MODFLOW6的输出

Li

GWF和GWT模型的二进制输出文件介绍

MODFLOW6计算结果可输出为二进制文件，有几种不同类型的二进制输出文件；

（1）二进制网格文件，包含所有后处理程序的需要信息，快速重构模型网格及连接关系；

（2）相关变量文件，包含模拟结果的二进制文件，如水头；

（3）收支文件，包含连接单元之间的模拟水流和来自驱动软件的水流。

观测也写出到二进制输出文件。

输出变量为DOUBLE PRECISION变量和Integer变量。

二进制文件格式如后文介绍，输出频率和文件类型在Output Control Option和各个软件包输入文件中描述。

二进制网格文件

文件后缀.grb，名称是离散输入文件名字。内容与离散软件包类型有关。GWF模型输出二进制网格文件，GWT模型不会输出。

DIS Grids

Header 1: 'GRID DIS' CHARACTER(LEN=50)
Header 2: 'VERSION 1' CHARACTER(LEN=50)
Header 3: 'NTXT 16' CHARACTER(LEN=50)
Header 4: 'LENTXT 100' CHARACTER(LEN=50)

Read NTXT strings of size LENTXT. Set the number of data records (NDAT) equal to number of lines that do not begin with #.
Definition 0: '#Comment ...' CHARACTER(LEN=LENTXT), comments not presently written
Definition 1: 'NCELLS INTEGER NDIM 0 # ncells' CHARACTER(LEN=LENTXT)
Definition 2: 'NLAY INTEGER NDIM 0 # nlay' CHARACTER(LEN=LENTXT)
Definition 3: 'NROW INTEGER NDIM 0 # nrow' CHARACTER(LEN=LENTXT)
Definition 4: 'NCOL INTEGER NDIM 0 # ncol' CHARACTER(LEN=LENTXT)
Definition 5: 'NJA INTEGER NDIM 0 # nja' CHARACTER(LEN=LENTXT)
Definition 6: 'XORIGIN DOUBLE NDIM 0 # xorigin' CHARACTER(LEN=LENTXT)
Definition 7: 'YORIGIN DOUBLE NDIM 0 # yorigin' CHARACTER(LEN=LENTXT)
Definition 8: 'ANGROT DOUBLE NDIM 0 # angrot' CHARACTER(LEN=LENTXT)
Definition 9: 'DELR DOUBLE NDIM 1 ncol' CHARACTER(LEN=LENTXT)
Definition 10: 'DELC DOUBLE NDIM 1 nrow' CHARACTER(LEN=LENTXT)
Definition 11: 'TOP DOUBLE NDIM 1 nrow*ncol' CHARACTER(LEN=LENTXT)
Definition 12: 'BOTM DOUBLE NDIM 1 ncells' CHARACTER(LEN=LENTXT)
Definition 13: 'IA INTEGER NDIM 1 ncells+1' CHARACTER(LEN=LENTXT)
Definition 14: 'JA INTEGER NDIM 1 nja' CHARACTER(LEN=LENTXT)
Definition 15: 'IDOMAIN INTEGER NDIM 1 ncells' CHARACTER(LEN=LENTXT)
Definition 16: 'ICELLTYPE INTEGER NDIM 1 ncells' CHARACTER(LEN=LENTXT)

DIS Grid

Read NDAT data variables using the definitions defined above.

Record 1: NCELLS INTEGER

Record 2: NLAY INTEGER

Record 3: NROW INTEGER

Record 4: NCOL INTEGER

Record 5: NJA INTEGER

Record 6: XORIGIN DOUBLE

Record 7: YORIGIN DOUBLE

Record 8: ANGROT DOUBLE

Record 9: DELR DOUBLE PRECISION ARRAY SIZE(NCOL)

Record 10: DELC DOUBLE PRECISION ARRAY SIZE (NROW)

Record 11: (TOP(J),J=1,NROW*NCOL) DOUBLE PRECISION ARRAY SIZE(NROW*NCOL)

Record 12: (BOTM(J),J=1,NCELLS) DOUBLE PRECISION ARRAY SIZE(NCELLS)

Record 13: (IA(J),J=1,NCELLS+1) INTEGER ARRAY SIZE(NCELLS+1)

Record 14: (JA(J),J=1,NJA) INTEGER ARRAY SIZE(NJA)

Record 15: (IDOMAIN(J),J=1,NCELLS) INTEGER ARRAY SIZE(NCELLS)

Record 16: (ICELLTYPE(J),J=1,NCELLS) INTEGER ARRAY SIZE(NCELLS)

DISV Grid

DISV网格的二进制网格文件包含关于节点和哪些节点组成一个单元的信息。存储各节点的x,y坐标，存储在VERTICES数组。组成所有单元的节点列表存储于JAVERT数组。

使用IAVERT数组找到任意单元的节点列表。

节点列表是“封闭的”，各单元的第1个节点编号等于最后一个节点编号。

```
DO K = 1, NLAY
  DO N = 1, NCPL
    PRINT *, 'THIS IS CELL (LAYER, ICELL2D): ', K, N
    NVCELL = IAVERT(N+1) - IAVERT(N)
    PRINT*, 'NUMBER OF VERTICES FOR CELL IS', NVCELL
    DO IPOS = IAVERT(N), IAVERT(N + 1) - 1
      IVERT = JAVERT(IPOS)
      X = VERTICES(1,IVERT)
      Y = VERTICES(2,IVERT)
      PRINT *,'  VERTEX PAIR: ', X, Y
    ENDDO
  ENDDO
ENDDO
```

对单元做循环的伪代码

DISV Grid

在DISV二进制网格文件中还存储IA和JA数组。这些数组描述单元连接关系。JA数组中的连接直接对应使用FLOW-JA-FACE记录，写出到收支文件。

DISV二进制网格文件的内容如下：

Header 1: 'GRID DISV' CHARACTER(LEN=50)
Header 2: 'VERSION 1' CHARACTER(LEN=50)
Header 3: 'NTXT 20' CHARACTER(LEN=50)
Header 4: 'LENTXT 100' CHARACTER(LEN=50)

Read NTXT strings of size LENTXT. Set the number of data records (NDAT) equal to number of lines that do not begin with #.

Definition 0: '#Comment ...' CHARACTER(LEN=LENTXT), comments not presently written
Definition 1: 'NCELLS INTEGER NDIM 0 # ncells' CHARACTER(LEN=LENTXT)
Definition 2: 'NLAY INTEGER NDIM 0 # nlay' CHARACTER(LEN=LENTXT)
Definition 3: 'NCPL INTEGER NDIM 0 # ncpl' CHARACTER(LEN=LENTXT)
Definition 4: 'NVERT INTEGER NDIM 0 # nvert' CHARACTER(LEN=LENTXT)
Definition 5: 'NJAVERT INTEGER NDIM 0 # njavert' CHARACTER(LEN=LENTXT)
Definition 6: 'NJA INTEGER NDIM 0 # nja' CHARACTER(LEN=LENTXT)
Definition 7: 'XORIGIN DOUBLE NDIM 0 # xorigin' CHARACTER(LEN=LENTXT)
Definition 8: 'YORIGIN DOUBLE NDIM 0 # yorigin' CHARACTER(LEN=LENTXT)
Definition 9: 'ANGROT DOUBLE NDIM 0 # angrot' CHARACTER(LEN=LENTXT)
Definition 10: 'TOP DOUBLE NDIM 1 ncpl' CHARACTER(LEN=LENTXT)
Definition 11: 'BOTM DOUBLE NDIM 1 ncells' CHARACTER(LEN=LENTXT)
Definition 12: 'VERTICES DOUBLE NDIM 2 2 nvert' CHARACTER(LEN=LENTXT)
Definition 13: 'CELLX DOUBLE NDIM 1 ncpl' CHARACTER(LEN=LENTXT)
Definition 14: 'CELLY DOUBLE NDIM 1 ncpl' CHARACTER(LEN=LENTXT)
Definition 15: 'IAVERT INTEGER NDIM 1 ncpl+1' CHARACTER(LEN=LENTXT)
Definition 16: 'JAVERT INTEGER NDIM 1 njavert' CHARACTER(LEN=LENTXT)

DISV Grid

Definition 17: 'IA INTEGER NDIM 1 ncells+1' CHARACTER(LEN=LENTXT)
Definition 18: 'JA INTEGER NDIM 1 nja' CHARACTER(LEN=LENTXT)
Definition 19: 'IDOMAIN INTEGER NDIM 1 ncells' CHARACTER(LEN=LENTXT)
Definition 20: 'ICELLTYPE INTEGER NDIM 1 ncells' CHARACTER(LEN=LENTXT)

Read NDAT data variables using the definitions defined above.
Record 1: NCELLS INTEGER
Record 2: NLAY INTEGER
Record 3: NCPL INTEGER
Record 4: NVERT INTEGER
Record 5: NJAVERT INTEGER
Record 6: NJA INTEGER
Record 7: XORIGIN DOUBLE
Record 8: YORIGIN DOUBLE
Record 9: ANGROT DOUBLE
Record 10: (TOP(J),J=1,NCPL) DOUBLE PRECISION ARRAY SIZE(NCPL)
Record 11: ((BOTM(J),J=1,NCELLS) DOUBLE PRECISION ARRAY SIZE(NCELLS)
Record 12: ((VERTICES(J,K),J=1,2),K=1,NVERT) DOUBLE PRECISION ARRAY SIZE(2,NVERT)
Record 13: (CELLX(J),J=1,NCPL) DOUBLE PRECISION ARRAY SIZE(NCPL)
Record 14: (CELLY(J),J=1,NCPL) DOUBLE PRECISION ARRAY SIZE(NCPL)
Record 15: (IAVERT(J),J=1,NCPL+1) INTEGER ARRAY SIZE(NCPL+1)
Record 16: (JAVERT(J),J=1,NJAVERT) INTEGER ARRAY SIZE(NJAVERT)
Record 17: (IA(J),J=1,NCELLS+1) INTEGER ARRAY SIZE(NCELLS+1)
Record 18: (JA(J),J=1,NJA) INTEGER ARRAY SIZE(NJA)
Record 19: (IDOMAIN(J),J=1,NCELLS) INTEGER ARRAY SIZE(NCELLS)
Record 20: (ICELLTYPE(J),J=1,NCELLS) INTEGER ARRAY SIZE(NCELLS)

DISU Grids

DISU网格的二进制网格文件包含关于节点和哪些节点组成一个单元的信息，但这与是否用户提供了DISU软件包的信息有关。除了在NPF软件包中指定了XT3D或SAVE_SPECIFIC_DISCHARGE选项以外，都不需要。存储各节点的x,y坐标，存储在VERTICES数组。组成所有单元的节点列表存储于JAVERT数组。使用IAVERT数组找到任意单元的节点列表。节点列表是“封闭的”，各单元的第1个节点等于最后一个节点编号。

DISU Grids

Header 1: 'GRID DISU' CHARACTER(LEN=50)
Header 2: 'VERSION 1' CHARACTER(LEN=50)
Header 3: 'NTXT 10' or 'NTXT 15' CHARACTER(LEN=50)
Header 4: 'LENTXT 100' CHARACTER(LEN=50)

Read NTXT strings of size LENTXT. Set the number of data records (NDAT) equal to number of lines that do not begin with #.
Definition 0: '#Comment ...' CHARACTER(LEN=LENTXT), comments not presently written
Definition 1: 'NODES INTEGER NDIM 0 # nodes' CHARACTER(LEN=LENTXT)
Definition 2: 'NJA INTEGER NDIM 0 # nja' CHARACTER(LEN=LENTXT)
Definition 3: 'XORIGIN DOUBLE NDIM 0 # xorigin' CHARACTER(LEN=LENTXT)
Definition 4: 'YORIGIN DOUBLE NDIM 0 # yorigin' CHARACTER(LEN=LENTXT)
Definition 5: 'ANGROT DOUBLE NDIM 0 # angrot' CHARACTER(LEN=LENTXT)
Definition 6: 'TOP DOUBLE NDIM 1 nodes' CHARACTER(LEN=LENTXT)
Definition 7: 'BOT DOUBLE NDIM 1 nodes' CHARACTER(LEN=LENTXT)
Definition 8: 'IA INTEGER NDIM 1 ncells+1' CHARACTER(LEN=LENTXT)
Definition 9: 'JA INTEGER NDIM 1 nja' CHARACTER(LEN=LENTXT)
Definition 10: 'ICELLTYPE INTEGER NDIM 1 ncells' CHARACTER(LEN=LENTXT)

If vertices are provided in the DISU Package, then 5 additional definitions are included:
Definition 11: 'VERTICES DOUBLE NDIM 2 2 nvert' CHARACTER(LEN=LENTXT)
Definition 12: 'CELLX DOUBLE NDIM 1 nodes' CHARACTER(LEN=LENTXT)
Definition 13: 'CELLY DOUBLE NDIM 1 nodes' CHARACTER(LEN=LENTXT)
Definition 14: 'IAVERT INTEGER NDIM 1 nodes+1' CHARACTER(LEN=LENTXT)
Definition 15: 'JAVERT INTEGER NDIM 1 njavert' CHARACTER(LEN=LENTXT)

Read NDAT data variables using the definitions defined above.
Record 1: NODES INTEGER
Record 2: NJA INTEGER
Record 3: XORIGIN DOUBLE
Record 4: YORIGIN DOUBLE
Record 5: ANGROT DOUBLE
Record 6: (TOP(J),J=1,NODES) DOUBLE PRECISION ARRAY SIZE(NODES)
Record 7: ((BOT(J),J=1,NODES) DOUBLE PRECISION ARRAY SIZE(NODES)
Record 8: (IA(J),J=1,NODES+1) INTEGER ARRAY SIZE(NODES+1)
Record 9: (JA(J),J=1,NJA) INTEGER ARRAY SIZE(NJA)
Record 10: (ICELLTYPE(J),J=1,NCELLS) INTEGER ARRAY SIZE(NCELLS)

DISU Grids

If vertices are provided in the DISU Package, then 5 additional records are included:

Record 11: ((VERT(J,K),J=1,2),K=1,NVERT) DOUBLE PRECISION ARRAY SIZE(2,NVERT)

Record 12: (CELLX(J),J=1,NODES) DOUBLE PRECISION ARRAY SIZE(NODES)

Record 13: (CELLY(J),J=1,NODES) DOUBLE PRECISION ARRAY SIZE(NODES)

Record 14: (IAVERT(J),J=1,NODES+1) INTEGER ARRAY SIZE(NODES+1)

Record 15: (JAVERT(J),J=1,NJAVERT) INTEGER ARRAY SIZE(NJAVERT)

相关变量文件

MODFLOW6中，对于GWF模型TEXT值定义为HEAD，GWT模型的TEXT值定义为CONECNTRATION

单元分配一个IDOMAIN值为0或$1.0x10^{30}$，干的单元分配为值$-1.0x10^{30}$

当使用Newton-Raphson公式时不使用干的单元值。

DIS Grids

对每个驱动周期，时间步和层，数据保存为二进制输出文件，写出下面2个记录：

```
Record 1: KSTP,KPER,PERTIM,TOTIM,TEXT,NCOL,NROW,ILAY
Record 2: ((DATA(J,I,ILAY),J=1,NCOL),I=1,NROW)
```

其中，

KSTP is the time step number;
KPER is the stress period number;
PERTIM is the time value for the current stress period;
TOTIM is the total simulation time;
TEXT is a character string (character*16);
NCOL is the number of columns;
NROW is the number of rows;
ILAY is the layer number; and
DATA is the head data of size (NCOL,NROW,NLAY).

DISV Grids

对每个驱动周期，时间步和层，数据保存为二进制输出文件，写出下面2个记录：

Record 1: KSTP,KPER,PERTIM,TOTIM,TEXT,NCPL,1,ILAY
Record 2: (DATA(J,ILAY),J=1,NCPL)

where

KSTP is the time step number;
KPER is the stress period number;
PERTIM is the time value for the current stress period;
TOTIM is the total simulation time;
TEXT is a character string (character*16);
NCPL is the number of cells per layer;
ILAY is the layer number; and
DATA is the head data of size (NCPL,NLAY).

DISU Grids

对每个驱动周期，时间步和层，数据保存为二进制输出文件，写出下面2个记录：

Record 1: KSTP,KPER,PERTIM,TOTIM,TEXT,NODES,1,1
Record 2: (DATA(N),N=1,NODES)

where

KSTP is the time step number;
KPER is the stress period number;
PERTIM is the time value for the current stress period;
TOTIM is the total simulation time;
TEXT is a character string (character*16);
NODES is the number cells in the model grid;
DATA is unstructured head data of size (NODES).

先进的水流和输移软件包

GWF模型的LAK, SFR和MAW软件包，GWT模型的LKT, SFT, MWT和UZT软件包，的相关变量可保存为二进制文件。表33显示了这些软件包的相关变量的文本识别号与描述。

Table 33. Dependent variable written for advanced flow and transport packages.

Model/Package	TEXT	Description
GWF/LAK	STAGE	Simulated lake stage
GWF/SFR	STAGE	Simulated stream reach stage
GWF/MAW	HEAD	Simulated well head
GWF/UZF	WATER-CONTENT	Simulated unsaturated zone cell water content
GWT/LKT	CONCENTRATION	Simulated lake concentration
GWT/SFT	CONCENTRATION	Simulated stream reach concentration
GWT/MWT	CONCENTRATION	Simulated well concentration
GWT/UZT	CONCENTRATION	Simulated unsaturated zone cell concentration

先进的水流和输移软件包

对每个驱动周期，时间步和层，数据保存为二进制输出文件，写出下面2个记录：

Record 1: KSTP,KPER,PERTIM,TOTIM,TEXT,MAXBOUND,1,1
Record 2: (DATA(N),N=1,MAXBOUND)

where

KSTP is the time step number;
KPER is the stress period number;
PERTIM is the time value for the current stress period;
TOTIM is the total simulation time;
TEXT is a character string (character*16);
MAXBOUND is the number advanced boundary items in the package;
DATA is unstructured dependent variable data of size (MAXBOUND).

模型收支文件

•MODFLOW6可选择性地输出收支(BUDGET)文件，也称之为cell-by-cell水流文件。收支文件保存为二进制格式，可使用如ZONEBUDGET程序做后处理。

•GWF和GWT模型的收支文件包含单元间水流和溶质流动、由于储水变化的流动、来自驱动软件包和先进的驱动软件的流动、与其他模型的交换流动等。收支文件的内容包含所有进出单元的流动。

•用户必须激活Output Control Package和某软件包中的保存流动项开关。

收支文件的格式

下面介绍收支文件的一般形式，这样能创建工具读取收支文件。下面介绍不同网格类型的文件内容。

Record 1: KSTP,KPER,TEXT,NDIM1,NDIM2,-NDIM3
Record 2: IMETH,DELT,PERTIM,TOTIM

IMETH=1: *Read 1D array of size NDIM1*NDIM2*NDIM3.*
Record 3: (DATA(J),J=1,NDIM1*NDIM2*NDIM3)

IMETH=6: *Read text identifiers, auxiliary text labels, and list of information.*
Record 3: TXT1ID1
Record 4: TXT2ID1
Record 5: TXT1ID2
Record 6: TXT2ID2
Record 7: NDAT
Record 8: (AUXTXT(N),N=1,NDAT-1)
Record 9: NLIST
Record 10: ((ID1(N),ID2(N),(DATA2D(I,N),I=1,NDAT)),N=1,NLIST)

where

KSTP is the integer time step number;
KPER is the integer stress period number;
TEXT is a character string (character*16) indicating the flow type;
PERTIM is the double precision time value for the current stress period;
TOTIM is the double precision total simulation time;
NDIM1 is the integer size of first dimension;
NDIM2 is the integer size of second dimension;
NDIM3 is the integer size of third dimension;

收支文件的格式

Record 1: KSTP,KPER,TEXT,NDIM1,NDIM2,-NDIM3
Record 2: IMETH,DELT,PERTIM,TOTIM

IMETH=1: *Read 1D array of size NDIM1*NDIM2*NDIM3.*
Record 3: (DATA(J),J=1,NDIM1*NDIM2*NDIM3)

IMETH=6: *Read text identifiers, auxiliary text labels, and list of information.*
Record 3: TXT1ID1
Record 4: TXT2ID1
Record 5: TXT1ID2
Record 6: TXT2ID2
Record 7: NDAT
Record 8: (AUXTXT(N),N=1,NDAT-1)
Record 9: NLIST
Record 10: ((ID1(N),ID2(N),(DATA2D(I,N),I=1,NDAT)),N=1,NLIST)

IMETH is an integer code that specifies the form of the remaining data;
DELT is the double precision length of the timestep;
PERTIM is the double precision time value for the current stress period;
TOTIM is the double precision total simulation time;
DATA is a double precision array of budget values;
TXT1ID1 is a character string (character*16) containing the first text identifier for information in ID1;
TXT2ID1 is a character string (character*16) containing the second text identifier for information in ID1;
TXT1ID2 is a character string (character*16) containing the model name for information in ID2;
TXT2ID2 is a character string (character*16) containing the package or model name for information in ID2;
NDAT is the number of columns in DATA2D, which is the number of auxiliary values plus 1;
AUXTXT is an array of size NDAT - 1 containing character*16 text names for each auxiliary variable;
NLIST is the size of the list;
ID1 is the first identifying number;
ID2 is the second identifying number, and
DATA2D is a double precision 2D array of size (NDAT,NLIST). The first column in DATA2D is the budget term; any remaining columns are auxiliary variable values.

- 单元间流动
- 离散类型的变化
- GWF模型的收支文件内容
- GWF模型的CSUB软件
- GWF模型的LAK, MAW, SFR和UZF模型
- GWT模型的收支文件内容
- GWT模型的LAT,MWT,UZT软件

观测输出文件

当使用BINARY选项打开一个观测输出文件（见OBS工具），输出文件有如下形式。Record 1有100bytes长度。

Record 1: `TYPE, PRECISION, LENOBSNAME` *(Record 1 includes 85 blanks following LENOBSNAME.)*
Record 2: `NOBS`
Record 3: `OBSNAME(1), OBSNAME(2), ..., OBSNAME(NOBS)`

Repeat for each time step.

Record 4: `TIME, SIMVALUE(1), SIMVALUE(2), ..., SIMVALUE(NOBS)`

where

`TYPE` (bytes 1–4 of Record 1) is "cont " — "cont" indicates the file contains continuous observations;
`PRECISION` (bytes 6–11 of Record 1) will always be "double" to indicate that floating-point values are written in double precision (8 bytes);
`LENOBSNAME` (bytes 12–15 of Record 1) is an integer indicating the number of characters used to store each observation name in following records (in the initial release of MODFLOW 6, LENOBSNAME equals 40);
`NOBS` (4-byte integer) is the number of observations recorded in the file;
`OBSNAME` (LENOBSNAME bytes) is an observation name;
`TIME` (floating-point) is the simulation time; and
`SIMVALUE` (floating-point) is the simulated value.