

Offset (OFS) input file format .....	2
Parameter ID file format.....	2
Main input (IN) file format.....	3
Extracting grid cell centre coordinates .....	5
Extracting MapStacks of aquifer properties .....	5
Extracting MapStacks of (input) recharge rates .....	6
Extracting TimeSeries of (input) groundwater abstraction rates .....	7
Extracting TimeSeries of (input) surface water discharge/abstraction rates ....	8
Extracting a map of stream flow directions .....	10
Extracting maps of boundary condition cell locations.....	11
Extracting maps of MODBRANCH branch cell locations .....	11
Extracting diagnostic information from a model simulation .....	12
Extracting MapStacks of model output (heads, flows, etc).....	14
Extracting compressed MapStacks of MODBRANCH output .....	15
Extracting time series of the ground water flow balance .....	16
Updating recharge rates .....	17
Updating Evaporation rates .....	17
Updating groundwater abstraction rates .....	18
Adding new groundwater abstractions .....	19
Updating the surface water discharges / abstractions.....	20
Updating the initial heads.....	21
Removing initial stress periods from input files .....	21
Updating the stress period lengths.....	22

## Offset (OFS) input file format

The offset (OFS) file contains information relating to the real world coordinates of the model and the start date of the simulation. The file should contain the following items:

```
Xoffset, Yoffset  
Rotation  
Start date  
length unit  
geoDatum  
ULXMAP, ULYMAP (optional NW coords)  
XDIM, YDIM (optional cell dimensions)  
XML timestep descriptor (optional)
```

Xoffset and Yoffset are the (e.g. UTM) coordinates of the south western corner of the model domain (the bottom left hand corner of the cell in the first column of the last row).

The rotation is specified in degrees (anticlockwise), with the pivot point of rotation also being around the south western corner of the model domain.

The start date is used to specify the date and time of the start of the simulation in the following format: YYYY-MM-DD hh:mm:ss, with a *single* space between the date and the time.

The length unit can be any character string up to ten characters long (e.g. m).

The geoDatum specifies the coordinate system that is being used. Usually this is "Ordnance Survey Great Britain 1936" (without the quote marks).

The XML timestep descriptor is an optional character string which is written to the TimeSeries and MapStacks XML files created by the Module Adaptor. If omitted, the default (unit="nonequidistant") is used.

## Parameter ID file format

The parameter ID file (ParameterIDs.dat) contains a list of the parameter names used by the NGMS and the corresponding text strings that are used to denote those parameters within the Modflow program. An example of the contents of this file is given below:

```
QSTOR          '          STORAGE '  
QHBND          '    CONSTANT HEAD '  
QX             ' FLOW RIGHT FACE '  
QY             ' FLOW FRONT FACE '  
QZ             ' FLOW LOWER FACE '  
QSRC_GW        '          WELLS '  
QDRA           '          DRAINS '  
QRCH_OUT       '    RECHARGE '  
QEV            '          ET '  
QRIV           '    RIVER LEAKAGE '  
QSTR           '    STREAM LEAKAGE '
```

QSTR_ACCU	'	STREAM FLOW OUT	'
QGHB	'	HEAD DEP BOUNDS	'
H	'	HEAD	'
DDN	'	DRAWDOWN	'
HDIFF_CONV	'	MAXIMUM HEAD CHANGE	'
QRESIDU	'	MAXIMUM RESIDUAL	'
BRANCH_LEAKAGE	'	BRANCH LEAKAGE	'
BRANCH_FLOW	'	BRANCH FLOW OUT	'
BRANCH_STAGE	'	BRANCH STAGE	'
QRCH	'	RECHARGE	'
RCH_LAY	'	RECHARGE LAYER INDEX	'
EVT_LAY	'	ET LAYER INDEX	'
EVT_SURF	'	ET SURFACE	'
EVT_MAX	'	EVAPOTRANSPIRATION RATE	'
EVT_DEPTH	'	EXTINCTION DEPTH	'
STOR1	'	PRIMARY STORAGE COEF	'
KDX	'	TRANSMIS. ALONG ROWS	'
KX	'	HYD. COND. ALONG ROWS	'
C	'	VERT HYD COND /THICKNESS	'
BAS	'	BOTTOM	'
TOP	'	TOP	'
STOR2	'	SECONDARY STORAGE COEF	'
ANTRPY	'	COLUMN TO ROW ANISOTROPY	'
WETDRY	'	WETDRY PARAMETER	'
VKD_GRDK	'	HYD COND GRADIENT FACTOR	'
VKD_UPPR	'	THICKNESS OF UPPER ZONE	'
KDY	'	TRANSMIS. ALONG COLUMNS	'
KY	'	HYD. COND. ALONG COLUMNS	'
VKD_ELVK	'	ELEVATION OF CHANGE IN K	'
VKD_MAXK	'	MAXIMUM HYD COND FACTOR	'
VKD_GRDS	'	STORAGE GRADIENT FACTOR	'
VKD_MAXS	'	MAXIMUM STORAGE FACTOR	'
VKD_ELVS	'	ELEVATION OF CHANGE IN S	'
CELLACT	'	BOUNDARY ARRAY	'
H.ini	'	INITIAL HEAD	'

The shorter text strings (16 characters) relate to parameters which are written to the Modflow binary output files, such as groundwater heads and cell-by-cell flow terms. These should all be contained within the first 40 lines of the ParameterIDs.dat file so that they can be used by the BINARY\_MAPSTACKS command (see Extracting MapStacks of model output (heads, flows, etc)). The longer text strings (24 characters) relate to input parameters (e.g. parameters provided in the Block Centred Flow (BCF) input file).

## Main input (IN) file format

The Module Adaptor has been designed so that it can be run from a command prompt, and input is requested from the user via a number of screen prompts. Usually, however, the Module Adaptor is run by substituting the keyboard input with an input file. The input file (input.in) is specified on the command that runs the Module Adaptor (MFma\_support.exe) in the following way:

```
MFma_support.exe < input.in
```

The contents of the input (IN) file are as follows:

```
Filename of the Diagnostic file
Filename of the Name file
Location ID
Filename of the Offset file
Filename of the file containing parameter IDs
Name of the 'From NGMS' directory
Name of the 'To NGMS' directory
'No data' value
List of Module Adaptor commands
```

A diagnostic file is written every time the Module Adaptor runs. A filename for this file can be specified in the first line of the IN file. If the first line is left blank, the default filename (MFma\_support\_diagnostics.xml) is used.

The second line of the IN file should contain the name of the Modflow "Name" (or NAM) file. This is same file that is required by Modflow to run a Modflow simulation. The file contains a list of the filenames of the input files required for the simulation along with unit numbers and identifiers. Please see the Modflow-96 and Modflow VKD documentation for details of the format of this and other Modflow input files.

The location ID is a character string (up to ~500 characters long) which is used to identify the model, and is used in the filenames of the files created by the Module Adaptor (e.g. "Test and Itchen").

The fourth line of the IN file should contain the filename of the Offset (OFS) file (described above).

The fifth line should contain the name of the file (e.g. ParameterIDs.dat) which contains the list of parameter IDs and the corresponding text strings used by Modflow (described above).

The 'From NGMS' directory is the folder that contains the XML files which have been created by the NGMS system for use by the Module Adaptor (output from NGMS, input to Module Adaptor).

The 'To NGMS' directory is the folder where XML files which have been created by the Module Adaptor will be written (output from Module Adaptor, input to NGMS).

The "no data" value is the value that is written to the XML files created by the Module Adaptor to signify that no data is available for that parameter (e.g. -1.11111e-11).

The list of Module Adaptor commands can be any of the commands described in the following sections of this guide. Any number of commands can be specified. Each command should be specified on a new line. The module adaptor stops processing commands once a blank line has been reached.

## Extracting grid cell centre coordinates

Command: **GRID\_OUT**

Routine(s): BCF\_GRID\_OUT

Source file: BCF\_support.for (MFma\_Write\_Grids\_XML.for)

This command creates a 'Grids' XML file containing the grid cell centre coordinates for each model layer. These coordinates are defined by the model offset and rotation specified in the OFS file, and by the cell dimensions and layer elevations specified in the BCF file.

The z coordinates of the grid cell centres are usually defined as being half way between the top and bottom elevations specified for each model cell (Modflow layer types 3 and 5). However, not all Modflow layer types have top and bottom elevations specified:

- Layer type 0 is a constant transmissivity layer, and neither top or bottom elevations are specified. In this case the z coordinate is given as zero.
- Layer types 1 and 4 are unconfined aquifer layers and no top elevation is specified for them. The z coordinates for these layer types are calculated as being half way between the bottom elevation and the initial head for each model cell. If the initial head value is equal to the dry cell value (representing a cell that has become inactive due to the head falling below the bottom elevation) or if the head is below the bottom elevation, the z coordinate is given as zero.
- Layer type 2 is a constant transmissivity layer where the storage value can change between confined and unconfined values depending on the elevation of the head compared to the top of the layer. In this case the z coordinate is given as the top elevations of the cells in the layer.

The command required in the IN file is GRID\_OUT.

The following output file is written to the "To\_NGMS" folder:

Grids XML file:            LocationID\_Grid.xml

Each grid layer has its own layer definition i.e. an <irregular> section. The order of cells follows the same ordering as a regular grid (i.e. first element is top left, last element is bottom right, row by row).

## Extracting MapStacks of aquifer properties

Command: **AQUIFER\_PROPS** list of parameter IDs

Routine(s): BCF\_OUT

Source file: BCF\_support.for (MFma\_Write\_MapStacks\_XML.for,  
MFma\_Write\_BIL.for)

This command extracts MapStacks XML files of aquifer properties from the block-centred flow (BCF) input file for Modflow.

The command required in the IN file is `AQUIFER_PROPS` followed by the list of parameter IDs for which MapStacks XML files are required. The parameter IDs and their relationship to the text strings used by Modflow are defined in the `ParameterIDs.dat` file. At least one parameter ID should be specified on the command line.

The longer text strings (24 characters long) in the `ParameterIDs.dat` file are those that are used to match up parameter IDs to Modflow input. These strings and their associated parameter IDs can be anywhere in the `ParameterIDs.dat` file. The parameters corresponding to the following text strings are contained within the BCF file and are therefore supported by this command:

```
' PRIMARY STORAGE COEF '  
' TRANSMIS. ALONG ROWS '  
' HYD. COND. ALONG ROWS '  
' VERT HYD COND /THICKNESS '  
' BOTTOM '  
' TOP '  
' SECONDARY STORAGE COEF '  
' COLUMN TO ROW ANISOTROPY '  
' WETDRY PARAMETER '  
' HYD COND GRADIENT FACTOR '  
' THICKNESS OF UPPER ZONE '  
' TRANSMIS. ALONG COLUMNS '  
' HYD. COND. ALONG COLUMNS '  
' ELEVATION OF CHANGE IN K '  
' MAXIMUM HYD COND FACTOR '  
' STORAGE GRADIENT FACTOR '  
' MAXIMUM STORAGE FACTOR '  
' ELEVATION OF CHANGE IN S '
```

The following output files are written to the “To\_NGMS” folder:

```
MapStacks files:      LocationID_ParameterID_MapStacks.xml  
USGS BIL files:      LocationID_ParameterID_L001.bil  
BIL header files:    LocationID_ParameterID_L001.hdr
```

One MapStacks XML file will be produced for each of the Parameter IDs, with corresponding BIL and HDR files for each parameter and each model layer. The numbers in the BIL files are 4-byte floating point values arranged in a series of blocks corresponding to the times specified in the XML files. The non-standard use of 4-byte floating point values is denoted in the HDR files by setting `nBits` equal to zero.

## Extracting MapStacks of (input) recharge rates

```
Command: RECHARGE_OUT  
Routine(s): RCH_OUT  
Source file: RCH_support.for
```

This command extracts a MapStacks XML file of recharge rates from the recharge (RCH) input file for Modflow.

The command required in the IN file is RECHARGE\_OUT.

The following output files are written to the "To\_NGMS" folder:

MapStacks files:       LocationID\_ParameterID\_MapStacks.xml  
USGS BIL files:        LocationID\_ParameterID\_L001.bil  
BIL header files:      LocationID\_ParameterID\_L001.hdr

One MapStacks XML file will be produced, with corresponding BIL and HDR files for each model layer. The Parameter ID will be the one listed in the ParameterIDs.dat file that corresponds to the text '        RECHARGE' (24 characters long).

The numbers in the BIL files are 4-byte floating point values arranged in a series of blocks corresponding to the times specified in the XML files. The non-standard use of 4-byte floating point values is denoted in the HDR files by setting nBits equal to zero.

The volumetric recharge rates that actually contribute to the model water balance can be extracted using the BINARY\_MAPSTACKS command (the overall rates may be different as recharge that is specified over inactive cells does not contribute to the model water balance).

## **Extracting TimeSeries of (input) groundwater abstraction rates**

Command:   **WELLS\_OUT** parameterID [NO\_NAME\_CHECK] [LOCATIONS]  
            [\*MERGE\*]  
Routine(s): WELLS\_OUT, WELLS\_OUT\_AL, WEL\_5RP, INT\_STRING  
Source file: WEL\_support.for

This command extracts a TimeSeries XML file of groundwater abstraction rates as specified in the Modflow WEL input file.

The command required in the IN file is WELLS\_OUT followed by the NGMS parameter ID (e.g. QSRC\_GW.sim.his). Two optional arguments are also allowed, the first, NO\_NAME\_CHECK, is to disable checking of names. If the argument is not included, the routine will check the WEL file for names (column 41 onwards of each line), which can be used to distinguish between abstractions that are specified in the same model cell (Groundwater Vistas includes these names in the WEL file if the wells are specified as 'analytic elements'). The second optional argument is LOCATIONS. If this is included a separate file of abstraction locations is created.

The locationId for each time series is written in the form:  
"ModelName\_R87\_C108\_L4\_(2)\_Name\_Public\_water\_supply"

This defines the row, column and layer of the model cell containing the abstraction, along with the name (if available) for each abstraction series. The number in brackets is specified if there is more than one abstraction in a model cell.

The time series are all written to a single file in the "To\_NGMS" folder:  
LocationID\_ParameterID.xml

This file contains individual time series for all the groundwater abstractions in the model.

If the LOCATION argument is included then an additional file is also created in the "To\_NGMS" folder:

LocationID\_ GW\_Abstraction\_Locations.xml

This file contains the locationIds described above with the associated X and Y coordinates of the model cell in which they are active (the Z coordinate is the model layer number).

If the \*MERGE\* argument is included then a time series (and locations) XML file is created with the same name structure as above, but with the text "\_MultiLayer" added just before the file extension identifier (.xml). This file contains the total abstraction rate for all the groups of wells that are located at the same row and column in the model. This is mainly to enable multi-layer wells to be grouped together, but it also means that wells within the same cell in the same layer are also grouped together. Wells are not included if they are the only well at a particular model row and column (omit the \*MERGE\* argument to produce a list of these). The locationIds for these wells are similar to the form described above, but include a list of the layers in which the wells are active. For example:

"ModelName\_R87\_C108\_L1+2+1(2)+4+4(2)\_1stName\_PWS1"

This locationId refers to the combined abstraction from five wells all located at row 87, column 108. Two of the wells are located in layer 1, one in layer 2 and two more in layer 4. The text that appears after "\_1stName\_" is the name assigned to the first well (in layer 1 in this case). This name is only included if it appears in the original Modflow WEL file.

## **Extracting TimeSeries of (input) surface water discharge/abstraction rates**

Command: **STR\_DISCH\_OUT** parameterID [MAPSTACKS] [LOCATIONS  
[Stress period no.]] [GRID [Stress period no.]]

Routine(s): STR\_DISCH\_2XML, STR\_DISCH\_AL, STR\_1RP

Source file: STR\_support.for

This command extracts a TimeSeries (or MapStacks) XML file of surface water discharge/abstraction rates as specified in the Modflow STR input file.

The command required in the IN file is STR\_DISCH\_OUT followed by the NGMS parameter ID (e.g. QSRC\_SW.sim.his). The first optional argument, MAPSTACKS, can be used to specify that the output should be created in the MapStacks XML format rather than the default TimeSeries XML format. The MapStacks XML file contains the discharge information on a single layer. An



additional optional argument is also allowed called LOCATIONS (or GRID), which allows the creation of separate files of stream cell locations and stream discharge locations. The LOCATIONS (or GRID) argument can be followed by a number which specifies the stress period to use for the output of stream cell locations. If no number is entered, the first stress period is used.

The locationId for each time series is written in the form:  
"ModelName\_discharge\_Seg5\_Rch1\_Name\_Sewageworks"

This defines the segment and reach numbers of the stream cell where the discharge (or abstraction) is located, along with the name (if available) for each time series.

The time series are all written to a single TimeSeries XML file in the "To\_NGMS" folder:  
LocationID\_ParameterID.xml

This file contains individual time series for all the surface water discharges / abstractions in the model.

If the MAPSTACKS argument is included on the command line, the TimeSeries XML file (in the "To\_NGMS" folder) is replaced by a MapStacks XML file and its associated BIL and HDR files:

LocationID\_ParameterID\_MapStacks.xml  
LocationID\_ParameterID.bil  
LocationID\_ParameterID.hdr

If the LOCATION argument is included then two more files are also created in the "To\_NGMS" folder:

LocationID\_Stream\_Cell\_Locations.xml  
LocationID\_Stream\_Discharge\_Locations.xml

These files contain the locationIds described above with the associated X and Y coordinates of the model cell in which they are active. For the discharge locations the Z coordinate is the model layer number, whereas for the stream cell locations the Z coordinate is the stream stage (unless the stream stage is calculated, in which case the elevation of the top of the stream bed is used instead). For the stream cell locations the "\_discharge\_" keyword in the locationID is replaced by "\_streamcell\_".

If the GRID argument is included (instead of the LOCATION argument) then the following two files are created in the "To\_NGMS" folder:

LocationID\_Stream\_Cell\_Locations\_Grid.xml  
LocationID\_Stream\_Discharge\_Locations\_Grid.xml

These files contain the X and Y coordinates of the model cell in which the stream cells are active, but do *not* contain the locationIds described above. For the discharge locations the Z coordinate is the model layer number, whereas for the stream cell locations the Z coordinate is the stream stage (unless the stream stage is calculated, in which case the elevation of the top

of the stream bed is used instead). For the discharge locations the locationId and description are “ModelName\_StreamDischarges”, whereas for the stream cell locations, they are “ModelName\_StreamCells”.

## Extracting a map of stream flow directions

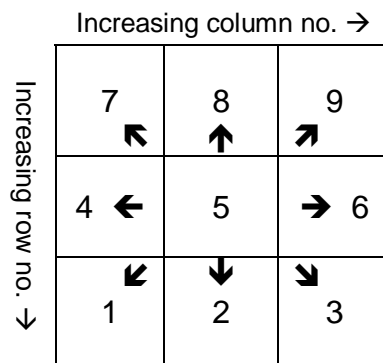
Command: **STREAM\_DIRECTIONS** [output filename] [stress period no.]  
 Routine(s): STR\_directions, STR\_directions\_AL, Write\_ASC\_int\_header, Write\_ASC\_int\_data  
 Source file: STR\_support.for, MFma\_Write\_ASC.for

This command extracts an ASC file of stream flow directions as determined from the stream network defined in the Modflow STR input file.

The command required in the IN file is STREAM\_DIRECTIONS, optionally followed by the name of the output filename. If the output filename is not specified, the following output file is written to the “To\_NGMS” folder:  
 LocationID\_Streamflow\_Directions.asc

If a filename has been specified, then a second argument can be included. This second argument specifies which stress period to use from the STR file. If it is omitted, the first stress period is used (in most models the actual stream network will not vary over time – only the surface water discharges / abstractions are likely to change).

The following convention is used to define the stream flow directions:



Where 5 denotes an “internal sink”, which is usually the most downstream cell of a stream network that is represented in the model.

For stream cells which do not connect to a stream cell in an adjacent model cell, the code -1 is used and a warning is written to the diagnostics file which gives details of the non-adjacent connection between the cells.

If the last stream cell in a segment is on a model boundary (the first or last row or column in the model grid) then the stream flow direction is assumed to be out of the model, perpendicular to the boundary.

## Extracting maps of boundary condition cell locations

Commands: **DRN\_OUT** ParameterID [GRID] [stress period no.]  
**RIV\_OUT** ParameterID [GRID] [stress period no.]  
**GHB\_OUT** ParameterID [GRID] [stress period no.]  
Routine(s): GEN\_GRID\_OUT  
Source file: Generic\_support.for

This command extracts a Grid XML file of boundary condition cell locations along with the elevation (stage) of the boundary condition specified at that location. It can be used for drains (DRN), river (RIV) or general head boundaries (GHB), depending on the command used.

The following file is written to the 'To\_NGMS' folder:

Grid xml file: LocationID\_ParameterID\_Cell\_Locations\_Grid.xml

If the stress period number is omitted, the first stress period from the model simulation is used. The GRID keyword is not required but may be included before the stress period number (if entered).

The ParameterID is only used to determine the xml filename, and does not need to appear in the ParameterIDs.dat file.

## Extracting maps of MODBRANCH branch cell locations

Commands: **BRANCH\_GRID** ParameterID  
Routine(s): BRC\_GRID\_OUT  
Source file: BRC\_support.for

This command extracts a Grid XML file of MODBRANCH branch cell locations along with the elevation (stage) of the branch specified at that location. It can only be used for MODBRANCH models.

The following file is written to the 'To\_NGMS' folder:

Grid xml file: LocationID\_ParameterID\_Cell\_Locations\_Grid.xml

The ParameterID is used to determine the xml filename and the location ID associated with the grid. It does not need to appear in the ParameterIDs.dat file. The location ID associated with the grid is set as:

LocationID\_ParameterID\_Cells

## Extracting diagnostic information from a model simulation

Command: **DIAGNOSTIC** 'Diagnostic\_Search\_Strings.dat' [output folder]  
Routine(s): LIST\_Diag  
Source file: LIST\_support.for (MFma\_Write\_Diag\_XML.for)

This command extracts diagnostics information from the main Modflow output file (sometimes called the list file) and writes it to a Diagnostics XML file.

The command required in the IN file is DIAGNOSTIC followed by the filename (including the path if necessary) of a file which contains the character strings which will be recognised by the routines. The format of this file is as follows:

IERR(1)	NLINES(1)	POS(1)	STRING(1)	[FORMAT(1)]
IERR(2)	NLINES(2)	POS(2)	STRING (2)	[FORMAT(2)]
IERR(3)	NLINES(3)	POS(3)	STRING (3)	[FORMAT(3)]
...	...	...	...	...
IERR(n)	NLINES(n)	POS(n)	STRING (n)	[FORMAT(n)]

IERR(x) is the warning level of the information which will be written to the DIAGNOSTICS file. These warning levels can be one of the following:

- 1 = Critical (critical problems, e.g.: "Illegal option code. Simulation aborting")
- 2 = Warn (warning info, e.g.: "Convergence stopped after 999 consecutive outer iterations with error criteria achieved.")
- 3 = Info (information all is well, e.g.: "Stress period No. 1, length = 31")
- 4 = Debug (debugging info, e.g.: "21 total iterations")

NLINES(x) is the number of lines that will be echoed from the Modflow output file to the Diagnostics XML file (including the line which contains the text that matches STRING).

POS(x) is the character location (from left to right along a line) where the first letter of STRING(x) will be searched for in the Modflow output file.

STRING(x) is the character string that will be searched for in the file. If STRING(x) is specified as '<ParameterIDs>', the program will search for all the text strings that are contained in the ParameterIDs.dat file at the specified position. However, using the <ParameterIDs> option can make the execution of this command quite slow.

FORMAT(x) is an optional input that is only required for the special cases described below.

In addition to the generic cases which are handled in the simple way described above, there are a number of special cases which are indicated by a negative value for IERR(x) and which result in some additional processing and output to the Diagnostics XML file. These are:

- 1 = Stress period counter.
- 2 = Time step counter.

- 3 = Search for the maximum head change at the end of each time step and write the values to a TimeSeries XML file in the "To\_NGMS" directory with the filename: LocationID\_ParameterID\_TimeSeries.xml, where the ParameterID is the one that corresponds to text string "MAXIMUM HEAD CHANGE". FORMAT(x) defines the Fortran format that is required to read in the head change values. NLINES(x) is the number of lines down from the line in the Modflow output file that contains STRING(x), where the first line of maximum head change values should be read from.
- 4 = Search for the maximum flow residual at the end of each time step and write the values to a TimeSeries XML file in the "To\_NGMS" directory with the filename: LocationID\_ParameterID\_TimeSeries.xml, where the ParameterID is the one that corresponds to text string "MAXIMUM RESIDUAL ". FORMAT(x) defines the Fortran format that is required to read in the head change values. NLINES(x) is the number of lines down from the line in the Modflow output file that contains STRING(x), where the first line of maximum head change values should be read from.
- 5 = Skip boundary condition information (echo of input). This significantly speeds up the processing of this command by skipping the echo of different boundary condition input information. FORMAT(x) defines the Fortran format that is required to read in the number of boundary condition cells for which information can be skipped. NLINES(x) is the number of additional lines that can be skipped.
- 6 = Check for the number of cell conversions that occurred during this time step (conversions from wet to dry and vice versa). FORMAT(x) is the format required to read in the number of cell conversions. NLINES(x) is the number of lines down from the line in the Modflow output file that contains STRING(x), where the number of cell conversions is to be read from (if this is zero then the number of cell conversions is read from the same line as STRING(x)).
- 7 = Read in the locations of wet and dry cells so that they can be checked against the locations of groundwater abstractions to ensure that none of the abstractions have become inactive. FORMAT(x) is the format required to read in the locations of the converted cells. NLINES(x) is the number of lines down from the line in the Modflow output file that contains STRING(x), where the first cell location is to be read from (if this is zero then the number of cell conversions is read from the same line as STRING(x)).
- 8 = Read in the locations of groundwater abstractions so that they can be checked against the locations of dry cells to ensure that none of the abstractions have become inactive. FORMAT(x) is the format required to read in the locations and abstraction rates of the well cells. NLINES(x) is the number of lines down from the line in the Modflow output file that contains STRING(x), where the first line of abstraction information is specified.
- 9 = End marker. If this string is found then the Modflow executable has stopped running and the simulation has finished.

You can increase the speed of execution of the DIAGNOSTIC command by removing any lines with <ParameterIDs> specified as the STRING(x) (or by inserting a blank line before these lines).

You can increase the speed further by removing all the lines where the first number on the line is 4 (these only return non-essential info).

However, it is strongly recommended that you do not remove any of the lines that start with a negative number, especially those starting with -1, -2 or -9.

Following the name of the file described above, a second optional argument can be specified which gives the location where the Diagnostics XML file will be saved. If omitted the file is saved to the "To\_NGMS" directory. The following output files are written to the specified folder:

Diagnostics XML file:	LocationID_Diagnostic.xml	
TimeSeries XML file:	LocationID_ParameterID.xml	[heads]
TimeSeries XML file:	LocationID_ParameterID.xml	[residuals]

## Extracting MapStacks of model output (heads, flows, etc)

Command: **BINARY\_MAPSTACKS** [\*MERGE\*] [list of parameter IDs] [unit Nos.]

Routine(s): BIN\_MapStacks, BIN\_ReadRecord, BIN\_CheckPrecision

Source file: BIN\_support.for (MFma\_Write\_MapStacks\_XML.for)

This command extracts MapStacks XML files of heads, drawdowns and cell-by-cell flows from the binary output files produced by Modflow during a simulation.

The command required in the IN file is BINARY\_MAPSTACKS, which can optionally be followed by a list of parameter IDs for which MapStacks XML files are required. The parameter IDs and their relationship to the text strings used by Modflow are defined in the ParameterIDs.dat file. If no parameter IDs are specified on the command line, all the output from the binary output files will be extracted to MapStacks XML files.

If a parameter is included that is not a standard Modflow output (i.e. from a new package or extension for Modflow, such as MODBRANCH), the parameter ID on the command line should be followed by the unit number of the file containing the cell-by-cell flow terms for this parameter (if the output is in a heads format file rather than a cell-by-cell flow format file, the unit number should be negative). If the cell-by-cell flow terms are written to the same file as one of the other parameters on the command line, the unit number can be omitted. If no parameters are listed on the command line, new parameters will only be processed if they are contained in the same file as one of the standard Modflow outputs.

(The shorter text strings (16 characters long) in the ParameterIDs.dat file are those that are used to match up parameter IDs to Modflow binary output. All of these strings and their associated parameter IDs should be in the upper part of the ParameterIDs.dat file. If any of them are below line 40 of this file, they will not be included when the BINARY\_MAPSTACKS command is used without a list of parameter IDs.)

The following output files are written to the "To\_NGMS" folder:

MapStacks files:	LocationID_ParameterID_MapStacks.xml
USGS BIL files:	LocationID_ParameterID_L001.bil
BIL header files:	LocationID_ParameterID_L001.hdr

One MapStacks XML file will be produced for each of the Parameter IDs, with corresponding BIL and HDR files for each parameter and each model layer. The numbers in the BIL files are 4-byte floating point values arranged in a series of blocks corresponding to the times specified in the XML files. The non-standard use of 4-byte floating point values is denoted in the HDR files by setting nBits equal to zero.

If the \*MERGE\* keyword is included after the BINARY\_MAPSTACKS command in the IN file, the results from a multi-layer model will be combined into a single layer (for all the parameters specified on the command line). For heads and drawdowns the value from the uppermost active layer is used (corresponding to the water table). For flow output the flows are summed over all the layers. For accumulated stream flow, the maximum flow rate is used (in case two stream cells are specified at the same row and column location).

This command can be used to extract Branch leakage rates from a MODBRANCH simulation if the unit number of the "Out binary cbc flow file" specified in the MODBRANCH master file is set to the same as the main .CBB flow file specified in the MODFLOW .NAM file (often this is set as unit number 50). The specified filename must also be the same as in the .NAM file, or set to "\*" (without the quotes).

## Extracting compressed MapStacks of MODBRANCH output

Command: **BRANCH\_OUT** parameter ID [location ID]  
Routine(s): BRANCH\_OUT  
Source file: BRC\_support.for (MFma\_Write\_MapStacks\_XML.for)

This command extracts compressed MapStacks XML files of branch leakage, branch flows or branch stages (depending on the parameter ID) from the text output files produced by MODBRANCH during a simulation.

The command required in the IN file is BRANCH\_OUT, which must be followed by the parameter ID (determining which compressed MapStacks XML file is required) and optionally by the location ID (which should be the same as the location ID written to the grid XML file produced by the

BRANCH\_GRID command). If the location ID is omitted, then the default location ID is used as defined in the IN file. The parameter IDs and their relationship to the text strings used by Modflow are defined in the ParameterIDs.dat file.

The following output files are written to the “To\_NGMS” folder:

MapStacks file: LocationID\_ParameterID\_Compressed\_MapStacks.xml  
USGS BIL file: LocationID\_ParameterID\_Compressed\_MapStacks.bil  
BIL header file: LocationID\_ParameterID\_Compressed\_MapStacks.hdr

One MapStacks XML file will be produced, with corresponding BIL and HDR files. The numbers in the BIL files are 4-byte floating point values arranged in a series of blocks corresponding to the times specified in the XML files. The non-standard use of 4-byte floating point values is denoted in the HDR files by setting nBits equal to zero. The blocks of values for each time step are arranged in the same order as the list of cell locations produced by the BRANCH\_GRID command.

## **Extracting time series of the ground water flow balance**

Command: **WATER\_BALANCE**

Routine(s): LIST\_TVFlux

Source file: LIST\_support.for (MFma\_Write\_TimeSeries\_XML.for)

This command extracts a TimeSeries XML file of flows to and from the groundwater flow system. The command required in the IN file is WATER\_BALANCE.

When sent this command the Module Adaptor scans through the main Modflow output (OUT or LIST) file and extracts the water balance information for each time step for which output is provided.

The time series are all written to a single file in the “To\_NGMS” folder:

LocationID\_WaterBalance.xml

This file contains individual time series for inflows from and outflows to the various boundary conditions specified in the model. These boundary conditions have NGMS parameter IDs associated with them, as specified in the ParameterIDs.dat file (in the ParameterIDs.dat file, the shorter text strings (16 characters long) are those that are used to match up parameter IDs to the boundary condition descriptions).

The parameterId written to the header of each series in the TimeSeries XML file is based on the relevant parameter ID in the ParameterIDs.dat file, but with either an “\_in” or an “\_out” added to the end of the ID, depending on whether the flow is into or out of the groundwater flow system. This sign convention is relatively straight forward, although the flows to and from aquifer storage (currently denoted as QSTOR) require some clarification. The water balance is in relation to groundwater flow system, which does not include



water which is 'statically' stored in the aquifer. Therefore storage releases (due to lowering groundwater heads) result in water leaving aquifer storage and becoming available to the groundwater flow system. These flows are therefore described as *inflows* despite the fact that the water is coming *out* of storage. Conversely, water that is taken into storage (due to rising groundwater levels) is denoted as an outflow from the groundwater flow system.

## Updating recharge rates

Command: **RECHARGE\_IN** MapStacks.xml [format]

Routine(s): RCH\_IN

Source file: RCH\_preproc.f90 (and timeseries.f90)

This command over-writes the Modflow RCH file (which contains recharge rates) using information supplied in a MapStacks XML file.

The command required in the IN file is RECHARGE\_IN followed by the name of the MapStacks XML file, which is assumed to be in the "From\_NGMS" directory along with its associated BIL and HDR files.

This can then optionally be followed by a string specifying a Fortran format code to use when writing the RCH file. If the format code is omitted, then a default format is used '(10000e15.7)' which means that up to 10,000 columns of cell-by-cell recharge rates will be written on each line in exponential format, each taking up 15 character spaces, and with 7 figures after the decimal point.

## Updating Evaporation rates

Command: **EVT\_IN** MapStacks.xml [format]

Routine(s): EVT\_IN

Source file: EVT\_preproc.f90 (and timeseries.f90)

This command over-writes the Modflow EVT file (which contains evaporation rates) using information supplied in a MapStacks XML file.

The command required in the IN file is EVT\_IN followed by the name of the MapStacks XML file, which is assumed to be in the "From\_NGMS" directory along with its associated BIL and HDR files.

This can then optionally be followed by a string specifying a Fortran format code to use when writing the EVT file. If the format code is omitted, then a default format is used '(10000e15.7)' which means that up to 10,000 columns of cell-by-cell EVT rates will be written on each line in exponential format, each taking up 15 character spaces, and with 7 figures after the decimal point. The same format will also be used to write the arrays for the EVT surface and extinction depth (these two arrays from the first stress period of the original EVT file are used for all the stress periods of the new EVT file).

## Updating groundwater abstraction rates

Command: **WELLS\_IN** 'TimeSeries.xml'  
Routine(s): WEL\_IN, WEL\_DeCode\_LocationID  
Source file: WEL\_preproc.f90 (timeseries.f90, read\_xml\_prims.f90, xmlparse.f90, etc.)

This command over-writes the Modflow WEL file (which contains groundwater abstractions) using information supplied in a TimeSeries XML file.

The command required in the IN file is WELLS\_IN followed by the name of the TimeSeries XML file, which is assumed to be in the "From\_NGMS" directory.

The locationId for each time series needs to be in the form:  
"ModelName\_R87\_C108\_L4\_AnyOtherInfo"

Where "\_R", "\_C" and "\_L" should be in upper case. These three text strings should be followed immediately by the relevant number, which should in turn be followed by a "\_", a space or the end of the locationId text string. The three bits of information can be in any order. The Module Adaptor reads the locationId from left to right until all three values are found. If all three are not found an error is written to the Module Adaptor diagnostics file and the code moves on to the next time series in the XML file.

ModelName and AnyOtherInfo can be any text as long as the ModelName does not contain any of the three text strings mentioned above. If AnyOtherInfo contains the string "\_Name\_", any characters after this string are written to the WEL file at the end of every line that contains an abstraction rate for the specified location. These characters can be useful for differentiating between two different abstractions which may be active in the same cell (otherwise this can be denoted by a number in brackets, as is done by the WELLS\_OUT command, although this number will not be written to the WEL file – see section on Extracting TimeSeries of (input) groundwater abstraction rates).

It is assumed that the dates within each time series are the same as the dates within all the other time series in the TimeSeries XML file.

The WELLS\_IN command will overwrite any previous abstraction information (e.g. from the WELLS\_ADD command – always use WELLS\_IN before WELLS\_ADD if both are to be used in the same instruction list).

## Adding new groundwater abstractions

Command: **WELLS\_ADD** 'ModuleParameters.xml' 'TimeSeries.xml'  
Routine(s): WEL\_ADD  
Source file: WEL\_preproc.f90 (PARAMS\_NEW.F90, timeseries.f90,  
read\_xml\_prims.f90, xmlparse.f90, etc.)

This command adds abstractions to the Modflow WEL file using location information supplied in a Module Parameter XML file, and abstraction rates supplied in a TimeSeries XML file.

The command required in the IN file is WELLS\_ADD followed by the names of the Module Parameters XML file and the TimeSeries XML file, which are assumed to be in the "From\_NGMS" directory.

The Module Parameters XML file needs to include the following parameters for each new abstraction:

<name>	WellName_x_cellcentre	<doubleData>	value
<name>	WellName_y_cellcentre	<doubleData>	value
<name>	WellName_layer	<intData>	value

Where WellName can be any text string (up to 57 characters long), and must be identical for each of the three parameters for any given abstraction location. The parameter identifiers "\_x\_centre" "\_y\_centre" and "\_layer" should all be in lower case. The three bits of information can be in any order. The Module Adaptor reads the Module Parameter XML file until all three values are found for each WellName. If all three are not found a warning is written to the Module Adaptor diagnostics file and the code continues with any points for which all the required information is present.

The locationId for each time series in the TimeSeries XML file should correspond to the text string used for WellName above. No RCL (row column layer) information is required.

The WellName is written to the WEL file at the end of every line that contains an abstraction rate for the specified location.

It is assumed that the dates of each time series in the TimeSeries XML file are the same as the dates being used in the current simulation. Therefore it is important that there are as many times specified in the time series as there are stress periods in the current simulation (see "Removing initial stress periods from input files

Command: **SKIP** FILETYPE DATE/State.xml  
Routine(s): GEN\_SKIP  
Source file: Generic\_preproc.f90

This command can be used to update input files so that a simulation can be started midway through an existing simulation. It over-writes existing Modflow input file type specified on the command line by removing information from the

start of the files relating to early stress periods. It is not required for any boundary condition type which remains constant throughout the simulation.

The FILETYPE argument must be included and can be one of the following:

WEL	to update groundwater abstractions
DRN	to update drains
RIV	to update rivers
GHB	to update general head boundaries
CHD	to update time variant constant heads

The FILETYPE should then be followed either by a date in the format YYYY-MM-DD, or by the filename of a State XML file which contains the relevant date at which the new simulation is required to start.

The routine searches to find a model stress period that starts at the specified date. If the date is found the file is updated, otherwise an error is returned.

This command should be used BEFORE the BAS\_IN command.

This command cannot be used to modify MODBRANCH input files.

Updating the stress period lengths” for information on how to change the number of stress periods in the simulation).

Any information written using the WELLS\_ADD command will be overwritten if it is followed by the WELLS\_IN command. Always use WELLS\_IN before WELLS\_ADD if both are to be used in the same instruction list.

## Updating the surface water discharges / abstractions

Command: **STR\_IN** [MAPSTACKS] 'TimeSeries/Mapstacks.xml'  
Routine(s): STR\_IN, STR\_DeCode\_LocationID  
Source file: STR\_preproc.f90 (timeseries.f90, read\_xml\_prims.f90, xmlparse.f90, etc.)

This command over-writes the Modflow STR file (which contains surface water discharges and/or abstractions) using information supplied in a TimeSeries XML file.

The command required in the IN file is STR\_IN followed by the name of the TimeSeries XML file, which is assumed to be in the “From\_NGMS” directory.

The locationId for each time series needs to be in the form:  
"ModelName\_discharge\_Seg5\_Rch1\_AnyOtherInfo"

Where "\_Seg" and "\_Rch" should be in title case. These two text strings should be followed immediately by the relevant number, which should in turn be followed by a "\_", a space or the end of the locationId text string. The two bits of information can be in any order. The Module Adaptor reads the locationId from left to right until both values are found. If both are not found

an error is written to the Module Adaptor diagnostics file and the code moves on to the next time series in the XML file.

ModelName and AnyOtherInfo can be any text as long as the ModelName does not contain any of the two text strings mentioned above. If AnyOtherInfo contains the string “\_Name\_”, any characters after this string are written to the STR file at the end of every line that contains a discharge/abstraction rate for the specified location. These characters can be useful for quickly identifying the discharges in the STR file.

(The “\_discharge” part of the locationId is not actually required, but is included here as it is written this way by the STR\_DISCH\_OUT command – see the section on Extracting TimeSeries of (input) surface water discharge/abstraction rates.)

It is assumed that the dates within each time series are the same as the dates within all the other time series in the TimeSeries XML file.

If the MAPSTACKS keyword is included then the filename of a MapStacks XML file should be entered instead of a TimeSeries XML file. In this case the BIL file referred to in the MapStacks XML file should contain records with one column and the same number of rows as there are stream cells in the original model. It is assumed that the number, locations and other properties of the stream cells remain constant through the simulation (i.e. same as in the first model stress period).

## Updating the initial heads

Command: **BAS\_TEMPLATE** [list of up to 10 dates] [LASTSTEP] [State XML filename]

Routine(s): BAS\_Temp

Source file: BAS\_support.for

This command should be executed at the end of a simulation to create one or more template BAS files containing the groundwater heads at different dates. These templates can then be used to create BAS files containing initial heads for future simulations (see section on Removing initial stress periods from input files

Command: **SKIP** FILETYPE DATE/State.xml

Routine(s): GEN\_SKIP

Source file: Generic\_preproc.f90

This command can be used to update input files so that a simulation can be started midway through an existing simulation. It over-writes existing Modflow input file type specified on the command line by removing information from the start of the files relating to early stress periods. It is not required for any boundary condition type which remains constant throughout the simulation.

The FILETYPE argument must be included and can be one of the following:

WEL to update groundwater abstractions

DRN	to update drains
RIV	to update rivers
GHB	to update general head boundaries
CHD	to update time variant constant heads

The FILETYPE should then be followed either by a date in the format YYYY-MM-DD, or by the filename of a State XML file which contains the relevant date at which the new simulation is required to start.

The routine searches to find a model stress period that starts at the specified date. If the date is found the file is updated, otherwise an error is returned.

This command should be used BEFORE the BAS\_IN command.

This command cannot be used to modify MODBRANCH input files.

Updating the stress period lengths). A State XML file is also created associated with each template BAS file.

The command required in the IN file is BAS\_TEMPLATE followed by an optional list of up to 10 dates. If no date is specified (or if the LASTSTEP argument is included), a template BAS file (and associated State XML file) is created for the heads output corresponding to the end date of the simulation. If dates are listed on the command line, the Module Adaptor reads through the binary heads output file and creates a template BAS file for each of the specified dates. The dates contained in the model output must be within one day of those specified.

The template BAS files are saved in the 'To\_NGMS' directory and have filenames which are the same as the original BAS filename, but with the date added at the end of the filename, before the file extension. A State XML file is created for each template BAS file with a similar filename structure. The following files are written to the 'To\_NGMS' directory:

State XML:	LocationID_State_YYYY-MM-DD.xml
Template BAS file:	Original_BAS_rootname_YYYY-MM-DD.bas

If the LASTSTEP argument is specified along with a State XML filename, then this filename (and path) is used instead of the default, and the original BAS filename is used for the template BAS file, which is placed in the same directory specified in the path of the State XML file.

If any of the groundwater head values are equal to the value specified for 'dry' cells (set in the BCF file), the head value is replaced with the value -9999. This ensures that the same model cells will start off dry in the new simulation (assuming that the base elevation of the model is above -9999).

## Removing initial stress periods from input files

Command: **SKIP** FILETYPE DATE/State.xml  
Routine(s): GEN\_SKIP  
Source file: Generic\_preproc.f90

This command can be used to update input files so that a simulation can be started midway through an existing simulation. It over-writes existing Modflow input file type specified on the command line by removing information from the start of the files relating to early stress periods. It is not required for any boundary condition type which remains constant throughout the simulation.

The FILETYPE argument must be included and can be one of the following:

WEL	to update groundwater abstractions
DRN	to update drains
RIV	to update rivers
GHB	to update general head boundaries
CHD	to update time variant constant heads

The FILETYPE should then be followed either by a date in the format YYYY-MM-DD, or by the filename of a State XML file which contains the relevant date at which the new simulation is required to start.

The routine searches to find a model stress period that starts at the specified date. If the date is found the file is updated, otherwise an error is returned.

This command should be used BEFORE the BAS\_IN command.

This command cannot be used to modify MODBRANCH input files.

## Updating the stress period lengths

Command: **BAS\_IN** format 'TimeSeries.xml' [or 'MapStacks.xml']  
'State.xml'  
Routine(s): BAS\_IN, BAS\_get\_Times\_from\_XML, BRC\_EXTEND  
Source file: BAS\_preproc.f90 (timeseries.f90, mapstacks.f90, statefiles.f90,  
BRC\_support.for)

This command over-writes the Modflow BAS file, updating the number of stress periods and the stress period lengths based on the dates and times specified in a MapStacks or TimeSeries XML file. If more than one time series occurs in the specified file, the first time series in the file is used.

This command should NOT be used before the SKIP command as it updates the start time and the number of stress periods.

The lengths of the stress periods are defined by the lengths of time between each of the dates specified in the XML file. The length of the first stress period is determined from the length of time between the date specified in the

'State.xml' file given on the command line of the IN file (StartDate) and the first date in the time series in the XML file.

The number of time steps and the time step multiplier for each stress period are copied from those specified for the first stress period of the original simulation. If updating a MODBRANCH model, the number of time steps per day used for the first stress period is used for all stress periods (this is because each Modflow time step length must be an integer multiple of the MODBRANCH time step length).

The command required in the IN file is BAS\_IN followed by the format (either 'timeseries' or 'mapstacks'), the name of the TimeSeries or MapStacks XML file (which is assumed to be in the "From\_NGMS" directory), and the filename of the State XML file (also in the "From\_NGMS" directory). The StartDate in the State XML file should be earlier than the first date in the TimeSeries or MapStacks XML file, and is used to calculate the length of the first stress period.

If the StartDate specified in the State XML file differs from the original model start date, then a template BAS file is required with the initial heads corresponding to the StartDate. These files are created for specific dates using the BAS\_TEMPLATE command following the completion of a model simulation (see section on Updating the initial heads).

If the number of stress periods specified in the XML file is greater than the number specified in the original model, then the other Modflow input files that require input for each stress period are also updated. This includes the input files for the following boundary conditions:

1. wells (WEL)
2. drains (DRN)
3. rivers (RIV)
4. evapotranspiration (EVT)
5. general head boundaries (GHB)
6. recharge (RCH)
7. streams (STR)
8. transient constant heads (CHD)
9. MODBRANCH branch boundary conditions (BRC/BRN)

The information specified for the last stress period in each of the files above is repeated for all of the additional stress periods. The one input file that is not currently supported in this way is the reservoir file (RES). If the simulation uses the reservoir package the user is not permitted to increase the number of stress periods.

The other input file that is updated when the number of stress periods is increased is the output control file (OC). The information added to this file specifies an output of the global water budget for every timestep, heads and cell-by-cell flows for the last time step of every stress period, and no output of drawdown (which can be calculated independently).



Finally, the model offset file (OFS) is updated to reflect any changes to the model start date.