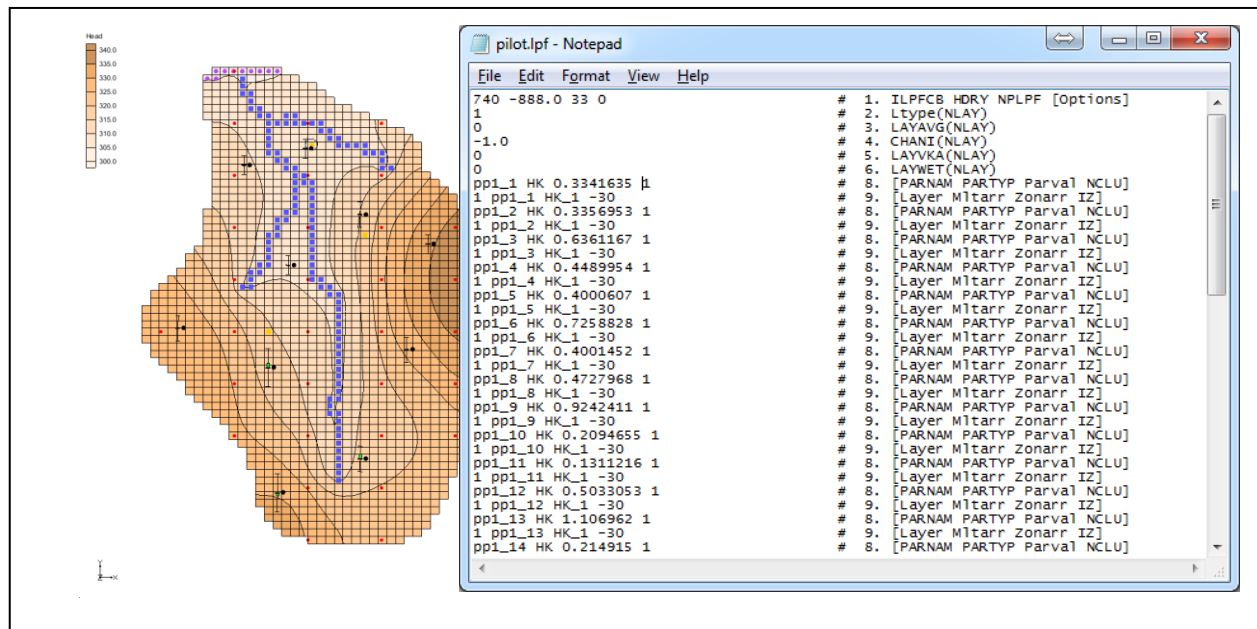


## Save a MODFLOW Simulation in Native Text Format



<b>1</b>	<b>Introduction .....</b>	<b>2</b>
1.1	Outline.....	2
<b>2</b>	<b>Getting Started .....</b>	<b>3</b>
<b>3</b>	<b>Reading in the Simulation .....</b>	<b>3</b>
<b>4</b>	<b>Saving the Project .....</b>	<b>4</b>
<b>5</b>	<b>Viewing MODFLOW Files.....</b>	<b>4</b>
<b>6</b>	<b>Saving Native Text from GMS.....</b>	<b>8</b>
<b>7</b>	<b>Viewing the Native Text.....</b>	<b>8</b>
<b>8</b>	<b>Reading in the Simulation with Parameters .....</b>	<b>10</b>
<b>9</b>	<b>Saving Native Text from GMS.....</b>	<b>12</b>
<b>10</b>	<b>Viewing the Native Text.....</b>	<b>12</b>
<b>11</b>	<b>Reading in the Simulation with Pilot Points .....</b>	<b>14</b>
<b>12</b>	<b>Saving Native Text from GMS.....</b>	<b>15</b>
<b>13</b>	<b>Viewing the Native Text.....</b>	<b>16</b>
<b>14</b>	<b>Saving Native Text from GMS with External Arrays.....</b>	<b>18</b>
<b>15</b>	<b>Conclusion.....</b>	<b>20</b>

# 1 Introduction

---

GMS uses a modified version of MODFLOW that can read data from HDF5<sup>1</sup> files. HDF5 is a binary, cross-platform file format for storing data. Large portions of the MODFLOW inputs are stored in an HDF5 file when creating a MODFLOW model in GMS. Large MODFLOW models are stored much more efficiently using HDF5. HDF5 compression ratios of 95% are common when compared to text files. Also, the read/write times are much faster with the binary HDF5 file when compared to text files.

Even though it has been shown that HDF5 with MODFLOW is extremely efficient, many users would also like to have their MODFLOW simulations in the native MODFLOW format. This is particularly true when completing a modeling project and the user needs to archive the project.

Native MODFLOW text files can be saved from GMS. Beginning with version 10.0, these text files also contain documentation comments that more clearly identify the contents of each line in the MODFLOW input file. Further, parameter definitions are also included in the text files.<sup>2</sup> This tutorial will go over the exporting options available in GMS using various MODFLOW simulations.

## 1.1 Outline

---

Here are the steps of the tutorial:

1. Read an existing MODFLOW simulation, and save it in native text format.

---

<sup>1</sup> See [www.hdfgroup.org](http://www.hdfgroup.org).

<sup>2</sup> Earlier versions of GMS also exported native text MODFLOW files but these files only removed references to the HDF5 file. There were no documenting comments and parameters were not written to the text files.

2. Read in an existing MODFLOW simulation with parameters, and save it in native text format.
3. Read in an existing MODFLOW simulation with pilot points, and save it in native text format.
4. Change the array saving options, and save the simulation in native text format.

## 2 Getting Started

---


Do the following to get started:

1. If necessary, launch GMS.
2. If GMS is already running, select the *File* / **New** command to ensure that the program settings are restored to their default state.

## 3 Reading in the Simulation

---

First, it is necessary read in a MODFLOW simulation. This is one of the example files that ships with the USGS MODFLOW source code.

1. Select the **Open**  button.
2. Locate and open the directory entitled *Tutorials\MODFLOW\text\mnw1*.
3. Select the file named “mnw1.nam,” then click **Open**.

The *MODFLOW Translator* dialog will come up. This dialog appears because the MODFLOW simulation was not created in GMS. When the user selects **OK**, MODFLOW will run and export a new set of files including an HDF5 file. These will be the files that GMS works with. Notice that no edits will be made to the original model files. The MODFLOW translator ensures that if MODFLOW can read the simulation, then GMS will be able to import the files.

4. Click **OK** in the *MODFLOW Translator* dialog to import the model using MODFLOW 2000.
5. Click **Done** when MODFLOW 2000 has finished the file conversion.

The user should see a MODFLOW model that looks similar to the following figure:

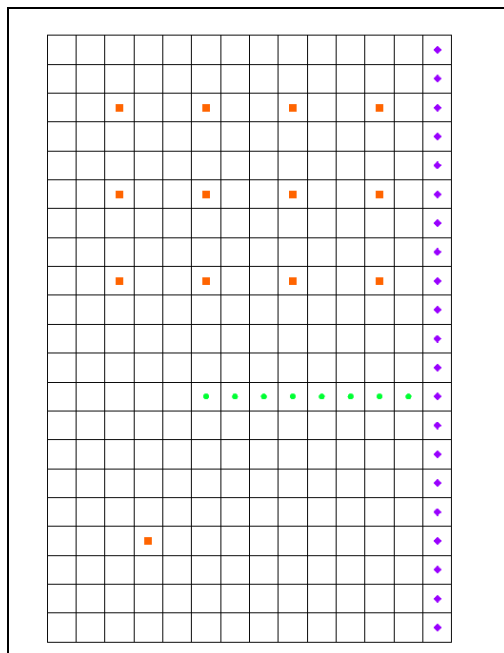


Figure 1 Imported MODFLOW model

## 4 Saving the Project

Now it is possible to save the project as if the user were going to run MODFLOW.

1. Select the *File* / **Save As** command.
2. Locate and open the directory entitled *Tutorials\MODFLOW\text*.
3. Using the **Save** button, save the project with the name “mnw1.gpr.”

When a project is saved in GMS, a GPR file is created. The GPR file contains GMS specific data. If the user has a MODFLOW simulation in the current project, then a folder is created that has all of the MODFLOW inputs. The folder will be named with the current project name and “\_MODFLOW.” In this project, a new folder called “mnw1\_MODFLOW” was created.

## 5 Viewing MODFLOW Files

If the user wants to view the MODFLOW input files, the user can do so quickly.

1. Right-click on the “MODFLOW”  item in the Project Explorer.
2. Select the **Open Containing Folder** command.

This command will open a file browser to the directory where the MODFLOW files are stored. The user should see files similar to those in the next figure.

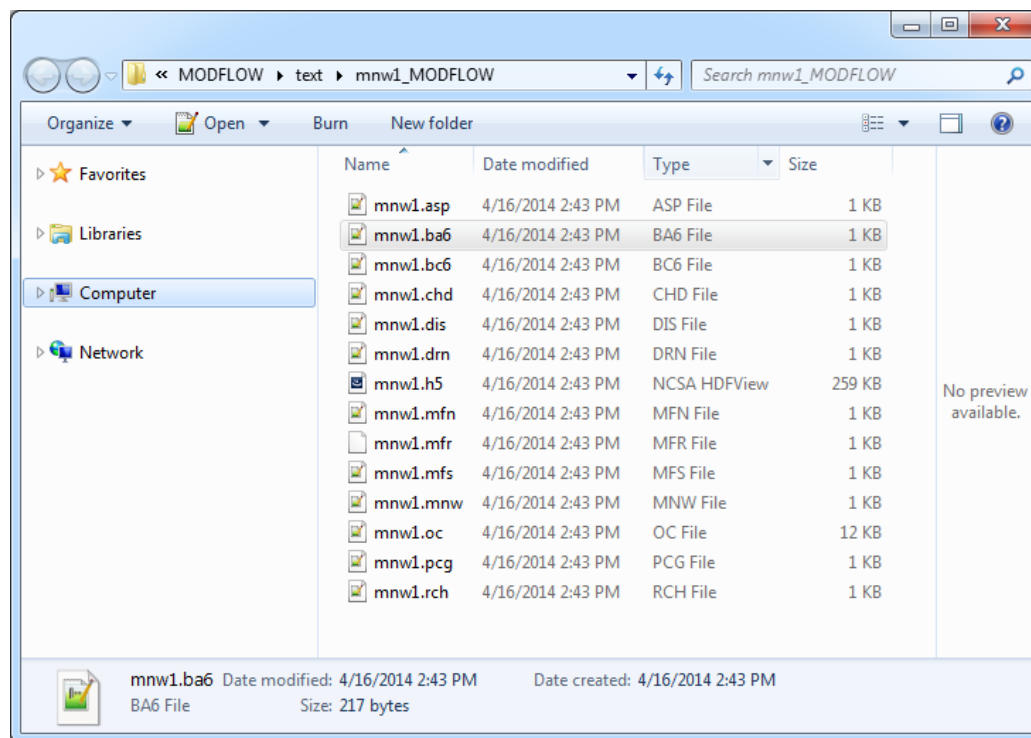


Figure 2 MODFLOW input files saved from GMS

All of the files in this directory are text files except for the “mnw1.h5” file. The next step is to examine the contents of the files.

- Using Notepad (or another text editor), open the “mnw1.bc6” file.

This is the BCF package file. This file contains information on the hydraulic properties of the grid cells. The file will look like the following figure.

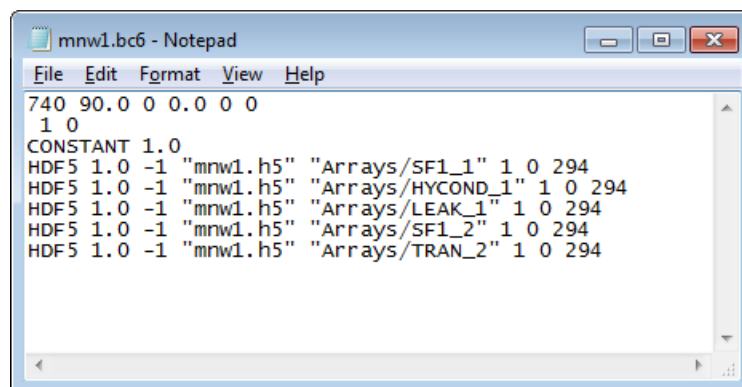


Figure 3 MODFLOW BCF package file saved from GMS

Notice the lines that begin with “HDF5.” When MODFLOW runs and reads these lines, it will go into the “mnw1.h5” file to get the data. The lines in the file that reference the H5 file are arrays that define the aquifer properties (Storage, Hydraulic Conductivity, and Transmissivity). There are many free viewers available to browse the contents of

HDF5 files. HDFView<sup>3</sup> is provided by the developers of HDF5 and allows viewing and editing of HDF5 files. The following image shows “mnw1.h5” in HDFView.

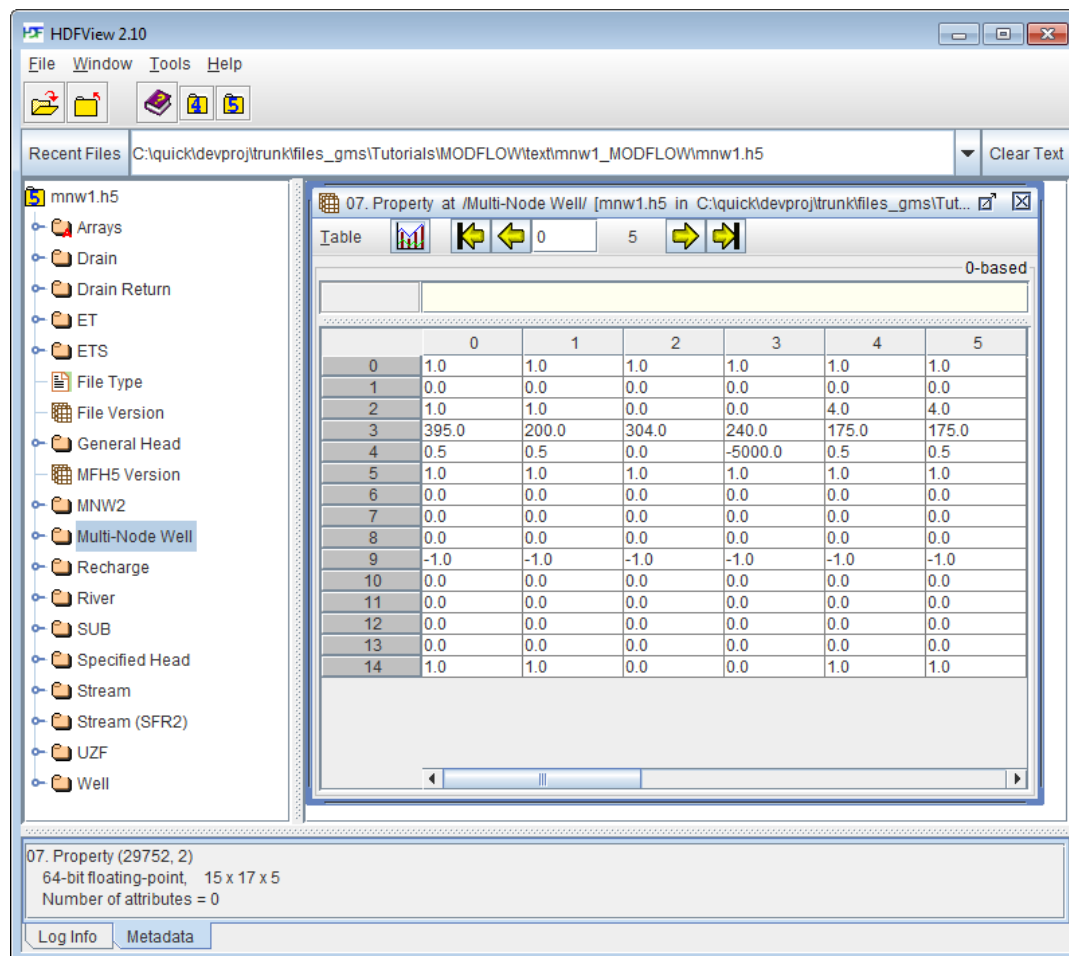


Figure 4 HDFView of “mnw1.h5”

This tutorial will now look at the original MODFLOW files. The user must go back to the file browser and find this same file in the original files.

4. Click on the **Open** button to go back to the file browser, and go up 1 directory.
5. Double-click on the “mnw1” directory.

This will take us to the original MODFLOW files. The user should see files similar to those in the next figure.

<sup>3</sup> See <http://www.hdfgroup.org/products/java/hdf-java-html/hdfview/>.

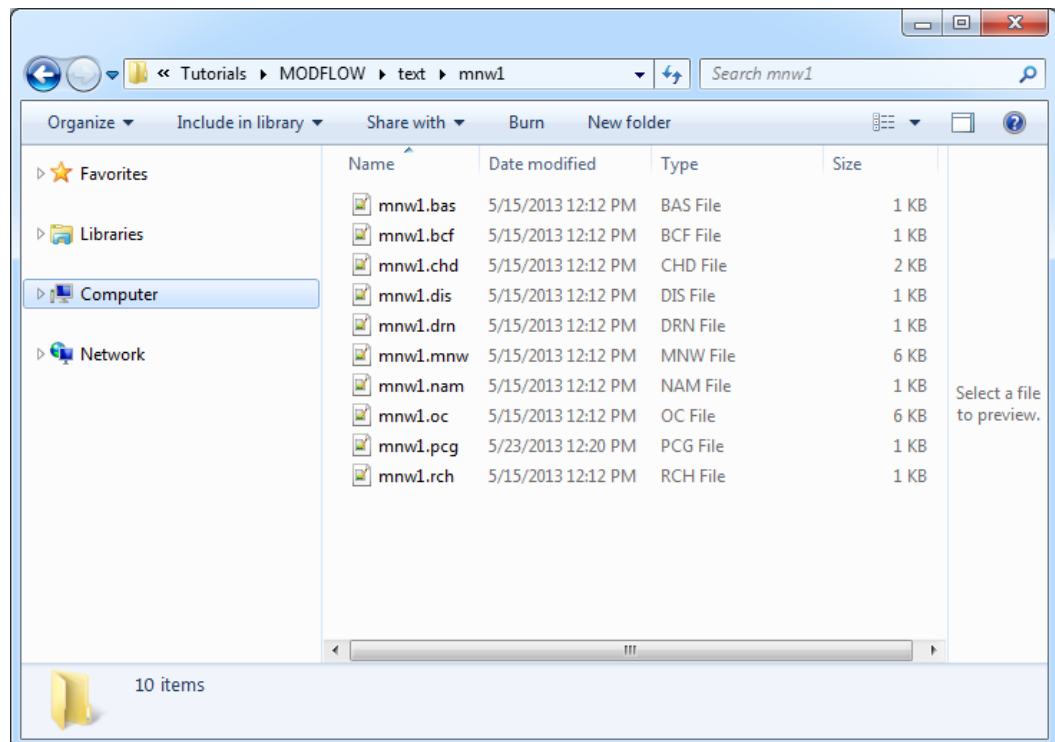


Figure 5 Original MODFLOW input files

6. Using a text editor, open the “mnw1.bcf” file.

The user should see a file like that shown in the following figure.

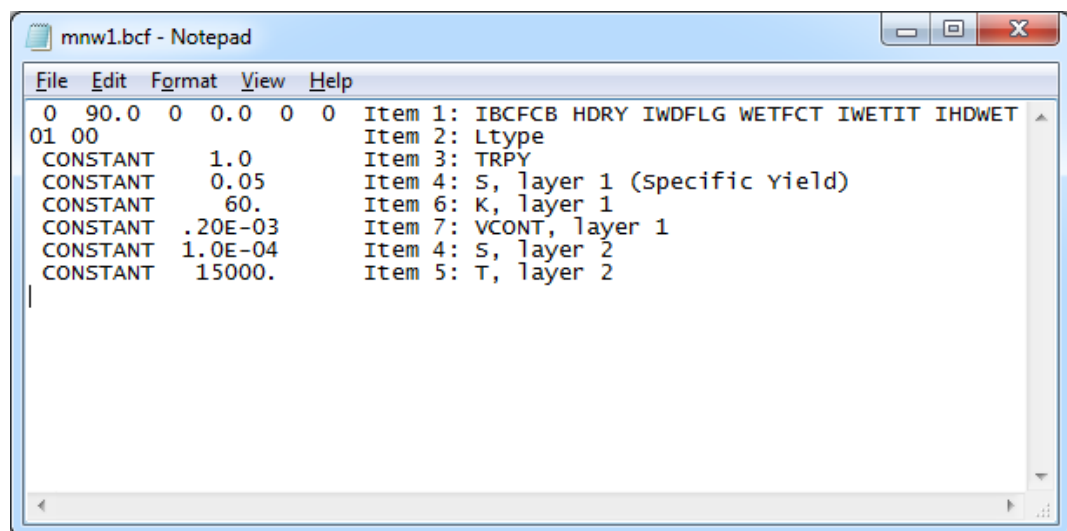


Figure 6. MODFLOW BCF file.

When the two files are compared, the user can see that the numbers on lines 1 and 2 of the file are the same. However, in the original files, the user can see some text on the right side of the file. MODFLOW will ignore this text because it is only expecting to

read numbers on those lines. Also notice that instead of referencing an HDF5 for the aquifer properties, the values for these arrays are listed in this file.

## 6 Saving Native Text from GMS

---

Now it is necessary to save out this simulation in MODFLOW native text format. The next step is to close the text editors and the file browser and return to GMS.


1. Close the text editors and file browser and return to GMS.
2. Select the *MODFLOW* / **Global Options** menu command.
3. In the *MODFLOW Global/Basic Package* dialog, under the *MODFLOW version* section of the dialog (near the top left), turn on the *Save native text copy* option.
4. Click **OK** to exit the dialog.
5. Select the *File* / **Save** command.

By turning on the *Save native text copy* option, the user has saved out an additional set of MODFLOW input files that are in the native text format.

## 7 Viewing the Native Text

---

The next step is to go look at the BCF package file in the native text files that were just saved.

1. Right-click on the “MODFLOW”  item in the Project Explorer.
2. Select the **Open Containing Folder** command.
3. Go up 1 directory in the file browser that opened.
4. Double-click on the “mnw1\_MODFLOW\_text” directory.

The user should now see the following files:



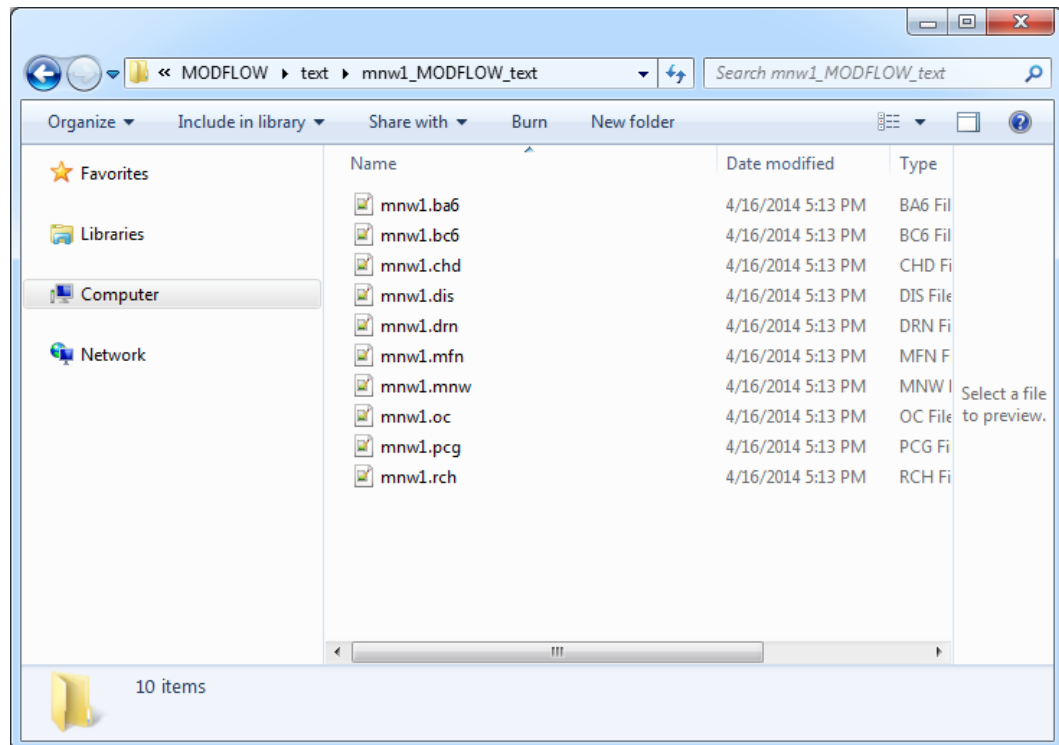


Figure 7 Native text MODFLOW files saved From GMS

5. Using a text editor, Open the “mnw1.bc6” file.

The file should look like the following figure.

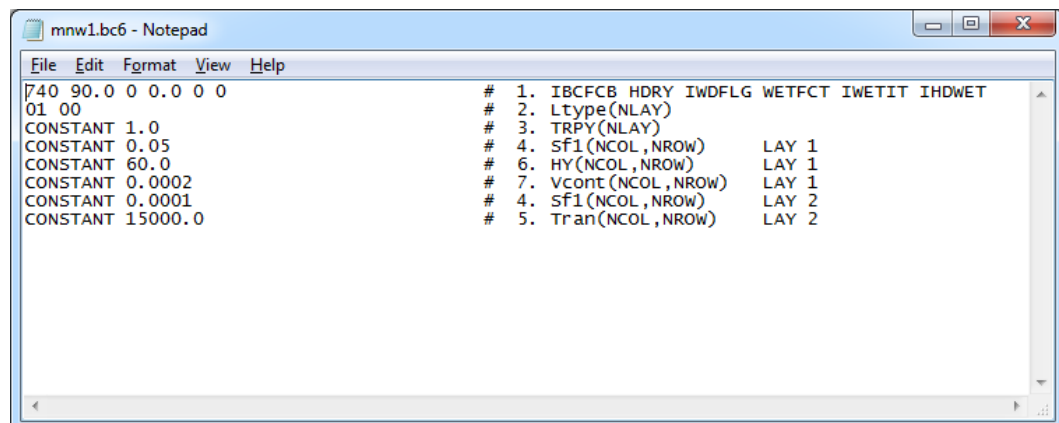


Figure 8 MODFLOW BCF file saved in native text format From GMS

Notice that this file is very similar to the original MODFLOW text file. There are some minor differences such as the format of the numbers. Look at the comments on the right side of the file. These all begin with the “#” symbol. The number on this line indicates the MODFLOW input dataset number for the BCF file. After the number, the user will see the documented variable name that is read from this line. These dataset numbers and variable names come straight from the MODFLOW documentation so if the user were to look at the BCF documentation, he or she will find these variables. The following image

shows the Online Guide to MODFLOW BCF documentation. Notice the dataset numbers and the variable names. These are the items written in the comments on the right side of the BCF file shown above.

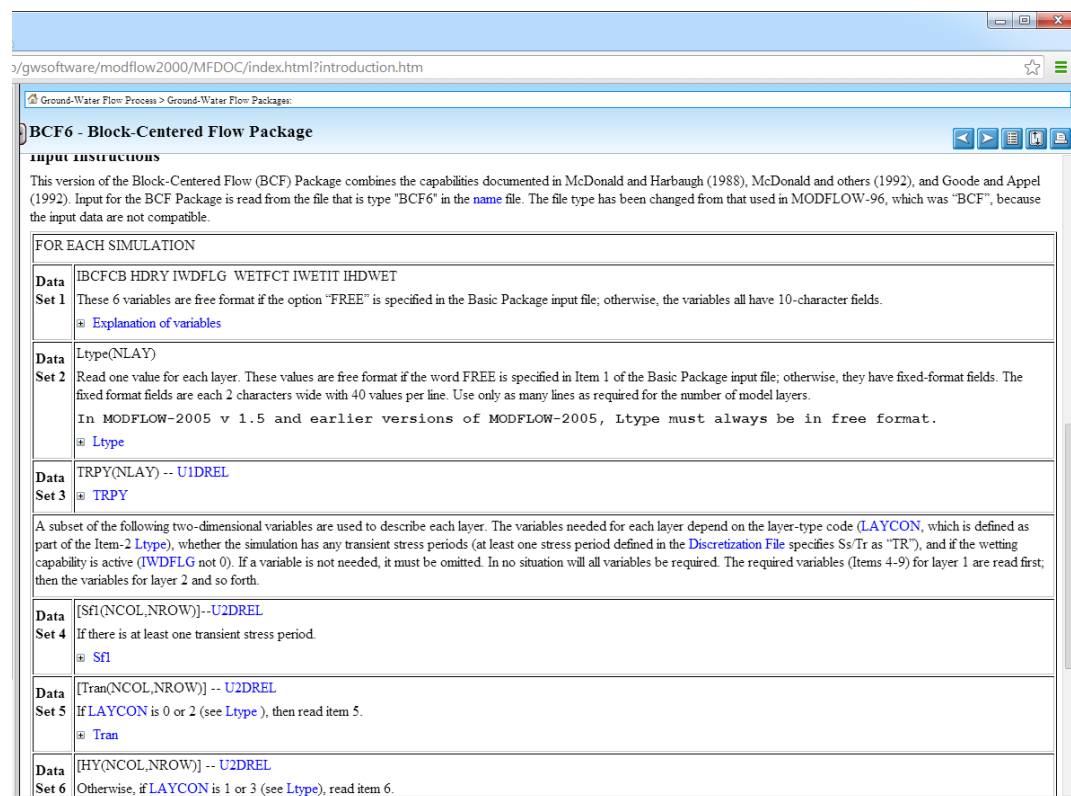



Figure 9 Online Guide to MODFLOW.

At this point, the user may wish to view some of the other MODFLOW input files. When finished, close the text editors and file browser and return to GMS.

## 8 Reading in the Simulation with Parameters

Now it is necessary to read in another MODFLOW simulation that has parameters.

1. Select the **File / New** command.
2. Select the **Open**  button.
3. Locate and open the directory entitled *Tutorials\MODFLOW\text\etsdrt*.
4. Select the file named “etsdrt.nam,” then click **Open**.
5. Click **OK** in the *MODFLOW Translator* dialog to import the model using MODFLOW 2005.
6. Click **Done** when MODFLOW 2005 has finished the file conversion.

The user should see a MODFLOW model similar to the figure below.

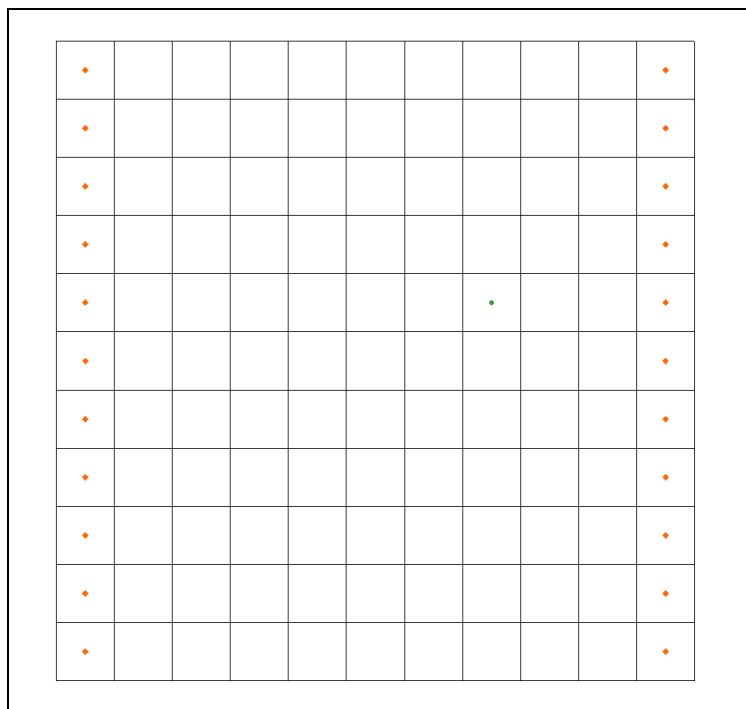


Figure 10 MODFLOW model of etsdrt simulation

7. Select the *MODFLOW* / **Parameters** menu command.

In the *Parameters* dialog, the user should see 4 parameters that were imported as part of the simulation as shown in the next figure.

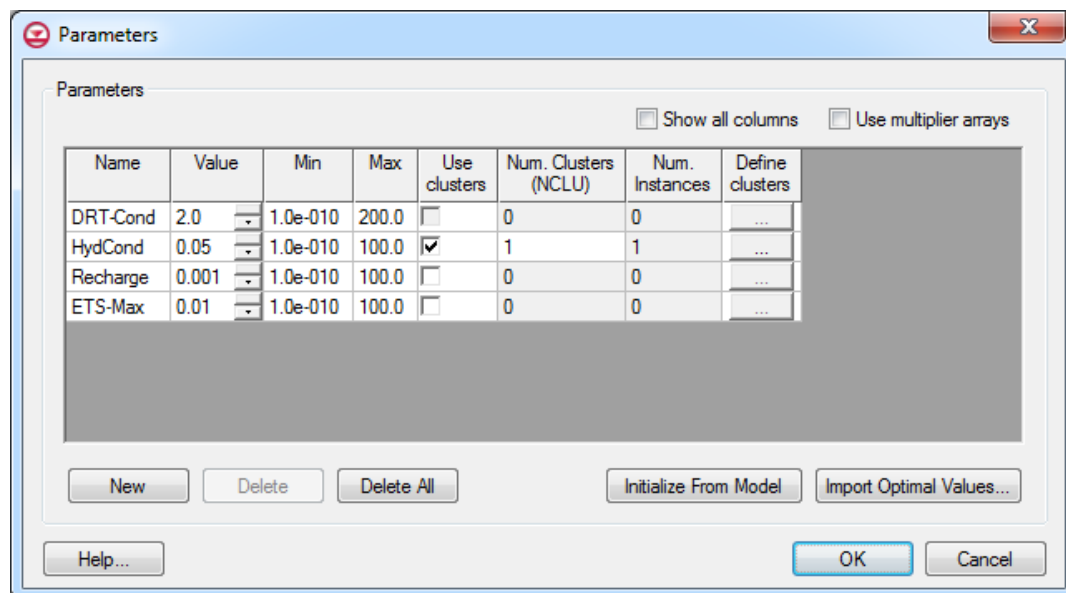


Figure 11 MODFLOW parameters

8. Click **OK** to exit the *Parameter* dialog.

When the user saves out the native text files, the parameters will be included in the text files as well.

## 9 Saving Native Text from GMS

---


The next step is to save this simulation in native text format.

1. Select the *MODFLOW* / **Global Options** menu command.
2. Under the *MODFLOW version* section of the dialog (near the top left), turn on the *Save native text copy* option.
3. Click **OK** to exit the dialog.
4. Select the *File* / **Save As** command.
5. Locate and open the directory entitled *Tutorials\MODFLOW\text*.
6. Using the **Save** button, save the project with the name “etsdrt.gpr.”

## 10 Viewing the Native Text

---

The next step is to view the text files to see how the parameters were included in the files.

1. Right-click on the “MODFLOW”  item in the Project Explorer.
2. Select the **Open Containing Folder** command.
3. Go up 1 directory in the file browser that opened.
4. Double-click on the “etsdrt\_MODFLOW\_text” directory.

The user should now see the following files:

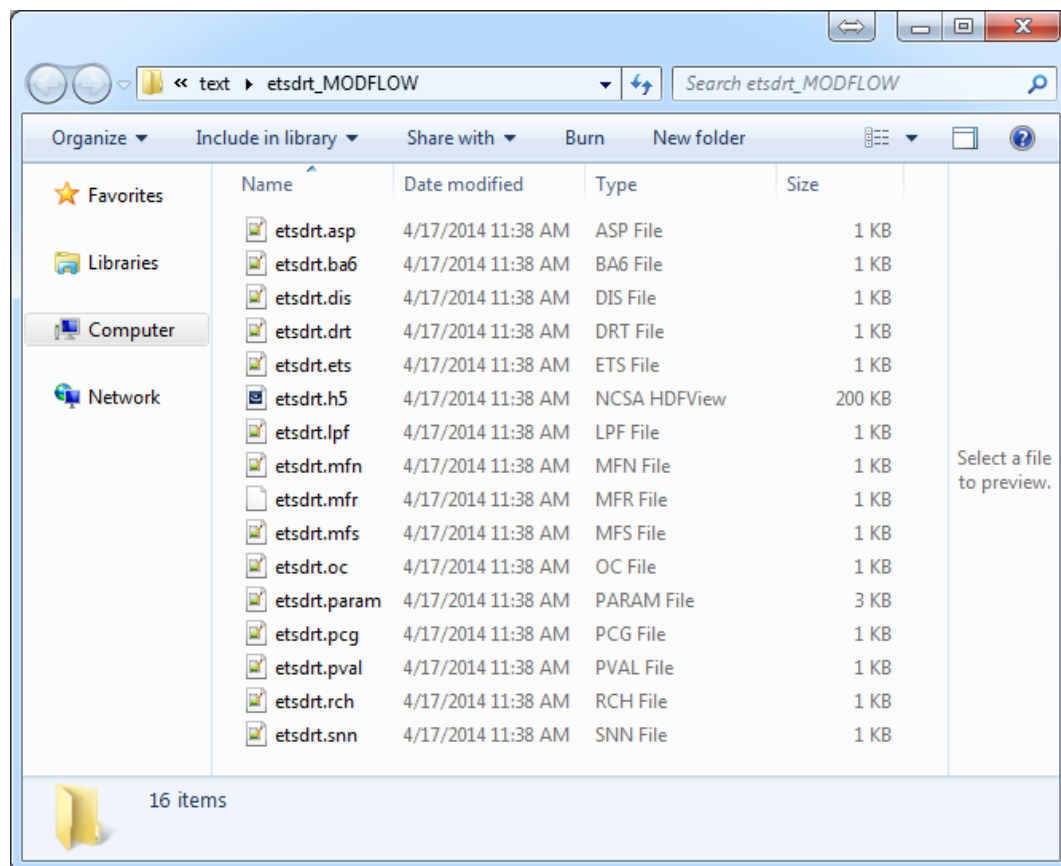


Figure 12 MODFLOW native text files

This tutorial will look at some of the files that have parameters defined. The ETS, DRT, RCH, and LPF packages all have parameters defined. The user will first look at the ETS package.

- Using a text editor, open the “etsdrt.ets” file.

The file should look like the following figure.

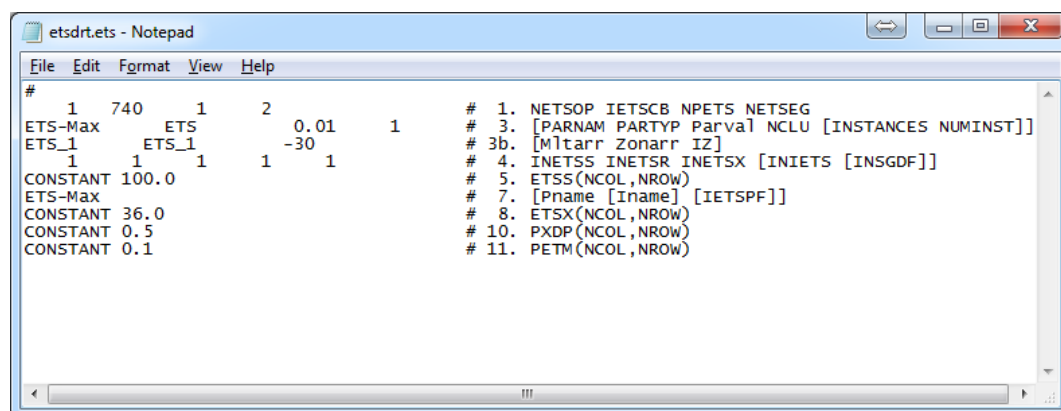


Figure 13      ETS package file

Note lines 3, 3b, and 7. The user can see that the parameter “ETS-MAX” has been included in the file. Now it is possible to look at the DRT file.

- Using a text editor, open the “etsdrt.drt” file.

The file should look like the following figure.

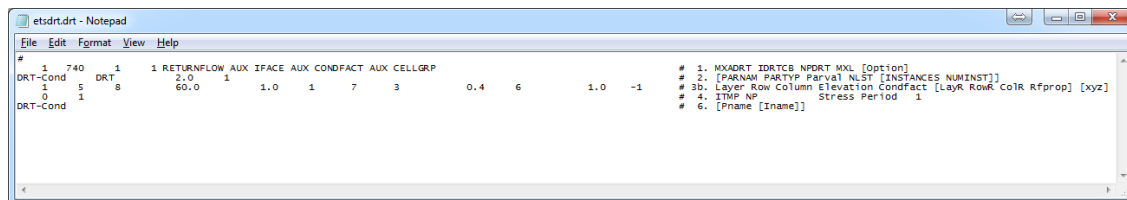



Figure 14 DRT package file

Notice lines 2, 3b, and 6. These lines define the parameter and its usage in the simulation. Also notice that the total width of the text in this file is much longer than the other files that the user has examined. The documentation comments are all aligned on the right side of the file and a small buffer is placed between the comments and the MODFLOW inputs. This makes reading the documentation easier than non-aligned comments.

## 11 Reading in the Simulation with Pilot Points

Now it is possible to read in a MODFLOW simulation that uses pilot points<sup>4</sup> with an array parameter.

- Select the *File* / **New** command.
- Select the **Open**  button.
- Locate and open the directory entitled *Tutorials\MODFLOW\text\pilot*.
- Select the file named “pilot.gpr,” then click **Open**.

The user should see a model similar to the next figure.

<sup>4</sup> Pilot Points are used with PEST as part of automated model calibration. To learn more about Pilot Points, complete the “MODFLOW – PEST Pilot Points” tutorial.

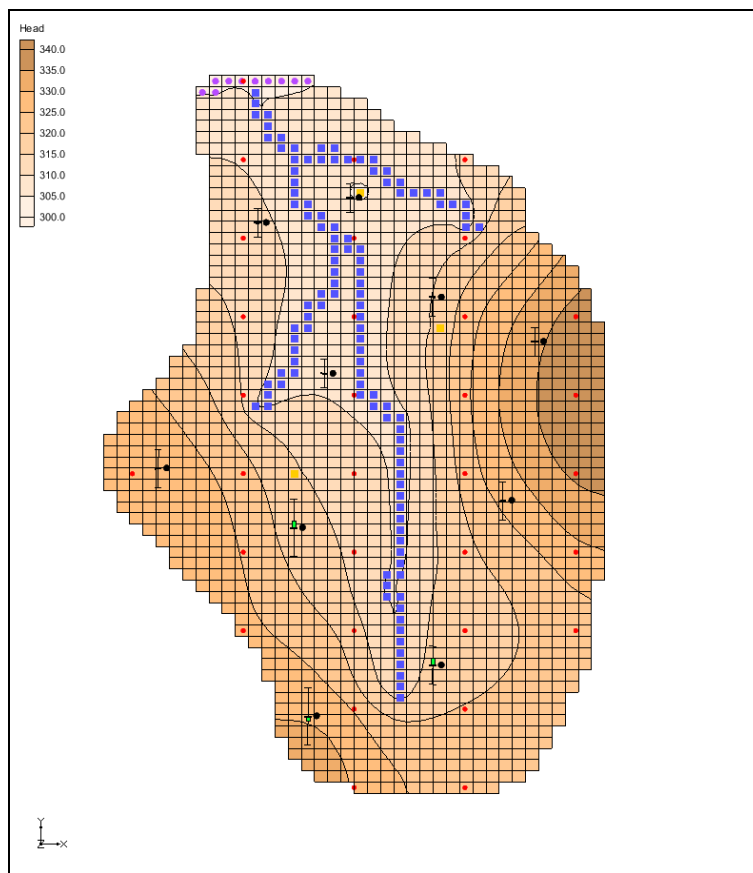


Figure 15 MODFLOW model with pilot points

## 12 Saving Native Text from GMS

The next step is to save this simulation in native text format.

1. Select the *MODFLOW* / **Global Options** menu command.


Notice that the *Save native text copy* option is not available. This option will only work with a “Forward” run MODFLOW simulation. So typically, the user would do the model calibration and then the result would be a final forward run model with all of the calibrated parameter values. Then the user can save out the native text. The same is true with the stochastic options; the user would have to switch back to a forward run to save the native text.

2. Under the *Run options* section (below *MODFLOW version* section) select the **Forward** option.
3. Under the *MODFLOW version* section of the *MODFLOW Global/Basic Package* dialog (near the top left), turn on the *Save native text copy* option.
4. Click **OK** to exit the dialog.

5. Select the *File* / **Save As** command.
6. Locate and open the directory entitled *Tutorials\MODFLOW\text*.
7. Using the **Save** button, save the project with the name “pilot.gpr.”

## 13 Viewing the Native Text

Now it is possible to view the text files and see how the parameters were included in the files.

1. Expand the “3D Grid Data” folder and the “grid” item.
2. Right-click on the “MODFLOW”  item in the Project Explorer and select the **Open Containing Folder** command.
3. Go up 1 directory in the file browser that opened.
4. Double-click on the “pilot\_MODFLOW\_text” directory.

The user should now see the following files:

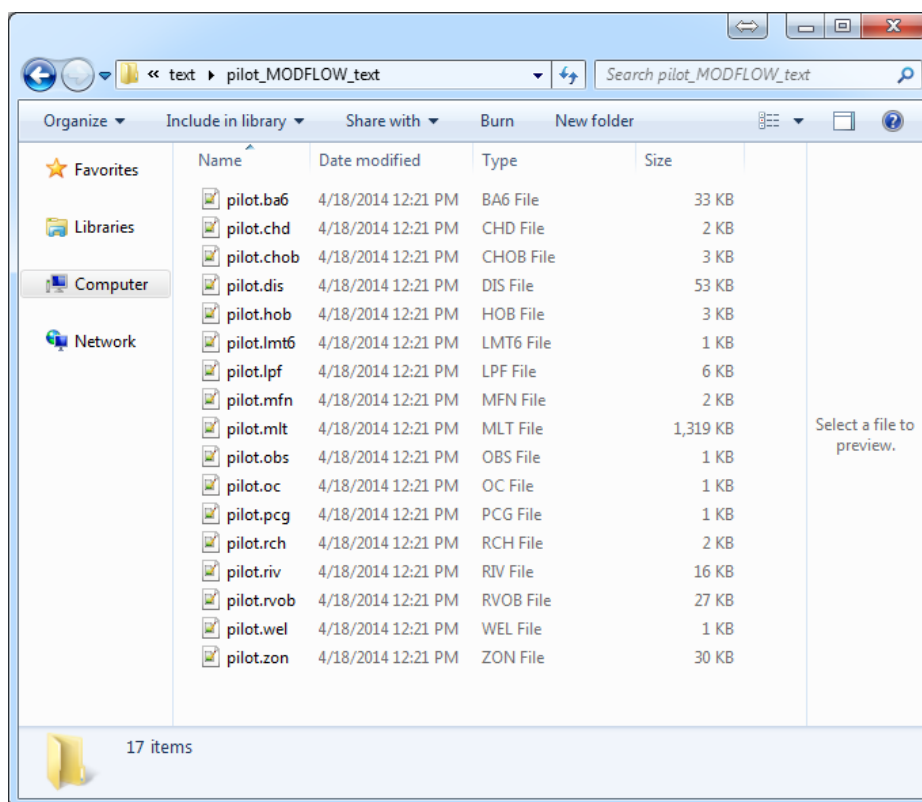


Figure 16 MODFLOW native text files



This tutorial will look at some of the files that are related to the pilot points. The user will first look at the LPF package.

- Using a text editor, open the “pilot.lpf” file.

The file should look like the following figure:

pilot.lpf - Notepad

File Edit Format View Help

```

740 -888.0 33 0
1
0
-1.0
0
0
pp1_1 HK 0.3341635 1
1 pp1_1 HK_1 -30
pp1_2 HK 0.3356953 1
1 pp1_2 HK_1 -30
pp1_3 HK 0.6361167 1
1 pp1_3 HK_1 -30
pp1_4 HK 0.4489954 1
1 pp1_4 HK_1 -30
pp1_5 HK 0.4000607 1
1 pp1_5 HK_1 -30
pp1_6 HK 0.7258828 1
1 pp1_6 HK_1 -30
pp1_7 HK 0.4001452 1
1 pp1_7 HK_1 -30
pp1_8 HK 0.4727968 1
1 pp1_8 HK_1 -30
pp1_9 HK 0.9242411 1
1 pp1_9 HK_1 -30
pp1_10 HK 0.2094655 1
1 pp1_10 HK_1 -30
pp1_11 HK 0.1311216 1
1 pp1_11 HK_1 -30
pp1_12 HK 0.5033053 1
1 pp1_12 HK_1 -30
pp1_13 HK 1.106962 1
1 pp1_13 HK_1 -30
pp1_14 HK 0.214915 1

```

# 1. ILPFCB HDYR NPLPF [Options]  
# 2. Ltype(NLAY)  
# 3. LAYAVG(NLAY)  
# 4. CHANI(NLAY)  
# 5. LAYVKA(NLAY)  
# 6. LAYWET(NLAY)  
# 7. [PARNAM PARTYP Parval NCLU]  
# 8. [Layer Mltarr Zonarr IZ]  
# 9. [PARNAM PARTYP Parval NCLU]  
# 10. [Layer Mltarr Zonarr IZ]  
# 11. [PARNAM PARTYP Parval NCLU]  
# 12. [Layer Mltarr Zonarr IZ]  
# 13. [PARNAM PARTYP Parval NCLU]  
# 14. [Layer Mltarr Zonarr IZ]

Figure 17 LPF package file

Notice that the user has all the lines in the file with the 8 and 9 line numbers. These lines define the pilot point parameters. The pilot point parameters are named `pp#_#`, where the first `#` is the index of the pilot point set (it is possible to have multiple sets of pilot point parameters) and the second `#` is an index for the point number. Now this tutorial will look at the multiplier file. This is where the multiplier arrays related to each pilot point are written.

- Using a text editor, open the “pilot.mlt” file.

The file should look like the following figure.

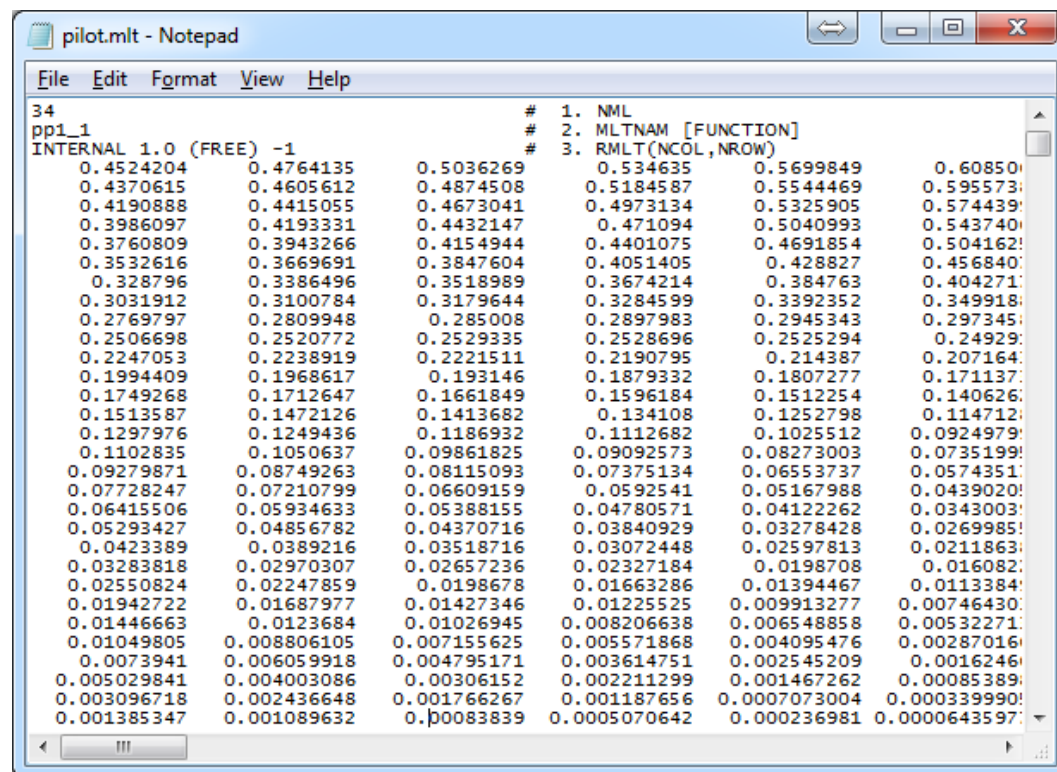


Figure 18 MLT package file

A multiplier array is created for each pilot point. When MODFLOW runs, the value at each point is multiplied by its multiplier array, resulting in one array for each pilot point. All of these arrays are then combined (added together) to create the final value at each cell. Looking at text files with array data can be difficult since the arrays contain such large amounts of data. GMS has exporting options to allow arrays to be external from the MODFLOW package files. This tutorial will try one of these options.

## 14 Saving Native Text from GMS with External Arrays

The next step is to save this simulation in native text format again with external arrays. Return to the GMS application.

1. Select the *Edit / Preferences* menu command.
2. In the *Preferences* dialog, select *MODFLOW* from the list on the left side of the dialog.
3. Turn on the *Save copy of MODFLOW simulation in native text format* option.
4. Below the *Save copy of MODFLOW...* check box, select the “Write external arrays to ‘arrays’ folder” option.
5. Select **OK** to exit the dialog.

6. Select the *File* / **Save** command.

Now return to the file browser where the native text files are shown. A new directory called “arrays” should have been created. First it is necessary to look at the MLT file again and then it will be possible to examine the arrays that were created in the “arrays” folder.

7. Using a text editor, Open the “pilot.mlt” file.

The file should look like the following figure.

```

34
pp1_1      # 1. NML
OPEN/CLOSE pilot_array_MLT_pp1_1.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_2      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_2.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_3      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_3.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_4      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_4.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_5      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_5.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_6      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_6.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_7      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_7.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_8      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_8.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_9      # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_9.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_10     # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_10.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_11     # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_11.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_12     # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_12.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_13     # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_13.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_14     # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_14.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_15     # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_15.txt 1.0 (FREE) -1 # 2. MLTNAM [FUNCTION]
pp1_16     # 3. RMLT(NCOL, NROW)
OPEN/CLOSE pilot_array_MLT_pp1_16.txt 1.0 (FREE) -1 # 3. RMLT(NCOL, NROW)

```

Figure 19 MLT package file with external arrays

Notice that, instead of seeing large amounts of floating point numbers in this file, the user will just see a reference to an external file. This is what the **OPEN/CLOSE** statement in the file means. Also notice the format of the names of the array files.

External arrays are placed in their own text files and are named as follows

“myProject\_array\_PACKAGE\_VARIABLE\_LAYERorSP.” In the current example, the first pilot point multiplier array is named “pilot\_array\_MLT\_pp1\_1.txt.” So “pilot” is the project name. “MLT” is the package. And “pp1” is the variable (or parameter). Lastly, “1” refers to the pilot point index. Here are two more examples:

- The ibound array for layer 1 would have the following name:  
“myProject\_array\_BAS\_IBOUND\_1.txt.”
- The recharge array for stress period 3 would have the following name:  
“myProject\_array\_RCH\_RECH\_3.txt.”

If The user returns to the file browser, he or she can go into the arrays directory and see all of the arrays created when the files were saved.

To export the arrays, it was necessary to turn on a preference that will remain turned on in any new GMS project. Do the following to return GMS preference to its default state.

8. Select the *Edit* / **Preferences** menu command.
9. In the *Preferences* dialog, select *MODFLOW* from the list on the left side of the dialog.
10. Turn off the *Save copy of MODFLOW simulation in native text format* option.
11. Select **OK** to exit the dialog.

## 15 Conclusion

---

This concludes the tutorial. Here are the key points of this tutorial:

- HDF5 is a binary, cross-platform file format.
- GMS stores most of the MODFLOW data in an HDF5 file.
- GMS can save out native text format MODFLOW files.
- Native text format MODFLOW files from GMS also include useful documentation comments.
- It is possible to save out native text format MODFLOW files from GMS with external arrays.