

MODPATH Version 7: Description of Model Input and Output

By David W. Pollock

MODPATH Version 7.1.000
September 26, 2016

MODPATH is available online at:

<http://water.usgs.gov/ogw/modpath/>

Contents

Description of MODPATH Input Files.....	3
General Structure of Input Data	3
Free-Format Input	3
Array Input.....	3
Name File.....	4
MODPATH Basic Data File	5
Simulation File.....	6
Starting Locations Data.....	11
Starting Location Data – Input Style 1.....	12
Starting Location Data – Input Style 2.....	14
Starting Location Data – Input Style 3.....	16
Starting Location Data – Input Style 4.....	18
MODPATH Unstructured Grid File	20
Time Discretization File	22
Explanation of Variables.....	22
Description of MODPATH Output Files	23
Endpoint File.....	23
File Header.....	23
Endpoint Record	24
Pathline File	25
File Header.....	25
Pathline Record.....	25
Timeseries File.....	26
File Header.....	26
Timeseries Record	26

Description of MODPATH Input Files

General Structure of Input Data

Free-Format Input

Free-format input is used unless otherwise specified. With free-format input, the spacing of values within a record is not fixed—each value can consist of any number of characters. One or more spaces, or a single comma optionally combined with spaces, must separate adjacent values. A value of zero must be explicitly represented as “0” rather than one or more spaces because there is no way to detect the difference between a space that represents a null value and one that represents a value separator. If a character data item containing spaces is included on a line with other data items, that character data item must be enclosed by apostrophes. If the character data item does not contain spaces, the apostrophes are not necessary. In the case where a character data item is the only data item on the line, it should not be enclosed by apostrophes even if it contains spaces.

Array Input

MODPATH uses array reader subroutines similar to those used by MODFLOW (Harbaugh, 2005). MODFLOW array readers are designed to provide a flexible and efficient way of specifying array data. Instructions for reading array data items are provided by array control records. The array control records implemented by MODPATH do not support all of the options supported by MODFLOW.

MODPATH supports free-format control records for integer and real-number arrays. The following control record options are supported:

- Option 1. **CONSTANT** CNSTNT
All values in the array are set equal to the value specified in CNSTNT.
- Option 2. **INTERNAL** CNSTNT FMTIN IPRN
The individual array elements are read from the same file that contains the control record. The value of CNSTNT is used as a multiplier to scale the array values that are read in.
- Option 3. **OPEN/CLOSE** FNAME CNSTNT FMTIN IPRN
The array will be read from the file whose name is specified by FNAME. The file is opened just prior to reading the array data and closed immediately after the array is read. The OPEN/CLOSE option allows the same file to be reused for more than one array, if appropriate.

Explanation of Variables

CNSTNT—is a real-number or integer constant. If the array is being defined as a constant, CNSTNT is the constant value. If individual elements of the array are being read, the values are multiplied by CNSTNT after they are read. When CNSTNT is used as a multiplier and is specified as 0, it is changed to 1.

FMTIN—is the format for reading array elements. The format must contain 20 characters or less. The format must either be (1) a standard FORTRAN format that is enclosed in parentheses, or (2) “(Free)”, which indicates free-format data style.

IPRN—is a flag indicating that the array being read should be printed and a code for indicating the format that should be used. An IPRN value of 0 will produce readable output for all real and integer data. If IPRN is set to a negative number, the array will not be printed. Additional print codes can be specified to customize the format of the printed arrays. The available values of IPRIN are documented in Harbaugh (2005).

Name File

The name file specifies the file names of the flow-system data files used by MODPATH.

Item 1. `FileType`, [`UnitNumber`], `FileName`

`FileType` is a keyword that associates a data file type with the flow system file specified by `FileName`.

`UnitNumber` is an optional integer that specifies the Fortran unit number assigned to the data file.

`UnitNumber` is not required for MODPATH version 7. `UnitNumber` should be used only when it is necessary to maintain backward compatibility with data sets prepared for previous versions of MODPATH. The following file types are supported:

DIS – specifies a traditional structured grid discretization file of the type used by MODFLOW-2005 and MODFLOW-USG. The DIS file contains both spatial and time discretization data.

MPUGRID – specifies the MODPATH unstructured grid file. The MPUGRID file must be used for MODPATH simulations based on MODFLOW-USG simulations that use unstructured grids. Currently MODPATH only supports quad-refined, rectangular unstructured grids. The MODPATH unstructured grid file contains only spatial discretization data.

TDIS – specifies the time discretization file. A time discretization file of type TDIS must be specified for unstructured grid simulations. A separate time discretization file is not specified for structured grid simulations because time discretization data is contained in the DIS file.

MPBAS – specifies the MODPATH basic data file.

HEAD – specifies the binary head output file from the MODFLOW simulation.

BUDGET – is the cell-by-cell budget output file from the MODFLOW simulation.

Comment lines may be placed anywhere throughout the name file. Comment lines are denoted by the presence of characters #, !, or // beginning in column 1.

MODPATH Basic Data File

Item 0. [Text]

Item 0 is an optional comment line. Comment lines are denoted by the presence of characters #, !, or // beginning in column 1. Item 0 can be repeated multiple times.

Item 1. HNOFLO, HDRY

Item 2. DefaultIFaceCount

Item 3. PackageLabel

Item 4. DefaultIFaceValue

Repeat items 3 and 4 for each package for which a default IFACE is specified:

Item 5. LAYTYP (LayerCount)

Item 6. IBOUND (LayerCellCount) – UIDINT

Repeat item 6 for each model layer. For unstructured grids, the number of cells per layer (LayerCellCount) may not be the same for all layers.

Item 6. Porosity (LayerCellCount) – UIDREL

Repeat item 6 for each model layer. For unstructured grids, the number of cells per layer (LayerCellCount) may not be the same for all layers.

Explanation of Variables

HNOFLO – is the value of head used in the MODFLOW simulation to represent inactive cells.

HDRY – is the value of head used in the MODFLOW simulation to represent dry cells.

DefaultIFaceCount – is the number of stress packages for which default values of IFACE are specified.

PackageLabel – is the text string used by MODFLOW in the budget output file to label flow rates for a stress package.

DefaultIFaceValue – the IFACE value used as default for the specified stress package. Values must be in the range of 0 to 6.

LAYTYP – is an array of integer values that indicate how the model layer is represented in the MODFLOW simulation.

0 = Confined

1 = Convertible (confined/unconfined)

IBOUND – is the boundary array for the MODFLOW simulation.

Porosity – is an array of grid cell porosity values.

Simulation File

Item 0. [Text]

Item 0 is an optional comment line. Comment lines are denoted by the presence of characters #, !, or // beginning in column 1. Item 0 can be repeated multiple times.

Item 1. NameFileName

Item 2. ListingFileName

Item 3. SimulationType, TrackingDirection, WeakSinkOption, WeakSourceOption,
BudgetOutputOption, TraceMode

Item 4. EndpointFileName

Include item 5 only if SimulationType = 2 or 4:

Item 5. PathlineFileName

Include item 6 only if SimulationType = 3 or 4:

Item 6. TimeseriesFileName

Include item 7 and item 8 only if TraceMode = 1:

Item 7. TraceFileName

Item 8. TraceParticleGroup, TraceParticleID

Item 9. BudgetCellCount

Include item 10 only if BudgetCellCount > 0:

Item 10. BudgetCellNumbers (BudgetCellCount)

Item 11. ReferenceTimeOption

Include item 12 only if ReferenceTimeOption = 1:

Item 12. ReferenceTime

Include item 13 only if ReferenceTimeOption = 2

Item 13. StressPeriod, TimeStep, TimeStepFraction

Item 14. `StopTimeOption`

Include item 15 only if `StopTimeOption = 3`

Item 15. `StopTime`

Include item 16 only if `SimulationType = 3` or `4`:

Item 16. `TimePointOption`

Include item 17 only if `SimulationType = 3` or `4` and `TimePointOption = 1`:

Item 17. `TimePointCount`, `TimePointInterval`

Include item 18 and item 19 only if `SimulationType = 3` or `4` and `TimePointOption = 2`:

Item 18. `TimePointCount`

Item 19. `TimePoints (TimePointCount)`

Item 20. `ZoneDataOption`

Include item 21 and item 22 only if `ZoneDataOption = 2`

Item 21. `StopZone`

Item 22. `Zones (LayerCellCount) – U1DINT`

Repeat item 22 for each model layer.

Item 23. `RetardationFactorOption`

Include item 24 only if `RetardationFactorOption = 2`:

Item 24. `Retardation (LayerCellCount) – U1DREL`

Repeat item 24 for each model layer.

Item 25. `ParticleGroupCount`

Repeat the data sequence for items 26 through 32 for each particle group:

Item 26. `ParticleGroupName`

Item 27. `ReleaseOption`

Include item 28 only if `ReleaseOption = 1`:

Item 28. `ReleaseTime`

Include item 29 only if ReleaseOption = 2:

Item 29. ReleaseTimeCount, InitialReleaseTime, ReleaseInterval

Include item 30 and item 31 only if ReleaseOption = 3:

Item 30. ReleaseTimeCount

Item 31. ReleaseTimes (ReleaseTimeCount)

Item 32. StartingLocationsFileOption, [StartingLocationsFileName]

If StartingLocationsFileOption = INTERNAL, enter the starting locations data immediately following item 32. Starting locations input instructions are described in section Starting Locations Data.

Explanation of Variables

NameFileName – is the file name of the MODPATH name file.

ListingFileName – is the file name of the MODPATH listing file.

SimulationType – is an integer indicating the type of MODPATH simulation.

- 1 = Endpoint
- 2 = Pathline
- 3 = Timeseries
- 4 = Combined pathline and timeseries

TrackingDirection – is an integer indicating whether the simulation tracks particles forward or backward in time.

- 1 = Forward
- 2 = Backward

WeakSinkOption – is an integer indicating whether particles should be stopped when they enter weak sink cells.

- 1 = Pass through weak sink cells
- 2 = Stop at weak sink cells

WeakSourceOption – is an integer indicating whether particles should be stopped when they enter weak source cells.

- 1 = Pass through weak source cells
- 2 = Stop at weak source cells

BudgetOutputOption – is an integer indicating whether individual cell water balance errors are computed at each time step for all cells in the grid.

- 0 = Individual cell water balance errors are not computed and budget record headers are not printed.
- 1 = A summary of individual cell water balance errors for each time step is printed in the listing file. Budget record headers are not printed in the listing file.
- 2 = A summary of individual cell water balance errors for each time step is printed in the listing file. In addition, budget record headers are printed in the listing file.

TraceMode – is an integer indicating if trace mode is turned on. When trace mode is on, detailed particle tracking information is summarized for one user-specified particle.

- 0 = Trace mode is off
- 1 = Trace mode is on

EndpointFileName – is the name of the MODPATH endpoint file

PathlineFileName – is the name of the MODPATH pathline file

TimeseriesFileName – is the name of the MODPATH timeseries file

TraceFileName – is the name of the MODPATH trace file

TraceParticleGroup and TraceParticleID – are the particle group and particle ID of the specified particle that is followed in detail when trace mode is turned on.

BudgetCellCount – is the number of cells for which detailed water budgets are computed.

BudgetCellNumbers – is an array of cell numbers for which detailed water budgets are computed.

ReferenceTimeOption – is an integer indicating how the reference time is defined.

- 1 = Reference time is defined directly by specifying a value of time.
- 2 = Reference time is defined by specifying a stress period, time step, and relative time position within the time step.

ReferenceTime – is the value of MODFLOW simulation time that marks the beginning of a MODPATH particle tracking simulation. MODPATH tracking time is measured relative to the reference time.

StressPeriod, TimeStep, and TimeFraction – specify a time step and relative position within the time step that is used to calculate the reference time. TimeFraction varies from 0 to 1, where 0 is the beginning of the time step and 1 is the end of the time step.

StopTimeOption – is an integer indicating how a particle tracking simulation is terminated based on time.

- 1 = For forward tracking, stop the particle tracking simulation at the end the final time step of the MODFLOW simulation. For backward tracking, stop the particle tracking simulation at the beginning of the first time step of the MODFLOW simulation.
- 2 = Extend initial or final steady-state time steps and track all particles until they reach a termination location.
- 3 = Specify a specific value of tracking time at which to stop the particle tracking simulation.

StopTime – is a user-specified value of tracking time at which to stop a particle tracking simulation if StopTimeOption 3 is selected.

TimePointOption – is an integer indicating how time point data should be specified for MODPATH simulations that involve timeseries output (simulation types 3 and 4).

TimePointCount – is the number of time points.

TimePointInterval – is the uniform interval of time between time points when TimePointOption 1 is selected.

TimePoints – is an array of time points that is used when TimePointOption 2 is selected.

ZoneDataOption – is an integer indicating how zone data is specified.

- 1 = Zone array data is not read. The zone value for all cells is automatically assigned a value of 1
- 2 = Zone array data is read.

StopZone – a specified integer zone value that indicates an automatic stopping location for particles. A value of 0 indicates no automatic stop zone. Negative values are not allowed.

Zones – is an array of positive integer zones. The zones array is read by layer when ZoneDataOption 2 is selected.

RetardationFactorOption – is an integer indicating how retardation data is specified.

- 1 = Retardation data is not read. The retardation factor for all cells is automatically assigned a value of 1.
- 2 = Retardation array data is read

Retardation – is an array of retardation factors. The retardation array is read by layer when RetardationFactorOption 2 is selected.

ParticleGroupCount – is the number of particle groups.

ParticleGroupName – is a user-defined name for a particle group. Particle group names are limited to 16 characters or less.

ReleaseOption – is an integer indicating how release times are specified for the particle group.

- 1 = A single value of tracking time is specified as the release time for the particle group.
- 2 = Particles are released at multiple points in time spaced at uniform time intervals over a period of time.
- 3 = Particles are released at multiple points in time based on an array of release time values.

ReleaseTime – is the value of tracking time that defines when particles in a particle group are released when ReleaseOption is 1.

ReleaseTimeCount – is the number of multiple release times. ReleaseTimeCount is specified if ReleaseOption is 2 or 3.

InitialReleaseTime – is the value of tracking time at which the first particles are released when multiple releases are specified using ReleaseOption 2.

ReleaseInterval – is the uniform interval of time between particle releases when ReleaseOption is 2.

ReleaseTimes – is an array of release times that is specified when ReleaseOption is 3.

StartingLocationsFileOption – is a keyword indicating what file the starting locations data should be read from.

- EXTERNAL = starting locations are read from and external file
- INTERNAL = starting locations are contained in the simulation file.

StartingLocationsFileName – is the name of the starting locations file that is read when the EXTERNAL option is specified for the variable StartingLocationsFileOption.

Starting Locations Data

The starting locations data input described in the following sections defines the starting locations of particles in a particle group. For each specific particle group defined in the simulation file, starting location data can be contained in a separate file that is referenced by name, or the data can be included directly in the simulation file (see section Simulation File).

Item 0. [Text]

Item 0 is an optional comment line. Comment lines are denoted by the presence of characters #, !, or // beginning in column 1. Item 0 can be repeated multiple times.

Item 1. InputStyle

InputStyle – is an integer flag indicating how particle starting locations are specified.

- | | |
|----------------|--|
| InputStyle = 1 | A list of individual particle starting locations is specified using either layer-row-column grid indices or cell numbers. Input style 1 is the most general input style and provides the most flexibility in customizing starting locations. |
| InputStyle = 2 | Particle starting locations are generated automatically by MODPATH using a particle placement template and a list of grid cell regions specified by layer, row, and column. The template specifies how arrays of particles are placed on cell faces or within a cell. Particles are generated for each cell in the grid cell region based on the specifications of the template. Automatic particle generation based on a particle placement template provides a method for generating large numbers of particles efficiently with minimal data input. Input style 2 only can be used with structured grids. |
| InputStyle = 3 | Particle starting locations are generated automatically by MODPATH using a particle placement template and a list of grid cells specified by cell number. The template specifies how arrays of particles are placed on cell faces or within a cell. Particles are generated for each cell in the list based on the specifications of the template. Input style 3 is analogous to input style 2 except that cell locations are specified by cell number rather than grid index. Input style 3 can be used with both structured and unstructured grids. |
| InputStyle = 4 | Particle starting locations are generated automatically using the same type of particle placement template described above for input styles 2 and 3. However, for input style 4 the cells associated with the template are defined by a three dimensional mask array rather than a list of cells. For many situations, the use of a mask array is a more compact way of defining the cells associated with the template. The use of a mask array also is convenient for pre-processing applications in many cases. Input style 4 can be used with both structured and unstructured grids. |

Depending on the values of InputStyle, include data from one of the following sections:

- If InputStyle = 1 – section *Starting Location Data – Input Style 1*
- If InputStyle = 2 – section *Starting Location Data – Input Style 2*
- If InputStyle = 3 – section *Starting Location Data – Input Style 3*
- If InputStyle = 4 – section *Starting Location Data – Input Style 4*

Starting Location Data – Input Style 1

Item 2. `LocationStyle`

Item 3. `ParticleCount, ParticleIdOption`

Include item 4 only if `LocationStyle` is 1:

Item 4. `[ID], Layer, Row, Column, LocalX, LocalY, LocalZ, TimeOffset, Drape`

Repeat item 4 `ParticleCount` times (once for each particle).

Include item 5 only if `LocationStyle` is 2:

Item 5. `[ID], CellNumber, LocalX, LocalY, LocalZ, TimeOffset, Drape`

Repeat item 5 `ParticleCount` times (once for each particle).

Explanation of Variables

`ParticleCount` – is the number of particles in the particle group.

`ParticleIdOption` – is a flag indicating if a user-specified particle ID is provided.

If `ParticleIdOption` = 0 No particle ID is specified. MODPATH automatically generates an ID for each particle.

If `ParticleIdOption` = 1 A user-specified particle ID is read for each starting location in item 4.

`LocationStyle` – is an integer flag indicating if grid cells are specified by layer, row, column index, or by cell number. Traditional structured grids may specify starting locations using either cell index or cell number. Unstructured grids must specify starting locations using cell number.

`LocationStyle` = 1 Layer, row, and column indices are used to specify starting locations

`LocationStyle` = 2 Cell numbers are used to specify starting locations

`ID` – is an optional user-specified particle ID that is included or omitted depending on the value of `ParticleIdOption`.

`Layer, Row, and Column` – are the indices of the cell containing the particle.

`LocalX, LocalY, and LocalZ` – are local coordinates that define the location of the particle within the cell. Local coordinates vary from 0 to 1 in the positive x, y, and z directions.

`TimeOffset` – is a value of tracking time that is added to the release time specified for the particle group in the simulation file.

In most cases, the value of `TimeOffset` should be set to 0. However, by setting the release time for the particle group to 0 in the simulation file, the `TimeOffset` value can be used to directly specify the release time for each particle. That approach makes it possible for data pre-processing applications to create nearly any configuration of starting locations using this style of starting location data.

Drape – is an integer indicating how particles are treated when starting locations are specified for cells that are dry.

If Drape = 0	Particles are placed in the specified cell. If the cell is dry at the time of release, the status of the particle is set to unreleased and removed from the simulation.
--------------	---

If Drape = 1	Particles are placed in the upper most active grid cell directly beneath the specified layer, row, column location.
--------------	---

Starting Location Data – Input Style 2

Item 2. ParticleTemplateCount, TotalCellRegionCount

Repeat the input sequence described below for items 3 through 6 ParticleTemplateCount times (once for each particle template).

Item 3. TemplateSubdivisionType, TemplateCellRegionCount, Drape

Include item 4 only if TemplateSubdivisionType = 1:

Item 4. VerticalDivisions1, HorizontalDivisions1, VerticalDivisions2, HorizontalDivisions2, VerticalDivisions3, HorizontalDivisions3, VerticalDivisions4, HorizontalDivisions4, RowDivisions5, ColumnDivisions5, RowDivisions6, ColumnDivisions6

Include item 5 only if TemplateSubdivisionType = 2,

Item 5. ColumnCellDivisions, RowCellDivisions, LayerCellDivisions

Repeat item 6 TemplateCellRegionCount times (once for each cell region associated with the template).

Item 6. MinLayer, MinRow, MinColumn, MaxLayer, MaxRow, MaxColumn

Each cell region in item 6 is entered on a separate line.

Explanation of Variables

ParticleTemplateCount – is the number of particle templates.

TotalCellRegionCount – is the total number of cell regions. The value of TotalCellRegionCount is equal to the sum of the number of cell regions associated with each particle template.

TemplateSubdivisionType – is a flag that specifies how particles are placed in the cells.

TemplateSubdivisionType = 1 Two-dimensional arrays of particles are assigned to cell faces.

TemplateSubdivisionType = 2 Three-dimensional arrays of particles are placed within cells.

TemplateCellRegionCount – is the number of rectangular cell regions that define the cells associated with the template.

Drape – is a flag indicating how particles are treated when starting locations are specified for cells that are dry.

If Drape = 0 Particles are placed in the specified cell. If the cell is dry at the time of release, the status of the particle is set to unreleased and removed from the simulation.

If Drape = 1 Particles are placed in the upper most active grid cell directly beneath the specified layer, row, column location.

VerticalDivisions1 and HorizontalDivisions1 – are the number of vertical and horizontal subdivisions that define the two-dimensional array of particles on cell face 1 when TemplateSubdivisionType = 1 . Similar definitions hold for faces 2, 3, and 4.

RowDivisions5 and ColumnDivisions5 – are the number of row and column subdivisions that define the two-dimensional array of particles on the bottom cell face (face 5) when TemplateSubdivisionType = 1. Similar definitions apply for the top face (face 6). The row and column orientation of the subdivisions correspond to the row and column orientation of the model grid. If the grid is unstructured, the row and column orientation of the subdivisions correspond to the row and column orientation of the base grid used to define the unstructured grid.

ColumnCellDivisions, RowCellDivisions, and LayerCellDivisions – define the three dimensional array of particles placed within cells when TemplateSubdivisionType = 2.

MinLayer, MinRow, MinColumn, MaxLayer, MaxRow, and MaxColumn – are the grid cell indices that define a three-dimensional region of cell associated with a particle template.

Starting Location Data – Input Style 3

Item 2. `ParticleTemplateCount, TotalCellCount`

Repeat the input sequence described below for items 3 through 6 `ParticleTemplateCount` times (once for each particle template).

Item 3. `TemplateSubdivisionType, TemplateCellCount, Drape`

Include item 4 only if `TemplateSubdivisionType = 1`:

Item 4. `VerticalDivisions1, HorizontalDivisions1, VerticalDivisions2,
HorizontalDivisions2, VerticalDivisions3, HorizontalDivisions3,
VerticalDivisions4, HorizontalDivisions4, RowDivisions5,
ColumnDivisions5, RowDivisions6, ColumnDivisions6`

Include item 5 only if `TemplateSubdivisionType = 2`,

Item 5. `ColumnCellDivisions, RowCellDivisions, LayerCellDivisions`

Item 6. `TemplateCellNumbers (TemplateCellCount)`

For item 6, more than one cell number may be entered per line. Use as many lines as necessary to include all of the cells associated with the template. Values are entered in free format style.

Explanation of Variables

`ParticleTemplateCount` – is the number of particle templates.

`TotalCellRegionCount` – is the total number of cell regions. The value of `TotalCellRegionCount` is equal to the sum of the number of cell regions associated with each particle template.

`TemplateSubdivisionType` – is a flag that specifies how particles are placed in the cells.

`TemplateSubdivisionType = 1` Two-dimensional arrays of particles are assigned to cell faces.

`TemplateSubdivisionType = 2` Three-dimensional arrays of particles are placed within cells.

`TemplateCellRegionCount` – is the number of rectangular cell regions that define the cells associated with the template.

`Drape` – is a flag indicating how particles are treated when starting locations are specified for cells that are dry.

If `Drape = 0` Particles are placed in the specified cell. If the cell is dry at the time of release, the status of the particle is set to unreleased and removed from the simulation.

If `Drape = 1` Particles are placed in the upper most active grid cell directly beneath the specified layer, row, column location.

VerticalDivisions1 and HorizontalDivisions1 – are the number of vertical and horizontal subdivisions that define the two-dimensional array of particles on cell face 1 when TemplateSubdivisionType = 1 . Similar definitions hold for faces 2, 3, and 4.

RowDivisions5 and ColumnDivisions5 – are the number of row and column subdivisions that define the two-dimensional array of particles on the bottom cell face (face 5) when TemplateSubdivisionType = 1. Similar definitions apply for the top face (face 6). The row and column orientation of the subdivisions correspond to the row and column orientation of the model grid. If the grid is unstructured, the row and column orientation of the subdivisions correspond to the row and column orientation of the base grid used to define the unstructured grid.

ColumnCellDivisions, RowCellDivisions, and LayerCellDivisions – define the three dimensional array of particles placed within cells when TemplateSubdivisionType = 2.

TemplateCellNumbers – is an array of cell numbers that defines a group of cells associated with a particle template.

Starting Location Data – Input Style 4

Item 2. `TemplateSubdivisionType, Drape`

Item 3. `MaskValueCount`

Include item 4 only if `MaskValueCount` is greater than 0.

Item 4. `MaskValues (MaskValueCount)`

Repeat item 5 for each layer in the grid.

Item 5. `Mask (LayerCellCount)` – read with array reader utility U2DINT

Include item 6 only if `TemplateSubdivisionType = 1`:

Item 6. `VerticalDivisions1, HorizontalDivisions1, VerticalDivisions2,
HorizontalDivisions2, VerticalDivisions3, HorizontalDivisions3,
VerticalDivisions4, HorizontalDivisions4, RowDivisions5,
ColumnDivisions5, RowDivisions6, ColumnDivisions6`

Include item 7 only if `TemplateSubdivisionType = 2`,

Item 7. `ColumnCellDivisions, RowCellDivisions, LayerCellDivisions`

Explanation of Variables

`TemplateSubdivisionType` – is a flag that specifies how particles are placed in the cells.

`TemplateSubdivisionType = 1` Two-dimensional arrays of particles are assigned to cell faces.

`TemplateSubdivisionType = 2` Three-dimensional arrays of particles are placed within cells.

`Drape` – is a flag indicating how particles are treated when starting locations are specified for cells that are dry.

If `Drape = 0` Particles are placed in the specified cell. If the cell is dry at the time of release, the status of the particle is set to unreleased and removed from the simulation.

If `Drape = 1` Particles are placed in the upper most active grid cell directly beneath the specified layer, row, column location.

`MaskValueCount` – is the number of unique, positive values in the `Mask` array that are used to define the cells associated with the particle placement template. When `MaskValueCount` is equal to 0, all positive values in the `Mask` array are used to define the cells associated with the template.

`Mask` – is an integer array whose values are used to define the cells in a model layer that are associated with the template. The cell values specified in the mask array may refer to structured or unstructured grids.

LayerCellCount – is the number of cells in a model layer. In unstructured grids the number of cells in a model layer may vary from one layer to another.

MaskValues – are the positive integer values used to define the mask.

VerticalDivisions1 and HorizontalDivisions1 – are the number of vertical and horizontal subdivisions that define the two-dimensional array of particles on cell face 1 when TemplateSubdivisionType = 1 . Similar definitions hold for faces 2, 3, and 4.

RowDivisions5 and ColumnDivisions5 – are the number of row and column subdivisions that define the two-dimensional array of particles on the bottom cell face (face 5) when TemplateSubdivisionType = 1. Similar definitions apply for the top face (face 6). The row and column orientation of the subdivisions correspond to the row and column orientation of the model grid. If the grid is unstructured, the row and column orientation of the subdivisions correspond to the row and column orientation of the base grid used to define the unstructured grid.

ColumnCellDivisions, RowCellDivisions, and LayerCellDivisions – define the three dimensional array of particles placed within cells when TemplateSubdivisionType = 2.

MODPATH Unstructured Grid File

The MODPATH unstructured grid file contains spatial discretization data. It does not contain time discretization data. MODPATH is restricted to a narrow range of unstructured grids that meet the following criteria:

- The grid is a rectangular-based network of 3-dimensional cells.
- The grid network must be based on a traditional, structured MODFLOW base grid. Unstructured grid cells are generated by subdividing base grid cells laterally within a grid layer using quad-refinement to produce the desired distribution of grid cells. For a grid that is quad-refined laterally within a layer, each successive level of refinement decreases the horizontal cell size by a factor of $\frac{1}{2}$ in each direction. The refinement process can be repeated as many times as necessary until a sufficient level of grid refinement is achieved. Vertical subdivisions within grid layers are not allowed. Cells are numbered sequentially by grid layer beginning with layer 1.
- The difference between the level of refinement of adjacent grid cells must be less than or equal to 1. That means each grid cell may be connected to 0, 1, or 2 potential neighbor cells adjacent to each of its four lateral cell faces. A grid cell may be connected to 0, 1, or 4 potential neighbor cells adjacent to its bottom and top faces.
- Sub-face areas of connected cells must be equal in size and must add up to the total area of the cell face to which they are connected. To illustrate this requirement, consider a cell (n) with dimensions (DX_n , DY_n , and DZ_n). If n is connected to two cells on its right face, each of those connecting cells must have a sub-face area equal to $0.5DY_n * DZ_n$. If n is connected to 4 cells on its top face, each of those connecting cells must have a sub-face area equal to $0.5DX_n * 0.5DY_n$.
- Reduced grids are allowed. A reduced grid is one in which grid cells in inactive areas are completely removed from the grid so that they do not appear in the MODFLOW simulation.

All scalar data is space delimited. Arrays are read using the standard 1-dimensional MODFLOW array reader (U1DINT) that allows array values to be read directly from the discretization file or from a separate external file. The X-Y node coordinates are relative to a model coordinate system that has its X-Y origin at the lower-left corner of the MODFLOW base grid as seen in map view.

Item 0. [Text]

Item 0 is an optional comment line. Comment lines are denoted by the presence of characters #, !, or // beginning in column 1. Item 0 can be repeated multiple times.

Item 1. CellCount LayerCount RowCount ColumnCount ConnectionsCount

CellCount = number of grid cells

LayerCount = number of model layers

RowCount = number of rows in the base grid

ColumnCount = number of columns in the base grid

ConnectionsCount = sum total of potential connections for all cells in the grid

Item 2. DX (COLUMNCOUNT) – U1DREL

Grid spacing in the X-direction for the base grid cells. DX corresponds to the spacing along a row in a structured MODFLOW grid (DELR).

Item 3. DY (ROWCOUNT) – U1DREL

Grid spacing in the Y-direction for the base grid cells. DY corresponds to the spacing along a column in a structured MODFLOW grid (DELC).

Item 4. CellNumber Layer Row Column Level X Y Bottom Top

Repeat item 4 for each cell in the grid.

CellNumber = unstructured grid cell number

Layer, Row, Column = cell coordinates of the base grid cell containing the unstructured grid cell
 Level = refinement level of the unstructured grid cell which is used to compute the size of cell relative to the base grid cell that contains it
 X, Y = areal coordinates of the grid cell node located at the cell centroid
 Bottom = elevation of cell bottom
 Top = elevation of cell top

- Item 5. CellNumber LeftCount RightCount FrontCount BackCount BottomCount TopCount
 LeftCon(LeftCount) RightCon(RightCount) FrontCon(FrontCount) BackCon(BackCount)
 BottomCon(BottomCount) TopCon(TopCount)
 Repeat item 5 for each cell in the grid.

LeftCount specifies the number of potential cell connections for the left cell face. Similar definitions apply to the connection count variables for the other faces. LeftCon is an integer array containing the cell numbers of the cells that are potential connections for the left face. Similar definitions apply to the other faces. A potential connection is defined one that would exist in a full, unclipped unstructured rectangular grid. For a reduced grid, some of the cells present in the full grid are absent. Potential connections that correspond to missing cells must be represented by a placeholder value of 0 in the connection arrays. No connection data is written for faces that have no potential connections. Figure 1 shows the order of the connected faces for each of a cells 6 faces. Preprocessing programs may number grid cells in any order, but the location of those cell connections in the connection arrays (LeftCon, RightCon, FrontCon, BackCon, BottomCon, and TopCon) must correspond to the order shown in figure 1.

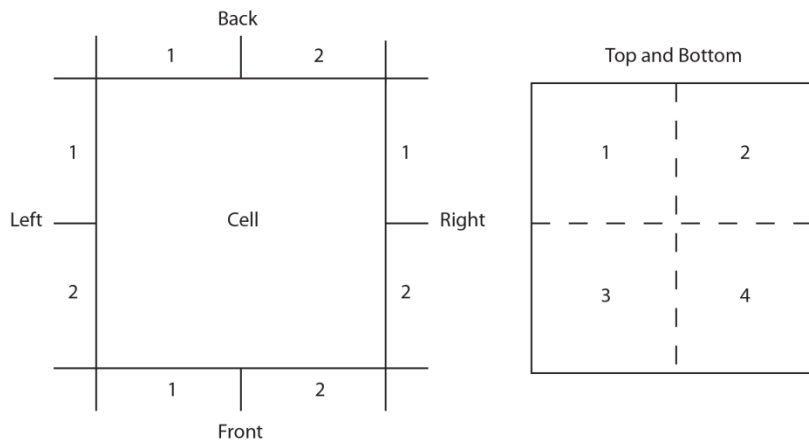


Figure 1. Plan view of grid cell illustrating how connecting grid cells are ordered.

Time Discretization File

For standard, structured grid simulations MODPATH relies on the MODFLOW discretization file for spatial and time discretization data and does not require the time discretization file. However, for unstructured grid simulations the spatial and time discretization data is split into separate files. Spatial discretization is contained in the MODPATH unstructured grid file and time discretization data is contained in the time discretization file. The format of time data in the MODPATH time discretization file is the same as that in the MODFLOW discretization file.

Item 0. [Text]

Item 0 is an optional comment line. Comment lines are denoted by the presence of characters #, !, or // beginning in column 1. Item 0 can be repeated multiple times.

Item 1. StressPeriodCount

Item 2. PeriodLength, TimeStepCount, TimeStepMultiplier, PeriodType

Repeat item 2 for each stress period in the simulation.

Explanation of Variables

StressPeriodCount – is the number of stress periods in the simulation

PeriodLength – is the length of the stress period in model time units

TimeStepCount – is the number of time steps in the stress period

TimeStepMultiplier – is the multiplication factor used to increase the length of each successive time step in the stress period

PeriodType – is a keyword indicating whether the stress period is steady-state or transient

SS = Steady state

TR = Transient

Description of MODPATH Output Files

Endpoint File

The endpoint file is a text file that consists of a series of header lines followed by a sequence of one-line endpoint records for each particle that was released during the simulation. All data are free-format. Multiple data items on a single line are separated by one or more spaces. An endpoint file is generated for all MODPATH simulations.

File Header

Item 1. Label, Version, Revision

Label = MODPATH_ENDPOINT_FILE

Version = 7

Revision = 0

Item 2. TrackingDirection, TotalCount, ReleaseCount, MaximumID, ReferenceTime

TrackingDirection – Forward = 1, Backward = 2

TotalCount—is the total number of particles allocated for the simulation, including particles that may not actually be released as active during the simulation.

ReleaseCount—is the number of particles that were actually released as active during the simulation. Only particles released as active are recording in the endpoint file. Therefore, the number of particle records is always equal to the value of ReleaseCount.

MaximumID—is the maximum particle ID value recorded in the file.

ReferenceTime – is the reference time for the simulation

Item 3. StatusCount(10)

StatusCount—is an integer array containing the number of particles in each of the 10 status categories at the end of the simulation. The status categories, in the order they appear in the array, are:

0 = Pending release

1 = Active

2 = Terminated at boundary face

3 = Terminated in weak sink cell

4 = Terminated in weak source cell

5 = Terminated in cell with no exit face

6 = Terminated in cell with specified zone number

7 = Terminated in an inactive cell

8 = Permanently unreleased

9 = Terminated for unknown reason

Item 3 consists of a single line with ten integer values corresponding to the elements of StatusCount.

Item 4. ParticleGroupCount

ParticleGroupCount – is the number of particle groups for the MODPATH simulation

Repeat item 5 ParticleGroupCount times

Item 5. ParticleGroupName

ParticleGroupName – is the name of the particle group as specified in the simulation file. The names are listed in the numerical order that they appear in the simulation file.

Item 6. The end of the header is marked by the following line of text that begins in column 1:

END HEADER

Endpoint Record

The endpoint file contains an endpoint record for each particle released during the simulation. A particle record consists of a single line of text that contains 26 space-delimited data items. The data items, listed in the order that they appear on the line, are defined as follows:

- | | | |
|--------------------------|-----------------------|--------------------|
| 1. Sequence number | 11. Initial local z | 21. Final local z |
| 2. Particle Group | 12. Initial global x | 22. Final global x |
| 3. Particle ID | 13. Initial global y | 23. Final global y |
| 4. Status | 14. Initial global z | 24. Final global z |
| 5. Initial tracking time | 15. Initial zone | 25. Final zone |
| 6. Final tracking time | 16. Initial face | 26. Final face |
| 7. Initial cell number | 17. Final cell number | |
| 8. Initial layer | 18. Final layer | |
| 9. Initial local x | 19. Final local x | |
| 10. Initial local y | 20. Final local y | |

Endpoint records are not written for particles that are pending release at the end of the simulation or permanently unreleased for any reason (status category codes 0 and 8, respectively).

Pathline File

The pathline file is a text file that consists of a series of file header lines followed by a sequence of particle pathline records. All data are free-format. Multiple data items on a single line are separated by one or more spaces. Each pathline record defines the path of a single particle. The pathline record consists of a header record followed by a series of one-line data records that contain coordinate and tracking time information for each point in the pathline. Examples of pathline files are provided with the example problems included with the software distribution.

File Header

Item 1. Label, Version, Revision

Label = MODPATH_PATHLINE_FILE

Version = 7

Revision = 0

Item 2. TrackingDirection, ReferenceTime

TrackingDirection – Forward = 1, Backward = 2

ReferenceTime – is the reference time for the simulation

Item 3. The end of the header is marked by the following line of text that begins in column 1:

END HEADER

Pathline Record

Item 4. SequenceNumber, Group, ParticleID, PathlinePointCount

SequenceNumber – is a globally unique number generated by MODPATH for each particle.

Group – is the particle group number.

ParticleID – is an integer that is either specified by the user or automatically generated by MODPATH. In contrast to sequence numbers, ParticleID values are unique within a single particle group but the same ParticleID may be present in more than one group.

PathlinePointCount – is the number of points contained in the pathline.

Item 5. CellNumber, GlobalX, GlobalY, GlobalZ, TrackinTime, LocalX, LocalY, LocalZ, Layer, StressPeriod, TimeStep

Item 5 is repeated ParticlePointCount times.

Timeseries File

The timeseries file is a text file that consists of a series of header lines followed by a sequence of one-line data records of particle locations at time points specified in the MODPATH simulation file. All data are free-format. Multiple data items on a single line are separated by one or more spaces. A timeseries file is produced for MODPATH simulation types 3 (timeseries) and 4 (combined pathline and timeseries).

File Header

Item 1. Label, Version, Revision

Label = MODPATH_TIMESERIES_FILE

Version = 7

Revision = 0

Item 2. TrackingDirection, ReferenceTime

TrackingDirection – Forward = 1, Backward = 2

ReferenceTime – is the reference time for the simulation

Item 3. The end of the header is marked by the following line of text that begins in column 1:

END HEADER

Timeseries Record

A timeseries record consists of a single line of text containing 14 data items that are separated by one or more spaces. The data items are defined as follows:

- | | |
|-------------------------|--------------|
| 1. Time Point Index | 8. Local x |
| 2. Cumulative Time Step | 9. Local y |
| 3. Tracking Time | 10. Local z |
| 4. Sequence number | 11. Global x |
| 5. Particle Group | 12. Global y |
| 6. Particle ID | 13. Global z |
| 7. Cell number | 14. Layer |