There are two ways to specify Output Control data: words or numeric codes. One of these methods must be used throughout any simulation (Harbaugh et al., 2000).

Output Control Using Words

Recognized words are shown in bold italics; these words must be entered exactly as shown (either uppercase or lowercase). Optional parts of records are shown in brackets. One or more spaces must separate each word or variable, and the total record length must not exceed 199 characters.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. Any combination of the following records:

HEAD PRINT FORMAT IHEDFM	Omit to obtain an unformatted file.
HEAD SAVE FORMAT CHEDFM [LABEL]	
HEAD SAVE UNIT IHEDUN	
DRAWDOWN PRINT FORMAT IDDNFM	
DRAWDOWN SAVE FORMAT CDDNFM [LABEL]	Omit to obtain an unformatted file.
DRAWDOWN SAVE UNIT IDDNUN	
IBOUND SAVE FORMAT CBOUFM [LABEL]	
IBOUND SAVE UNIT IBOUUN	
COMPACT BUDGET [AUX or AUXILIARY]	

If the final "COMPACT BUDGET" option is used, cell-by-cell budget files produced require less disk space than files written by MODFLOW-88; however, programs that read these data using the specifications of MODFLOW-88 will need to be modified. If this option is not used, MODFLOW-2000 will write the files using the MODFLOW-88 method. The optional word "AUX" (or "AUXILIARY") after "COMPACT BUDGET" indicates that auxiliary data that are defined in packages (see input data for the RIV, WEL, DRN, and GHB packages) should be saved in the budget file along with budget data.

PRINT indicates that the model output is written to the LIST file.

SAVE indicates that output is written to the indicated file unit. Drawdown and head are saved as text files if the SAVE FORMAT record is included. If the SAVE FORMAT record is not used, then the

file is written as a binary (unformatted) file. Binary files are usually more compact than text files, but they are not generally transportable among different computer operating systems or different Fortran compilers.

FOR EACH TIME STEP FOR WHICH OUTPUT IS DESIRED

2. PERIOD IPEROC STEP ITSOC

3. Any combination of the following records:

   PRINT HEAD [list layers if all layers not desired]
   PRINT DRAWDOWN [list layers if all layers not desired]
   PRINT BUDGET Write overall volumetric budget to the LIST file.
   SAVE HEAD [list layers if all layers not desired]
   SAVE DRAWDOWN [list layers if all layers not desired]
   SAVE IBOUND [list layers if all layers not desired]
   SAVE BUDGET Write cell-by-cell budget data to a file.

Item 2 and one or more item-3 records are specified for each time that an output is desired. These records must be in increasing order of simulation time.

EXPLANATION OF VARIABLES READ BY OUTPUT CONTROL USING WORDS

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read

.

IHEDFM	Format Used	IHEDFM	Format Used
0	10G11.4	11	20F5.4
1	11G10.3	12	10G11.4
2	9G13.6	13	10F6.0
3	15F7.1	14	10F6.1
4	15F7.2	15	10F6.2
5	15F7.3	16	10F6.3
6	15F7.4	17	10F6.4
7	20F5.0	18	10F6.5
8	20F5.1	19	5G12.5
9	20F5.2	20	6G11.4
10	20F5.3	21	7G9.2

IHEDFM—is a code for the format in which heads will be printed. (Positive values are associated with wrap format, whereas negative values are associated with strip format.)

CHEDFM—is a character value that specifies the format for saving heads, and can only be specified if the word method of output control is used. The format must contain 20 characters or less and must be a valid Fortran format that is enclosed in parentheses. The format must be enclosed in apostrophes if it contains one or more blanks or commas. The optional word LABEL after the format is used to indicate that each

layer of output should be preceded with a line that defines the output (simulation time, the layer being output, and so forth). If there is no record specifying CHEDFM, then heads are written to a binary (unformatted) file.

IHEDUN—is the unit number on which heads will be saved.

IDDNFM—is a code for the format in which drawdowns will be printed. The codes are the same as for IHEDFM.

CDDNFM—is a character value that specifies the format for saving drawdown, and can only be specified if the word method of output control is used. The format must contain 20 characters or less and must be a valid Fortran format that is enclosed in parentheses. The format must be enclosed in apostrophes if it contains one or more blanks or commas. The optional word LABEL after the format is used to indicate that each layer of output should be preceded with a line that defines the output (simulation time, the layer being output, and so forth). If there is no record specifying CDDNFM, then drawdown is written to a binary (unformatted) file.

IDDNUN—is the unit number on which drawdowns will be saved.

CBOUFM—is a character value that specifies the format for saving IBOUND, and can only be specified if the word method of output control is used. The format must contain 20 characters or less and must be a valid Fortran format that is enclosed in parentheses. The format must be enclosed in apostrophes if it contains one or more blanks or commas. The optional word LABEL is used to indicate that each layer of output should be preceded with a line that defines the output (simulation time, the layer being output, and so forth). If there is no record specifying CBOUFM, then IBOUND is written using format (20I4). IBOUND is never written as a binary (unformatted) file.

IBOUUN—is the unit number on which drawdowns will be saved.

IPEROC—is the stress period number at which output is desired.

ITSOC—is the time step number (within a stress period) at which output is desired.

Output Control Using Numeric Codes

All variables are free format if the word FREE is specified in Item 1 of the Basic package input file; otherwise, the variables all have 10-character fields.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. IHEDFM IDDNFM IHEDUN IDDNUN

FOR EACH TIME STEP

2. INCODE IHDDFL IBUDFL ICBCFL

3. Hdpr Ddpr Hdsv Ddsv

(Item 3 is read 0, 1, or NLAY times, depending on the value of INCODE.)

EXPLANATION OF VARIABLES READ BY OUTPUT CONTROL USING NUMERIC CODES

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

IHEDFM—is a code for the format in which heads will be printed. See the description above in the explanation of variables read by output control using words.

IHEDUN—is the unit number on which heads will be saved.

IDDNFM—is a code for the format in which drawdowns will be printed. The codes are the same as for IHEDFM.

IDDNUN—is the unit number on which drawdowns will be saved.

INCODE—is the code for reading Item 3.
- If INCODE < 0, Item 3 flags are used from the last time step and Item 3 is not read.
- If INCODE = 0, all layers are treated the same way and Item 3 will consist of one record.
- If INCODE > 0, Item 3 will consist of one record for each layer.

IHDDFL—is a head and drawdown output flag. This flag allows Item 3 flags to be specified in an early time step and then used or not used in subsequent time steps. Thus, it may be possible to use IHDDFL to avoid resetting Item 3 flags every time step.
- If IHDDFL = 0, no heads or drawdowns will be printed or saved regardless of which Item 3 flags are specified.
- If IHDDFL ≠ 0, heads and drawdowns will be printed or saved according to the Item 3 flags.

IBUDFL—is a budget print flag.
- If IBUDFL = 0, overall volumetric budget will not be printed.
- If IBUDFL ≠ 0, overall volumetric budget will be printed.

ICBCFL—is a flag for writing cell-by-cell flow data.
- If ICBCFL = 0, cell-by-cell flow terms are not written to any file.
- If ICBCFL ≠ 0, cell-by-cell flow terms are written to the LIST file or a budget file depending on flags set in the component of flow packages, that is, IWELCB, IRCHCB, and so forth.

Using the summation flux flag (Rodberg, K., 1999) in the SFWMD Global Flag file (see input instructions page 20):

ICBCFL - cell-by-cell flow-term flag.
- if ICBCFL =0 cell by cell flow are not saved
- if ICBCFL = 1 cell by cell flow are written to disk without summing (no change here)
- if ICBCFL =4 add current cell by cell flows to those in the accumulator (on the first call, flows in the accumulator are zeroed out first)
- if ICBCFL = 5 add current cell by cell flows to those in the accumulator, write out flows, zero the accumulator

Hdpr—is the output flag for head printout.
- If Hdpr = 0, head is not printed for the corresponding layer.
- If Hdpr $\neq$ 0, head is printed for the corresponding layer.

Ddpr—is the output flag for drawdown printout.
- If Ddpr = 0, drawdown is not printed for the corresponding layer.
- If Ddpr $\neq$ 0, drawdown is printed for the corresponding layer.

Hdsv—is the output flag for head save.
- If Hdsv = 0, head is not saved for the corresponding layer.
- If Hdsv $\neq$ 0, head is saved for the corresponding layer.

Ddsv—is the output flag for drawdown save.
- If Ddsv = 0, drawdown is not saved for the corresponding layer.
- If Ddsv $\neq$ 0, drawdown is saved for the corresponding layer.

Block-Centered Flow (BCF) Package

Input for the BCF package is read from the file that is type "BCF6" in the name file. The file type has been changed from that used in MODFLOW-96, which was "BCF", because the input data are not compatible (Harbaugh et al., 2000).

FOR EACH SIMULATION

0. IBCFCB   HDRY   IWDFLG   WETFCT   IWETIT   IHDWET

These 6 variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the variables all have 10-character fields.

1. Ltype(NLAY)

Read one value for each layer. These values are free format if the word FREE is specified in Item 1 of the Basic package input file; otherwise, they have fixed-format fields. The fixed format fields are each 2 characters wide with 40 values per line. Use only as many lines as required.

2. TRPY(NLAY)--U1DREL

A subset of the following two-dimensional variables is used to describe each layer. The variables needed for each layer depend on the layer-type code (LAYCON, which is defined as part of the Item-2 Ltype), whether the simulation has any transient stress periods (at least one stress period defined in the Discretization File specifies Ss/Tr as "TR"), and if the wetting capability is active (IWDFLG not 0). If a variable is not needed, it must be omitted. In no situation will all variables be required. The required variables (Items 4-9) for layer 1 are read first; then the variables for layer 2 and so forth.

3. [Sf1(NCOL,NROW)]--U2DREL

Use item 3 if there is at least one transient stress period.

4. If LAYCON is 0 or 2 (see Ltype), then read item 5.

5. [Tran(NCOL,NROW)]--U2DREL.

Otherwise, if LAYCON is 1 or 3 (see Ltype), read item 6.

6. [HY(NCOL,NROW)] -- U2DREL

7. [Vcont(NCOL,NROW)] -- U2DREL

Use item 7 if the current layer is not the bottom layer.

8. [Sf2(NCOL,NROW)]--U2DREL

Use item 8 if there is at least one transient stress period and LAYCON (see Ltype) is 2 or 3.

9. [WETDRY(NCOL,NROW)]--U2DREL

Use item 9 if IWDFLG is not 0 and LAYCON is 1 or 3 (see Ltype).

EXPLANATION OF VARIABLES READ BY THE BCF PACKAGE

IBCFCB—is a flag and a unit number.
- If IBCFCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. The terms that are saved are storage, constant-head flow, and flow between adjacent cells.
- If IBCFCB = 0, cell-by-cell flow terms will not be written.
- If IBCFCB < 0, cell-by-cell flow for constant-head cells will be written in the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. Cell-by-cell flow to storage and between adjacent cells will not be written to any file.

HDRY—is the head that is assigned to cells that are converted to dry during a simulation. Although this value plays no role in the model calculations, it is useful as an indicator when looking at the resulting heads that are output from the model. HDRY is thus similar to HNOFLO in the Basic package, which is the value assigned to cells that are no-flow cells at the start of a model simulation.

IWDFLG—is a flag that determines if the wetting capability is active.
- If IWDFLG = 0, the wetting capability is inactive.
- If IWDFLG is not 0, the wetting capability is active.

WETFCT—is a factor that is included in the calculation of the head that is initially established at a cell when it is converted from dry to wet. (See IHDWET.)

IWETIT—is the iteration interval for attempting to wet cells. Wetting is attempted every IWETIT iteration. If using the PCG solver (Hill, 1990), this applies to outer iterations, not inner iterations. If IWETIT is 0, it is changed to 1.

IHDWET—is a flag that determines which equation is used to define the initial head at cells that become wet:
- If IHDWET = 0, equation is used:
  $h = BOT + WETFCT (h_n - BOT)$.
       h_n is the head at the neighboring cell that causes the cell to wet.
- If IHDWET is not 0, equation 33B (equation 3b in McDonald and others, 1992) is used:
  $h = BOT + WETFCT (WETDRY)$.

Ltype—contains a combined code for each layer that specifies both the layer type (LAYCON) and the method of computing interblock conductance. Use as many records as needed to enter a value for each layer. Values are two-digit numbers:

The left digit defines the method of calculating interblock transmissivity.

0 or blank—harmonic mean.
1—arithmetic mean

36

2—logarithmic mean

3—arithmetic mean of saturated thickness and logarithmic-mean hydraulic conductivity.

The right digit defines the layer type (LAYCON):

0—*confined*—Transmissivity and storage coefficient of the layer are constant for the entire simulation.
1—*unconfined*—Transmissivity of the layer varies. It is calculated from the saturated thickness and hydraulic conductivity. The storage coefficient is constant. This type code is valid only for layer 1.
2—*confined/unconfined*—Transmissivity of the layer is constant. The storage coefficient may alternate between confined and unconfined values. Vertical flow from above is limited if the layer becomes unstaturated.
3—*confined/unconfined*—Transmissivity of the layer varies. It is calculated from the saturated thickness and hydraulic conductivity. The storage coefficient may alternate between confined and unconfined values. Vertical flow from above is limited if the aquifer becomes unstaturated..

TRPY—is a one-dimensional variable containing a horizontal anisotropy factor for each layer. It is the ratio of transmissivity or hydraulic conductivity (whichever is being used) along a column to transmissivity or hydraulic conductivity along a row. Set to 1.0 for isotropic conditions. This is a single variable with one value per layer. Do not read a variable for each layer—that is, include only one array control record for the entire variable.

Sf1—is the primary storage coefficient. Read only if there are one or more transient stress periods specified in the Discretization File. For LAYCON equal to 1, Sf1 will always be specific yield, whereas for LAYCON equal to 2 or 3, Sf1 will always be confined storage coefficient. For LAYCON equal to 0, Sf1 would normally be confined storage coefficient; however, a LAYCON value of 0 can also be used to simulate water-table conditions where drawdowns are expected to remain everywhere a small fraction of the saturated thickness, and where there is no layer above, or flow from above is negligible. In this case, specific yield values would be entered for Sf1.

Tran—is the transmissivity along rows. Tran is multiplied by TRPY to obtain transmissivity along columns. Read only for layers where LAYCON is 0 or 2.

HY—is the hydraulic conductivity along rows. HY is multiplied by TRPY to obtain hydraulic conductivity along columns. Read only for layers where LAYCON is 1 or 3.

Vcont—is the vertical hydraulic conductivity divided by the thickness from a layer to the layer below. The value for a cell is the hydraulic conductivity divided by thickness for the material between the node in that cell and the node in the cell below. Because there is not a layer beneath the bottom layer, Vcont cannot be specified for the bottom layer.

Sf2—is the secondary storage coefficient. Read only for layers where LAYCON is 2 or 3 and only if there are one or more transient stress periods specified in the Discretization File. The secondary storage coefficient is always specific yield.

WETDRY—is a combination of the wetting threshold and a flag to indicate which neighboring cells can cause a cell to become wet.
- If WETDRY < 0, only the cell below a dry cell can cause the cell to become wet.

- If WETDRY > 0, the cell below a dry cell and the four horizontally adjacent cells can cause a cell to become wet.
- If WETDRY is 0, the cell cannot be wetted.

The absolute value of WETDRY is the wetting threshold. When the sum of BOT and the absolute value of WETDRY at a dry cell is equaled or exceeded by the head at an adjacent cell, the cell is wetted. Read only if LAYCON is 1 or 3 and IWDFLG is not 0.

Layer-Property Flow (LPF) Package

Input for the Layer-Property Flow (LPF) package is read from the file that is type "LPF" in the name file. Free format is used for reading all values. The LPF package is an alternative to the BCF package. Both packages should not be used simultaneously (Harbaughet al., 2000).

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. ILPFCB HDRY NPLPF

2. LAYTYP(NLAY)

3. LAYAVG(NLAY)

4. CHANI(NLAY)

5. LAYVKA(NLAY)

6. LAYWET(NLAY)

7. [WETFCT IWETIT IHDWET]

(Include item 7 only if LAYWET indicates at least one wettable layer.)

8. [PARNAM PARTYP Parval NCLU]

9. [Layer Mltarr Zonarr IZ]

Each repetition of Item 9 is called a parameter cluster. Repeat Item 9 NCLU times. Repeat Items 8-9 for each parameter to be defined (that is, NPLPF times).

A subset of the following two-dimensional variables is used to describe each layer. All the variables that apply to layer 1 are read first, followed by layer 2, followed by layer 3, and so forth. If a variable is not required due to simulation options (for example, Ss and Sy for a completely steady-state simulation), then it must be omitted from the input file. These variables are either read by the array-reading utility module, U2DREL, or they are defined through parameters. If a variable is defined through parameters, then the variable itself is not read; however, a single record containing a print code is read in place of the array control record. The print code determines the format for printing the values of the variable as defined by parameters. The print codes are the same as those used in an array control record. If any parameters of a given type are used, parameters must be used to define corresponding variable for all layers in the model.

10. HK(NCOL,NROW)

If there are any HK parameters, read only a print code

11. [HANI(NCOL,NROW)]

Include item 11 only if CHANI is less than or equal to 0. If there are any HANI parameters, read only a print code.

12. VKA(NCOL,NROW)

If there are any VK or VANI parameters, read only a print code.

13. [Ss(NCOL,NROW)]

Include item 13 only if at least one stress period is transient. If there are any SS parameters, read only a print code.

14. [Sy(NCOL,NROW)]

Include item 14 only if at least one stress period is transient and LAYTYP is not 0. If there are any SY parameters, read only a print code.

15. [VKCB(NCOL,NROW)]

Include item 15 only if LAYCBD (in the Discretization File) is not 0. If there are any VKCB parameters, read only a print code.

16. [WETDRY(NCOL,NROW)]

Include item 16 only if LAYWET is not 0 and LAYTYP is not 0.

EXPLANATION OF VARIABLES READ BY THE LPF PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

ILPFCB—is a flag and a unit number.
- If ILPFCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. The terms that are saved are storage, constant-head flow, and flow between adjacent cells.
- If ILPFBC = 0, cell-by-cell flow terms will not be written.
- If ILPFCB < 0, cell-by-cell flow for constant-head cells will be written in the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. Cell-by-cell flow to storage and between adjacent cells will not be written to any file.

HDRY—is the head that is assigned to cells that are converted to dry during a simulation. Although this value plays no role in the model calculations, it is useful as an indicator when looking at the resulting

heads that are output from the model. HDRY is thus similar to HNOFLO in the Basic package, which is the value assigned to cells that are no-flow cells at the start of a model simulation.

NPLPF—is the number of LPF parameters.

LAYTYP—contains a flag for each layer that specifies the layer type. Use as many records as needed to enter a value for each layer. 0—confined not 0—convertible.

LAYAVG—contains a flag for each layer that defines the method of calculating interblock transmissivity. Use as many records as needed to enter a value for each layer.
      0—harmonic mean
      1—logarithmic mean
      2—arithmetic mean of saturated thickness and logarithmic-mean hydraulic conductivity.

CHANI—contains a value for each layer that is a flag or the horizontal anisotropy. If CHANI is less than or equal to 0, then variable HANI defines horizontal anisotropy. If CHANI is greater than 0, then CHANI is the horizontal anisotropy for the entire layer, and HANI is not read. If any HANI parameters are used, CHANI for all layers must be less than or equal to 0. Use as many records as needed to enter a value of CHANI for each layer.

LAYVKA—contains a flag for each layer that indicates whether variable VKA is vertical hydraulic conductivity or the ratio of horizontal to vertical hydraulic conductivity. Use as many records as needed to enter a value for each layer.
- Zero (0)—indicates VKA is vertical hydraulic conductivity.
- Non-zero—indicates VKA is the ratio of horizontal to vertical hydraulic conductivity, where the horizontal hydraulic conductivity is specified as HK in item 10.

LAYWET—contains a flag for each layer that indicates if wetting is active. Use as many records as needed to enter a value for each layer.
- Zero (0)—indicates wetting is inactive
- Non-zero—indicates wetting is active

WETFCT—is a factor that is included in the calculation of the head that is initially established at a cell when it is converted from dry to wet. (See IHDWET.)

IWETIT—is the iteration interval for attempting to wet cells. Wetting is attempted every IWETIT iteration. If using the PCG solver (Hill, 1990), this applies to outer iterations, not inner iterations. If IWETIT $\leq$ 0, it is changed to 1.

IHDWET—is a flag that determines which equation is used to define the initial head at cells that become wet:
- If IHDWET = 0, equation 33A is used: h = BOT + WETFCT (h_n - BOT).
- If IHDWET is not 0, equation 33B is used: h = BOT + THRESH(WETDRY) .

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the LPF package, the allowed parameter types are:

HK—defines variable HK, horizontal hydraulic conductivity

HANI—defines variable HANI, horizontal anisotropy

VK—defines variable VKA for layers for which VKA represents vertical hydraulic conductivity (LAYVKA=0)

VANI—defines variable VKA for layers for which VKA represents vertical anisotropy (LAYVKA≠0)

SS—defines variable Ss, the specific storage

SY—defines variable Sy, the specific yield

VKCB—defines variable VKCB, the vertical hydraulic conductivity of a Quasi-three-dimensional confining layer.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NCLU—is the number of clusters required to define the parameter. Each repetition of Item 9 is a cluster (variables Layer, Mltarr, Zonarr, and IZ). There is usually only one cluster for each layer that is associated with a parameter. For example, if the parameter applies to cells in a single layer, there will generally be just one cluster. However, it is acceptable to have more than one cluster for the same layer.

Layer—is the layer number to which a cluster definition applies.

Mltarr—is the name of the multiplier array to be used to define variable values that are associated with a parameter. The name "NONE" means that there is no multiplier array, and the variable values will be set equal to Parval.

Zonarr—is the name of the zone array to be used to define the cells that are associated with a parameter. The name "ALL" means that there is no zone array, and all cells in the specified layer are part of the parameter.

IZ—is up to 10 zone numbers (separated by spaces) that define the cells that are associated with a parameter. These values are not used if ZONARR is specified as "ALL". Values can be positive or negative, but 0 is not allowed. The end of the line, a zero value, or a non-numeric entry terminates the list of values.

HK—is the hydraulic conductivity along rows. HK is multiplied by horizontal anisotropy (see CHANI and HANI) to obtain hydraulic conductivity along columns.

HANI—is the ratio of hydraulic conductivity along columns to hydraulic conductivity along rows, where HK of item 10 specifies the hydraulic conductivity along rows. Thus, the hydraulic conductivity along columns is the product of the values in HK and HANI. Read only if CHANI ≤ 0.

VKA—is either vertical hydraulic conductivity or the ratio of horizontal to vertical hydraulic conductivity depending on the value of LAYVKA. If LAYVKA is 0, VKA is vertical hydraulic conductivity. If LAYVKA is not 0, VKA is the ratio of horizontal to vertical hydraulic conductivity. In this case, HK is divided by VKA to obtain vertical hydraulic conductivity, and values of VKA typically are greater than or equal to 1.0.

Ss—is specific storage. Read only for a transient simulation (at least one transient stress period).

Sy—is specific yield. Read only for a transient simulation (at least one transient stress period) and if the layer is convertible (LAYTYP is not 0).

VKCB—is the vertical hydraulic conductivity of a Quasi-three-dimensional confining bed below a layer. Read only if there is a confining bed. Because there cannot be a confining bed beneath the bottom layer, VKCB cannot be specified for the bottom layer.

WETDRY—is a combination of the wetting threshold and a flag to indicate which neighboring cells can cause a cell to become wet.
- If WETDRY < 0, only the cell below a dry cell can cause the cell to become wet.
- If WETDRY > 0, the cell below a dry cell and the four horizontally adjacent cells can cause a cell to become wet.
- If WETDRY is 0, the cell cannot be wetted. The absolute value of WETDRY is the wetting threshold. When the sum of BOT and the absolute value of WETDRY at a dry cell is equaled or exceeded by the head at an adjacent cell, the cell is wetted. Read only if LAYTYP is not 0 and LAYWET is not 0.

Horizontal Flow Barrier (HFB) Package

Input for the Horizontal Flow Barrier (HFB) package is read from the file that has file type "HFB6" in the name file. The file type has been changed from that used in MODFLOW-96, which was "HFB", because the input data are not compatible. All variables are read in free format (Hsieh and Freckleton, 1993).

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. NPHFB   MXFB   NHFBNP

2. PARNAM   PARTYP   PARVAL   NLST

3. Layer   IROW1 ICOL1   IROW2   ICOL2   Factor

NLST repetitions of Item 3 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Factor).

Repeat Items 2 and 3 for each NPHFB parameter. Items 2 and 3 are not read if NPHFB is negative or zero.

FOR EACH STRESS PERIOD

4. Layer   IROW1   ICOL1   IROW2   ICOL2   HYDCHR

NHFBNP repetitions of Item 4 are read. Item 4 is not read if NHFBNP is negative or zero.

5. NACTHFB

6. PNAME

NACTHFB repetitions of Item 6 are read. Item 6 is not read if NACTHFB is negative or zero.

EXPLANATION OF VARIABLES READ BY THE HFB PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPHFB—is the number of horizontal-flow barrier parameters to be defined in Items 2 and 3. Note that for an HFB parameter to have an effect in the simulation, it must be defined in Items 2 and 3 and made active using Item 6.

MXFB—is the maximum number of HFB barriers that will be defined using parameters.

NHFBNP—is the number of HFB barriers not defined by parameters.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter. For the HFB package, the only allowed parameter type is HFB, which defines values of the hydraulic characteristic of the barrier.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NLST—is the number of horizontal-flow barrier cells included in the parameter.

Layer—is the number of the model layer in which the horizontal-flow barrier is located.

IROW1—is the row number of the cell on one side of the horizontal-flow barrier.

ICOL1—is the column number of the cell on one side of the horizontal-flow barrier.

IROW2—is the row number of the cell on the other side of the horizontal-flow barrier.

ICOL2—is the column number of the cell on the other side of the horizontal-flow barrier.

Factor—is the factor used to calculate hydraulic characteristic from the parameter value. The hydraulic characteristic is the product of Factor and the parameter value.

Hydchr—is the hydraulic characteristic of the horizontal-flow barrier. The hydraulic characteristic is the barrier hydraulic conductivity divided by the width of the horizontal-flow barrier.

NACTHFB—is the number of active HFB parameters.

Pname—is the name of a parameter to be used in the simulation. NACTHFB parameter names will be read.

Utility Generation (UGEN) Package

Input for the Generation process (UGEN) package is read from the file that is type "UGEN" in the name file. Free format is used for reading all values. The UGEN package is an alternative to the regular input for several MODFLOW packages, a list is shown in **Table** 4. Both input types should not be used simultaneously.

The UGEN package has three files associated with the input in addition to the MODFLOW package files using UGEN (Restrepo et al., 2003, 2001): The first one, the UGEN control file, is the generation utility input file. It defines the number of stations for stages/flows, the unit input file for stages/flows, and the flags that activate UGEN for any MODFLOW package with dynamic input data. This file is used in the MODFLOW's <name file>.

The other two associated files are the observation files. They contain stage and flow values for each stress period for all their respective station names. The first record of each file contains the location name. The following records contain the values of each stress period for all the stations during the whole simulation. If data for both flows and stages are being used, a separate file must be created for each case. If data for only one case (i.e. stage/flows) are used, just one observation file is necessary. Note: A MODFLOW package cannot be run using UGEN and standard MODFLOW input format for a given package simultaneously.

Table 4 MODFLOW packages compatible with UGEN

HBXY	Interpolation Head Boundary package
DRN	Drain package
RIV	River package
GHB	General-Head Boundary package
WEL	Well package
WELT	Well Trigger package
MDIV	Diversion package
RDF	Reinjected Drain Flow package
CHD	Constant-Head package
UCHD	CHD package using UGEN and HBXY
OBSG	Observation package

MODFLOW Package Input Files Using the UGEN

Input files contain one record for each cell/reach. The record defines cell location, station name, and associated parameters. Unlike the original version of MODFLOW, stage/flow values are not specified within those files. The station name points to the stage/flow values for every stress period in the observation file.

The modified MODFLOW package files include stress period-related MODFLOW input files, such as RIV, DRN, WEL, GHB, CHD, MDIV and RDF. These files contain the same information that MODFLOW requires. However, instead of having an input field for stage or flow values, the associated location (station) name must be provided with a length of no more than 21 characters. The MODFLOW input packages may be constant during all simulations or may be updated if the static information is changed.

During a MODFLOW simulation, UGEN begins each stress period by reading observation values (stages, flows). The stage/flow station name in the MODFLOW package input file using the UGEN is used to replace (in memory) the numerical value associated with the station name. The program "points out" the

same station name in the observation file and assigns the observation value to each station name. Then, MODFLOW finds the solution for the stress period. Then, a new stress period is started.

An example of the name file using UGEN is shown in **Table** 5. This example reads the names of files being used by UGEN. It includes the UGEN file, the MODFLOW package input files using UGEN, and the associated observation stage file.

Table 5 Sample input for a name file using UGEN

Type	Unit	Name	Description
LIST	6	Test.LST	
BAS	1	Test.BAS	
BCF	11	Test.BCF	
DRN	13	Test.DRN	MODFLOW package using UGEN
RIV	14	Test.RIV	MODFLOW package using UGEN
SIP	31	Test.SIP	
UGEN	27	Test.GEN	Activates the UGEN (UGEN file)
DATA	21	Stage.dat	Stages for RIV and DRN packages

UGEN file

The UGEN input file requires of two main sets of input. The first one specifies the number of stations, input units, and display options. The second one consists of the flags that indicate which MODFLOW packages will be used with UGEN during the simulation.

The UGEN includes an option to "skip" records in the observation file time series, i.e. if the current time series has daily records and the user wants to run a weekly simulation there is no need to change the original time series. There is also a flag in the UGEN that tells the program to skip a given amount of records so the program will run on a weekly basis. Nevertheless, the user has to be aware of matching the new number of stress periods and stress period-lengths in the DIS and BTN files. Stage data is not averaged, just skipped and the last value is used. Flow data is summed and averaged, taking into account the skipped values.

The UGEN allows the user to have flexibility with the data set up. However, if you want to use the CHD package, the CHD input requires the heads at the beginning and end of the stress period. In the new modification of the CHD, the program uses the head at the end of previous stress period as the initial head for the current stress period and the user must provide the value at the end of the stress period in the UGEN package stage file. For the River, Drain and General Head packages, this assumption (using the head at the end of stress period) could be an issue and can be solved by creating an additional array to keep the initial value and do the interpolation as in the CHD package for each stress period. Or the user still has the option in UGEN to not use the skip variable and provide the average value; however, in this case, it would not be reasonable to use the CHD with the average option.

FOR EACH SIMULATION

Read number of stations, input units, and display options:

1) nstat_st, nstat_fl, in_st, in_fl, irpflg_gen, kper_cycl

Read flags that indicate which MODFLOW packages will be used with UGEN:

2) igen_welon, igen_drnon, igen_rivon, igen_ghbon, igen_divon, igen_dense, igen_rdfon, igen_chdon, igen_hbxyon

EXPLANATION OF VARIABLES READ BY THE UGEN FILE

nstat_st—is the number of stage stations.

nstat_fl—is the number of flow stations.

in_st—is the input unit for the stages station names and head values of each stress period.

in_st—is the input unit for the flow station names and flow values of each stress period.

irpflg_gen—is a flag that, if positive does not display an echo, if zero displays basic information, and if negative (not recommended in a large model) displays full information about cells and stages.

kper_cycl—is the skip interval (measured in number of stress periods) period used to update the heads in all packages. The time unit of model data and the length of the stress period must be consistent for all data in the UGEN files. It is recommended that the UGEN observation values are provided daily.

igen_welon—is a flag that, if one (active) uses UGEN to handle the Well dynamic information, otherwise uses the unmodified MODFLOW input file.

igen_drnon—is a flag that if one (active) uses UGEN to handle the Drain dynamic information, otherwise uses the unmodified MODFLOW input file.

igen_rivon—is a flag that, if one (active) uses UGEN to handle the River dynamic information otherwise uses the unmodified MODFLOW input file.

igen_ghbon—is a flag that, if one (active) uses UGEN to handle the General Head Boundary dynamic information, otherwise uses the unmodified MODFLOW input file.

igen_divon—is a flag that, if one (active) uses the UGEN package to handle the diversion dynamic information, otherwise uses the unmodified MODFLOW input file.

igen_dense—is a flag that, if one (active) uses the UGEN package to handle a piezometric head correction due to salt-water interface.

igen_rdfon—is a flag that, if one (active) uses the UGEN package to handle the Reinjection DrainFlow dynamic information, otherwise uses the unmodified MODFLOW input file.

igen_chdon—is a flag that, if one (active) uses UGEN to handle the Constant Head Boundary dynamic information, otherwise uses the unmodified MODFLOW input file.

igen_hbxyon—is a flag that, if one (active) uses UGEN to handle the interpolation package for Constant Head Boundary or General Head Boundary dynamic information.

A sample input for the UGEN file is shown in **Table** 6.

Table 6. Sample input for the UGEN file

80	0	21	0	1					
0	1	1	0	0	0	0	0	0	0

In the example shown in **Table** 6, the simulation is interpreted as having eighty stage stations, having zero flow stations, assigning computer input unit number 21 to stage station names, not assigning any computer unit to flow station names, and not displaying any echoes. Note: the value of 21 assigned in the example to the variable in_st could be consistently replaced by any other positive integer as well.

According to the same sample file, the drain (igen_drnon), and the river (igen_rivon) input flags are set to one; therefore, changes should be made in the original MODFLOW input files for Drain and River packages based on the information provided below in this documentation.

Observation files: Stage file and Flow file

Any MODFLOW input file associated with the UGEN contains records for each cell/reach with a given property. Specifically, a record defines a cell location, station name, and associated parameters. Unlike the original version of MODFLOW, stage/flow values are not specified within those files. The station name points to the stage/flow values for every stress period in the observation files. The observation file and type of dynamic information associated with each MODFLOW package are shown in Table 7.

Table 7. Observation file and dynamic information associated with each MODFLOW package

MODFLOW package using the UGEN	Observation file	Dynamic information
DRN	Stage	Elevation of the drain
RIV	Stage	Head in the river
GHB	Stage	Head on the boundary
WEL	Flow	Pumping rate
DIV at a pumping station	Flow	Diversion flow
RDF	Stage	Weir elevation
CHD	Stages	Head on the boundary
HBXY	Stages	Head at stations

The observation files have the following structure (see Table 8):
- The first column is an ID.
- The second to fourth columns contain the date of data (year, month and day), and
- The subsequent columns list (horizontally) the stages or flow values for every station name during the whole simulation.

Table 8. Sample input for stages observation file

ID	Year	Month	Day	S-197-H	S-176-T
1	1988	1	31	4.007	2.999
2	1988	2	60	3.998	2.999
3	1988	3	91	3.990	3.000

The location name must match the one contained in the MODFLOW package that uses UGEN.

River (RIV) Package

Input to the River (RIV) package is read from the file that has file type "RIV" in the name file. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format. The River package can be used either with standard MODFLOW (Harbaughet al., 2000) or with the Utility Generation package (UGEN).

Using Standard MODFLOW

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPRIV MXL]

This optional item must start with the word "PARAMETER".

2. MXACTR IRIVCB [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer Row Column Stage Condfact Rbot [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Condfact). Repeat Items 3 and 4 for each NPRIV parameter.

FOR EACH STRESS PERIOD

5. ITMP NP

6. Layer Row Column Stage Cond Rbot [xyz]

ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Cond.) Item 6 is not read if ITMP is negative or 0.

7. [Pname]

(Item 7 is repeated NP times. It is not read if NP is negative or 0.)

Using the UGEN Package

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPRIV MXL]

This optional item must start with the word "PARAMETER".

2. MXACTR IRIVCB [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer Row Column Station_name Condfact Rbot [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Condfact). Repeat Items 3 and 4 for each NPRIV parameter. **Note: to ensure the adequate function of this package, an additional blank space must be entered between the fourth and fifth fields (i.e. Station_name and Condfact) when Using the UGEN package**.

FOR EACH STRESS PERIOD

5. ITMP NP

6. Layer Row Column Cond Rbot Station_name [xyz]

ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Cond.) Item 6 is not read if ITMP is negative or 0.

7. [Pname]

(Item 7 is repeated NP times. It is not read if NP is negative or 0.)

The stage data for the variable Station_name must be provided in the Observation Stage file. The flag for the River package must be active in the UGEN file. If the variable nstat_st (i.e. the number of stage stations) is changed, it is important to update its value in the UGEN file to avoid any related errors.

EXPLANATION OF VARIABLES READ BY THE RIV PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPRIV—is the number of river parameters.

MXL—is the maximum number of river reaches that will be defined using parameters.

MXACTR—is the maximum number of river reaches in use during any stress period, including those that are defined using parameters.

IRIVCB—is a flag and a unit number.
- If IRIVCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.
- If IRIVCB = 0, cell-by-cell flow terms will not be written.
- If IRIVCB < 0, river leakage for each reach will be written to the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

Option—is an optional list of character values.
- "AUXILIARY abc" or "AUX abc"—defines an auxiliary variable, named "abc", which will be read for each river reach as part of Items 4 and 6. Up to five variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." The auxiliary variable values will be read after the Rbot variable.
- "CBCALLOCATE" or "CBC"—indicates that memory should be allocated to store cell-by- cell flow for each river reach in order to make these flows available for use in other packages.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter. For the RIV package, the only allowed parameter type is RIV, which defines values of riverbed hydraulic conductance.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NLST—is the number of river reaches that are included in the parameter.

ITMP—is a flag and a counter. If ITMP < 0, non-parameter river data from the last stress period will be reused. If ITMP ≥ 0, ITMP will be the number of non-parameter reaches read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Layer—is the layer number of the cell containing the river reach.

Row—is the row number of the cell containing the river reach.

Column—is the column number of the cell containing the river reach.

Stage—is the head in the river.

Station_name—is the name or reference assigned to a particular station. This variable is associated with a set of data in the UGEN observation file(s).

Condfact—is the factor used to calculate riverbed hydraulic conductance from the parameter value. The conductance is the product of Condfact and the parameter value.

Cond—is the riverbed hydraulic conductance.

Rbot—is the elevation of the bottom of the riverbed.

[xyz]—represents any auxiliary variables for a river reach that have been defined in Item 2. The auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in Item 2.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

Interpolation Head Boundary (HBXY) Package

The Interpolation Head boundary package (HBXY) was developed by FAU (Restrepo, 2007) as a tool to interpolate time-related MODFLOW input during model execution. Although this package has not been fully implemented yet, FAU is in the process of updating this utility. HBXY interpolates the heads in the input files "on the fly" by linking the static input parameters, the physical location (X, Y), with the dynamic temporal data, much like a relational database. The static and dynamic data are linked by "location name" identifiers.

The HBXY package can be used to generate the model initial conditions and to generate the piezometric heads for the following MODFLOW-2000 packages: General Head Boundary (GHB) and Constant Head Boundary (CHD).

The HBXY package must be used with the UGEN package. If HBXY is not activated, MODFLOW is equally capable of running the input files constructed in the original form. Generating the input with HBXY increases the functionality of MODFLOW by reducing the pre-processing. HBXY is compatible with the modular structure and programming language (FORTRAN) of MODFLOW-2000.

Using the UGEN Package

FOR EACH SIMULATION

Read maximum number of observation points at any stress period and flag for echo:

1. MXUCHD, irpflg_hbxy

Read flags to active HBXY for initial piezometric conditions, GHB, and CHD packages

2. ihbxy_hzr, ihbxy_ghb, ihbxy_chd

Read bottom left corner (x,y) and rotation angle around the bottom left corner of the grid system, θ. The angle is assumed positive in the clockwise direction:

3. xcoo_bot_lft, ycoo_bot_lft, ang_rot_to_NS

Read variable to define the procedure to compute the initial head array for a layer in the system, if ihbxy_hzr is activated:

4. ly_hbxy_hzr(1), ly_hbxy_hzr(2), ly_hbxy_hzr(3), … , ly_hbxy_hzr(nlay).

Read variable to define the observation wells used to compute head within top and bottom layers for the corresponding layer in the system

5. Top layer: ly_hbxy_lys(1,1), ly_hbxy_lys(1,2), ….,ly_hbxy_lys(1, nlay)

6. Bottom Layer: ly_hbxy_lys(2,1), ly_hbxy_lys(2,2), ….,ly_hbxy_lys(2, nlay)

FOR EACH STRESS PERIOD

Read actual number of observation head points in any layer. For missing observation values, a value equal to -999.99 should be used in order to skip the interpolation (or for the initial condition if ihbxy_hzr is not equal to zero) :

7. ITMP

For each observation point read location (layer, x and y coordinates), station name, and the maximum radius of influence:

8. Layer_t, Layer_b, xcoo_pnt , ycoo_pnt , Station_name, rad_val, type

EXPLANATION OF VARIABLES READ BY THE HBXY FILE:

MXUCHD—is the maximum number of x and y points associated with stage observations active in any stress period.

irpflg_hbxy —is a flag that,
- irpflg_hbxy=0, limited echo for all hbxy variables
- irpflg_hbxy=1, no echo at all for hbxy variables
- irpflg_hbxy=-1, full echo for all hbxy variables

ihbxy_hzr—is a flag that, if one (active) uses UGEN to handle dynamic information and the HBXY for the interpolation procedure for the initial piezometric conditions, otherwise uses the unmodified MODFLOW input file.

ihbxy_ghb—is a flag that, if one (active) uses UGEN to handle the General Head Boundary dynamic information and the HBXY for the interpolation procedure, otherwise use the unmodified MODFLOW input file or just the UGEN.

ihbxy_chd—is a flag that, if one (active) uses UGEN to handle the Constant Head Boundary dynamic information and the HBXY for the interpolation procedure, otherwise uses the unmodified MODFLOW input file or just the UGEN.

xcoo_tp_lft —is the x coordinate of the most bottom left point within the grid domain.

ycoo_tp_lft—is the y coordinate of the most bottom left point within the grid domain.

ang_rot_to_NS—is the rotation angle around the bottom left corner of the system; the angle is assumed to be positive in the clockwise direction.

ly_hbxy_hzr(i)—this variable defines the procedure to compute the starting head array for a layer in the system. This variable is used only if ihbxy_hzr is equal to 1. Any layer in between the top and bottom layers may be computed with the average of the layer heads just above and below in order to define the array (e.g., in case of lack of observation values). In this case, ly_hbxy_hzr(i) should be equal to1.

- ly_hbxy_hzr(i)=0, if the head array read in the standard MODFLOW input is to be used for a specific layer
- ly_hbxy_hzr(i)=1, if the interpolation of the layer is going to be based on the layers above and below
- ly_hbxy_hzr(i)=-1, if the interpolation of the layer is going to be based on given points for that specific layer

ly_hbxy_lys(1,i)—Read variable to define the top layer range that may be used to identify observation wells that can be used to compute head for the corresponding layer in the system. This variable is used only if used only if both ihbxy_hzr and ly_hbxy_hzr(i) are equal to one.

ly_hbxy_lys(2,i)— Read variable to define the bottom layer range that may be used to identify observation wells that can be used to compute head for the corresponding layer in the system. This variable is used only if used only if both ihbxy_hzr and ly_hbxy_hzr(i) are equal to one.

ITMP—is the number of xy points associated with stage observations active in any stress period.

Layer_t—this variable identifies the top layer number of the location of the observation value.

Layer_b— this variable identifies the bottom layer number of the location of the observation value.

xcoo_pnt—is the x coordinate of the observation point

ycoo_pnt—is the y coordinate of the observation point

Station_name—is the station name of the observation point at the start of the stress period.

rad_val—value of radius of influence for the head observation point in the interpolation process

Type—is the type of observation station that is used to define the minimum radius of influence for that value.
- Type=TID, for tidal
- Type=CAN, for canal
- Type=MON, for monitoring
- Type=STR, for structure
- Type=OTH for others

Example of the HBXY Input File in the Lower West Coast Floridan Aquifer System model.

Table 9 shows the HBXY input for the Lower West Coast Floridan Aquifer System Model. Each record on the file is described below.

- Record 1: The simulation is interpreted as having 600 maximum number of xy points associated with stage observations active in any stress period, and not displaying any echoes.

- Record 2, field 1: The initial head (ihbxy_hzr) input flag is set to zero. That means, no interpolation procedure is done for the initial piezometric conditions. In other words, starting head array is provided in the model as is required in the original MODFLOW input file.
- Record 2, fields 2 and 3: The general head boundary (ihbxy_ghb), and the constant head boundary (ihbxy_chd ghb), input flags are set to one; therefore, changes should be made in the original MODFLOW input files for General Head Boundary and Constant Head Boundary packages based on the information provided below in this documentation.
- Record 3: The third record specified the x and y coordinate of the most bottom left point within the grid domain and rotation angle around the bottom left corner of the system, the angle is assumed positive in the clockwise direction (currently not used in the hbxy package).
- Record 4: define the procedure to compute the initial head cell values for each layer in the system if flag ihbxy_hzr is active (record 2, field1). Since ihbxy_hzr is inactive in the LWCFAS model, then this record can be ignored.
- Record 5 and 6: define the top and bottom layer range that may be used to identify observation wells that can be used to compute head for the corresponding layer in the system. This is only active if flag ihbxy_hzr is active and ihbxy_hzr(layer) (record 4) is 1. Since ihbxy_hzr is inactive in the LWCFAS model, then these records can be ignored.
- Record 7: The simulation is interpreted as having 589 number of x and y points associated with stage observations active in any stress period.
- Records 8 to 12: This record defines an observation point location, x and y coordinates, defined by a structure that can be applied from layers 1 to 2. A radius of influence for the head observation point of 7,200 feet is used in the interpolation process (see next slide for more details). Note that x and y corresponds to a "fake" point un-rotated in a vertical plane. Therefore, x and y coordinates are not the actual x and y values. Any grid resolution model (e.g., 3,000 ft, 12,000ft) can work with this file. Observation points used in stress period 1 are the same for the rest of stress period.

Table 9. HBXY input for the LWCFAS Model.

Record/field												
1	600	1										
2	0	1	1									
3	585000	190000	330									

Layer/Record	1	2	3	4	5	6	7	8	9	10	11	12
4	-1	1	-1	1	-1	1	-1	1	1	-1	1	1
5	0	1	0	3	0	5	0	7	7	0	10	10
6	0	3	0	5	0	7	0	10	10	0	10	12

Record/Field	Layert_t	Layer_b	xcoo_pnt	ycoo_pnt	Station_name_1	rad_val	type
7	589	!Stress Period 1					
8	1	2	707985	818806	6MI.10	72000	STR
9	1	2	738400	835508	6MI.I75	72000	STR
10	1	2	714452	722944	951EXT	72000	STR
11	1	2	749038	561636	BARRON	72000	STR
12	1	2	853963	531331	BCA15	72000	STR
...	…						
	-1	!Stress Period 2					
	-1	!Stress Period 3					

Recharge (RCH) Package

Input to the Recharge (RCH) package (Harbaugh et al., 2000) is read from the file that has type "RCH" in the name file. All single-valued variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the variables have 10-character fields.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPRCH]

This optional item must start with the word "PARAMETER".

2. NRCHOP IRCHCB

3. [PARNAM PARTYP Parval NCLU]

4. [Mltarr Zonarr IZ]

Repeat Item 4 NCLU times. Each repetition of Item 4 is called a parameter cluster. Repeat Items 3 and 4 for each parameter to be defined (that is, NPRCH times).

FOR EACH STRESS PERIOD

5. INRECH INIRCH

6. [RECH(NCOL,NROW)] -- U2DREL if NPRCH=0 and if INRECH ≥ 0

7. [Pname [IRCHPF]] -- if NPRCH > 0 and if INRECH > 0

Either Item 6 or Item 7 may be read, but not both. If Item 7 is read, it is repeated INRECH times.

8. [IRCH(NCOL,NROW)] -- U2DINT If NRCHOP=2 and if INIRCH > = 0

EXPLANATION OF VARIABLES READ BY THE RCH PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPRCH—is the number of recharge parameters.

NRCHOP—is the recharge option code. Recharge fluxes are defined in a layer variable, RECH, with one value for each vertical column. Accordingly, recharge is applied to one cell in each vertical column, and the option code determines which cell in the column is selected for recharge.

> 1—Recharge is only to the top grid layer.
> 2—Vertical distribution of recharge is specified in layer variable IRCH.
> 3—Recharge is applied to the highest active cell in each vertical column. A constant-head node intercepts recharge and prevents deeper infiltration.

IRCHCB—is a flag and a unit number. If IRCHCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. If IRCHCB ≤ 0, cell-by-cell flow terms will not be written.

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the RCH package, the only allowed parameter type is RCH, which defines values of the recharge flux.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NCLU—is the number of clusters required to define a parameter. Each repetition of Item 4 is a cluster (variables Mltarr, Zonarr, and IZ). There is usually only one cluster used to define a RCH parameter; however, it is acceptable to have more than one cluster.

Mltarr—is the name of the multiplier array to be used to define cell values that are determined by a parameter. The name "NONE" means that there is no multiplier array, and the cell values will be set equal to Parval.

Zonarr—is the name of the zone array to be used to define the cells that are associated with a parameter. The name "ALL" means that there is no zone array, and all cells in the layer are associated with the parameter.

IZ—is up to 10 zone numbers (separated by spaces) that define the cells that are associated with a parameter. These values are not used if Zonarr is specified as "ALL." Values can be positive or negative, but 0 is not allowed. The end of the line, a zero value, or a non-numeric entry terminates the list of values.

INRECH—is the RECH read flag. Its function depends on whether or not parameters are being used. If no parameters are being used (NPRCH = 0): If INRECH ≥ 0, a layer variable of recharge fluxes, RECH, is read. If INRECH < 0, recharge rates from the preceding stress period are used. If parameters are being used (NPRCH > 0): If INRECH > 0, INRECH is the number of parameters that will be used to define RECH in the current stress period. Item 7 defines the names of the parameters. If INRECH < 0, recharge parameters from the preceding stress period are used. INRECH = 0 is not allowed. That is, when parameters are used, at least one parameter must be specified each stress period.

INIRCH—is the IRCH read flag, which is read only if NRCHOP is two:
- If INIRCH ≥ 0, a layer variable of layer numbers (IRCH) is read.
- If INIRCH < 0, the variable (IRCH) used in the preceding stress period is reused.

RECH—is the recharge flux (LT-1). Read only if INRECH is greater than or equal to zero and if NPRCH equals 0.

Pname—is the name of a parameter that will be used to define the RECH variable in the current stress period. Read INRECH values if NPRCH > 0 and INRECH > 0.

IRCHPF—is an optional format code for printing the RECH variable after it has been defined by parameters. The format codes are the same as those used in the U2DREL array reading utility module.

IRCH—is the layer number variable that defines the layer in each vertical column where recharge is applied. Read only if NRCHOP is two and if INIRCH is greater than or equal to zero.

Well (WEL) Package

Input to the Well (WEL) package (Harbaugh et al., 2000) is read from the file that has type "WEL" in the name file. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format. This package can also be used within the UGEN package. In addition, multiple type of wells can defined in the input as part of the description of each well.

Using Standard MODFLOW

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPWEL MXL]

This optional item must start with the word "PARAMETER".

2. MXACTW IWELCB [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer Row Column Qfact [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Qfact). Repeat Items 3 and 4 for each NPWEL parameter.

FOR EACH STRESS PERIOD

5. ITMP NP

6. Layer Row Column Q [xyz]

(ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Q.) Item 6 is not read if ITMP is negative or zero.

7. [Pname]

(Item 7 is repeated NP times. It is not read if NP is negative or 0.)

Using the UGEN Package

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPWEL MXL]

This optional item must start with the word "PARAMETER".

2. MXACTW IWELCB [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer   Row   Column   Station_name   [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Qfact). Repeat Items 3 and 4 for each NPWEL parameter.

FOR EACH STRESS PERIOD

5. ITMP NP

6. Layer Row Column Station_name [xyz]

(ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Q.) Item 6 is not read if ITMP is negative or zero.

7. [Pname]

(Item 7 is repeated NP times. It is not read if NP is negative or 0.)

EXPLANATION OF VARIABLES READ BY THE WEL PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPWEL—is the number of well parameters.

MXL—is the maximum number of wells that will be defined using parameters.

MXACTW—is the maximum number of wells in use during any stress period, including those that are defined using parameters.

IWELCB—is a flag and a unit number. If IWELCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. If IWELCB = 0, cell-by-cell flow terms will not be written. If IWELCB < 0, well recharge for each well will be written to the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

Option—is an optional list of character values.

- "AUXILIARY abc" or "AUX abc"—defines an auxiliary variable, named "abc", which will be read for each well as part of Items 4 and 6. Up to five variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." The auxiliary variable values will be read after the Q variable.
- "CBCALLOCATE" or "CBC"—indicates that memory should be allocated to store cell-by-cell flow for each well in order to make these flows available for use in other packages.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter. For the WEL package, the only allowed parameter type is Q, which defines values of the volumetric recharge rate.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NLST—is the number of wells that are included in the parameter.

ITMP—is a flag and a counter. If ITMP < 0, non-parameter well data from the last stress period will be reused. If ITMP ≥ 0, ITMP will be the number of non-parameter wells read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Layer—is the layer number of the model cell that contains the well.

Row—is the row number of the model cell that contains the well.

Column—is the column number of the model cell that contains the well.

Qfact—is the factor used to calculate well recharge rate from the parameter value. The recharge rate is the product of Qfact and the parameter value.

Q—is the volumetric recharge rate. A positive value indicates recharge and a negative value indicates discharge (pumping).

[xyz]—represents any auxiliary variables for a well that have been defined in Item 2. The auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in Item 2.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

Station_name—is the name or reference assigned to a particular station. This variable is associated with a set of data in the UGEN observation file(s).

Well Trigger/Multiple Wells (WELT) Package

This package is only available using the UGEN package. This means it cannot be used with standard MODFLOW. Input to the Well Trigger (WELT) package (SFWMD, 1999; Restrepo, 2008) is read from the file that has type "WELT" in the name file. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format. Note: The Well package and the Well Trigger package cannot be read simultaneously.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPWEL MXL]

This optional item must start with the word "PARAMETER".

2. NUSE_TYPE [Option_T]

3. MAXACTW_T(NUSE_TYPE)*

4. NTYPE_WELL(NUSE_TYPE)*

NUSE_TYPE repetitions of items 3 and for are required.

5. NT_OF_WELLS

6. Layer   Row Column Station_name Type

MXWELL_T repetitions of Item 6 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Qfact). Repeat Items 3 and 4 for each NTYPE_WELL parameter. The order of this list does not affect the outcome of the program. For convenience, it is strongly suggested that this information be entered by type in the respective order specified in the variable NTYPE_WELL.

* A record with all the maximum number of wells per category of well type use.

FOR EACH STRESS PERIOD

7. ITMP NP

8. Layer   Row Column Station_Name Type

(ITMP repetitions of Item 7 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Q.) Item 7 is not read if ITMP is negative or zero.

9. [Pname]

(Item 8 is repeated NP times. It is not read if NP is negative or 0.)

EXPLANATION OF VARIABLES READ BY THE WELT PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPWEL—is the number of well parameters.

MXL—is the maximum number of wells that will be defined using parameters.

NUSE_TYPE—is the number of different types of wells, according to water usage, employed in the model, i.e. public water supply, agriculture, industrial, etc.

OPTION_T—determines how the well locations are read.
        0—Read rows and columns.
        1—Read [x,y] coordinates.

XACTW_T—is the maximum number of wells per type in use, during any stress period, including those that are defined using parameters.

NT_OF_WELLS—is the total number of wells in any type use, during the current stress period.

NTYPE_WELL—is the numerical reference for the categories of the different well types, according to water use.
        1—ASR (Aquifer storage and recovery)
        2—PWS (Public water supply)
        3—LAN (Landscape)
        4—GOL (Golf course irrigation)
        5—NUR (Nursing)
        6—AGR-FL (Agriculture)
        7—AG1-OH (Horticulture)
        8—AG2-LV (Livestock)
        9—IND (Industrial)

IWELCB—is a flag and a unit number. If IWELCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. If IWELCB = 0, cell-by-cell flow terms will not be written. If IWELCB < 0, well recharge for each well will be written to the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

Option—is an optional list of character values.
- "AUXILIARY abc" or "AUX abc"—defines an auxiliary variable, named "abc", which will be read for each well as part of Items 4 and 6. Up to five variables can be specified, each of which

66

must be preceded by "AUXILIARY" or "AUX." The auxiliary variable values will be read after the Q variable.

- "CBCALLOCATE" or "CBC"—indicates that memory should be allocated to store cell-by-cell flow for each well in order to make these flows available for use in other packages.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter. For the WEL package, the only allowed parameter type is Q, which defines values of the volumetric recharge rate.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NLST—is the number of wells that are included in the parameter.

ITMP—is a flag and a counter. If ITMP < 0, non-parameter well data from the last stress period will be reused. If ITMP ≥ 0, ITMP will be the number of non-parameter wells read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Layer—is the layer number of the model cell that contains the well.

Row—is the row number of the model cell that contains the well.

Column—is the column number of the model cell that contains the well.

Station_name—is the name or reference assigned to a particular station. This variable is associated with a set of data in the UGEN observation file(s).

Type—is the category of water use for the well, as defined in NTYPE_WELL.

Qfact—is the factor used to calculate well recharge rate from the parameter value. The recharge rate is the product of Qfact and the parameter value.

Q—is the volumetric recharge rate. A positive value indicates recharge and a negative value indicates discharge (pumping).

[xyz]—represents any auxiliary variables for a well that have been defined in Item 2. The auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in Item 2.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

Drain (DRN) Package

Input to the Drain (DRN) package (Harbaugh et al., 2000) is read from the file that has type "DRN" in the name file. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

Using Standard MODFLOW

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPDRN MXL]

This optional item must start with the word "PARAMETER".

2. MXACTD IDRNCB [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer Row Column Elevation Condfact [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Condfact). Repeat Items 3 and 4 for each NPDRN Parameter.

FOR EACH STRESS PERIOD

5. ITMP NP

6. Layer Row Column Elevation Cond [xyz]

ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Cond.) Item 6 is not read if ITMP is negative or 0.

7. [Pname]

(Item 7 is repeated NP times. It is not read if NP is negative or 0.)

Using the UGEN Package

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPDRN MXL]

This optional item must start with the word "PARAMETER".

2. MXACTD IDRNCB [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer Row Column Elevation Condfact [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Condfact). Repeat Items 3 and 4 for each NPDRN Parameter. **Note:** to ensure the adequate function of this package, an additional blank space must be entered between the fourth and fifth fields (i.e. Elevation and Condfact) when Using the UGEN package.

FOR EACH STRESS PERIOD

5. ITMP   NP

6. Layer Row Column Station_name Cond [xyz]

ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Cond.) Item 6 is not read if ITMP is negative or 0.

7. [Pname]

(Item 7 is repeated NP times. It is not read if NP is negative or 0.)

The stages for the variable Station_name must be provided in the Observation stage file.

EXPLANATION OF VARIABLES READ BY THE DRN PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPDRN—is the number of drain parameters.

MXL—is the maximum number of drain cells that will be defined using parameters.

MXACTD—is the maximum number of drain cells in use during any stress period, including those that are defined using parameters.

IDRNCB—is a flag and a unit number.

- If IDRNCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.
- If IDRNCB = 0, cell-by-cell flow terms will not be written.
- If IDRNCB < 0, drain leakage for each drain cell will be written to the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

Option—is an optional list of character values.
- "AUXILIARY abc" or "AUX abc"—defines an auxiliary variable, named "abc", which will be read for each drain as part of Items 4 and 6. Up to five variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX."
- CBCALLOCATE or CBC—indicates that memory should be allocated to store cell-by-cell flow for each drain in order to make these flows available for use in other packages.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive. That is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter. For the DRN package, the only allowed parameter type is DRN, which defines values of the drain hydraulic conductance.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NLST—is the number of drain cells that are included in the parameter.

ITMP—is a flag and a counter. If ITMP < 0, non-parameter drain data from the last stress period will be reused. If ITMP ≥ 0, ITMP will be the number of non-parameter drains read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Layer—is the layer number of the cell containing the drain.

Row—is the row number of the cell containing the drain.

Column—is the column number of the cell containing the drain.

Elevation—is the elevation of the drain.

Condfact—is the factor used to calculate drain hydraulic conductance from the parameter value. The conductance is the product of Condfact and the parameter value.

Cond—is the hydraulic conductance of the interface between the aquifer and the drain.

[xyz]—represents any auxiliary variables for a drain that have been defined in Item 2. The auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in Item 2.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

Evapotranspiration (EVT) Package

Input to the Evapotranspiration (EVT) package (Harbaugh et al., 2000) is read from the file that is type "EVT" in the name file. All single-valued variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the variables have 10-character fields.

FOR EACH SIMULATION

   0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

   1. [PARAMETER NPEVT]

This optional item must start with the word "PARAMETER".

   2. NEVTOP IEVTCB

   3. [PARNAM PARTYP Parval NCLU]

   4. [Mltarr Zonarr IZ]

Repeat Item 4 NCLU times. Each repetition of Item 4 is called a parameter cluster. Repeat Items 3-4 for each parameter to be defined (that is, NPEVT times).

FOR EACH STRESS PERIOD

   5. INSURF INEVTR INEXDP INIEVT

   6. [SURF(NCOL,NROW)]U2DREL If INSURF ≥ 0

   7. [EVTR(NCOL,NROW)]---U2DREL If NPEVT = 0 and if INEVTR ≥ 0

   8. [Pname [IEVTPF]] -- if NPEVT > 0 and if INEVTR > 0

Either Item 7 or Item 8 may be read, but not both. If Item 8 is read, it is repeated INEVTR times.

   9. [EXDP(NCOL,NROW)] -- U2DREL If INEXDP ≥ 0

  10. [IEVT(NCOL,NROW)] -- U2DINT If NEVTOP = 2 and if INIEVT ≥ 0

EXPLANATION OF VARIABLES READ BY THE EVT PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPEVT—is the number of evapotranspiration parameters.

NEVTOP—is the evapotranspiration (ET) option code. ET variables (ET surface, maximum ET rate, and extinction depth) are specified in layer variables, SURF, EVTR, and EXDP, with one value for each vertical column. Accordingly, ET is calculated for one cell in each vertical column. The option codes determine the cell within a column for which ET will be calculated.

     1—ET is calculated only for cells in the top grid layer.

     2—The cell for each vertical column is specified by the user in variable IEVT.

IEVTCB—is a flag and a unit number.
- If IEVTCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.
- If IEVTCB ≤ 0, cell-by-cell flow terms will not be written.

PARNAM—is the name of a parameter to be defined. This name can consist of 1 to 10 characters and is not case sensitive; that is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the EVT package, the only allowed parameter type is EVT, which defines values of the maximum ET flux.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NCLU—is the number of clusters required to define a parameter. Each repetition of Item 4 is a cluster (variables Mltarr, Zonarr, and IZ). There is usually only one cluster used to define an EVT parameter; however, it is acceptable to have more than one cluster.

Mltarr—is the name of the multiplier array to be used to define the values that are determined by a parameter. The name "NONE" means that there is no multiplier array, and the values will be set equal to Parval.

Zonarr—is the name of the zone array to be used to define the cells that are associated with a parameter. The name "ALL" means that there is no zone array, and all cells are associated with the parameter.

IZ—is up to 10 zone numbers (separated by spaces) that define the cells that are associated with a parameter. These values are not used if Zonarr is specified as "ALL." Values can be positive or negative, but 0 is not allowed. The end of the line, a zero value, or a non-numeric entry terminates the list of values.

INSURF—is the ET surface (SURF) read flag.
- If INSURF ≥ 0, a layer variable containing the ET surface elevation (SURF) will be read.
- If INSURF < 0, the ET surface from the preceding stress period will be reused.

INEVTR—is the EVTR read flag. Its function depends on whether or not parameters are being used. If no parameters are being used (NPEVT=0):
- If INEVTR ≥ 0, a layer variable containing the maximum ET rate (EVTR) will be read.
- If INEVTR < 0, the maximum ET rate from the preceding stress period will be reused.
- If parameters are being used (NPEVT>0):

- If INEVTR > 0, INEVTR is the number of parameters that will be used to define EVTR in the current stress period. Item 8 defines the names of the parameters.
- If INEVTR < 0, EVT parameters from the preceding stress period are used.
- INEVTR = 0 is not allowed. That is, when parameters are used, at least one parameter must be specified each stress period

INEXDP—is the extinction depth (EXDP) read flag.
- If INEXDP ≥ 0, a layer variable containing the extinction depth (EXDP) will be read.
- If INEXDP < 0, the extinction depth from the preceding stress period will be reused.

INIEVT—is the layer indicator (IEVT) read flag. It is read only if the ET option (NEVTOP) is equal to two.
- If INIEVT ≥ 0, a layer variable containing the layer indicators (IEVT) will be read.
- If INIEVT < 0, layer indicators used during the preceding stress period will be reused.

SURF—is the elevation of the ET surface. This variable is read only if INSURF is greater than or equal to zero (0).

EVTR—is the maximum ET flux [volumetric flow rate per unit area (LT-1)]. This variable is read only if INEVTR is greater than or equal to zero (0) and if NPEVT equals zero (0).

Pname—is the name of a parameter that will be used to define the EVTR variable in the current stress period. Read INEVTR values if NPEVT is greater than zero (0) and INEVTR is greater than zero (0).

IEVTPF—is an optional format code for printing the EVTR variable after it has been defined by parameters. The format codes are the same as those used in the U2DREL array reading utility module.

EXDP—is the ET extinction depth. This variable is read only if INEXDP is greater than or equal zero (0).

IEVT—is the layer indicator variable. For each horizontal location, it indicates the layer from which ET is removed. It is read only if the ET option is equal to two and if INIEVT is greater than or equal to zero (0).

General-Head Boundary (GHB) Package

Input to the General-Head Boundary (GHB) package (Harbaugh et al., 2000) is read from the file that has file type "GHB" in the name file. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

Using Standard MODFLOW

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPGHB MXL]

This optional item must start with the word "PARAMETER".

2. MXACTB IGHBCB [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer Row Column Bhead Condfact [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Condfact). Repeat Items 3 and 4 for each NPGHB parameter.

FOR EACH STRESS PERIOD

5. ITMP NP

6. Layer Row Column Bhead Cond [xyz]

ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Cond.) Item 6 is not read if ITMP is negative or 0.

7. [Pname]

Item 7 is repeated NP times. It is not read if NP is negative or 0.

Using the UGEN Package

FOR EACH SIMULATION:

1. MXACTB,IGHBCB

FOR EACH STRESS PERIOD:

2. ITMP

3. Layer, Row, Column, Cond, Station_name

Using the UGEN and HBXY Packages

FOR EACH SIMULATION:

1. MXACTB, IGHBCB, IGPFLG

FOR EACH STRESS PERIOD:

2. ITMP

3. Layer, Row, Column, Cond*, Station_name, IFLAG

Note: to ensure the adequate function of this package, an additional blank space must be entered between the fourth and fifth fields (i.e. Cond* and Station_name) when Using the UGEN package.

EXPLANATION OF VARIABLES READ BY THE GHB PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPGHB—is the number of general-head boundary parameters.

MXL—is the maximum number of general-head-boundary cells that will be defined using parameters.

MXACTB—is the maximum number of general-head boundary cells in use during any stress period, including those that are defined using parameters.

IGHBCB—is a flag and a unit number.
- If IGHBCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.
- If IGHBCB = 0, cell-by-cell flow terms will not be written.
- If IGHBCB < 0, boundary leakage for each GHB cell will be written to the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

Option—is an optional list of character values.
- "AUXILIARY abc" or "AUX abc"—defines an auxiliary variable, named "abc", which will be read for each general-head boundary as part of Items 4 and 6. Up to five variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." The auxiliary variable values will be read after the Cond variable.

- "CBCALLOCATE" or "CBC"—indicates that memory should be allocated to store cell-by-cell flow for each general-head boundary to make these flows available for use in other packages.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive; that is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the GHB package, the only allowed parameter type is GHB, which defines values of the general-head boundary hydraulic conductance.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NLST—is the number of head-dependent boundary cells that are included in the parameter.

ITMP—is a flag and a counter. If ITMP < 0, non-parameter GHB data from the preceding stress period will be reused. If ITMP ≥ 0, ITMP is the number of non-parameter general-head boundaries read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Layer—is the layer number of the cell affected by the head-dependent boundary.

Row—is the row number of the cell affected by the head-dependent boundary.

Column—is the column number of the cell affected by the head-dependent boundary.

Bhead—is the head on the boundary.

Condfact—is the factor used to calculate hydraulic conductance from the parameter value. The conductance is the product of Condfact and the parameter value.

Cond—is the hydraulic conductance of the interface between the aquifer cell and the boundary.

[xyz]—represents any auxiliary variables for a boundary that have been defined in Item 2. The auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in item 2.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

IFLAG–is the flag option to interpolate or not interpolate.
- IFLAG=0, head for the prescribed head is defined by the GHB with UGEN package
- IFLAG>0, head for the general head is defined by the HBXY package by means of interpolation of a set observation values in space for a given time interval; "Station_name" variable is not used.

Kxy—is the layer number of the head value is taken from

Station_name—is the name or reference assigned to a particular station. This variable is associated with a set of data in the UGEN observation file(s).

Constant-Head Boundary (CHD) Package

Input to the Constant-Head Boundary (CHD) package (Leake and Lilly, 1997) is read from the file that has file type "CHD" in the name file. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

Using Standard MODFLOW

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. [PARAMETER NPCHD MXL]

This optional item must start with the word "PARAMETER".

2. MXACTC [Option]

3. [PARNAM PARTYP Parval NLST]

4. Layer   Row   Column   Shdfact   Ehdfact [xyz]

NLST repetitions of Item 4 are required; they are read by module ULSTRD. (SFAC of the ULSTRD utility module applies to Shdfact and Ehdfact). Repeat Items 3 and 4 for each NPGHB parameter.

FOR EACH STRESS PERIOD

5. ITMP NP

6. Layer Row Column Shead Ehead [xyz]

ITMP repetitions of Item 6 are read by module ULSTRD if ITMP > 0. (SFAC of the ULSTRD utility module applies to Shead and Ehead.) Item 6 is not read if ITMP is negative or 0.

7. [Pname]

(Item 7 is repeated NP times. It is not read if NP is negative or 0.)

Using the UGEN Package

FOR EACH SIMULATION:

1. MXACTC, ICHDCB

FOR EACH STRESS PERIOD:

2. ITMP

3. Layer, Row, Column, Station_name_1,Station_name_2

Using the UGEN and HBXY Packages

FOR EACH SIMULATION:

1. MXACTC, ICHDCB

FOR EACH STRESS PERIOD:

2. ITMP

3. Layer, Row, Column, Station_name_1, Station_name_2, IFLAG

EXPLANATION OF VARIABLES READ BY THE CHD PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

NPCHD—is the number of constant-head boundary parameters.

MXL—is the maximum number of constant-head-boundary cells that will be defined using parameters.

MXACTC—is the maximum number of constant-head boundary cells in use during any stress period, including those that are defined using parameters.

Option—is an optional list of character values.
- "AUXILIARY abc" or "AUX abc"—defines an auxiliary variable, named "abc", which will be read for each constant-head boundary as part of Items 4 and 6. Up to five variables can be specified, each of which must be preceded by "AUXILIARY" or "AUX." The auxiliary variable values will be read after the Ehead variable.

PARNAM—is the name of a parameter. This name can consist of 1 to 10 characters and is not case sensitive; that is, any combination of the same characters with different case will be equivalent.

PARTYP—is the type of parameter to be defined. For the CHD package, the only allowed parameter type is CHD, which defines values of the start and end head at the boundary.

Parval—is the parameter value. This parameter value may be overridden by a value in the Sensitivity process input file or by a value generated by the Parameter-Estimation process.

NLST—is the number of constant-head cells that are included in the parameter.

ITMP—is a flag and a counter.
- If ITMP < 0, non-parameter CHD data from the preceding stress period will be reused.
- If ITMP ≥ 0, ITMP is the number of non-parameter constant-head boundaries read for the current stress period.

NP—is the number of parameters in use in the current stress period.

Layer—is the layer number of the constant-head boundary.

Row—is the row number of the constant-head boundary.

Column—is the column number of the constant-head boundary.

Shdfact—is the factor used to calculate the head at the boundary at the start of the stress period from the parameter value. The head is the product of Shdfact and the parameter value.

Ehdfact—is the factor used to calculate the head at the boundary at the end of the stress period from the parameter value. The head is the product of Ehdfact and the parameter value.

Shead—is the head at the boundary at the start of the stress period.

Ehead—is the head at the boundary at the end of the stress period.

[xyz]—represents any auxiliary variables for a constant-head boundary that have been defined in Item 2. The auxiliary variables must be present in each repetition of Items 4 and 6 if they are defined in Item 2.

Pname—is the name of a parameter that is being used in the current stress period. NP parameter names will be read.

Station_name_1—Station name for head at the boundary at the start of the stress period.

Station name_2—Station name for head at the boundary at the end of the stress period.

IFLAG—is the flag option to interpolate or not interpolate.
- IFLAG = 0, the head for the constant head is defined by the CHD package.
- IFLAG > 0, the head for the constant head is defined by the HBXY package by means of interpolation of a set observation values in space for a given time interval. Currently, the head at the start and end of stress period is assumed the same in this CHD package.

Kxy—is the layer number of the head value is taken from.

Reinjected DrainFlow (RDF) Package

The Reinjected DrainFlow (RDF) package (McDonald Morrisey Associates, Inc., 1999) redirects the outflow from the drain package into a sink area provided by cells. This package was modified by FAU in 2007 in order to work with the UGEN and the rules of operation based on the head of the sink and source. Input to these packages is:

Using Standard MODFLOW

FOR EACH SIMULATION:

0. MXRDF, IRDFCB,

FOR EACH STRESS PERIOD:

1. ITMP

2. Layer, Row, Column, Cond, Layer, Row, Column, Head_1, Head_2

Using the UGEN Package

FOR EACH SIMULATION:

0. MXRDF,IRDFCB,

FOR EACH STRESS PERIOD:

1. ITMP

2. Layer[1], Row[1], Column[1], Cond, Layer[2], Row[2], Column[2], Station_name_1[3],Station_name_2[4]

    [1] Source location

    [2] Sink (destination) location

    [3] Station name of the source's draining level

    [4] Station name of the sink's minimum level. In the case that there is not a second station, a hypothetical station is set up with a high value, which will not allow the minimum to ever be reached

EXPLANATION OF VARIABLES READ BY THE RDF PACKAGE

MXRDF—is the maximum number of reinjection drain cells active in any stress period.

IRDFCB—is a flag and a unit number. If IRDFCB > 0, it is the unit number to which cell-by-cell flow terms will be written when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control. If IRDFCB = 0, cell-by-cell flow terms will not be written. If IRDFCB < 0, drain leakage for each drain cell will be written to the listing file when "SAVE BUDGET" or a non-zero value for ICBCFL is specified in Output Control.

Layer—is the layer number of the constant-head boundary.

Row—is the row number of the constant-head boundary.

Column—is the column number of the constant-head boundary.

Head_1—is the head at the boundary at the start of the stress period.

Head_2—is the head at the boundary at the end of the stress period.

Station_name_1—Station name for head at the boundary at the start of the stress period.

Station name_2—Station name for head at the boundary at the end of the stress period.

There must be one blank space between Column value and Station_name if fixed format is specified.

Multi-Operation Diversion (MDIV) Package

The Diversion (MDIV) package (Restrepo et al., 1998) was developed by the DISTRICT and Florida Atlantic University (FAU) as a tool to simulate the effects of man-made structures such as pumping stations or drains on groundwater and surface water levels. The current version has been revised and improved by FAU in association with the DISTRICT as the package has been applied to sub-regional models in south Florida.

Using the UGEN Package

FOR EACH SIMULATION:

Read maximum number of source and sink points at any stress period and flag for echoes:

1. mxsr, mxsk, nstrctr, idivcb , idivflg, in_tab

Read information for nstrctr number of structures, and operation rules associated with that specific structure name (see Table 10).

Table 10. Information for nstrctr number of structures and operation rules associated with that specific structure name

Item	Type of Information	Variable Name	Type	Description
2.	Structure:	strctr_name	char(16)	name of the structure for flow values
3.	Surface Water Management	basin_name_src	char(16)	Level target name for the source area
4.		basin_name_snk		Level target name for the sink area
5.	Flow conditions:	source within aquifer active region?	{Y,N} char(3)	Source may be located inner/outer aquifer active domain (e.g., out of domain when using a surface storage)
6.		sink within aquifer active region?	{Y,N} char(3)	Sink may be located inner/outer aquifer active domain (e.g., out of domain when using a canal as a sink)
7.		eff_snk	real	Flow efficiency
8.	rules of operation offset based on level/flow:	lv_sr_on	real	lower limit of source head level
9.		lv_sk_of	real	upper limit of sink head level

Item	Type of Information	Variable Name	Type	Description
10.		lv_sk_on	real	lower limit of sink head level
11.		lv_sr_of	real	upper limit of source head level

FOR EACH STRESS PERIOD:

Read cell locations associated with a given source or sink and its structure name. A source or sink can be located in either a wetland cell or any other active/inactive cell within the model grid system. Flow for each pumping station is provided by the UGEN package through the flow file. Level target for each pumping station is provided by the UGEN package through the stage file.

12. itmp_srsk

The following record should appear itmp_srsk times:

13. lay, row, col, type, strctr_name

Type has two options: source or sink. strctr_name is the name of the structure controlling a specific diversion. This name may be up to 16 characters. Free formatting is used for the input instructions. One or more spaces, or a single comma optionally combined with spaces, must separate adjacent values.

EXPLANATION OF VARIABLES READ BY THE MDIV PACKAGE

mxsr—Maximum number of source cells that can be active during the simulation.

mxsk—Maximum number of sink cells that can be active during the simulation.

nstrctr—Current number of structures that can be used during the simulation

idivcb—Both a flag and a unit number.
- If idivcb > 0, unit number on which cell by cell flow terms will be recorded.
- If idivcb = 0, cell by cell flow terms will not be printed or recorded.
- If idivcb < 0, diversions will be printed.

irpflg —is a flag that,
- irpflg_hbxy=0, limited echo for all hbxy variables
- irpflg_hbxy=1, no echo at all for hbxy variables
- irpflg_hbxy=-1, full echo for all hbxy variables

in_tab—Unit number from which static data (items 2-11) is read for each simulation. By default, MODFLOW reads static data from the diversion input data set.

strctr_name—Name of the flow structure controlling a specific diversion. This name can no longer than 16 characters in length.

basin_name_src —Name of the stage target level at source area controlling a specific diversion. This name can no longer than 16 characters in length.

basin_name_snk — Name of the stage target level at sink area controlling a specific diversion. This name can no longer than 16 characters in length.

eff_snk—Flow sink efficiency.

For flow at the structure from source to sink:

lv_sr_on—Lower limit of the source head to be able to release water.

lv_sk_of—Upper limit of sink head to be able to receive water.

lv_sk_on—Lower limit of the sink head to be able to receive water.

lv_sr_of—Upper limit of source head to be able to release water.

For flow at the structure from sink to source:

lv_sr_on—Lower limit of the source head level to be able to receive water.

lv_sk_of—Upper limit of sink head level to be able to release water.

lv_sk_on—Lower limit of the sink head level to be able to release water.

lv_sr_of—Upper limit of source head level to be able to receive water.

itmp_srsk—Both a flag and counter.
- If itmp_srsk < 0, diversion source and sink cell from the preceding stress period will be reused.
- If itmp_srsk >= 0, itmp_srsk is the number of both source and sink cells during the current stress period. It is recommended that large sink/source flows be distributed to several smaller source or sink if the values become very large.

lay,row,col—Location of the cell in the model grid containing the source or sink diversion cell. In the case, the location of diversion is in an inactive cell or even outside of the model area, a value of zero should be provided for its layer. In a case a layer gets dry, water is only injected to underneath layer specified on input data set.

Type—Indicates if a cell is located in a source or sink area.
- SRC: Source associated with a specific structure.
- SNK: Sink associated with a specified structure.

Example of the MDIV Input File in the Lower East Coast Sub-Regional model.

Table 11 shows the MDIV input taken from the Lower East Coast Sub-Regional Model. Each record on the file is described below.

- Record 1: The simulation is interpreted as having 38563 maximum number of source cells associated with stage/flow observations active in any stress period, 38563 maximum number of sink cells, and 25 structures that can be used during the simulation. Cell by cell flow terms will not be printed or recorded. Full echo for all variables are listed in the output file. The file in unit number 51 (see name file) contains static data (items 2-11).
- Record 2: This record defines the name of the structure that is linked with the UGEN flow file for a given diversion.
- Record 3: This record defines the name of the structure that is linked with the UGEN target stage file for the source area. In this case "ZERO" stage is defined as zero constant in the UGEN stage file.
- Record 4: This record defines the name of the structure that is linked with the UGEN target stage file for the sink area. In this case "ZERO" stage is defined as zero in the UGEN stage file.
- Record 5: This record is interpreted as having 4 diversions pumping water in from outside of the model and 5 diversions pumping water in from inside of the model domain.
- Record 6: This record is interpreted as having 1 diversion pumping water out of model domain and 8 diversions pumping water in to the model domain.
- Record 7: The efficiency of the diversion is 100%. There could be only a reduction in the diversion flow due to the target level condition.
- Records 8-11: These records define the deviation of the rules of operation for the structure. In this case, this record defines the target stage values. As an example, structure "WPBWS" is diverting water from outside of the model to the model domain if stage in the sink area is lower than 18.5 ft and higher than -99999 ft.
- Record 12: The simulation is interpreted as having 38563 total number of source and sink cells.
- Records 13 to …: This record defines cell point location, layer, row and column values, defined by a structure that can be applied from layers 1 to 2. The type of diversion cells and structure flow associated is also specified in each record.
- Diversion cells used in stress period 1 are the same for the rest of stress period.

Table 11. MDIV input example taken from the LECsR Model.

Record									
1	38563	38563	25	0	-1	51			
2	WPBWS	WCA2B	WCA3B	WCA1	WCA3A	WCA2A	G92	JFARM	C-18W
3	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO
4	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO	ZERO
5	N	Y	Y	N	N	N	Y	Y	Y
6	Y	Y	Y	Y	Y	Y	Y	N	Y
7	100	100	100	100	100	100	100	100	100
8	-25	11.5	0	-99999	-99999	-99999	13.5	11	17.6
9	18.5	11	99.9	99.9	99.9	99.9	16	99.9	99.9
10	-99999	-99999	-99999	-99999	-99999	-99999	-99999	-99999	-99999
11	99	99	99	99	99	99	99	99	99
12	38563								
13	1	206	338	SNK	WPBWS				
14	1	206	339	SNK	WPBWS				
15	1	206	340	SNK	WPBWS				
16	1	206	341	SNK	WPBWS				
17	1	206	342	SNK	WPBWS				
18	1	207	338	SNK	WPBWS				
19	1	207	339	SNK	WPBWS				
20	1	207	340	SNK	WPBWS				
21	1	207	341	SNK	WPBWS				
22	….								
…	-1								

Wetland (WTL) Package

The first layer type specified in the BCF must be an unconfined layer (laycon=1). The second layer type must be a confined/unconfined with layer type 3 (laycon=3). Muck/peat is considered to be contained within the wetland surface water top layer (i.e., layer one).

FOR EACH SIMULATION;

  0. mxwtld, mxgrdsl, iwtlout

Herein, a subset of two-dimensional arrays is used to describe the parameters of the wetland system (see Table 12). Only values in the array within the wetland areas are redefined (or updated) by the wetland package. However you must provide a value for every cell in the layer.

Table 12. WTL parameters of the wetland system.

Item	Array(ncol,nrow)	Utilily	Description
1.	ibnd_wtl	u2dint	Define wetland areas greater or equal to one and non-wetland areas equal to zero.
2.	hymuc	u2drel	Horizontal hydraulic conductivity of the muck/peat.
3.	vhymucr	u2drel	Anisotropy ratio in the muck/peat areas.
4.	vhyly2r	u2drel	Anisotropic ratio in the model layer underlying the wetland.
5.	capfrng	u2drel	Capillary fringe.
6.	hyc_kdl	u2drel_wtl	Kadlec conductance coefficient for overland flow -kadlec conductivity- without taking into account any preferential flow ways or barriers.

If the simulation is transient (see Table 13):

Table 13. WTL parameters of the wetland system for a transient simulation.

Item	Array(ncol,nrow)	Utilily	Description
7.	symuc	u2drel_wtl	Specific yield of the muck/peat.
8.	sywtl	u2drel_wtl	Specific yield of the surface water body.
9.	alfac, betac	u2drel_wtl	Exponents of the hydraulic slope in the kadlec equation, and of the hydraulic head in the kadlec equation, respectively

The iloc flag, embedded in the same line record of the cell, is used to describe both surface and groundwater flow ways and/or interactions with levees in wetland areas. If a surface water feature (e.g., slough, channel, levee, lake, air boat trail) is present in a wetland cell, then iloc is read as a value greater than zero, in the niloc variable. A wetland cell with no features present is not listed in Table 14.

10. niloc

The following record should appear niloc times; each record format depends on iloc value as shown in Table 14.

Table 14. Parameters for surface water feature present in a wetland cell.

Item	Format	If iloc:	Surface water feature description
11.	row, col, iloc, whyrat, mxlay, lvfp, lvfs, f_slugh	2	slough or channel with interphase provided
12.	row, col, iloc, whyrat	3	air-boat trail
13.	row, col, iloc, mxlay, lvfp, lvfs, f_barrier	4	levee with interphase provided
14.	row, col, iloc, whyrat	5	lake

In the case of a fixed format, a value has a 10 spaces field width within a record. If an integer number does not require the entire field width, then the number should be right justified within the field. If a real number does not require the entire field width, then the number must be within the field. In the case of a free format, the field of the values is not fixed. Each value can consist of any number of digits/characters. One or more spaces, or a single comma optionally combined with spaces, must separate adjacent values.

EXPLANATION OF VARIABLES READ BY THE WETLAND PACKAGE

mxwtld—Maximum number of actual surface wetland features cells active during the simulation.

mxgrdsl—Maximum number of cells at any place within the width of the slough that are going to be considered in the direction calculation of a given cell. Its value should be less than 6 (if any iloc=1).

iwtlout—Both a flag and a unit number for wetland detail echo data set. Warning and basic information always are echoed in the MODFLOW list file (main ascii output).
   • If iwtlout = 0, no echoes written.
   • If iwtlout > 0, echo full output is written in a specific wetlands (ASCII) list file (wetlands.out).
   • If iwtlout < 0, echo with an abbreviated output is written in MODFLOW list file.

ibnd_wtl—Is the wetland boundary array. One value is read for every model cell at the top layer.
   • If ibnd_wtl = 0, non-wetland areas.
   • If ibnd_wtl > 0, wetland areas.

hymuc—Horizontal hydraulic conductivity of the wetland muck/peat.

symuc—Specific yield of the muck/peat array, read only for a transient simulation.

vhymucr—Ratio of vertical hydraulic conductivity to horizontal hydraulic conductivity array in the muck/peat areas.

vhyly2r—Ratio of vertical hydraulic conductivity to horizontal hydraulic conductivity array in the layer underlying wetlands.

capfrng—Muck/peat capillary fringe array.

hyc_kdl—Equivalent hydraulic conductance array for each wetland cell.

sywtl—Specific yield array of the wetland water body, read only for a transient simulation.

alfac—Exponent of the hydraulic slope in the kadlec equation (it is not implemented).

betac—Exponent of the hydraulic head in the kadlec equation.

nwtlnd—Number of actual surface wetland features cells active during the simulation, defined by the user.

row,col—Wetland cell location.

iloc—Flag for each cell which indicates water ways and levees.
- If iloc = 2, slough-channel with interface provided (4).
- If iloc = 3, air-boat trail (2).
- If iloc = 4, levee with interface provided (5).
- If iloc = 5, lake (6).

mxlay—Maximum number of layers penetrated by water body (it can be either one or two).

lvfp—Code indicating primary block interface for the slough or levee.
- If lvfp = 0: not needed;
- If lvfp = 1: right face;
- If lvfp = 2: bottom face,
- If lvfp =3: left face,
- If lvfp = 4: top face.

lvfs—Code indicating secondary block interface for the slough or levee.
- If lvfs = 0: not needed;
- If lvfs = 1: right face;
- If lvfs = 2: bottom face,
- If lvfs = 3: left face,
- If lvfs = 4: top face.

f_barrier—Factor used to multiply the aquifer interblock conductance to represent a levee face. Faces are perpendicular to the direction of flow. (If a wall is to be represented, a value less than 0.005 should be selected).

f_slough—Factor used to multiply the aquifer interblock conductance to represent a slough face. Faces are toward the direction of flow (should be between 1.0 and 2.0).

whyrat—Factor used to multiply the cell horizontal hydraulic conductivity to increase the slough, wetland channels, or air boat horizontal hydraulic conductivity in wetland cells.

Trigger (TRG) Package

This package is named after the mechanism built into the program to trigger cutbacks in well pumpage when heads at certain cells ('trigger cells') in the model reach some specified level (SFWMD, 1999). Input to the Trigger (TRG) package is read from the file that has type "TRG" in the name file. Optional variables are shown in brackets. All variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the non-optional variables have 10-character fields and the optional variables are free format.

FOR EACH SIMULATION:

0. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

1. unit_output   primary_out_file

2. unit_pump_output pumping_out_file

3. n_phase

4. n_type

5. head_or_draw

6. def_cutback

7. n_dry

8. dry_start   dry_end

n_dry repetitions of item 8 are required.

9. n_zone

FOR EACH ZONE:

10. zone

11. zone_corner

12. n_trigger

13. method_string   dry_season_flag

14. tcell_lay   tcell_row   tcell_col

15. trig_value

16. trig_delay

17. cutback

Items 13-17 must be repeated n_trigger times.

EXPLANATION OF VARIABLES READ BY THE TRG PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

unit_output—Fortran unit number for primary trigger package output, input.

primary_out_file—input character string with name of the primary output file, a100.

unit_pump_output—Fortran unit number for pumpage reduction input, output.

pumping_out_file—Input character string with name of the pumpage reduction output file, a100.

n_phase—input integer, number of cutback phases.

n_type—integer, number of usage types.

head_or_draw—integer, input switch to implement triggering by heads (=1)_ or drawdowns (=2).

def_cutback—input, real array of default cutback values (type, phase).

n_dry—input integer, number of dry seasons.

dry_start—input integer array of stress periods at the start of dry seasons (dry seasons).

dry_end—input integer array of stress periods at the end of dry seasons (dry seasons).

n_zone—number of zones as counted by the program.

zone—index referring to current zone.

zone_corner—input integer array of the row and column numbers of the upper left and lower right corners of a zone.

n_trigger—input array of the number of trigger cells in a zone.

method_string—input character string indicating if a trigger uses the cell method or time series method.

dry_season_flag—input integer array indicating the dry season protocol is used for a trigger (zone, trig).

tcell_lay—input integer array of layer numbers of the trigger cells (zone, trigger).

tcell_row—input integer array of row numbers of the trigger cells (zone, trigger).

tcell_col—input integer array of column numbers of the trigger cells (zone, trigger).

trig_value—input real array of trigger values of heads or drawdowns (zone, trigger, and phase).

trig_delay—input integer array of number of periods between setting a trigger and cutting back pumping (zone, trigger, phase).

cutback—real array of values that become the cutback amounts for each well (zone, trig, type, phase).

VARIABLE-DENSITY FLOW PROCESS INPUT INSTRUCTIONS

Variable-Density Flow (VDF) Process Input File

Input for the Variable-Density Flow (VDF) process is read from the file that is specified in the name file with "VDF" as the file type. All single-valued variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, the variables have 10-character fields.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional and the symbol # must be in column 1. Item 0 can be repeated multiple times.

1. MT3DRHOFLG MFNADVFD NSWTCPL IWTABLE

2. DENSEMIN DENSEMAX

   Read Item 3 if NSWTCPL > 1 or NSWTCPL = -1

3. DNSCRIT

   Read Item 4 if MT3DRHOFLG ≥ 0

4. DENSEREF DRHODC(1)

   Read Items 4a through 4c if MT3DRHOFLG = -1

      a. DENSEREF DRHODPRHD PRHDREF

      b. NSRHOEOS

      c. MTRHOSPEC(NSRHOEOS) DRHODC(NSRHOEOS) CRHOREF(NSRHOEOS)

   Item 4c is read as a table with one row representing each MT3DMS species to be included in the equation of state for fluid density. There is no requirement that this table be sorted by MTRHOSPEC.

5. FIRSTDT

FOR EACH STRESS PERIOD

Read items 6 and 7 only if MT3DRHOFLG = 0

6. INDENSE

Read item 7 only if INDENSE is greater than zero

7. [DENSE(NCOL,NROW)] – U2DREL

Item 7 is read for each layer in the grid.

EXPLANATION OF VARIABLES READ FROM THE VDF PROCESS INPUT FILE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Text is printed when the file is read.

MT3DRHOFLG—is the MT3DMS species number that will be used in the equation of state to compute fluid density. This input variable was formerly referred to as MTDNCONC (Langevin and others, 2003).
- If MT3DRHOFLG = 0, fluid density is specified using Items 6 and 7, and flow will be uncoupled with transport if the IMT process is active.
- If MT3DRHOFLG > 0, fluid density is calculated using the MT3DMS species number that corresponds with MT3DRHOFLG. A value for MT3DRHOFLG greater than zero indicates that flow will be coupled with transport.
- If MT3DRHOFLG = -1, fluid density is calculated using one or more MT3DMS species. Items 4a, 4b, and 4c will be read instead of Item 4.

The dependence of fluid density on pressure head can only be activated when MT3DRHOFLG = -1. A value for MT3DRHOFLG of -1 indicates that flow will be coupled with transport.

MFNADVFD—is a flag that determines the method for calculating the internodal density values used to conserve fluid mass.
- If MFNADVFD = 2, internodal conductance values are calculated using a central-in-space algorithm.
- If MFNADVFD ≠ 2, internodal conductance values are calculated using an upstream-weighted algorithm.

NSWTCPL—is a flag used to determine the flow and transport coupling procedure.
- IF NSWTCPL = 0 or 1, flow and transport will be explicitly coupled using a one timestep lag. The explicit coupling option is normally much faster than the iterative option and is recommended.
- If NSWTCPL > 1, NSWTCPL is the maximum number of non-linear coupling iterations for the flow and transport solutions. SEAWAT-2000 will stop execution after NSWTCPL iterations if convergence between flow and transport has not occurred.
- If NSWTCPL = -1, the flow solution will be recalculated only for: a) if: it is the first transport step of the simulation, or b) the last transport step of the MODFLOW timestep, or c) the maximum density change at a cell is greater than DNSCRIT.

IWTABLE—is a flag used to activate the variable-density water-table corrections (The correction term for the water-table case adds nonlinearity to the flow equation because head is a function of equivalent freshwater head. This is partially treated by using the equivalent freshwater head from the previous iteration of the flow equation to solve for $Z_{WT/2}$ terms prior to calculating the correction term, $Z_{WT/2}$ is the elevation of vertical midpoint between the water table and bottom of the model cell [L], see eq. 82, Guo and Langevin, 2002).

- If IWTABLE = 0, the water-table correction will not be applied.
- If IWTABLE > 0, the water-table correction will be applied.

DENSEMIN—is the minimum fluid density. If the resulting density value calculated with the equation of state is less than DENSEMIN, the density value is set to DENSEMIN.

- If DENSEMIN = 0, the computed fluid density is not limited by DENSEMIN (this is the option to use for most simulations).
- If DENSEMIN > 0, a computed fluid density less than DENSEMIN is automatically reset to DENSEMIN.

DENSEMAX—is the maximum fluid density. If the resulting density value calculated with the equation of state is greater than DENSEMAX, the density value is set to DENSEMAX.

- If DENSEMAX = 0, the computed fluid density is not limited by DENSEMAX (this is the option to use for most simulations).
- If DENSEMAX > 0, a computed fluid density larger than DENSEMAX is automatically reset to DENSEMAX.

DNSCRIT—is a user specified density value. DNSCRIT is the convergence criterion, in units of fluid density, for convergence between flow and transport. If the maximum fluid density difference between two consecutive implicit coupling iterations is not less than DNSCRIT, the program will continue to iterate on the flow and transport equations, or will terminate if NSWTCPL is reached. If NSWTCPL is -1, DNSCRIT is the maximum density threshold, in units of fluid density. If the fluid density change (between the present transport timestep and the last flow solution) at one or more cells is greater than DNSCRIT, then SEAWAT_V4 will update the flow field (by solving the flow equation with the updated density field).

DENSEREF—is the fluid density at the reference concentration, temperature, and pressure. For most simulations DENSEREF is specified as the density of freshwaterat 25 °C and at a reference pressure of zero. The length units of DENSEREF should be consistent with the length units used elsewhere in the MODFLOW and MT3DMS input. For example, if the model is developed using meters, fluid density should be expressed as kg/m^3.

DRHODC—formerly referred to as DENSESLP (Langevin and others, 2003), is the slope of the linear equation of state that relates fluid density to solute concentration. In SEAWAT_V4, separate values for DRHODC can be entered for as many MT3DMS species as desired. If DRHODC is not specified for a species, then that species does not affect fluid density. Any measurement unit can be used for solute concentration, provided DENSEREF and DRHODC are set properly. DRHODC can be approximated by the user by dividing the density difference over the range of endmember fluids by the difference in concentration between the end-member fluids

For many applications, the end-member fluids are freshwater and seawater. Therefore, if the measurement units for both density and solute concentration are either kilograms per cubic meter or pounds per cubic

feet, the value for DRHODC is about 0.7143. Users may also prefer to use relative concentrations, such that freshwater has a concentration of zero, and seawater has a concentration of 1.0. If relative concentrations are used with density values in units of kilograms per cubic meter, the appropriate DRHODC value is about 25 (Langevin and others, 2008).

SEAWAT_V4 offers the flexibility to use a variety of different solute concentrations to calculate fluid density, provided DRHODC is specified correctly. For example, users can utilize MT3DMS to simulate the concentration of total dissolved solids (TDS), chloride concentration, or the relative seawater concentration on a scale from 0 to 1. Use of TDS as the solute species for the fluid density calculation is straightforward, and is described in Guo and Langevin (2002, p. 18). If chloride is used for the fluid density calculation, a constant ratio between chloride concentration and TDS normally has to be assumed. For seawater and mixtures of seawater with freshwater, the ratio between chloride concentration and TDS is about 55 percent (19 g/L of chloride per 35 g/L of TDS). In some cases, particularly at relatively low TDS concentrations, this constant ratio may no longer hold. In these cases, caution is necessary in defining the relation between concentration and fluid density. If the linear relation implemented in SEAWAT_V4 is not appropriate, users may need to add a new equation of state and recompile the program. Regardless of the approach, simple hand calculations of fluid density are suggested as a way of ensuring that the correct input values are being used.

DRHODPRHD—is the slope of the linear equation of state that relates fluid density to the height of the pressure head (in terms of the reference density).

Note that DRHODPRHD can be calculated from the volumetric expansion coefficient for pressure using the equation below. If the simulation is formulated in terms of kilograms and meters, DRHODPRHD has an approximate value of 4.46×10^{-3} kg/m^4. A value of zero, which is the option that would be used for most problems, inactivates the dependence of fluid density on pressure.

$$\frac{\partial \rho}{\partial \ell} = \rho_0^2 g \beta_P$$

PRHDREF—is the reference pressure head. This value should normally be set to zero.

NSRHOEOS—is the number of MT3DMS species to be used in the equation of state for fluid density.

Note that this value is read only if MT3DRHOFLG = -1.

MTRHOSPEC—is the MT3DMS species number corresponding to the adjacent DRHODC and CRHOREF.

CRHOREF—is the reference concentration (C0) for species, MTRHOSPEC. For most simulations, CRHOREF should be specified as zero.

Note that if MT3DRHOFLG > 0, CRHOREF is assumed by SEAWAT to equal zero (as in previous versions of SEAWAT).

FIRSTDT—is the length of the first transport timestep used to start the simulation if both of the following two conditions are met:

1. the IMT process is active, and
2. transport timesteps are calculated as a function of the user-specified Courant number (the MT3DMS input variable, PERCEL, is greater than zero)

INDENSE—is a flag. INDENSE is read only if MT3DRHOFLG is equal to zero.
- If INDENSE < 0, values for the DENSE array will be reused from the previous stress period. If it is the first stress period, values for the DENSE array will be set to DENSEREF.
- If INDENSE = 0, values for the DENSE array will be set to DENSEREF.
- If INDENSE ≥ 1, values for the DENSE array will be read from Item 7.
- If INDENSE = 2, values read for the DENSE array are assumed to represent solute concentration, and will be converted to density values using the equation of state.

DENSE—is the fluid density array read for each layer using the MODFLOW-2000 U2DREL array reader. The DENSE array is read only if MT3DRHOFLG is equal to zero.

Note that the DENSE array may also be entered in terms of solute concentration, or any other units, if INDENSE is set to 2, and the constants used in the density equation of state are specified appropriately.

Viscosity (VSC) Package Input File

Input for the Viscosity (VSC) package is read from the file that is specified in the name file with "VSC" as the file type. Note that the VDF process must be active in order to use the VSC package. Also, the VSC package works **only** with the LPF package; the VSC package is not compatible with the BCF or HUF2 packages. All variables in the VSC input file are read using free format.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional and the symbol # must be in column 1. Item 0 can be repeated multiple times.

1. MT3DMUFLG

2. VISCMIN VISCMAX

Read Item 3 if MT3DMUFLG $\geq$ 0

3. VISCREF DMUDC(1) CMUREF(1)

Read Item 3a through 3c if MT3DMUFLG = -1

   a. VISCREF

   b. NSMUEOS MUTEMPOPT

   c. MTMUSPEC(NSMUEOS) DMUDC(NSMUEOS) CMUREF(NSMUEOS)

Item 3c is read NSMUEOS times, once for each MT3DMS species to be included in the viscosity equation of state.

   d. MTMUTEMPSPEC AMUCOEFF(MUNCOEFF)

Item 3d is read if MUTEMPOPT > 0. The size of the AMUCOEFF coefficient array depends on the selected equation of state for viscosity.

FOR EACH STRESS PERIOD

Read items 4 and 5 only if MT3DMUFLG = 0

4. INVISC

Read item 5 only if INVISC is greater than zero

5. [VISC(NCOL,NROW)] – U2DREL

Item 5 is read for each layer in the grid.

EXPLANATION OF VARIABLES READ FROM THE VSC PACKAGE INPUT FILE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

MT3DMUFLG—is the MT3DMS species number that will be used in the equation of state to compute fluid viscosity.
- If MT3DMUFLG ≥ 0, fluid density is calculated using the MT3DMS species number that corresponds with MT3DMUFLG.
- If MT3DMUFLG = -1, fluid viscosity is calculated using one or more MT3DMS species.

VISCMIN—is the minimum fluid viscosity. If the resulting viscosity value calculated with the equation of state is less than VISCMIN, the viscosity value is set to VISCMIN.
- If VISCMIN = 0, the computed fluid viscosity is not limited by VISCMIN (this is the option that should be used for most simulations).
- If VISCMIN > 0, a computed fluid viscosity less than VISCMIN is automatically reset to VISCMIN.

VISCMAX—is the maximum fluid viscosity. If the resulting viscosity value calculated with the equation of state is greater than VISCMAX, the viscosity value is set to VISCMAX.
- If VISCMAX = 0, the computed fluid viscosity is not limited by VISCMAX (this is the option that should be used for most simulations).
- If VISCMAX > 0, a computed fluid viscosity larger than VISCMAX is automatically reset to VISCMAX.

VISCREF—is the fluid viscosity at the reference concentration and reference temperature. For most simulations VISCREF is specified as the viscosity of freshwater.

DMUDC—is the slope of the linear equation that relates fluid viscosity to solute concentration.

NSMUEOS—is the number of MT3DMS species to be used in the linear equation of state for fluid viscosity (this number does not include the temperature species if the nonlinear option is being used).

   Note that this value is read only if MT3DMUFLG = -1. A value of zero indicates that none of the MT3DMS species have a linear affect on fluid viscosity (note that the nonlinear temperature dependence may still be activated); nothing should be entered for item 3c in this case.

MUTEMPOPT—is a flag that specifies the option for including the effect of temperature on fluid viscosity.
- If MUTEMPOPT = 0, the effect of temperature on fluid viscosity is not included, or it is a simple linear relation that is specified in item 3c.
- If MUTEMPOPT = 1, fluid viscosity is calculated using the equation below. The size of the AMUCOEFF array in item 3e is 4 (MUNCOEFF = 4).

$$\mu(T) = A_1 \cdot A_2^{\left(\frac{A_3}{T+A_4}\right)}$$

If MUTEMPOPT = 2, fluid viscosity is calculated using the equation below. The size of the AMUCOEFF array in item 3e is 5 (MUNCOEFF = 5).

$$\mu(T) = A_1 \cdot \left[A_2 + A_3(T + A_4)\right]^{A_5}$$

If MUTEMPOPT = 3, fluid viscosity is calculated using the equation below. The size of the AMUCOEFF array in item 3e is 2 (MUNCOEFF = 2).

$$\mu(T) = A_1 \cdot T^{A_2}$$

Note that if NSMUEOS and MUTEMPOPT are both set to zero, all fluid viscosities are set to VISCREF.

MTMUSPEC—is the MT3DMS species number corresponding to the adjacent DMUDC and CMUREF.

CMUREF—is the reference concentration for species, MTMUSPEC.

DMUDC—is the slope of the linear equation that relates fluid viscosity to solute concentration.

MTMUTEMPSPEC—is the MT3DMS species number that corresponds to temperature. This value must be between 1 and NCOMP and should not be listed in MTMUSPEC of item 3c.

AMUCOEFF—is the coefficient array of size, MUNCOEFF. AMUCOEFF is "A" in three equations shown above.

MUNCOEFF—is the size of the AMUCOEFF array.

INVISC—is a flag. INVISC is read only if MT3DMUFLG is equal to zero.
- If INVISC < 0, values for the VISC array will be reused from the previous stress period. If it is the first stress period, values for the VISC array will be set to VISCREF.
- If INVISC = 0, values for the VISC array will be set to VISCREF.
- If INVISC ≥ 1, values for the VISC array will be read from item 5.
- If INVISC = 2, values read for the VISC array are assumed to represent solute concentration, and will be converted to viscosity values using the equation of state.

VISC—is the fluid viscosity array read for each layer using the MODFLOW-2000 U2DREL array reader. The VISC array is read only if MT3DMUFLG is equal to zero. Note that the VISC array may also be entered in terms of solute concentration, or any other units, if INVISC is set to 2, and the simple linear expression in item 3 can be used to represent the relation to viscosity.

Time-Variant Constant-Head (CHD) Package Input File USING Density

The new options for specifying the density value assumed to correspond to the head values entered in the CHD Package are activated using auxiliary variables (Harbaugh and others, 2000). If the auxiliary variable "CHDDENSOPT" (in all uppercase) is detected in the CHD input file, then the program will read the CHD density option as an additional field. If all the values assigned to CHDDENSOPT are either 0 or 2, then no additional auxiliary variable is required. If, however, one or more CHD cells are assigned a value of 1 for CHDDENSOPT, then SEAWAT_V4 requires a user-specified density value for those cells. This density value must be entered through an additional auxiliary variable with the name "CHDDENS" (again in all uppercase). All three CHDDENSOPT options may be used within a single simulation. If option 1 is used for one or more CHD cells, the CHDDENS value will be required for all the CHD cells. The CHDDENS value will only be used, however, for those CHD cells with a CHDDENSOPT of 1. An example of a simple CHD input file is shown in Table 15.

Table 15. An example of a simple CHD input file using density.

3	AUX	CHDDENSOPT	CHDDEN			
3	0					
1	1	1	0.00	0.00	0	1025.0
2	1	1	0.00	0.00	1	1025.0
3	1	1	0.00	0.00	2	1025.0

This CHD input file will result in the assignment of CHD cells in layers 1, 2, and 3 of row 1 and column 1. CHD option 0 will be used for the CHD cell in layer 1. This option means that the assigned head value will be converted to reference head using the simulated density at that cell. CHD option 1 will be used for the CHD cell in layer 2. This option means that the assigned head value will be converted to a reference head using a density value of 1,025 kg/m3 (entered in auxiliary variable "CHDDEN"). CHD option 2, used for the CHD cell in layer 3, indicates that the assigned head value is an environmental head that will be converted to a reference head using an average density value of the overlying CHD cells.

Evaluating Temperature Output

Evaluating simulated temperatures is straightforward, and can be done using the same types of post-processing programs that are used to evaluate concentrations. Calculation of thermal energies, however, can be confusing because the "mass" values reported in the listing and mass budget files do not directly correspond with temperature or heat energy. The "mass" values reported by SEAWAT_V4 are calculated by multiplying temperature by a volumetric flux and the length of the transport timestep. Thus, for the temperature species, a reported mass value might have units of m^3 °C (or $(m^3$ °C)/d for a mass flux). These "mass" values (M_T) can be converted into thermal energy (E), using the following simple relation: E = M_T ρc_{Pfluid}

SOLUTE TRANSPORT PROCESS INPUT INSTRUCTIONS

These input forms and the array readers are re-described in this section for easy reference in the preparation of input files. A list-directed record terminates when the number of the input values equals the number of items in the input list or until a slash (/) is encountered. A list-directed record may occupy several lines of an input file, but each new record should start at a new line. List-directed input permits the use of a repeat count in the form, n*d where n is an unsigned-nonzero integer constant, and the input n*d causes n consecutive values of d to be entered.

Array Readers RARRAY and IARRAY

Most of the input data to be entered to the model by the user will consist of one- or two-dimensional real or integer arrays. To perform these tasks, two utility subroutines, RARRAY for reading one- or two-dimensional real arrays, and IARRAY for reading one- or two-dimensional integer arrays, are provided in the program. These two array readers are compatible to array readers U2DREL and U2DINT provided in MODFLOW (McDonald and Harbaugh 1988). However, RARRAY and IARRAY also permit the input of array values by block, zonal, and list-directed (or free) formats.

For each array to be entered, RARRAY or IARRAY reads an array—control record first on the unit reserved for the major option which calls the array reader. The content and form of the array control record are as follows:

> **FOR REAL ARRAY READER (RARRAY):**
> Record: IREAD CNSTNT FMTIN IPRN
> Format: I10 F10.0 A20 I10
> **FOR INTEGER ARRAY READER (IARRAY)**:
> Record: IREAD ICONST FMTIN IPRN
> Format: I10 I10 A20 I10

> a. **IREAD**—determines how array values are read.
> (1) If IREAD = 0, every element in array will be set equal to the value CNSTNT for RARRAY or to ICONST for IARRAY.
>
> (2) If IREAD = 100, an array of input values follows the control record. The array values are read in the format specified in the third field of the array-control record (FMTIN) from the same unit used for reading the array-control record.
>
> (3) If IREAD = 101, an array of values organized in "block" format follows the array-control record. The block format consists of a record specifying the number of blocks, NBLOCK, followed by NBLOCK records of input values (all in free format), specifying the first row (I1), last row (I2), first column (J1), and last column (J2) of each block as well as the value (ZZ/IZ)

to be assigned to the cells within the block (ZZ is a real value for RARRAY and IZ is an integer value for IARRAY), as shown below:

NBLOCK (free format)

    I1, I2, J1, J2, ZZ/IZ (block 1) (free format)

    I1, I2, J1, J2, ZZ/IZ (block 2) (free format)

    ......

    I1, I2, J1, J2, ZZ/IZ (block NBLOCK) (free format)

If a subsequent block overlaps any preceding blocks, the subsequent block overrides the preceding blocks. It is always good practice to have the first block cover the entire grid and allow subsequent blocks to override portions of the grid as this averts the possibility that any portion of the grid will be left unassigned.

(4) If IREAD = 102, an array of values organized in "zonal" format follows the array-control record. Each zone represents one value of the input variable, and each zone is identified by a zone number -- an integer which may actually be thought of as a code for the corresponding value of the input variable. The zonal format consists of a record specifying the number of zones, NZONE, followed by an array of values for the input variables, ZV(NZONE) or IZV (NZONE), which are listed in sequence according to zone numbers (1, 2, 3, ..., etc.); ZV is a real array for RARRAY and IZV is an integer array for IARRAY as explained below. Following ZV/IZV is the zone indicator array A(NCOL,NROW)/IA(NCOL,NROW) specifying the zone number assigned to each node (cell) in a given layer of the model. The elements in this array are actually always integers (i.e., the integer codes identifying the zone for each cell). However, they are stored in the array reserved for the input variable itself and thus follow the format of the input variable. Hence, the zone indicators are read using the format as specified in the third field of the array-control record (FMTIN) which must be a real format specifier for RARRAY and an integer format specifier for IARRAY. If the zone indicator for a cell is equal to zero, the value for that cell is set to zero.

    The zonal format for RARRAY:

        NZONES (free format)

        ZV(1),.ZV(2),........, ZV(NZONE) (free format)

        A(NCOL,NROW) (using format FMTIN)

    The zonal format for IARRAY:

        NZONE (free format)

        IZV(1), IZV(2), ..., IZV(NZONE) (free format)

        IA(NCOL,NROW) (using format FMTIN)

(5) If IREAD = 103, an array of values follows the array-control record.

The array values are read using list-directed or free format.

(6) If IREAD = any value other than 0, 100, 101, 102, and 103, array values are read from a separate file.

    (a) If IREAD > 0, it is the unit number on which the external file is read using the format specified in FMTIN.

    (b) If IREAD < 0, the absolute value of IREAD gives the unit number on which the array values are read from an external unformatted file. The unformatted file contains one record of header, followed by an unformatted record of NCOL*NROW values.

b. **CNSTNT/ICONST**—is a constant. Its use depends on the value of IREAD.
(1) If IREAD = 0, every element in the array is set equal to CNSTNT/ICONST.
(2) If IREAD z 0, and CNSTNT/ICONST z 0, elements in the array are multiplied by CNSTNT/ICONST.

c. **FMTIN**—is the user-specified format to read array values or the zonal indicator array. The format must be enclosed in parentheses; for example, (15F5.0) for real values and (15I5) for integer input values.

d. **IPRN**—is a flag indicating whether or not the array being read should be printed out for checking, and it also serves as a code indicating the format that should be used in printing. It is used only if IREAD is not equal to zero. IPRN is set to zero if the specified value exceeds those defined in Table 3. If IPRN is less than zero, the array will not be printed.

Basic Transport (BTN) Package

The BTN package is required for every simulation using MT3D.

FOR EACH SIMULATION

A1 Record: HEADNG(1)

Format: A80

HEADNG(1) is the first line of any title or heading for the simulation run.

A2 Record: HEADNG(2)

Format: A80

HEADNG(2) is the second line of any title or heading for the simulation run.

A3 Record: NLAY, NROW, NCOL, NPER, NCOMP, MCOMP

Format: 6I10
  NLAY—is the total number of layers;
  NROW—is the total number of rows;
  NCOL—is the total number of columns;
  NPER—is the total number of stress periods;
  NCOMP—is the total number of chemical species included in the current simulation. For single-species simulation, set NCOMP = 1;
  MCOMP— is the total number of "mobile" species. MCOMP must be equal to or less than **NCOMP**—. For single-species simulation, set MCOMP=1.

A4 Record: TUNIT, LUNIT, MUNIT

Format: 3A4
  TUNIT— is the name of unit for time, such as DAY or HOUR;
  LUNIT—is the name of unit for length, such as FT or M;
  MUNIT—is the name of unit for mass, such as LB or KG.

    Note that these names are used for identification purposes only and do not affect the model outcome.

A5 Record: Dummy input line

Format: n/a

TRNOP is no longer used since MT3DMS v5.0. However, a dummy input line must still be specified in the input file. A blank line is acceptable.

A6 Record: LAYCON(NLAY)

Format: 40I2

LAYCON—is a 1-D integer array indicating the type of model layers. Each value in the array corresponds to one model layer. Enter LAYCON in as many lines as necessary if NLAY > 40.LAYCON = 0, the model layer is confined. The layer thickness DZ to be entered in a subsequent record will be used as the saturated thickness of the layer.

LAYCON—0, the model layer is either unconfined or convertible between confined and unconfined. The saturated thickness, as calculated by the flow model and saved in the flow-transport link file, will be read and used by the transport model. (Note that this type corresponds to the LAYCON values of 1, 2, and 3 of MODFLOW; however, there is no need to distinguish between these layer types in the transport simulation.)

A7 Array: DELR(NCOL)

Reader: RARRAY

DELR—is a 1-D real array representing the cell width along rows ('x) in the direction of increasing column indices (j). Specify one value for each column of the grid.

A8 Array: DELC(NROW)

Reader: RARRAY

DELC—is a 1-D real array representing the cell width along columns (y) in the direction of increasing row indices (i).

Specify one value for each row of the grid.

A9 Array: HTOP(NCOL,NROW)

Reader: RARRAY

HTOP—is a 2-D array defining the top elevation of all cells in the first (top) model layer, relative to the same datum as the hydraulic heads.

If the first model layer is unconfined, HTOP can be set most conveniently to a uniform elevation above the water table.

Note that the concentrations for the cells in the first model layer are calculated at nodal points assumed to be midway between HTOP and the bottom of the first layer. Thus, HTOP should not be set much higher than the water table. Also note that the difference between HTOP and the bottom elevation of the first layer must be equal to the layer thickness DZ to be defined in

the next record. If the first model layer is confined, HTOP is equal to the bottom elevation of the confining unit overlying the first model layer.

A10 Array: DZ(NCOL,NROW) (one array for each layer in the grid)

Reader: RARRAY

DZ—is the thickness of all cells in each model layer. DZ is a 3-D array. The input to 3-D arrays is handled as a series of 2-D arrays with one array for each layer, entered in the sequence of layer 1, 2, ..., NLAY. The thickness of the first layer must be equal to the difference between HTOP and its bottom elevation. When the grid is discretized into horizontal layers, HTOP for the first layer and DZ within each layer are uniform. However, if a vertically distorted grid is used, both HTOP and DZ may be variable for cells within the same layer.

A11 Array: PRSITY(NCOL,NROW) (one array for each layer)

Reader: RARRAY

PRSITY is the "effective" porosity of the porous medium in a single porosity system (see discussions in Chapter 2, MT3D). A12 Array: ICBUND(NCOL,NROW) (one array for each layer)

Reader: IARRAY

ICBUND is an integer array specifying the boundary condition type (inactive, constant-concentration, or active) for every model cell. For multispecies simulation, ICBUND defines the boundary condition type shared by all species.

   Note that different species are allowed to have different constant-concentration conditions through an option in the Source and Sink Mixing package.
   - If ICBUND = 0, the cell is an inactive concentration cell for all species. Note that no-flow or "dry" cells are automatically converted into inactive concentration cells. Furthermore, active cells in terms of flow can be treated as inactive concentration cells to minimize the area needed for transport simulation, as long as the solute transport is insignificant near those cells.
   - If ICBUND < 0, the cell is a constant-concentration cell for all species. The starting concentration of each species remains the same at the cell throughout the simulation. (To define different constant-concentration conditions for different species at the same cell location, refer to the Sink/Source Mixing package.).
      Note that unless explicitly defined as a constant-concentration cell, a constant-head cell in the flow model is not treated as a constant-concentration cell.
   - If ICBUND > 0, the cell is an active (variable) concentration cell where the concentration value will be calculated.

   (***Enter A13 for each species***)

A13 Array: SCONC(NCOL,NROW) (one array for each layer)

Reader: RARRAY

SCONC—is the starting concentration (initial condition) at the beginning of the simulation (unit, ML-3). For multispecies simulation, the starting concentration must be specified for all species, one species at a time.

A14 Record: CINACT, THKMIN

Format: 2F10.0

CINACT—is the value for indicating an inactive concentration cell (ICBUND = 0). Even if inactive cells are not anticipated in the model, a value for CINACT still must be submitted.

THKMIN—is the minimum saturated thickness in a cell, expressed as the decimal fraction of the model layer thickness (DZ) below which the cell is considered inactive. The default value is 0.01 (i.e., 1 percent of the model layer thickness).

A15 Record: IFMTCN, IFMTNP, IFMTRF, IFMTDP, SAVUCN

Format: 4I10, L10

IFMTCN—is a flag indicating whether the calculated concentration should be printed to the standard output text file and also serves as a printing-format code if it is printed.
- IFMTCN > 0, concentration is printed in the wrap form.
- IFMTCN < 0, concentration is printed in the strip form.
- IFMTCN = 0, concentration is not printed.

IFMTNP— is a flag indicating whether the number of particles in each cell (integers) should be printed and also serves as a printing-format code if they are printed. The convention is the same as that used for IFMTCN.

IFMTRF—is a flag indicating whether the model-calculated retardation factor should be printed and also serves as a printing-format code if it is printed. The convention is the same as that used for IFMTCN.

IFMTDP—is a flag indicating whether the model-calculated, distance-weighted dispersion coefficient should be printed and also serves as a printing-format code if it is printed. The convention is the same as that used for IFMTCN.

SAVUCN—is a logical flag indicating whether the concentration solution should be saved in a default unformatted (binary) file named MT3Dnnn.UCN, where nnn is the species index number, for post-processing purposes or for use as the initial condition in a continuation run.

- If SAVUCN = T, the concentration of each species will be saved in the default file MT3Dnnn.UCN. In addition, the model spatial discretization information will be saved in another default file named MT3D.CNF to be used in conjunction with MT3Dnnn.UCN for post-processing purposes.

- If SAVUCN = F, neither MT3Dnnn.UCN nor MT3D.CNF is created.

A16 Record: NPRS

Format: I10

NPRS—is a flag indicating the frequency of the output and also indicating whether the output frequency is specified in terms of total elapsed simulation time or the transport step number. Note that what is actually printed or saved is controlled by the input values entered in the preceding record (Record A15).
- If NPRS > 0, simulation results will be printed to the standard output text file or saved to the unformatted concentration file at times as specified in record TIMPRS(NPRS) to be entered in the next record.
- If NPRS = 0, results will not be printed or saved except at the end of simulation.
- If NPRS < 0, results will be printed or saved whenever the number of transport steps is an even multiple of NPRS.

(***Enter A17 only if NPRS > 0***)

A17 Record: TIMPRS(NPRS)

Format: 8F10.0

TIMPRS is the total elapsed time at which the simulation results are printed to the standard output text file or saved in the default unformatted (binary) concentration file MT3Dnnn.UCN.

Note that if NPRS > 8, enter TIMPRS in as many lines as necessary.

A18 Record: NOBS, NPROBS

Format: 2I10

NOBS—is the number of observation points at which the concentration of each species will be saved at the specified frequency in the default MT3Dnnn.OBS where nnn is the species index number.

NPROBS—is an integer indicating how frequently the concentration at the specified observation points should be saved in the observation file MT3Dnnn.OBS. Concentrations are saved every NPROBS step.

(***Enter A19 NOBS times if NOBS > 0***)

A19 Record: KOBS, IOBS, JOBS

Format: 3I10

KOBS, **IOBS**, and **JOBS**—are the cell indices (layer, row, column) in which the observation point or monitoring well is located and for which the concentration is to be printed at every transport step in file MT3Dnnn.OBS. Enter one set of KOBS, IOBS, JOBS for each observation point.

A20 Record: CHKMAS, NPRMAS

Format: L10, I10

CHKMAS—is a logical flag indicating whether a one-line summary of mass balance information should be printed, for checking and post-processing purposes, in the default file MT3Dnnn.MAS where nnn is the species index number.
- If CHKMAS = T, the mass balance information for each transport step will be saved in file MT3Dnnn.MAS.
- If CHKMAS = F, file MT3Dnnn.MAS is not created.

NPRMAS—is an integer indicating how frequently the mass budget information should be saved in the mass balance summary file MT3Dnnn.MAS. Mass budget information is saved every NPRMAS step.

FOR EACH STRESS PERIOD

A21 Record: PERLEN, NSTP, TSMULT

Format: F10.0, I10, F10.0

PERLEN—is the length of the current stress period. If the flow solution is transient, PERLEN specified here must be equal to that specified for the flow model. If the flow solution is steady-state, PERLEN can be set to any desired length.

NSTP—is the number of time-steps for the transient flow solution in the current stress period. If the flow solution is steady-state, NSTP = 1.

TSMULT—is the multiplier for the length of successive timesteps used in the transient flow solution; it is used only if NSTP > 1.

- If TSMULT > 0, the length of each flow time-step within the current stress period is calculated using the geometric progression as in MODFLOW. Note that both NSTP and TSMULT specified here must be identical to those specified in the flow model if the flow model is transient.

- If TSMULT ≤0, the length of each flow time-step within the current stress period is read from the record TSLNGH

(**see A22**). This option is needed in case the length of timesteps for the flow solution is not based on a geometric progression in a flow model, unlike MODFLOW.

(***Enter A22 if TSMULT≤0***)

A22 Record: TSLNGH(NSTP)

Format: 8F10.0

TSLNGH—provides the length of time-steps for the flow solution in the current stress period. This record is needed only if the length of time-steps for the flow solution is not based on a geometric progression. Enter TSLNGH in as many lines as necessary if NSTP > 8.

A23 Record: DT0, MXSTRN, TTSMULT, TTSMAX

Format: F10.0, I10, 2F10.0

DT0—is the user-specified transport step size within each time-step of the flow solution. DT0 is interpreted differently depending on whether the solution option chosen is explicit or implicit:

- For explicit solutions (i.e., the GCG solver is not used), the program will always calculate a maximum transport step size which meets the various stability criteria. Setting DT0 to zero causes the model-calculated transport step size to be used in the simulation. However, the model-calculated DT0 may not always be optimal. In this situation, DT0 should be adjusted to find a value that leads to the best results. If DT0 is given a value greater than the model-calculated step size, the model-calculated step size, instead of the user-specified value, will be used in the simulation.

- For implicit solutions (i.e., the GCG solver is used), DT0 is the initial transport step size. If it is specified as zero, the model-calculated value of DT0, based on the userspecified Courant number in the Advection package, will be used. The subsequent transport step size may increase or remain constant depending on the user-specified transport step size multiplier TTSMULT and the solution scheme for the advection term.

MXSTRN—is the maximum number of transport steps allowed within one time step of the flow solution. If the number of transport steps within a flow time-step exceeds MXSTRN, the simulation is terminated.

TTSMULT—is the multiplier for successive transport steps within a flow time-step if the GCG solver is used and the solution option for the advection term is the standard finite-difference method. A value between 1.0 and 2.0 is generally adequate. If the GCG package is not used, the transport solution is solved explicitly as in the original MT3D code, and TTSMULT is always set to 1.0 regardless of the user-specified input.

Note that for the particle-tracking-based solution options and the third-order TVD scheme, TTSMULT does not apply.

TTSMAX—is the maximum transport step size allowed when transport step size multiplier TTSMULT > 1.0.

Note: Setting TTSMAX=0 imposes no maximum limit.

Advection (ADV) Package

This input file is needed only if the Advection package is used; however, this package is needed under almost all circumstances.

FOR EACH SIMULATION

B1 Record: MIXELM, PERCEL, MXPART, NADVFD

Format: I10, F10.0, 2I10

MIXELM—is an integer flag for the advection solution option.

- MIXELM = 0, the standard finite-difference method with upstream or central-in-space weighting, depending on the value of NADVFD;
- MIXELM = 1, the forward-tracking method of characteristics (MOC);
- MIXELM = 2, the backward-tracking modified method of characteristics (MMOC);
- MIXELM = 3, the hybrid method of characteristics (HMOC) with MOC or MMOC automatically and dynamically selected;
- MIXELM = -1, the third-order TVD scheme (ULTIMATE).

PERCEL—is the Courant number (i.e., number of cells, or a fraction of a cell) advection will be allowed in any direction in one transport step.

- For implicit finite-difference or particle-tracking-based schemes, there is no limit on PERCEL, but for accuracy reasons, it is generally not set much greater than one.

    Note, however, that the PERCEL limit is checked over the entire model grid. Thus, even if PERCEL > 1, advection may not be more than one cell's length at most model locations.

- For the explicit finite-difference or the third-order TVD scheme, PERCEL is also a stability constraint which must not exceed one and will be automatically reset to one if a value greater than one is specified.

MXPART—is the maximum total number of moving particles allowed and is used only when MIXELM = 1 or 3.

NADVFD—is an integer flag indicating which weighting scheme should be used; it is needed only when the advection term is solved using the implicit finite difference method.
- NADVFD = 0 or 1, upstream weighting (default);
- NADVFD = 2, central-in-space weighting.

(***Enter B2 if MIXELM = 1, 2, or 3***)

B2 Record: ITRACK, WD

Format: I10, F10.0

ITRACK—is a flag indicating which particle-tracking algorithm is selected for the Eulerian-Lagrangian methods.
- ITRACK = 1, the first-order Euler algorithm is used.
- ITRACK = 2, the fourth-order Runge-Kutta algorithm is used; this option is computationally demanding and may be needed only when PERCEL is set greater than one.
- ITRACK = 3, the hybrid first- and fourth-order algorithm is used; the Runge-Kutta algorithm is used in sink/source cells and the cells next to sinks/sources while the Euler algorithm is used elsewhere.

WD—is a concentration weighting factor between 0.5 and 1. It is used for operator splitting in the particletracking- based methods. The value of 0.5 is generally adequate. The value of WD may be adjusted to achieve better mass balance. Generally, it can be increased toward 1.0 as advection becomes more dominant.

(***Enter B3 if MIXELM = 1 or 3***)

B3 Record: DCEPS, NPLANE, NPL, NPH, NPMIN, NPMAX

Format: F10.0, 5I10

DCEPS—is a small Relative Cell Concentration Gradient below which advective transport is considered negligible. A value around 10^{-5} is generally adequate.

NPLANE—is a flag indicating whether the random or fixed pattern is selected for initial placement of moving particles.

- If NPLANE = 0, the random pattern is selected for initial placement. Particles are distributed randomly in both the horizontal and vertical directions by calling a random number generator. This option is usually preferred and leads to smaller mass balance discrepancy in nonuniform or diverging/converging flow fields.

- If NPLANE > 0, the fixed pattern is selected for initial placement. The value of NPLANE serves as the number of vertical "planes" on which initial particles are placed within each cell block. The fixed pattern may work better than the random pattern only in relatively uniform flow fields. For two-dimensional simulations in plan view, set NPLANE = 1. For cross sectional or three-dimensional simulations, NPLANE = 2 is normally adequate. Increase NPLANE if more resolution in the vertical direction is desired.

NPL—is the number of initial particles per cell to be placed at cells where the Relative Cell Concentration Gradient is less than or equal to DCEPS. Generally,NPL can be set to zero since advection is considered insignificant when the Relative Cell Concentration Gradient is less than or equal to DCEPS. Setting NPL equal to NPH causes a uniform number of particles to be placed in every cell over the entire grid (i.e., the uniform approach).

NPH—is the number of initial particles per cell to be placed at cells where the Relative Cell Concentration Gradient is greater than DCEPS. The selection of NPH depends on the nature of the flow field and computer memory limitation. Generally, a smaller number should be used in uniform flow fields and a larger number should be used in nonuniform flow fields. Values exceeding 16 in 2D simulation or 32 in 3D simulation are rarely necessary. If random pattern is chosen, NPH particles are randomly distributed within the cell block. If fixed pattern is chosen, NPH is divided by NPLANE to yield the number of particles to be placed per vertical plane.

NPMIN—is the minimum number of particles allowed per cell. If the number of particles in a cell at the end of a transport step is fewer than NPMIN, new particles are inserted into that cell to maintain a sufficient number of particles. NPMIN can be set to zero in uniform flow fields and to a number greater than zero in diverging/converging flow fields. Generally, a value between zero and four is adequate.

NPMAX—is the maximum number of particles allowed per cell. If the number of particles in a cell exceeds NPMAX, all particles are removed from that cell and replaced by a new set of particles equal to NPH to maintain mass balance. NPMAX can be set to approximately two times of NPH.

(***Enter B4 if MIXELM = 2 or 3***)

B4 Record: INTERP, NLSINK, NPSINK

Format: 3I10

INTERP—is a flag indicating the concentration interpolation method for use in the MMOC scheme. Currently, only linear interpolation is implemented.

Enter INTERP = 1.

NLSINK—is a flag indicating whether the random or fixed pattern is selected for initial placement of particles to approximate sink cells in the MMOC scheme. The convention is the same as that for NPLANE. It is generally adequate to set NLSINK equivalent to NPLANE.

NPSINK—is the number of particles used to approximate sink cells in the MMOC scheme. The convention is the same as that for NPH. It is adequate to set NPSINK equivalent to NPH.

(***Enter B5 if MIXELM = 3***)

B5 Record: DCHMOC

Format: F10.0

DCHMOC—is the critical Relative Concentration Gradient for controlling the selective use of either MOC or MMOC in the HMOC solution scheme. The MOC solution is selected at cells where the Relative Concentration Gradient is greater than DCHMOC. The MMOC solution is selected at cells where the Relative Concentration Gradient is less than or equal to DCHMOC.

Dispersion (DSP) Package

This input file is needed only if the Dispersion package is used in the simulation.

FOR EACH SIMULATION:

C0 Record: One or more optional keywords.

  Format: [Free]

  The only keyword currently supported is **MultiDiffusion**, case insensitive. **The keyword must be specified in an input line with the $ sign in the first column.** The keyword input record is optional.

C1 Array: AL(NCOL,NROW) (one array for each layer).

  Reader: RARRAY

  AL—is the longitudinal dispersivity, α_L , for every cell of the model grid (unit, L).

C2 Array: TRPT(NLAY)

  Reader: RARRAY

  TRPT—is a 1D real array defining the ratio of the horizontal transverse dispersivity, α_{TH}, to the longitudinal dispersivity, α_L. Each value in the array corresponds to one model layer. Some recent field studies suggest that TRPT is generally not greater than 0.1.

C3 Array: TRPV(NLAY)

  Reader: RARRAY

  TRPV—is the ratio of the vertical transverse dispersivity, α_{TV} , to the longitudinal dispersivity, L. Each value in the array corresponds to one model layer. Some recent field studies suggest that TRPT is generally not greater than 0.01.

  Set TRPV equal to TRPT to use the standard isotropic dispersion model. Otherwise, the modified isotropic dispersion model is used (Equation 10 and 11 in Chapter 2, Zheng and Wang, 1999).

(*If no keyword is defined*):

C4 Array: DMCOEF(NLAY)

  Reader: RARRAY

DMCOEF—is the effective molecular diffusion coefficient (unit: L^2T^{-1}). Set DMCOEF = 0 if the effect of molecular diffusion is considered unimportant. Each value in the array corresponds to one model layer.

Enter one array for all solute components.

(*If keyword [MultiDiffusion] is defined*):

<u>C4 Array</u>: DMCOEF(NCOL,NROW) (One array for each layer)

Reader: RARRAY

DMCOEF—is the effective molecular diffusion coefficient (unit: L^2T^{-1}). Set DMCOEF = 0 if the effect of molecular diffusion is considered unimportant. Each value in the array corresponds to one model cell.

Repeat the input for each mobile component.

Sink & Source Mixing (SSM) Package

This input file is needed if any sink or source option is used in the flow model, including the constant-head or generalhead- dependent boundary conditions. The classification of the sink/source types used in MT3DMS is the same as that used by MODFLOW (McDonald and Harbaugh 1988).

FOR EACH SIMULATION

D1 Record: Dummy input line

Format: n/a

These logical flags are no longer needed as the status of various flow sink/source packages is obtained by MT3DMS through MODFLOW. However, a dummy input line must still be specified in the input file. A blank line is acceptable.

D2 Record: MXSS, ISSGOUT

Format: I10

MXSS—is the maximum number of all point sinks and sources included in the flow model. Point sinks and sources include constant-head cells, wells, drains, rivers, and general-head-dependent boundary cells. Recharge and evapotranspiration are treated as areally distributed sinks and sources; thus, they should not be counted as point sinks and sources. MXSS should be set close to the actual number of total point sinks and sources in the flow model to minimize the computer memory allocated to store sinks and sources. **This input value is used for the memory allocation purpose only**.

ISSGOUT—is the unit number for an optional output file to save the calculated flux-averaged composite concentrations at multi-node wells. The name of the output file must be specified through the Name File as in "**DATA ISSGOUT FileName**".

FOR EACH STRESS PERIOD:

(*Enter D3 if FRCH = T*)

D3 Record: INCRCH

Format: I10

INCRCH—is a flag indicating whether an array containing the concentration of recharge flux for each species will be read for the current stress period.
- If INCRCH≥ 0, an array containing the concentration of recharge flux for each species will be read.
- If INCRCH < 0, the concentration of recharge flux will be reused from the last stress period. If INCRCH < 0 is specified for the first stress period, then by default, the

concentration of positive recharge flux (source) is set equal to zero and that of negative recharge flux (sink) is set equal to the aquifer concentration.

(***Enter D4 for each species if FRCH = T and INCRCH≥0***)

D4 Array: CRCH(NCOL,NROW)

Reader: RARRAY

CRCH—is the concentration of recharge flux for a particular species. If the recharge flux is positive, it acts as a source whose concentration can be specified as desired. If the recharge flux is negative, it acts as a sink (discharge) whose concentration is always set equal to the concentration of groundwater at the cell where discharge occurs.

Note that the location and flow rate of recharge/discharge are obtained from the flow model directly through the unformatted flow-transport link file.

(***Enter D5 if FEVT = T***)

D5 Record: INCEVT

Format: I10

INCEVT—is a flag indicating whether an array containing the concentration of evapotranspiration flux for each species will be read for the current stress period.
- If INCEVT≥0, an array containing the concentration of evapotranspiration flux for each species will be read.
- If INCEVT< 0, the concentration of evapotranspiration flux for each species will be reused from the last stress period. If INCEVT < 0 is specified for the first stress period, then by default, the concentration of negative evapotranspiration flux (sink) is set to the aquifer concentration, while the concentration of positive evapotranspiration flux (source) is set to zero.

(***Enter D6 for each species if FEVT = T and INCEVT≥0***)

D6 Array: CEVT(NCOL,NROW)

Reader: RARRAY

CEVT—is the concentration of evapotranspiration flux for a particular species. Evapotranspiration is the only type of sink whose concentration may be specified externally. Note that the concentration of a sink cannot be greater than that of the aquifer at the sink cell. Thus, if the sink concentration is specified greater than that of the aquifer, it is automatically set equal to the concentration of the aquifer. Also note that the location and flow rate of evapotranspiration are obtained from the flow model directly through the unformatted flow-transport link file.

D7 Record: NSS Format: I10

NSS—is the number of point sources whose concentrations need to be specified. By default, unspecified point sources are assumed to have zero concentration. (The concentration of point sinks is always set equal to the concentration of groundwater at the sink location.)

Note that in MT3DMS, point sources are generalized to include not only those associated with a flow rate in the flow model, but also those independent of the flow solution. This type of "mass-loading" sources may be used to include contaminant sources which have minimal effects on the hydraulics of the flow field.

(***Enter D8 NSS times if NSS > 0***)

D8 Record: KSS, ISS, JSS, CSS, ISSTYPE, (CSSMS(n), n=1,NCOMP)

Format: 3I10, F10.0, I10, [free]

Note that those fluid sources included in the MODFLOW simulation but not specified herein are assumed to have zero concentration by default.

KSS, **ISS**, **JSS**—are the cell indices (layer, row, column) of the point source for which a concentration needs to be specified for each species.

CSS—is the specified source concentration or massloading rate, depending on the value of ISSTYPE, in a single-species simulation,. (For a multispecies simulation, CSS is not used, but a dummy value still needs to be entered here.)

Note that for most types of sources, CSS is interpreted as the source concentration with the unit of mass per unit volume (ML^{-3}), which, when multiplied by its corresponding flow rate (L^3T^{-1}) from the flow model, yields the mass-loading rate (MT^{-1}) of the source.

For a special type of sources (ISSTYPE= 15), CSS is taken directly as the mass-loading rate (MT^{-1}) of the source so that no flow rate is required from the flow model.

Furthermore, if the source is specified as a constantconcentration cell (ISSTYPE = -1), the specified value of CSS is assigned directly as the concentration of the designated cell. If the designated cell is also associated with a sink/source term in the flow model, the flow rate is not used.

ITYPE—is an integer indicating the type of the point source as listed below:
- ISSTYPE = 1, constant-head cell;
- ISSTYPE = 2, well;
- ISSTYPE = 3, drain (note that in MODFLOW conventions, a drain is always a sink, thus, the concentration for drains cannot be specified if the flow solution is from MODFLOW);
- ISSTYPE = 4, river (or stream);
- ISSTYPE = 5, general-head-dependent boundary cell;
- ISSTYPE = 15, mass-loading source;
- ISSTYPE = -1, constant-concentration cell.

(**CSSMS(n), n=1, NCOMP**)—defines the concentrations of a point source for multispecies simulation with NCOMP>1. In a multispecies simulation, it is necessary to define the concentrations of all species associated with a point source. As an example, if a chemical of a certain species is injected into a multispecies system, the concentration of that species is assigned a value greater than zero while the concentrations of all other species are assigned zero. CSSMS(n) can be entered in free format, separated by a comma or space between values. Several important notes on assigning concentration for the constant-concentration condition (ISSTYPE = -1) are listed below:

- The constant-concentration condition defined in this input file takes precedence to that defined in the Basic Transport package input file. In a multiple stress period simulation, a constantconcentration cell, once defined, will remain a constantconcentration cell in the duration of the simulation, but its concentration value can be specified to vary in different stress periods.

- In a multispecies simulation, if it is only necessary to define different constant-concentration conditions for selected species at the same cell location, specify the desired concentrations for those species, and assign a negative value for all other species. The negative value is a flag used by MT3DMS to skip assigning the constant-concentration condition for the designated species.

- No changes in the meaning and number of the input variables as well as their input format from version 5.0; however, the range of allowed ISSTYPE codes has increased since the initial release of MT3DMS. Since version 5.2, users can specify the input concentration of an injection well (ISSTYPE=2), i.e., CSS or CSSMS, as a negative integer code (IC). The absolute value of the integer code is the single cell location indicator of the extraction well whose output concentration is used as the input concentration for the injection well. For an extraction well located at layer K, row I, and column J, IC is computed as,

$$IC = NCOL \times NROW \times (K-1) + NCOL \times (I-1) + J .$$

where NCOL and NROW are the total numbers of columns and rows.

SOLVER PACKAGES INPUT INSTRUCTIONS

The user may select one of five solver packages to solve the variable-density ground-water flow equation. These solver packages include the Strongly Implicit Procedure, Slice-Successive Over-Relaxation, Preconditioned Conjugate-Gradient, Direct Solver, and the Link-Algebraic Multi-Grid packages. Input to these packages is read from files that are specified as type SIP, SOR, PCG, DE4, and LMG, respectively, in the name file. Input instructions for these packages can be found in Harbaugh et al (2000), with exception to the LMG package, which is documented in Mehl and Hill (2001). A more detailed discussion of the DE4 package can be found in Harbaugh (1995).

The defined convergence criteria are too large if the global ground-water flow budget errors calculated are unacceptably large. For most ground-water flow problems, global budget errors greater than one percent are unacceptable.

Much like the prior release of SEAWAT, it may be necessary to set a smaller value for the convergence criteria for head, and a larger value for the convergence criteria for flow. For example, with the PCG package, the convergence criteria for head is HCLOSE, and the convergence criteria for flow is RCLOSE. With the VDF process, HCLOSE may need to be about four to six orders of magnitude smaller than HCLOSE as compared to a constant-density flow simulation. If HCLOSE is not set small enough, then SEAWAT will not be able to resolve vertical freshwater head gradients, and vertical fluxes calculated by the program may not be accurate. RCLOSE has the dimensions of a mass flux with the VDF process, which means that RCLOSE may be set larger by about a factor of the average fluid density.

Strongly Implicit Procedure (SFWMD-SIP) Package

Input to the Strongly Implicit Procedure (SFWMD-SIP) package (Harbaugh et al., 2000), modified by the SFWMD, is read from the file that is type "SIP" in the name file. All numeric variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, all the variables have 10-character fields.

FOR EACH SIMULATION

1. [#Text]

Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

2. MXITER NPARM MNITER

3. ACCL HCLOSE IPCALC WSEED IPRSIP HCLOSEMAX, NOSTOP

EXPLANATION OF VARIABLES READ BY THE SIP PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

MXITER—is the maximum number of times through the iteration loop in one time step in an attempt to solve the system of finite-difference equations.

MNITER—is the minimum number of iterations in one time step.

NPARM—is the number of iteration variables to be used. Five variables are generally sufficient.

MNITER—is the minimum forced number of iterations in one time step.

ACCL—is the acceleration variable. It must be greater than zero and is generally equal to one. If a zero is entered, it is changed to one.

HCLOSE—is the head change criterion for convergence. When the maximum absolute value of head change from all nodes during an iteration is less than or equal to HCLOSE, iteration stops.

IPCALC—is a flag indicating where the seed for calculating iteration variables will come from.
- If 0—the seed entered by the user will be used.
- If 1—the seed will be calculated at the start of the simulation from problem variables.

WSEED—is the seed for calculating iteration variables. It is always read, but it is used only if IPCALC is equal to zero.

IPRSIP—is the printout interval for SIP. If IPRSIP is equal to zero, it is changed to 999. The maximum head change (positive or negative) is printed for each iteration of a time step whenever the time step is an even multiple of IPRSIP. This printout also occurs at the end of each stress period regardless of the value of IPRSIP.

HCLOSEMAX—is the maximum closure criterion; the simulation will terminate if the closure criterion exceeds HCLOSEMAX.

NOSTOP—is a flag:
- If NOSTOP=0— the heads in the cells which violated HCLOSEMAX will be reset to their values at the end of the last previous time step and a warning message will be written to the output file.
- If NOSTOP=1— the program will not terminate if the HCLOSEMAX criterion has been violated

Slice-Successive Overrelaxation (SOR) Package

Input to the Slice-Successive Overrelaxation (SOR) package (Harbaugh et al., 2000) is read from the file that is type "SOR" in the name file. All numeric variables are free format if the option "FREE" is specified in the Basic package input file; otherwise, all the variables have 10-character fields.

FOR EACH SIMULATION

1. [#Text]

 Item 0 is optional—"#" must be in column 1. Item 0 can be repeated multiple times.

2. MXITER

3. ACCL   HCLOSE   IPRSOR

EXPLANATION OF VARIABLES READ BY THE SOR PACKAGE

Text—is a character variable (199 characters) that starts in column 2. Any characters can be included in Text. The "#" character must be in column 1. Except for the name file, lines beginning with # are restricted to these first lines of the file. Text is printed when the file is read.

MXITER—is the maximum number of iterations allowed in a time step.

ACCL—is the acceleration variable, usually between 1.0 and 2.0.

HCLOSE—is the head change criterion for convergence. When the maximum absolute value of head change from all nodes during an iteration is less than or equal to HCLOSE, iteration stops.

IPRSOR—is the printout interval for SOR. IF IPRSOR is equal to zero, it is changed to 999. The maximum head change (positive or negative) is printed for each iteration of a time step whenever the time step is an even multiple of IPRSOR. This printout also occurs at the end of each stress period regardless of the value of IPRSOR.

Preconditioned Conjugate-Gradient (PCG2) Package

In MODFLOW-2000, input for PCG2 is read from a file that is type "PCG" in the name file. In PCG2, the difference between results of successive iterations, for convergence criteria, is measured using both the maximum absolute value of the change in hydraulic head and the maximum absolute value of the all-by-all residual for that iteration. Typical values for these error criteria are 0.01 ft and 0.01 ft^3/s, respectively. The convergence criteria are dimensional, so different values are required when different units are used. For example, if the time unit is days instead of seconds, the equivalent maximum absolute value of cell-by-cell residual for that iteration would be 864 ft^3/d. If unacceptably large global budget errors occur, the error criteria should be reduced. Hill (1990) found that reducing both error criteria from 10^{-3} to 10^{-6} increased the execution time by as much as 55 percent, and resulted in lack of convergence in some test cases.

FOR EACH SIMULATION

0. [#Text]

Item 0 is optional. The "#" symbol needs to be in column 1. Item 0 can be repeated many times.

1. Data: MXITER ITER NPCOND

2. Data: HCLOSE RCLOSE RELAX NBPOL IPRPCG MUTPCG DAMP

EXPLANATION OF VARIABLES READ BY THE PCG PACKAGE

MXITER—is the maximum number of outer iterations -- that is, calls to the solution routine. For a linear problem MXITER should be 1, unless more than 50 inner iterations are required, when MXITER could be as large as 10. A larger number (generally less than 100) is required for a nonlinear problem.

ITERI—is the maximum number of inner iterations. For nonlinear problems, ITERI usually ranges from 3 to 10; a value of 30 will be sufficient for most linear problems.

NPCOND—is the flag used to select the matrix preconditioning method. The following options are available.
- NPCOND = 1 Modified Incomplete Cholesky (for use on scalar computers)
- NPCOND = 2 Polynomial (for use on parallel computers)

HCLOSE—is the head change criterion for convergence, in units of length. When the maximum absolute value of the head change at all nodes during an iteration is less than or equal to HCLOSE, and the criterion for RCLOSE is satisfied (see below), iteration stops. Commonly, HCLOSE equals 0.01.

RCLOSE—is the residual criterion for convergence, in units of cubic length per time. When the maximum absolute value of the residual at all nodes during an iteration is less than or equal to RCLOSE, and the criterion for HCLOSE is satisfied (see above), iteration stops. Commonly, RCLOSE equals HCLOSE.

For nonlinear problems, convergence is achieved when the convergence criteria are satisfied on the first inner iteration.

RELAX—is the relaxation parameter used with NPCOND=L (MICCG). Usually, RELAX=1.0, but for some problems a value of 0.99, 0.98, or 0.97 will reduce the number of iterations required for convergence. RELAX is not used if NPCOND≠I.

NBPOL—is used when NPCOND=2 to indicate whether the estimate of the upper bound on the maximum eigenvalue is 2.0, or whether the estimate will be calculated. NBPOL=2 is used to specify the value as 2.0; for any other value of NBPOL, the estimate is calculated. Convergence is generally insensitive to this parameter. NBPOL is not used if NPCOND does not equal 2.

IPRPCG—is the printout interval for PCG. If IPRPCG is equal to zero, it is changed to 999. The extreme head change and residual (positive or negative) are printed for each iteration of a time step whenever the time step is an even multiple of.IPRPCG. The printout also occurs at the end of each stress period regardless of the value of IPRPCG.

MUTPCG—is a flag that controls printing of convergence information from the solver:
- MUTPCG = 0 - print tables of maximum head change and residual each iteration
- MUTPCG = 1 - print only the total number of iterations
- MUTPCG = 2 – print nothing from the solver
- MUTPCG = 3 - print as for 0, but only if solver convergence fails

DAMP—is a damping parameter. Values less than l.0 can be used to aid convergence in nonlinear problems.

Direct Solution (DE4) Package

The DE4 package (Harbaugh, 1995) reads its input data from the file-unit number specified in the file name. Input to the DE4 package is defined using two numbered "items." Each item consists of several parameters that are specified in one record. Descriptions of the parameters follow the list of items. The format of the parameters is free format, which means that the exact location of parameter values in a record is not predefined. The parameter values must be specified in the indicated order, and adjacent values must be separated by one or more spaces, or by a comma surrounded optionally by any number of spaces. A zero value must be explicitly specified as zero; that is, zero cannot be specified by using blanks. All parameters are integers except for HCLOSE, which is a real number; integer values must not include a decimal point, but real numbers can. The items are read once at the beginning of the simulation:

FOR EACH SIMULATION

    1. ITMX MXUP MXLOW MXBW

    2. IFREQ MUTD4 ACCL HCLOSE IPRD4

EXPLANATION OF VARIABLES READ BY THE DE4 PACKAGE

ITMX—is the maximum number of iterations each time step. Specify ITMX = 1 if iteration is not desired. Ideally iteration would not be required for direct solution; however, it is necessary to iterate if the flow equation is non linear (see explanation for IFREQ=3) or if computer precision limitations result in inaccurate calculations as indicated by a large water budget error. For a non-linear flow equation, each iteration is equally time consuming because [A] is changed each iteration and Gaussian elimination is required after each change. This is called external iteration. For a linear equation, iteration is significantly faster because [A] is changed at most once per time step; thus, Gaussian elimination is required at most once per time step. This is called internal iteration.

MXUP—is the maximum number of equations in the upper part of the equations to be solved. This value impacts the amount of memory used by the DE4 package. If specified as 0, the program will calculate MXUP as half the number of cells in the model, which is an upper limit. The actual number of equations in the upper part will be less than half the number of cells whenever there are no-flow and constant-head cells because flow equations are not formulated for these cells. The DE4 package prints the actual number of equations in the upper part when it runs. The printed value can be used for MXUP in future runs in order to minimize memory usage.

MXLOW—is the maximum number of equations in the lower part of equations to be solved. This value impacts the amount of memory used by the DE4 package. If specified as 0, the program will calculate MXLOW as half the number of cells in the model, which is an upper limit. The actual number of equations in the lower part will be less than half the number of cells whenever there are no-flow and constant-head cells because flow equations are not formulated for these cells. The DE4 package prints the actual number of equations in the lower part when it runs. The printed value can be used for MXLOW in future runs in order to minimize memory usage.

MXBW—is the maximum band width plus 1 of the [AL] matrix. This value impacts the amount of memory used by the DE4 package. If specified as 0, the program will calculate MXBW as the product of the two smallest grid dimensions plus 1, which is an upper limit. The DE4 package prints the actual band width plus 1 when it runs. The printed value can be used for MXBW in future runs in order to minimize memory usage.

IFREQ—is a flag indicating the frequency at which coefficients in [A] change. This affects the efficiency of solution; significant work can be avoided if it is known that [A] remains constant all or part of the time.

> IFREQ = 1 indicates that the flow equations are linear and that coefficients of simulated head for all stress terms are constant for all stress periods. To meet the linearity requirement, all model layers must be confined (which is specified in the Block- Centered Flow package by setting parameter LAYCON equal to 0 for all layers), and there must be no formulations that change based upon head (such as seepage from a river changing from head dependent flow to a constant flow when head drops below the bottom of the riverbed). Examples of coefficients of simulated head for stress terms are riverbed conductance, drain conductance, maximum evapotranspiration rate, evapotranspiration extinction depth, and general-head boundary conductance.

> IFREQ = 2 indicates that the flow equations are linear, but coefficients of simulated head for some stress terms may change at the start of each stress period. (See IFREQ=1 for information about linear equations.) Examples of coefficients of simulated head for stress terms are riverbed conductance, drain conductance, maximum evapotranspiration rate, evapotranspiration extinction depth, and general-head boundary conductance. For a simulation consisting of only one stress period, IFREQ=2 has the same meaning as IFREQ=1.

> IFREQ = 3 indicates that a non-linear flow equation is being solved, which means that some terms in [A] depend on simulated head. Examples of head-dependent terms in [A] are transmissivity for water-table layers, which is based on saturated thickness; flow terms for rivers, drains, and evapotranspiration if they convert between head dependent flow and constant flow; and the change in storage coefficient when a cell converts between confined and unconfined. When a non-linear flow equation is being solved, external iteration (ITMX>1) is normally required in order to accurately approximate the non linearities.

> **Note** that when non linearities caused by water-table calculations are part of a simulation, there are not necessarily any obvious signs in the output from a simulation that does not use external iteration to indicate that iteration is needed. In particular, the budget error may be acceptably small without iteration even though there is significant error in head because of non linearity. To understand this, consider the water-table correction for transmissivity. For each iteration a new transmissivity is calculated based on the previous head. Then the flow equations are solved, and a budget is computed using the new head with the same transmissivities. No budget discrepancy results because heads are correct for the transmissivity being used at this point; however, the new heads may mean that there should be a significant change in transmissivity. The new transmissivity will not be calculated unless there is another iteration. Therefore, when one or more layers is under water-table conditions, iteration should always be tried. The maximum change in head

each iteration (printed by DE4 when IPRD4 = 1 and MUTD4 = 0) provides an indication of the impact of all non linearities.

MUTD4—is a flag that indicates the quantity of information that is printed when convergence information is printed for a time step.
- MUTD4 = 0 indicates that the number of iterations in the time step and the maximum head change each iteration are printed.
- MUTD4 = 1 indicates that only the number of iterations in the time step is printed.
- MUTD4 = 2 indicates no information is printed.

ACCL—is a multiplier for the computed head change for each iteration. Normally this value is 1. A value greater than 1 may be useful for improving the rate of convergence when using external iteration to solve non-linear problems (IFREQ=3). ACCL should always be 1 for linear problems. When ITMX=1, ACCL is changed to 1 regardless of the input value; however, a value must always be specified.

HCLOSE—is the head change closure criterion. If iterating (ITMX > 1), iteration stops when the absolute value of head change at every node is less than or equal to HCLOSE. HCLOSE is not used if not iterating, but a value must always be specified.

IPRD4—is the time step interval for printing out convergence information when iterating (ITMX > 1). For example, if IPRD4 is 2, convergence information is printed every other time step. A value must always be specified even if not iterating.

Multigrid solver (LMG4) Package

Algebraic multigrid solvers can attain significant reduction in computer execution time for many problems. Generally, the decrease in execution time is accompanied by the use of larger amounts of memory. When the Link-AMG (LMG) package is activated through the Name file using file type LMG, the AMG solver is used to solve the equations produced by MODFLOW-2000s Ground-Water Flow process for hydraulic head, and, if applicable, the equations produced by MODFLOW-2000s Sensitivity process for sensitivities of hydraulic head throughout the grid. Linear or nonlinear flow conditions may be simulated. The closure criteria for the solvers were adjusted such that, for each comparison, the solutions had approximately the same accuracy as indicated by the MODFLOW-2000 budget summary.

Typically, decreasing BCLOSE by one order of magnitude will decrease the global budget error by about one order of magnitude. A value of BCLOSE that is smaller than is needed will produce excessively long execution times. Such a situation probably exists if the global budget errors calculated for the solutions at all time steps and all parameters (if sensitivities are calculated) are less than 0.01 percent (0.00 will be printed by MODFLOW-2000 as the percent discrepancy for all budgets). The LMG package reads its input data from the file indicated in the Name file using File Type LMG.

FOR EACH SIMULATION

1. STOR1 STOR2 STOR3 ICG

2. MXITER MXCYC BCLOSE DAMP IOUTAMG

Item 3 is read if DAMP = -2

3. DUP DLOW

EXPLANATION OF VARIABLES READ BY THE DE4 PACKAGE

STOR1—is a variable controlling the amount of storage allocated in the Z array for the array A and the amount of storage allocated in the IX array for the array JA. For most problems, a value of 3.0 should be adequate.

STOR2—is a variable controlling the amount of storage allocated in the Z array for the arrays U and FRHS, and the amount of storage allocated in the IX array for the array IA. For most problems, a value of 2.2 should be adequate.

STOR3—is a variable controlling the amount of storage allocated in the IX array for the array IG. For most problems, a value of 5.4 should be adequate.

ICG—is a variable controlling whether or not conjugate gradient iterations are used at the end of each multigrid cycle. A value of 1 indicates that conjugate gradient iterations will be performed, while a value of 0 indicates no conjugate gradient iterations will be performed. All other values are automatically reset to 0. For some problems, using conjugate gradient iterations can improve convergence, but it will increase the memory used by the solver.

MXITER—is the maximum number of iterations – that is, calls to the AMG solver. For linear problems, MXITER can be set equal to 2. For nonlinear problems,

MXITER—generally needs to be larger, but rarely more than 50.

MXCYC—is the maximum number of cycles allowed per call to the solver. This is similar to the variable ITER1 in PCG2 (Hill, 1990, p. 13). A value of 50 is suggested. For some nonlinear problems, however, faster convergence may be achieved by reducing MXCYC and increasing MXITER.

BCLOSE—is the budget closure criterion for the scaled L2 norm of the matrix equations (eq. 3). A value similar to RCLOSE of PCG2 (Hill, 1990, p. 12-14) should be used. If the global budget error is too large, decrease BCLOSE by one order of magnitude to reduce the global budget error by about one order of magnitude. This approximation can be used to adjust BCLOSE until a satisfactory solution is attained. See the sections on Convergence Criterion and Tips for Achieving Convergence for more information.

DAMP—is a damping/accelerating parameter identical to ACCL of the DE4 (Harbaugh, 1995, p. 12) solver. Generally, a value of 1.0 is sufficient for most problems. However, for nonlinear problems, values less than 1.0 may be necessary to achieve convergence.
- DAMP>0 This value of DAMP is applied for all iterations.
- DAMP=-1 Cooley's method for adaptive damping is implemented.
- DAMP=-2 The relative reduced residual method for adaptive damping is implemented. All other values of DAMP are automatically reset to 1.0 (no damping).

IOUTAMG—is a flag that controls the information printed each time step from the solver to the MODFLOW-2000 LIST output file (Harbaugh and others, 2000). Diagnostic messages from the solver are sent to a temporary file called "lmg_err.tmp" except if IOUTAMG=3, when they are printed along with the other iteration information. The "lmg_err.tmp" file is deleted upon successful termination of the solver. If the solver should fail, the output in this file may help to identify solver problems. The possible values of IOUTAMG and the information printed to the LIST file are as follows.
- IOUTAMG=0 No printing from the solver to the LIST file.
- IOUTAMG=1 Print scaling for residuals and residuals before and after cycling.
- IOUTAMG=2 Print scaling for residuals, residuals before and after cycling, the computer storage used, and the computation times if the CTIME subroutine has been adapted to the computer operating system (see Compatibility and Portability).
- IOUTAMG=3 Print solver messages, scaling for residuals, residuals after each cycle the computer storage used, and the computation times if the CTIME subroutine has been adapted to the computer operating system (see Compatibility and Portability).

DUP—is the maximum value of DAMP that should be applied at any iteration. If the adaptive scheme calculates a value of DAMP that is greater than DUP, DAMP will be reset to DUP. A value of 1.0 is reasonable for most problems.

DLOW—is the minimum value of DAMP that should be applied at any iteration. If the adaptive scheme determines that the value of DAMP should be decreased, it will calculate a new value of DAMP based on DLOW being the minimum (see eq. 7). A value of 0.2 is reasonable for most problems.

Generalized Conjugate Gradient Solver Package

Input to the Generalized Conjugate Gradient (GCG) package is read on unit INGCG = 9, which is preset in the main program. The input file is needed only if the GCG solver is used for implicit solution schemes. This package is a new addition in MT3DMS.

FOR EACH SIMULATION:

F1 Record: MXITER, ITER1, ISOLVE, NCRS

Format: Free

MXITER—is the maximum number of outer iterations; it should be set to an integer greater than one only when a nonlinear sorption isotherm is included in simulation.

ITER1—is the maximum number of inner iterations; a value of 30-50 should be adequate for most problems.

ISOLVE—is the type of preconditioners to be used with the Lanczos/ORTHOMIN acceleration scheme:
- ISOLVE = 1, Jacobi
- ISOLVE = 2, SSOR
- ISOLVE = 3, Modified Incomplete Cholesky (MIC) (MIC usually converges faster, but it needs significantly more memory)

NCRS—is an integer flag for treatment of dispersion tensor cross terms:
- NCRS = 0, lump all dispersion cross terms to the righthand- side (approximate but highly efficient).
- NCRS = 1, include full dispersion tensor (memory intensive).

F2 Record: ACCL, CCLOSE, IPRGCG

Format: Free

ACCL—is the relaxation factor for the SSOR option; a value of 1.0 is generally adequate.

CCLOSE—is the convergence criterion in terms of relative concentration; a real value between 10^{-4} and 10^{-6} is generally adequate.

IPRGCG—is the interval for printing the maximum concentration changes of each iteration. Set IPRGCG to zero as default for printing at the end of each stress period.

REFERENCES

Guo, Weixing, and Langevin, C.D., 2002. User's guide to SEAWAT: A computer program for simulation of three-dimensional variable-density ground-water flow. U.S. Geological Survey Techniques of Water-Resources Investigations, book 6, chap. A7, 77 p.

Harbaugh, A.W., 1995, Direct solution package based on alternating diagonal ordering for the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 95-288, 46 p.

Harbaugh, A.W., Banta, E.R., Hill, M.C., and McDonald, M.G., 2000. MODFLOW-2000, the U.S. Geological Survey modular ground-water model – User guide to modularization concepts and the ground-water flow process. U.S. Geological Survey Open-File Report 00-92, 121 p.

Harbaugh, A.W., and McDonald, M.G., 1996, User's documentation for MODFLOW-96, an update to the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 96-485, 56 p.

Harbaugh, A.W., 1995, Direct solution package based on alternating diagonal ordering for the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 95-288, 46 p.

Hill, M.C., 1990, Preconditioned Conjugate-Gradient 2 (PCG2), A computer program for solving ground-water flow equations: U.S. Geological Survey Water-Resources Investigations Report 90-4048, 43 p.

Hsieh, P.A., and Freckleton, J.R., 1993, Documentation of a computer program to simulate horizontal-flow barriers using the U.S. Geological Survey modular three- dimensional finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 92-477, 32 p. (Note: the input instructions in this documentation are outdated. Use the input instructions in Harbaugh and others (2000) instead.)

Jones, M. 1999. A Reinjection Drainflow Package for MODFLOW. Albuquerque, NM, 33 p.

Langevin, C.D., Shoemaker, W.B., and Guo, Weixing. 2003. SEAWAT-2000, the U.S. Geological Survey Modular Ground-Water Model–Documentation of the SEAWAT-2000 Version with the Variable-Density Flow Process (VDF) and the Integrated MT3DMS Transport Process (IMT). U.S. Geological Survey Open-File Report 03-426, 43 p.

Langevin, C.D., Thorne, D.T., Jr., Dausman, A.M., Sukop, M.C., and Guo, Weixing, 2007, SEAWAT Version 4: A Computer Program for Simulation of Multi-Species Solute and Heat Transport: U.S. Geological Survey Techniques and Methods Book 6, Chapter A22, 39 p.

Langevin, C.D., Thorne, D.T., Jr., Dausman, A.M., Sukop, M.C., and Guo, Weixing, 2008, SEAWAT Version 4: A Computer Program for Simulation of Multi-Species Solute and Heat Transport: U.S. Geological Survey Techniques and Methods Book 6, Chapter A22, 39 p.

Leak, S.A. and Prudic, D.E., 1991, Documentation of a computer program to simulate aquifer-system compaction using the modular finite-difference ground-water flow model: U.S. Geological Survey Techniques of Water-Resources Investigations, book 6, chap. A2, 68 p.

Leake, S.A., and Lilly, M.R., 1997, Documentation of a computer program (FHB1) for assignment of transient specified-flow and specified-head boundaries in applications of the modular finite-difference ground-water flow model (MODFLOW): U.S. Geological Survey Open-File Report 97-571, 50 p.

McDonald, M.G., A.W. Harbaugh. 1988. A Modular, Three-Dimensional Finite-Difference Groundwater Flow Model. Techniques of Water Resources Investigations of the United States Geological Survey, Book 6, U.S. Government Printing Office, Washington, D.C.

McDonald, M.G., Harbaugh, A.W., Orr, B.R., and Ackerman, D.J., 1992, A method of converting no-flow cells to variable head cells for the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 91-536, 99 p.

Mehl, S.E. and Hill, M.C., MODFLOW-2000, the U.S. Geological Survey modular ground-water model -- User guide to the LINK-AMG (LMG) Package for solving matrix equations using an algebraic multigrid solver: U.S. Geological Survey Open-File Report 01-177, 33 p.

Survey Modular Ground-Water Model–Documentation of the SEAWAT-2000 Version with the Variable-Density Flow Process (VDF) and the Integrated MT3DMS Transport Process (IMT). U.S. Geological Survey Open-File Report 03-426, 43 p.

Restrepo, J.I. and A.M Montoya. 1997. MODFLOW Wetland Module: Final Report. Department of Geography and Geology, Florida Atlantic University, Boca Raton, FL, 86 p.

Restrepo, J.I., A.M. Montoya, and J. Obeysekera. 1998. A Wetland Simulation Module for the MODFLOW Ground Water Model. Ground Water, 36(5):764-770 p.

Restrepo, J.I., A.M. Montoya, and D. Garces. 2001. The utility generation package (UGEN) for the groundwater flow model, MODFLOW. Submitted to the conference MODFLOW 2001. Golden, Colorado September 11-14, 2001.

Restrepo, J.I., A.M. Montoya and D. Garces. 2003. The Updated Utility Generation Package (USGEN) for the Groundwater Flow Model, MODFLOW. Developed by the Hydrological Research Center at Florida Atlantic University, Boca Raton, FL in cooperation with the South Florida Water Management District, West Palm Beach, FL.

Rodberg, K., 1999. Multibud Utility. South Florida Water Management District. Internal Report.

Rodberg, K., 1999. Modflow Summation Enhancement. South Florida Water Management District. Internal Report.

South Florida Water Management District. Diversion Systems: Structures and Pump Stations. Prepared by Florida Atlantic University for the South Florida Water Management District. [Online] Available http://iweb/iwebB501/omod/hsm/modflow/ div_system.htm, December 2005.

Wilson, J.D. and Naff, R.L., 2004, The U.S. Geological Survey modular ground-water model --
GMG linear equation solver package documentation: U.S. Geological Survey Open-File Report 2004-
1261, 47 p.

Zheng, Chunmiao, and Wang, P.P. 1999. MT3DMS—A modular three-dimensional multispecies
transport model for simulation of advection, dispersion and chemical reactions of contaminants in ground-
water systems; documentation and user's guide. Jacksonville, Fla., U.S. Army Corps of Engineers
Contract Report SERDP-99-1.

Zheng, Chunmiao, 2006, MT3DMS v5.2 Supplemental User's Guide, Technical Report to the U.S.
Army Engineer Research and Development Center, Department of Geological Sciences, University of
Alabama, 24 p.

Zheng, Chunmiao, Hill, M.C., and Hsieh, P.A., 2001, MODFLOW-2000, the U.S. Geological
Survey modular ground-water model - User guide to the LMT6 package, the linkage with MT3DMS for
multi- species mass transport modeling: U.S. Geological Survey Open-File Report 01-82, 43 p.