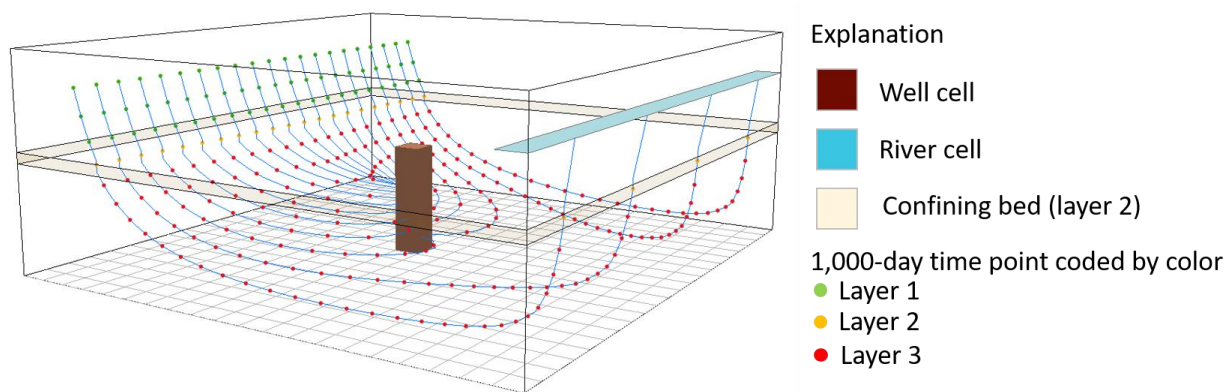




# MODPATH Version 7: Description of Model Input and Output

MODPATH Version 7.2.001  
December 15, 2017



MODPATH is available online at:

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



# Contents

|   |    |
|---|----|
| Overview of Release 7.2.001 .....                                     | 5  |
| Major Changes .....   | 5  |
| MODPATH-6 Support.....  | 5  |
| MODPATH-USG Support.....  | 5  |
| Bug Fixes .....   | 5  |
| Description of MODPATH Input Files.....                               | 5  |
| General Structure of Input Data .....                                 | 5  |
| Free-Format Input .....   | 5  |
| Array Input.....  | 5  |
| Grid Discretization.....  | 6  |
| DIS Grid Characteristics and Requirements .....                       | 7  |
| DISU Grid Characteristics and Requirements .....                      | 7  |
| GRBDIS Grid Characteristics and Requirements .....                    | 8  |
| GRBDISV Grid Characteristics and Requirements .....                   | 8  |
| Time Discretization.....  | 8  |
| Name File.....  | 8  |
| GRIDMETA Data File.....   | 10 |
| MODFLOW-USG Unstructured Grids (DISU) .....                           | 10 |
| MODFLOW-2005 and MODFLOW-USG Structured Grids (DIS) .....             | 10 |
| MODPATH Basic Data File .....   | 11 |
| Simulations based on MODFLOW-2005 and MODFLOW-USG (DIS and DISU)..... | 11 |
| Simulations based on MODFLOW-6 (GRBDIS, GRBDISV).....                 | 11 |
| Simulation File.....  | 13 |
| Starting Locations Data.....  | 19 |
| Starting Location Data – Input Style 1.....                           | 20 |
| Starting Location Data – Input Style 2.....                           | 22 |
| Starting Location Data – Input Style 3.....                           | 24 |
| Starting Location Data – Input Style 4.....                           | 26 |
| Description of MODPATH Output Files .....                             | 28 |
| Endpoint File.....  | 28 |
| File Header.....  | 28 |
| Endpoint Record .....   | 29 |
| Pathline File .....   | 30 |
| File Header.....  | 30 |

|                         |    |
|-------------------------|----|
| Pathline Record.....    | 30 |
| Timeseries File.....    | 31 |
| File Header.....        | 31 |
| Timeseries Record ..... | 31 |

# Overview of Release 7.2.001

## Major Changes

### MODPATH-6 Support

MODPATH release 7.2.001 adds support for MODFLOW-6 structured grids and rectangular quadtree and quadpatch unstructured grids. Unstructured grids are restricted to vertex-based grids of type DISV. DISV grids allow grid refinement in the horizontal direction but require the same horizontal grid to be used for all model layers.

MODPATH reads the spatial grid data for MODFLOW-6 simulations from the binary grid files generated as output by MODFLOW-6. Time discretization data for MODPATH simulations based on MODFLOW-6 is read directly from the MODFLOW-6 time discretization file. Binary grid files and the time discretization file are described in the “MODFLOW 6 – Description of Input and Output” document that is included with the MODFLOW-6 software distribution.

### MODPATH-USG Support

MODPATH-7 no longer supports the MODPATH Unstructured Grid data file (MPUGRID) for MODFLOW-USG unstructured grid data. Instead, MODPATH-7 now reads unstructured grid data for MODFLOW-USG simulations directly from the MODFLOW-USG unstructured discretization file (DISU). Additional, information not available in the DISU file that is needed to completely define the unstructured grid is provided in a supplemental data file referred to as the GRIDMETA data file. Reading the DISU file directly eliminates a large amount of redundant data that previously was read from the MPUGRID file. The DISU file is described in the “MODFLOW-USG Description of Model Input and Output” document that is included with the software distribution.

## Bug Fixes

- Fixed a problem reading the LAYCBD array in the MODFLOW-2005 discretization file.
- Fixed a problem reading forward slash (/) path delimiters in file names in the simulation data file.
- Fixed a problem that caused particles not to stop at cells with a zone number equal to the stop zone when the automatic stop zone option was selected.
- Other minor bug fixes and output format changes.

## 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). 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.

MODPATH does not support the EXTERNAL array control record option. The OPEN/CLOSE option is the only method available for accessing external files.

The way in which MODPATH reads layer data arrays depends on whether the MODPATH simulation is based on a structured or unstructured grid MODFLOW simulation. For structured grid simulations, layer data arrays are read as 2-dimensional row-column arrays, one row at a time for a fixed number of columns. For unstructured grids, layer data arrays are read as single, 1-dimensional arrays with the dimension set to the number of cells in each specific grid layer. Throughout this document, these two styles of array input are referred to as 2-dimensional layer array data and 1-dimensional layer array data, respectively.

### *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.

## **Grid Discretization**

This version of MODPATH supports MODFLOW-2005, MODFLOW-USG, and MODFLOW-6. The style of the grid discretization input data for MODPATH varies depending on the version of MODFLOW. In the case of MODFLOW-USG and MODFLOW-6, the grid discretization input data also depends on the grid type used in the MODFLOW simulation.

MODPATH supports traditional rectangular structured grids as well as specific types of rectangular-based unstructured grids. MODPATH requires the rectangular grid cells be aligned with the x-y axes so that the lateral grid cell sides are parallel to either the x or y axis. Table 1 shows a summary of the grid input data options supported by MODPATH.

Table 1. MODPATH-7 Grid Input Data

| Grid Type Keyword | Description  | Input Files                                       | MODFLOW Version             |
|-------------------|--|---|-----------------------------|
| DIS               | Structured rectangular grid                              | MODFLOW DIS File<br>GRIDMETA File (optional)      | MODFLOW-2005<br>MODFLOW-USG |
| DISU              | Unstructured, quad-refined rectangular grid              | MODFLOW-USG DISU File<br>GRIDMETA File (required) | MODFLOW-USG                 |
| GRBDIS            | Structured rectangular grid                              | MODFLOW-6 binary grid file, grid type DIS         | MODFLOW-6                   |
| GRBDISV           | Unstructured, vertex-based quad-refined rectangular grid | MODFLOW-6 binary grid file, grid type DISV        | MODFLOW-6                   |

## DIS Grid Characteristics and Requirements

DIS grids are the standard structured, rectangular grid type used by all versions of MODFLOW prior to MODFLOW-USG and MODFLOW-6. In plan view, DIS grids are based on an x-y coordinate system with an origin (0, 0) located in the lower-left corner of the grid. That is, for a grid that has NCOL columns and NROW rows, the origin of the grid would be located at the lower left corner of grid cell row NROW, column 1. MODPATH reads the MODFLOW DIS file to obtain grid data. For structured grids, MODFLOW-USG uses the same DIS file format as MODFLOW-2005.

MODPATH allows an optional, supplemental data file, referred to as the GRIDMETA file, to be included in addition to the DIS file. The GRIDMETA file allows users to specify an x-y coordinate offset for the grid origin and an angle of rotation around the origin point. If the GRIDMETA file is omitted, MODPATH assumes the x-y origin is (0, 0) and the angle of rotation is 0. MODPATH does not use the origin offset and rotation data in its particle tracking computations. However, the grid origin offset and rotation data is included in the headers the particle coordinate output files so that particle coordinate output from MODPATH can be transformed into user-defined real-world coordinates, if desired. The GRIDMETA file format is defined later in this document.

MODPATH does not support the following DIS file options:

- Quasi-3d layers are not supported. Set LAYCBD = 0 for all model layers.
- The EXTERNAL option is not supported in array control records.

## DISU Grid Characteristics and Requirements

MODPATH supports a subset of unstructured grids for MODFLOW-USG. MODPATH requires unstructured grids to be rectangular grids based on quadtree or quadpatch grid refinement. MODPATH reads most of the grid information for DISU grids from the MODFLOW-USG DISU file. The grid information read from the DISU file includes cell connectivity and top and bottom elevation data. The DISU file does not contain the cell location and size information necessary to build the grid. That additional information is provided in a supplemental GRIDMETA file. The GRIDMETA file is required for DISU grids.

As with DIS grids, the x-y grid cell locations that define a DISU grid are based on an x-y origin of (0, 0) with an angle of rotation of 0. Origin offset and angle of rotation data can be specified in the GRIDMETA file for inclusion in the particle coordinate output file headers. The GRIDMETA file also includes x-y coordinate data for grid cells as well as grid cell size data. The GRIDMETA file format is defined later in this document.

MODPATH does not support the following DISU file options:

- Symmetric data input for grid cell connections is not supported. Connection data must be read in for the full connection matrix. Set IDSYMRD = 0.
- Vertical subdivision of layers is not supported. Set IVSD = 0.
- Quasi-3d layers are not supported. Set LAYCBD = 0 for all model layers.
- The EXTERNAL option is not supported in array control records.

## GRBDIS Grid Characteristics and Requirements

MODPATH reads grid data for MODFLOW-6 structured grid simulations from the binary grid output file generated by MODFLOW-6. Binary grid files are described in the “MODFLOW 6 – Description of Input and Output” document that is included with the MODFLOW-6 software distribution.

As with structured grids for MODFLOW-2005 and MODFLOW-USG. The grid cell location and size information in the binary grid file is used to generate a grid that has an x-y origin of (0, 0) located in the lower left corner of the grid and a grid rotation angle of 0. The binary grid file for MODFLOW-6 grid type DIS also contains grid origin offset and rotation data that MODPATH writes in the header data of the particle coordinate output files, just as for MODFLOW-2005 and MODFLOW-USG grids.

## GRBDISV Grid Characteristics and Requirements

MODPATH supports unstructured grids for MODFLOW-6 simulations that are based on quadtree or quadpatch refined rectangular grids defined by the discretization-by-vertex (DISV) input format option of MODFLOW-6. The grid data is read from the binary grid output file generated by MODFLOW-6 for grid type DISV. The DISV grid option and binary grid files are described in the “MODFLOW 6 – Description of Input and Output” document that is included with the MODFLOW-6 software distribution.

The DISV input structure in MODFLOW-6 allows a wide range of flexibility in designing unstructured grids in the x-y dimensions, but requires the same x-y grid structure to be used for all model layers. MODPATH further requires that the unstructured DISV grids to be restricted to quadtree or quadpatch refined rectangular grid cells with an angle of rotation of 0 and an origin of (0, 0). Unlike structured grids, the origin point for DISV grids does not need to correspond to a specific location relative to the DISV grid. The binary grid file for MODFLOW-6 grid type DISV also contains grid origin offset and rotation data that MODPATH writes in the header data of the particle coordinate output files, just as for MODFLOW-2005 and MODFLOW-USG grids.

## Time Discretization

Time discretization data for simulations based on MODFLOW-2005 and MODFLOW-USG is read directly from the MODFLOW DIS and DISU files. For simulations based on MODFLOW-6, time discretization data is read directly from the MODFLOW-6 time discretization file (TDIS).

## 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.

DISU – specifies the MODFLOW-USG unstructured grid discretization file. The DISU file contains both spatial and time discretization data.

GRBDIS – specifies the MODFLOW-6 binary grid file for structured grids. The GRBDIS file only contains spatial discretization data.

GRBDISV – specifies the MODFLOW-6 binary grid file for DISV-style unstructured grids. The GRBDISV file only contains spatial discretization data.



GRIDMETA – specifies a supplemental grid discretization data file. The contents of the GRIDMETA file depends on the type of MODFLOW grid. The supplemental grid data file is required for MODPATH simulations based on MODFLOW-USG simulations that use unstructured grids (grid type DISU). The supplemental grid data file is optional for MODFLOW-2005 and MODFLOW-USG structured grids (grid type DIS). The supplemental grid data file is not used for MODFLOW-6 grids (grid types GRBDIS and GRBDISV).

TDIS – specifies the MODFLOW-6 time discretization file. A time discretization file of type TDIS must be specified for all MODPATH simulations based on MODFLOW-6. A separate time discretization file is not specified for MODPATH simulations based on MODFLOW-2005 and MODFLOW-USG because time discretization is included directly in the DIS and DISU files. The TDIS file read by MODPATH is the same file used by MODFLOW-6.

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.

## GRIDMETA Data File

The GRIDMETA file contains supplemental grid data. The GRIDMETA file is required for simulations based on MODFLOW-USG unstructured grids (DISU). The GRIDMETA file is optional for MODFLOW-2005 and MODFLOW-USG structured grid (DIS) simulations. Simulations based on MODFLOW-6 do not require a GRIDMETA file. The structure of GRIDMETA files is described below.

### MODFLOW-USG Unstructured Grids (DISU)

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. XORIGIN, YORIGIN, ANGROT

Item 2. NCELLS, NLAY

Item 3. NCPL(NLAY)

Read item 4 for each grid cell:

Item 4. ICELL, XC, YC, CELLDX, CELLDY

### MODFLOW-2005 and MODFLOW-USG Structured Grids (DIS)

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. XORIGIN, YORIGIN, ANGROT

### *Explanation of Variables*

XORIGIN -- is the x-offset coordinate for the origin.

YORIGIN -- is the y-offset coordinate for the origin.

ANGROT -- is the angle of rotation measured counter-clockwise from the x-axis.

NCELLS -- is the number of grid cells in the MODFLOW grid.

NLAY -- is the number of layers in the MODFLOW grid.

NCPL(NLAY) -- is an integer array containing the number of grid cells for each model layer.

ICELL -- is the cell number.

XC -- is the global x coordinate of the grid cell node.

YC -- is the global y coordinate of the grid cell node.

CELLDX, CELLDY -- are the grid cell sizes in the x and y directions, respectively. For the current version of MODPATH, unstructured grids are restricted to square grids cells (CELLDX = CELLDY).

## MODPATH Basic Data File

Simulations based on MODFLOW-2005 and MODFLOW-USG (DIS and DISU)

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 (NLAY)

Item 6. IBOUND (NCPL) – 1D layer array reader for unstructured grids, or  
IBOUND (NROW, NCOL) – 2D layer array reader for structured grids

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

Item 6. Porosity (NCPL) – 1D layer array reader for unstructured grids, or  
Porosity (NROW, NCOL) – 2D layer array reader for structured grids

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

## Simulations based on MODFLOW-6 (GRBDIS, GRBDISV)

HDRY, HNOFLO, and IBOUND data are omitted from the MODPATH basic data file for simulations based on MODFLOW-6.

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. DefaultIFaceCount

Item 2. PackageLabel

Item 3. DefaultIFaceValue

Repeat items 2 and 3 for each package for which a default IFACE is specified:

Item 4. LAYTYP (NLAY)

Item 5. Porosity (NCPL) – 1D layer array reader for unstructured grids, or  
Porosity (NROW, NCOL) – 2D layer array reader for structured grids

Repeat item 6 for each model layer. For MODFLOW-6 unstructured grids based on the DISV grid style, the number of cells per layer (NCPL) will 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.

NROW – is the number of rows in a structured MODFLOW grid.

NCOL – is the number of columns in a structured MODFLOW grid.

NLAY – is the number of model layers in a MODFLOW grid.

NCPL – is the number of grid cells in a model layer in an unstructured MODFLOW grid.

## 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 (NCPL)` – 1D layer array reader for unstructured grids, or  
`Zones (NROW, NCOL)` – 2D layer array reader for structured grids

Repeat item 22 for each model layer.

Item 23. `RetardationFactorOption`

Include item 24 only if `RetardationFactorOption = 2`:

Item 24. `Retardation (NCPL)` – 1D layer array reader for unstructured grids, or  
`Retardation (NROW, NCOL)` – 2D layer array reader for structured grids

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.

NROW – is the number of rows in a structured MODFLOW grid.

NCOL – is the number of columns in a structured MODFLOW grid.

NCPL – is the number of grid cells in a model layer in an unstructured MODFLOW grid.

## 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.

# 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 = 2

Item 2. TrackingDirection, TotalCount, ReleaseCount, MaximumID, ReferenceTime, XOrigin, YOrigin, AngRot

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

XOrigin—is the offset value of the x origin coordinate.

YOrigin—is the offset value of the y origin coordinate.

AngRot—is the angle of rotation in degrees, measured positive in the counter-clockwise direction around the origin point.

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 = 2

Item 2. TrackingDirection, ReferenceTime, XOrigin, YOrigin, AngRot

TrackingDirection – Forward = 1, Backward = 2

ReferenceTime – is the reference time for the simulation

XOrigin—is the offset value of the x origin coordinate.

YOrigin—is the offset value of the y origin coordinate.

AngRot—is the angle of rotation in degrees, measured positive in the counter-clockwise direction around the origin point.

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 = 2

Item 2. TrackingDirection, ReferenceTime, XOrigin, YOrigin, AngRot

TrackingDirection – Forward = 1, Backward = 2

ReferenceTime – is the reference time for the simulation

XOrigin—is the offset value of the x origin coordinate.

YOrigin—is the offset value of the y origin coordinate.

AngRot—is the angle of rotation in degrees, measured positive in the counter-clockwise direction around the origin point.

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    |