

ModelArchiver—A Program for Facilitating the Creation of Groundwater Model Archives

Open-File Report 2017–1149

ModelArchiver—A Program for Facilitating the Creation of Groundwater Model Archives

By Richard B. Winston

Open-File Report 2017–1149

**U.S. Department of the Interior
U.S. Geological Survey**

U.S. Department of the Interior

RYAN K. ZINKE, Secretary

U.S. Geological Survey

William H. Werkheiser, Deputy Director
exercising the authority of the Director

U.S. Geological Survey, Reston, Virginia: 2018

For more information on the USGS—the Federal source for science about the Earth, its natural and living resources, natural hazards, and the environment—visit <https://www.usgs.gov> or call 1–888–ASK–USGS.

For an overview of USGS information products, including maps, imagery, and publications, visit <https://store.usgs.gov>.

Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

Although this information product, for the most part, is in the public domain, it also may contain copyrighted materials as noted in the text. Permission to reproduce copyrighted items must be secured from the copyright owner.

Suggested citation:

Winston, R.B., 2018, ModelArchiver—A program for facilitating the creation of groundwater model archives: U.S. Geological Survey Open-File Report 2017–1149, 15 p., <https://doi.org/10.3133/ofr20171149>.

ISSN 2331-1258 (online)

Preface

The programs ModelArchiver and FgdcMetaEditor can be downloaded from the U.S. Geological Survey for free. The performance of the ModelArchiver and FgdcMetaEditor have been tested in a variety of applications. Future applications, however, might reveal errors that were not detected in the test simulations. Users are requested to send notification of any errors found in this software documentation report or in the program codes to the contact listed on the software release web page (<https://doi.org/10.5066/F73X85M1>). Updates might be made to the report and to the model program. Users can check for updates on the ModelArchiver and FgdcMetaEditor web page.

Contents

Preface	iii
Abstract	1
Introduction.....	1
Purpose and Scope	1
Archive Structure.....	2
Creating Model Archives.....	2
Getting Started Pane	4
Readme.txt File Pane.....	4
Modelgeoref.txt File Pane	4
Georef Pane	5
Browse Graphic Pane	5
Metadata Pane.....	5
File Extensions Pane.....	5
Archive Lists Pane	6
Files Pane	6
Archive Structure Pane	7
Create Archive Pane	8
FgdcMetaEditor.....	8
Examples of Modifications in .Archive Files	9
RunSutra.Bat File	9
MODFLOW Name File.....	10
MODPATH Name File.....	11
Modifications to ModelMuse to Support ModelArchiver.....	11
Summary	13
Acknowledgments	13
References Cited.....	13
Appendix 1. Structure of Archive List Files and ModelMuse Integration.....	14
References Cited in Appendix 1	15

Figures

1. Screenshot of a typical working directory that does not satisfy the USGS model archiving policy.....	3
2. Screenshot of model input, output, and ancillary files automatically assigned to archive directory structure.....	3
3. Screenshot of a pane in ModelArchiver for selecting the readme.txt file for the model archive.....	4
4. Screenshot showing the Archive Structure Pane in ModelArchiver illustrating the use of color to highlight problems in the archive.....	7
5. Screenshot showing FgdcMetaEditor, using color to indicate missing data.....	9
6. Screenshot showing an unmodified RunSutra.bat file.....	9
7. Screenshot showing RunSutra.bat file modified to create output directory and run the model from the bin directory.....	10
8. Screenshot showing an unmodified MODFLOW name file.....	10
9. Screenshot showing a MODFLOW name file modified to generate output files in a directory separate from the input files.....	10
10. Screenshot showing an unmodified MODPATH name file.....	11
11. Screenshot showing a MODPATH name file with directory paths modified.....	11
12. Screenshot showing an example Archive List file.....	12
1-1. Screenshot showing an unmodified RunModflow.bat file.....	14
1-2. Screenshot showing a RunModflow.bat.archive file.....	14

ModelArchiver—A Program for Facilitating the Creation of Groundwater Model Archives

By Richard B. Winston

Abstract

ModelArchiver is a program designed to facilitate the creation of groundwater model archives that meet the requirements of the U.S. Geological Survey (USGS) policy (Office of Groundwater Technical Memorandum 2016.02, <https://water.usgs.gov/admin/memo/GW/gw2016.02.pdf>, <https://water.usgs.gov/ogw/policy/gw-model/>). ModelArchiver version 1.0 leads the user step-by-step through the process of creating a USGS groundwater model archive. The user specifies the contents of each of the subdirectories within the archive and provides descriptions of the archive contents. Descriptions of some files can be specified automatically using file extensions. Descriptions also can be specified individually. Those descriptions are added to a readme.txt file provided by the user. ModelArchiver moves the content of the archive to the archive folder and compresses some folders into .zip files.

As part of the archive, the modeler must create a metadata file describing the archive. The program has a built-in metadata editor and provides links to websites that can aid in creation of the metadata. The built-in metadata editor is also available as a stand-alone program named FgdcMetaEditor version 1.0, which also is described in this report. ModelArchiver updates the metadata file provided by the user with descriptions of the files in the archive. An optional archive list file generated automatically by ModelMuse can streamline the creation of archives by identifying input files, output files, model programs, and ancillary files for inclusion in the archive.

Introduction

A key element of scientific research is that results should be reproducible. Reproducing the results of a groundwater model from a model report can be challenging because the report may not fully describe the principles of the construction of the model. Model archives can be used to overcome this limitation by storing the input and output of the model, along with other relevant information. The archive can be used in the future to better understand the model or to modify it to address new concerns.

The U.S. Geological Survey (USGS) Office of Groundwater Technical Memorandum 2016.02 (<https://water.usgs.gov/admin/memo/GW/gw2016.02.pdf>, <https://water.usgs.gov/ogw/policy/gw-model/>) sets the policy for the creation and distribution of groundwater model archives. A specific directory structure is required for the archives. The archive must contain files that identify the contents and location of the model or models included in the archive. The archives are made available to the public through the USGS Science Data Catalog at <https://data.usgs.gov/datacatalog>. Groundwater model archive metadata files are expected to include the keyword `usgsgroundwatermodel`. Older archives may not include that keyword.

ModelArchiver is a program that facilitates the creation of archives that meet policy requirements. ModelArchiver can be used by anyone. However, some of the websites to which ModelArchiver is linked may require a USGS login. Compiled versions of ModelArchiver, FgdcMetaEditor, source code, and example data may be obtained from Winston (2017).

Purpose and Scope

This report describes how to use ModelArchiver version 1.0 to create groundwater model archives that are consistent with USGS policy and guidance. The report also describes a separate program, FgdcMetaEditor version 1.0, that can be used to create metadata for groundwater model archives or for other types of geographic data. The steps needed to create model archives are presented.

Archive Structure

The required archive structure is described in detail at <https://water.usgs.gov/ogw/policy/gw-model/modelers-setup.html>. For convenience, the archive structure is summarized here.

Base directory: The base directory of the archive is a folder whose name should be related to the report in which the model results are presented. The base directory contains two text files—`readme.txt` and `modelgeoref.txt`—and a number of subdirectories.

Readme.txt: The `readme.txt` file is a human-readable text file describing, among other things, the operating system, the directory structure, and the contents of each subdirectory.

Modelgeoref.txt: The `modelgeoref.txt` specifies, among other things, the location of a model. Its purpose is to make it easier to find a model on the basis of its location.

Model directory: The model directory contains the input files for the model. If the archive contains more than one model, each model is placed in a separate subdirectory. For each model, a `usgs.model.reference` file containing information specific to the model is required. For purposes of archiving, child models in MODFLOW-LGR models are considered separate models. If postprocessors such as MODPATH (Pollock, 2012) or ZONEBUDGET (Harbaugh, 1990) are used, they are considered separate models. If parameter estimation software is used and a report presents subsequent analysis, based on the parameter estimation software results, the input files for the parameter estimation are put in the model directory. If the report does not present any subsequent analysis of the parameter estimation results, the input files could be put in the ancillary or nonpublic directories.

Output directory: For each model in the model directory, there is a corresponding output directory containing the output of the model. Like the model directory, the output directory must have a separate subdirectory for each model if the archive contains more than one model.

Bin directory: The bin directory contains compiled executable versions of the model programs.

Georef directory: The georef directory contains a polygon shapefile that outlines the active and inactive extents of the model.

Source directory: The source directory contains the source code for model programs used in the project.

Webrelease directory: The webrelease directory contains two files. One is a Federal Geographic Data Committee (FGDC) XML (Extensible Markup Language) metadata file (Federal Geographic Data Committee, 1998, 2000) describing the archive. The other is a browse graphic image of the study area.

Ancillary directory: The optional ancillary directory contains additional files that might be useful to users of the model archive.

Nonpublic directory: The optional nonpublic directory contains information that cannot be released publicly, such as the source code for proprietary programs or personally identifiable information.

Creating Model Archives

It is not uncommon for the working directory for a modeling project to be in a directory structure different than that required by USGS policy. Input, output, and ancillary files might be in the same working directory. The working directory might include files that are not intended for inclusion in the model archive. MODFLOW, MODPATH, and ZONEBUDGET files might be in the same directory, even though policy treats those as separate models, which would be placed in separate directories (fig. 1).

ModelArchiver is designed to lead the user step-by-step through the process of setting up the archive. However, the user can skip any of the steps and come back to them later, if desired. ModelArchiver allows the user to select multiple files or directories, and then uses file extensions to sort them into suitable input or output directories. It automatically assigns a description of each file that will be included in the `readme` and metadata files (fig. 2).

ModelArchiver presents the user with a series of panes, each of which is specialized for setting up one part of the archive. Navigation between panes can be done by using Next or Back buttons to switch between panes in the recommended order (fig. 3). The user also can navigate to a pane by selecting the name of the desired pane from a list on the left of the main form. Each pane allows the user to perform one step in the creation of an archive. Each pane has a text control with a detailed description of the purpose of the pane and how to use it. Additional help on each pane is available by pressing the F1 key. Next to the Next and Back buttons is a button that will open the USGS Groundwater Modeling Archive website in the user's default browser.

The user can save the information for creating an archive by selecting File|Save and the name of a file in which to store the information. The file can be opened later by selecting File|Open.

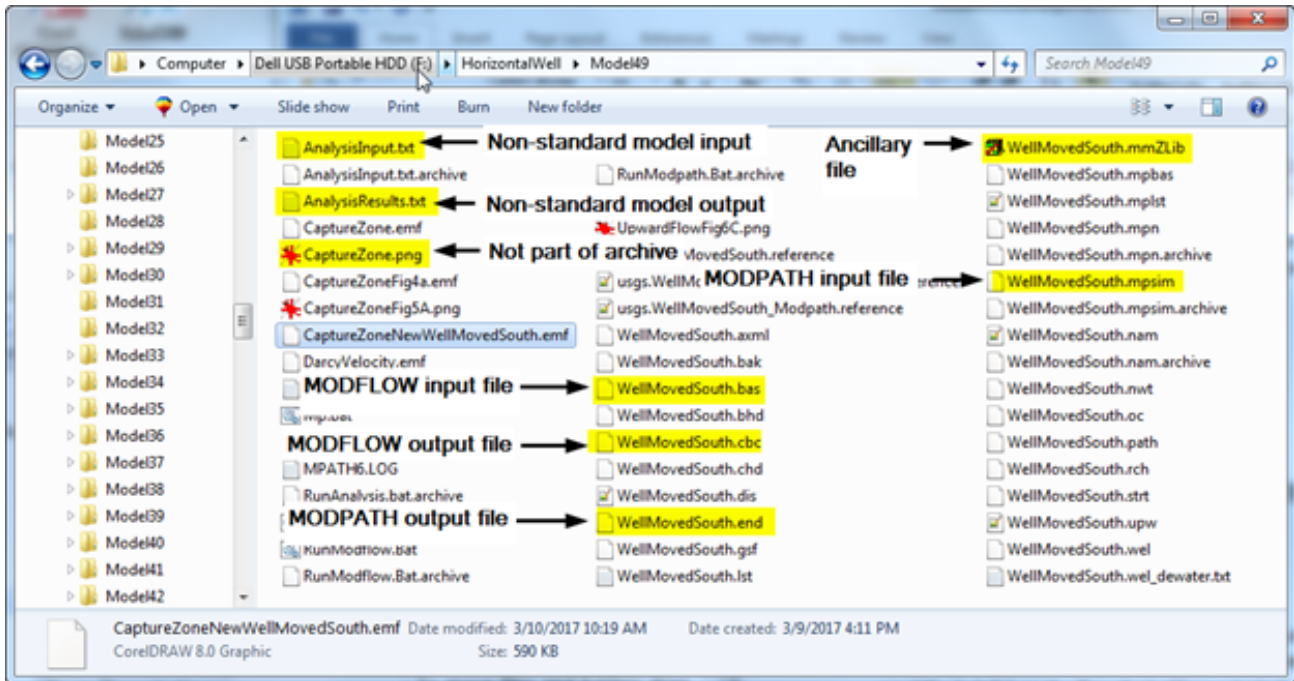


Figure 1. Screenshot of a typical working directory that does not satisfy the USGS model archiving policy.

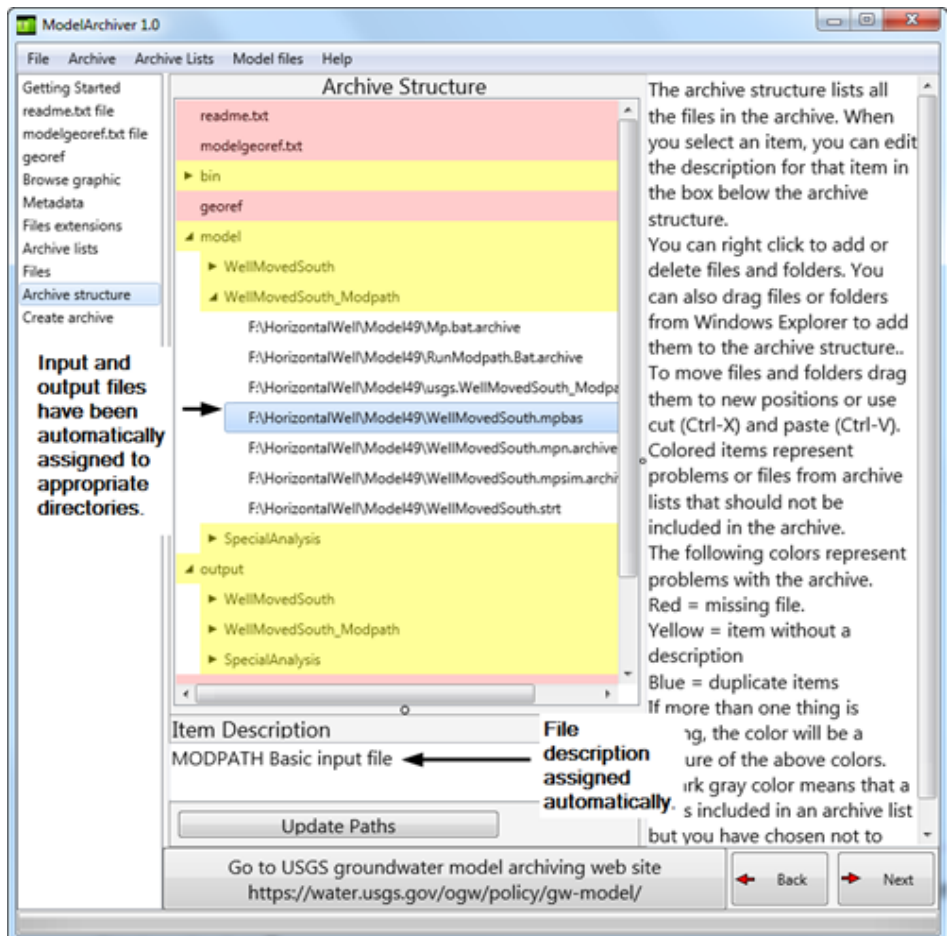


Figure 2. Screenshot of model input, output, and ancillary files automatically assigned to archive directory structure.

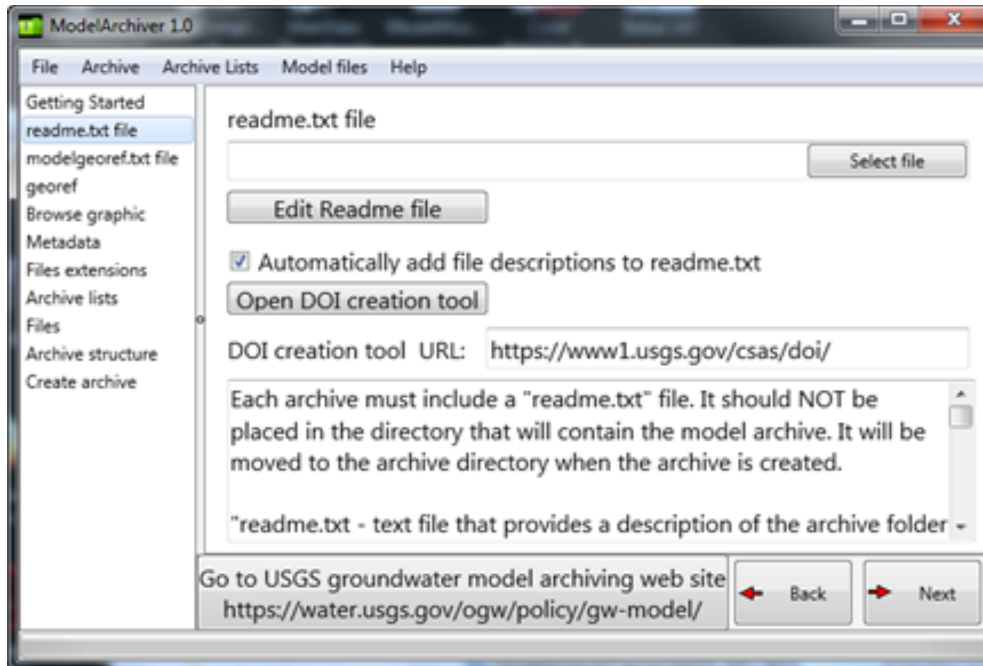


Figure 3. Screenshot of a pane in ModelArchiver for selecting the readme.txt file for the model archive.

Getting Started Pane

The Getting Started pane has a button with a link to the USGS Groundwater modeling website. Clicking the button will open the website in the users default web browser. The user is advised to read through the website to understand the process of creating and reviewing a groundwater model archive.

Readme.txt File Pane

Each model archive is required to have a file named “readme.txt” that gives a description of the archive. The user can select and edit the readme.txt file in this pane. The text on the pane provides a more complete description of the suggested contents of the readme.txt file. The readme.txt file will usually include the Digital Object Identifier (DOI) for the archive. USGS users can access a website where a DOI can be reserved by clicking the button Open DOI creation tool.

One important part of the readme.txt file is a description of all models and all the files included in the archive. This information will be specified on the Archive Structure pane. After the readme.txt file is copied to the archive directory, these descriptions will be added to the end of the copy of the readme.txt file in the archive directory. Therefore, the user does not need to manually include that information in the readme.txt file. If the user wishes to provide those descriptions manually, the user can uncheck the Automatically add file descriptions to readme.txt checkbox. By default, the checkbox is checked.

Modelgeoref.txt File Pane

The modelgeoref.txt file is a text file that gives the location of the model study area in decimal degrees. It can be used by web programs to locate the model area on a map. The user selects the modelgeoref.txt file in this pane. The text on the pane provides more detail on how to set up a modelgeoref.txt file.

Georef Pane

Each archive must include a shapefile that shows the active and inactive extent of the model domain in all layers. A shapefile is actually a collection of files. Required files have the extensions .shp, .shx, and .dbf (see <https://www.esri.com/library/whitepapers/pdfs/shapefile.pdf> accessed November 2, 2016). Other file extensions are used for optional files included in a shapefile. In the Georef pane, the user can select files with the extension .shp. The selected file and the related files that are part of the selected shapefile will be included in the archive. A suitable shapefile can be created using ModelMuse (Winston, 2009, 2014) with the command “File|Export|Shapefile|Model Outline or Gridlines...”

Browse Graphic Pane

Each archive must include a small graphic image of the model area. The user selects this file on the Browse graphic pane. This image may be displayed on cataloging sites. Users of the cataloging sites might use the image when deciding whether a particular archive is of interest. A suitable graphic might overlay the model boundary on a State or regional map.

Metadata Pane

Each archive must include a metadata XML file. The Metadata pane is used to select the metadata file. The metadata file can be created or edited by clicking the Edit metadata file button. This opens a built-in metadata editor. A stand-alone version of the editor is described in the “FgdcMetaEditor” section of this document.

The Metadata pane also includes buttons to open several websites that would be important in creating the metadata file. These sites include the following:

- The metadata web page in the USGS Groundwater Modeling Archive website, which explains special metadata requirements for model archives;
- The USGS DOI Creation Tool, which is used to generate Digital Object Identifiers that might be needed in the archive;
- The Open Metadata Editor website, which is an alternative to the built-in metadata editor for creating the metadata file; and
- The Geospatial Metadata Validation Service, which can be used to check for errors in the metadata structure.

Users typically would use those websites in the order in which the buttons are displayed.

If the Add file descriptions to metadata checkbox is checked, the descriptions of the files or folders in the Archive Structure pane will be automatically included in the Distribution Information section of the metadata file. The checkbox is checked by default.

File Extensions Pane

ModelArchiver provides several ways to add the model input and output files to the archive.

- The user can select a parent directory. All files in that directory or its subdirectories that have extensions recognized by ModelArchiver will be added to the archive structure.
- The user can select archive list files generated by preprocessors that identify files to include in the archive. (As of 2017, only ModelMuse (Winston, 2009, 2014) creates archive list files.)
- The user can select files individually to add to the archive.

ModelArchiver will assign descriptions to files automatically on the basis of the file extensions of the file names. The extensions recognized by ModelArchiver and the associated default descriptions can be modified by the user by clicking the Edit File Extensions button. This will open the File Extensions dialog box. Each recognized extension also has an associated file type. The file types are Model input, Model output, Modpath input, Modpath output, Zonebudget input, Zonebudget output, MT3D input, MT3D output, Other input, Other output, and Ancillary. The user can select one of these file types for each recognized file extension.

6 ModelArchiver—A Program for Facilitating the Creation of Groundwater Model Archives

Each extension is displayed in a cell in a table in the File Extensions dialog box. The number of extensions in the table is shown in an edit box at the bottom of the dialog box. To add a new extension to the table, the user increases the number of extensions shown in the edit box. The user then fills in the empty row or rows at the bottom of the table. Clicking the OK button will save any changes made in the File Extensions dialog box and will close the dialog box. The table of extensions, associated file types, and file descriptions is automatically saved in an initialization file on the computer when a ModelArchiver project is saved. Each time the user starts ModelArchiver, these data are loaded into ModelArchiver. Thus, all projects opened by ModelArchiver will use the same descriptions and file types for archive files newly added to a project. If the user wishes to revert to the default values for the extensions, the user can click the Use default file extensions button in the File Extensions pane of ModelArchiver.

Files that have the extension .archive receive special treatment when creating the archive. The .archive will be removed from the file name when the file is copied to the archive. This special treatment can be used to provide a modified version of an input file that is more useful than the original file in the context of a model archive. For example, the original MODFLOW name file of a model might create the output files in the same directory as the input files. A similar name file, for example AModel.nam.archive, could be created for the archive. It would be the same as the original name file except that it would be modified to create the output files in a separate directory. When generating the input files for models, ModelMuse (Winston, 2009, 2014) will create some files with the .archive extension for use in archives. For users who might wish to create such files manually, examples of the modifications in such files are included in the section of this report “Examples of Modifications in .Archive Files.”

Archive Lists Pane

Archive lists are XML files that identify other files that belong in the model, output, ancillary, and bin sections of a model archive. Archive lists files have the extension .axml. Archive lists are created by ModelMuse, version 3.9, and later. Users who wish to create archive lists in some other fashion can consult the appendix for detailed information on the structure of archive lists.

Archive lists are not required in order to use ModelArchiver so this pane may be skipped if not applicable. If archive lists are available, the user can select them in the Archive Lists pane. The names of the Archive list files are displayed at the top of the Archive Lists pane, and the files included in the archive lists will be included in the Archive Structure pane.

Archive lists can be selected in two ways. To select them individually, the user clicks the Select individual archive lists button and selects one or more .axml files. The other method is to click the Open archive list directory button. This method is useful when each one of several separate models is in a separate subdirectory of a parent directory. When this method is selected, ModelArchiver searches all of the subdirectories of the selected directory for .axml files and adds them to the archive lists.

Archive lists can be removed by selecting them, right-clicking, and selecting Delete from the pop-up menu. If any of the archive lists or model files have been moved to a new drive or directory, the user can update the paths by clicking the Update paths button. A dialog box will be displayed in which the user can specify the old and new directories for the files. The new directory name will be substituted for the old name in the paths for all the files that contain the old directory name.

Files Pane

The Files pane is used to include files in the archive by selecting a folder. (Files can also be added individually on the Archive Structure pane.) Any file in the selected folder or a subfolder with an extension recognized by ModelArchiver will be included in the archive. The file extensions will be used to assign each file to a file type, and the file types represented in each folder, along with the folder names, will be used to define folders in the archive. This automatic restructuring of the original directory structure to the directory structure required in the model archive is one of the key benefits of ModelArchiver. For example, if the folder is named Model1 and it contains input and output files for MODFLOW and MODPATH, four folders will be defined in the archive. There will be two folders—model.Model1 and model.Model1_Modpath—in the model directory and two more folders—output.Model1 and output.Model1_Modpath—in the output directory. The placement of files in the archive structure may need to be modified when this method is used. For example, the heads generated by one MODFLOW model might be used to specify the initial heads in another MODFLOW model. The same extension might be used for the heads file regardless of whether it was an input or output file. It would not be possible to determine the proper placement of the file by its extension alone. Another example is the usgs.<model name>.reference files. It would not be possible to identify whether such files belong in a MODFLOW, MODPATH, or ZONEBUDGET model directory from the extension alone. Other examples are files with the extension .csv. ZoneBudget creates such files as output files. However, .csv files are simply files of values separated by commas. The .csv files could be used as input for a model. Still another example would be files with the extension .GWMWFILE.wel, which are created as output by MODFLOW-GWM (Ahlfeld and others, 2005) but then can be used as input for the well package in MODFLOW.

Archive Structure Pane

The Archive Structure pane shows all the files to be included in the archive arranged in the folder structure that will be used in the archive. This pane can be used to add or remove individual files, to move them from one place to another in the archive structure, or to make other changes to the archive structure. The Archive Structure pane is also where the descriptions of individual items are assigned. To change which files are included in the archive structure, click on the location in the archive structure where you wish to make a change. Right click the mouse, and select the desired option from the pop-up menu. You can add or delete files or folders. To edit the description of an item, select the item, and enter the description in the edit box labeled Item Description (fig. 4).

ModelArchiver uses color to highlight problems in the model archive (fig. 4) or to highlight files in archive lists that will not be included in the final archive. Three colors are used to indicate problems: red, yellow, and blue. If a particular item has multiple problems, the color will be a blend of two or more of these colors. Red indicates that a required file in the archive has not been specified or does not exist at its specified location. In figure 4, the webrelease folder is red because USGS policy requires that it contain a browse graphic file and a metadata file, but it contains nothing. On the other hand, the nonpublic directory is not colored because it is optional and is allowed to be empty. Yellow indicates that a description for the item is missing. In figure 4, the Model6_Modpath item is yellow because no description has been provided for it. Its parent directory, output, is also yellow despite having a description because one of the items within it lacks a description. Blue indicates a duplicate file name. There are several items colored blue in figure 4. If you remove the .archive extension from RunModpath.bat.archive, its name would be RunModpath.bat, which is the name of another file in the same location in the archive. Because ModelArchiver removes .archive extensions from files when they are copied to the archive directory, the two files would end up having the same name in the archive. This is not allowed and would cause an error.

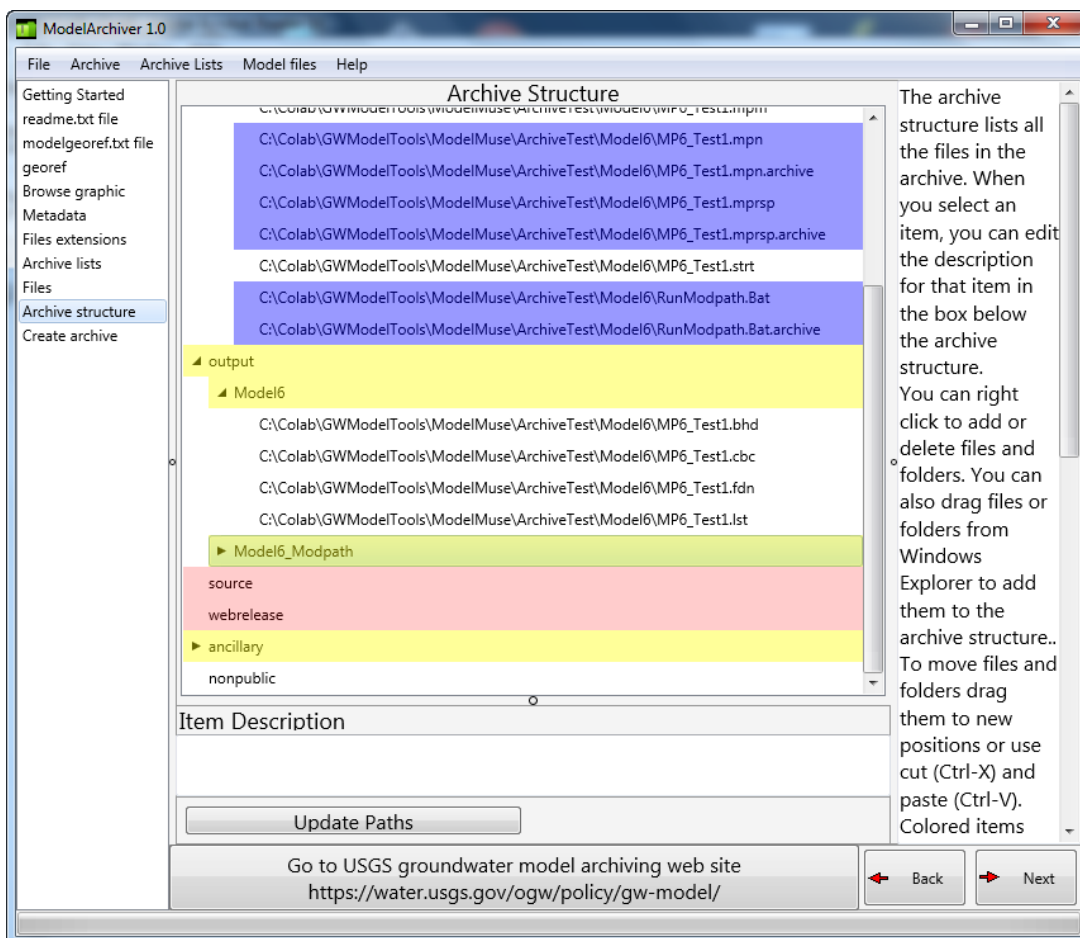


Figure 4. Screenshot showing the Archive Structure Pane in ModelArchiver illustrating the use of color to highlight problems in the archive.

8 ModelArchiver—A Program for Facilitating the Creation of Groundwater Model Archives

If a file is added to the archive structure by virtue of being included in an archive list, the file will not be removed from the archive structure if the user attempts to delete it. Instead, it will be marked as a file not to be included in the archive and will have a dark gray background. It might be desirable to delete ancillary files in an archive list if they are not actually relevant to the archive. For example, suppose the user imported data from a shapefile but subsequently removed the imported data from the model. The shapefile would still be included in the archive list, even though that shapefile was no longer relevant to the model.

ModelArchiver does not check that the correct files have been included in the archive; that is the user's responsibility. For example, a model might have been run with one version of MODFLOW, but the user might have included a different version of MODFLOW in the archive. Such changes might cause the output from the model to be different or even prevent the model from running at all.

The Update paths button on the Archive Structure pane behaves the same way as the button in the Archive Lists pane.

Create Archive Pane

The Create Archive pane has an edit box where the user can select the location for a new archive, and four buttons are provided to create or update an archive. The user is expected to ensure that the directory name is consistent with the naming convention for model archives at a particular USGS Science Center.

When the Create archive button is clicked, the program first checks that there is enough free space on the disk where the archive is to be created. If there is, the program copies all the files listed in the archive structure to the archive directory. If any of the files has an .archive file extension, that extension will be removed from the file name. After being copied, the readme.txt file will be modified by adding the file descriptions from the archive structure to the end of the copy of the readme.txt file in the archive. The model archiving policy requires that some of the folders be compressed into .zip files. ModelArchiver performs that compression. In the final step, the program updates the Distribution Information section of the metadata with the names and descriptions of the files in the archive. (Files inside .zip files are not listed in the metadata.)

When the Update Archive button is clicked, a dialog box will open that compares the files that are supposed to be in the archive, as shown in the Archive Structure pane, with those that are actually in the archive. A tree control will display both sets of files. Color is used to indicate those files that are different in the Archive Structure pane from those in the actual archive. A button in the dialog box can be used to update files in the actual archive and the readme.txt and metadata files.

When the Move Files button is clicked, the files in the archive are moved to the selected archive directory. No other tasks are performed. This option can be useful, if the user wishes to check that the model or models can be run correctly. It also might be used if the user has an alternative method for compressing the files.

The Update Metadata button is used to update the metadata without making any other changes to the archive. This button might be used if the user has manually added a new file to the archive after the archive was created. The Update Readme.txt button is used to update the readme.txt file in the archive by copying the originally specified readme.txt file and adding the file descriptions specified in the Archive Structure pane.

FgdcMetaEditor

FgdcMetaEditor is an editor for FGDC metadata (Federal Geographic Data Committee, 1998, 2000). It is a stand-alone version of the metadata editor built into ModelArchiver. (See the "Metadata Pane" section of this document.) The left side of the editor is a tree control that includes the names and numbers of the standard tags in FGDC metadata. Each item has either a checkbox or radiobutton. If the checkbox or radiobutton is checked and all its parent items are also checked, the associated item will be included in the metadata file. If an item is required, the checkbox will be disabled so that the user cannot uncheck it. When the user selects an item, the right hand side shows a description of the item and provides a way to edit that item (fig. 5). The background color of the description indicates whether an item is always required, required if applicable, or optional using the same color scheme as in Federal Geographic Data Committee (2000)—yellow for required items, green for required if applicable items, and blue for optional items. If an item will be included in the metadata file but has not been specified, its name has a blue background on the left hand side (fig. 5). If the FGDC standard requires the metadata to specify one of several options and none of those options has been selected, items have a yellow background on the left hand side (fig. 5). If the user has specified an invalid value for an option, its name has a red background on the left hand side. If multiple copies of an item can be included in the metadata file, the item has a blue "plus" symbol to the left of its name. Clicking on the blue "plus" symbol or selecting it and clicking the Duplicate button will add a new, empty copy of the item.

Metadata files can be opened or saved with the Open and Save options on the File menu. If a metadata file that includes items that are not part of the FGDC standard is opened, the values for those items will be editable as text in FgdcMetaEditor. However, FgdcMetaEditor does not allow the user to add items to the metadata file that are not part of the FGDC standard and are not already in the metadata file.

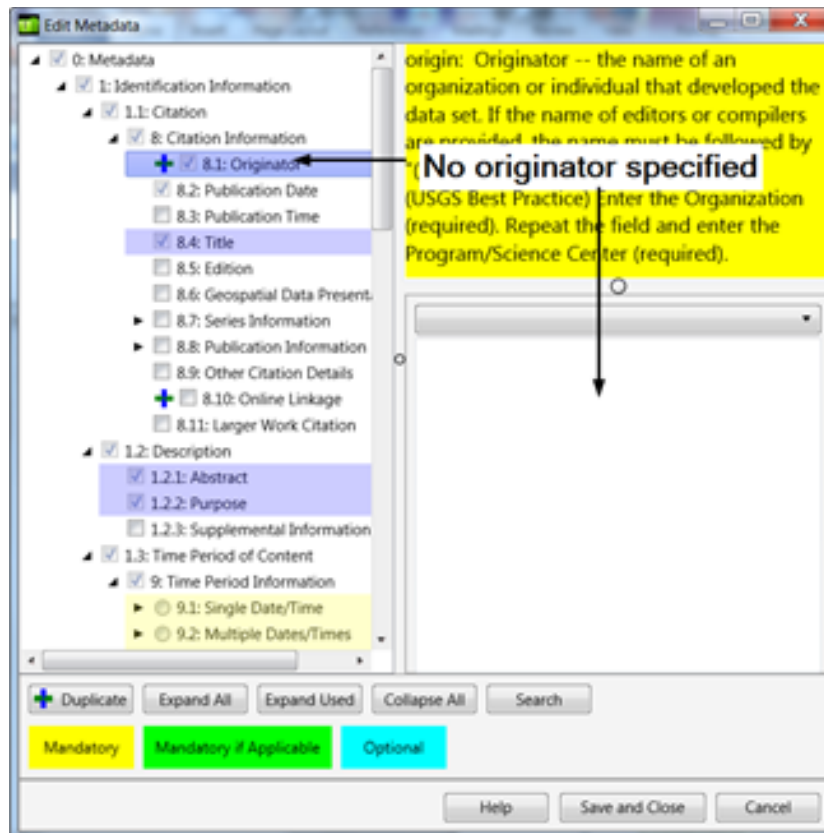


Figure 5. Screenshot showing FgdcMetaEditor, using color to indicate missing data.

Examples of Modifications in .Archive Files

As described previously, files with the extension .archive will have this extension stripped off when the file is copied to the archive. This allows the user to have one version of such files that is used when originally running the model and another version that is used only in the archive. Examples of the sort of modifications that might be used are presented below to guide users who wish to create such files manually. The modifications shown here ensure that (1) the model is run from the bin directory, (2) the output directory exists, and (3) the output files are created in or moved to the output directory.

RunSutra.Bat File

In the original RunSutra.bat file (fig. 6), SUTRA is run from the default installation location of SUTRA version 2.2. In the modified RunSutra.bat file (fig. 7), the output directory for the model is created if it does not already exist. Then Sutra is run from the bin directory of the archive.

```

1 "C:\SutraSuite\SUTRA_2_2\bin\sutra_2_2.exe"
2 Pause
3

```

Figure 6. Screenshot showing an unmodified RunSutra.bat file.

```

1 if not exist "..\..\output\NUL" mkdir "..\..\output"
2 if not exist "..\..\output\SutraTest1\NUL" mkdir "..\..\output\SutraTest1"
3 ..\..\bin\sutra_2_2.exe
4 Pause
5

```

Figure 7. Screenshot showing RunSutra.bat file modified to create output directory and run the model from the bin directory.

MODFLOW Name File

In the unmodified MODFLOW name file (fig. 8), all the input and output files are in the same directory. In the modified name file (fig. 9), the input files are in a subdirectory of the model directory, and the output files are created in a subdirectory of the output directory.

```

1 LIST 11 ImportedP1_Casela.lst REPLACE
2 DATA (BINARY) 9 ImportedP1_Casela.cbc REPLACE
3 DIS 12 ImportedP1_Casela.dis OLD
4 BAS6 13 ImportedP1_Casela.bas OLD
5 OC 39 ImportedP1_Casela.oc OLD
6 DATA 37 ImportedP1_Casela.fhd REPLACE
7 DATA 38 ImportedP1_Casela.fdn REPLACE
8 PCG 18 ImportedP1_Casela.pcg OLD
9 BCF6 64 ImportedP1_Casela.bcf OLD
10 LMT6 135 ImportedP1_Casela.lmt OLD
11 DATA (BINARY) 136 ImportedP1_Casela.ftl REPLACE
12 CHD 17 ImportedP1_Casela.chd OLD
13

```

Figure 8. Screenshot showing an unmodified MODFLOW name file.

```

1 LIST 11 ..\..\output\ImportedP1_Casela\ImportedP1_Casela.lst REPLACE
2 DATA (BINARY) 9 ..\..\output\ImportedP1_Casela\ImportedP1_Casela.cbc REPLACE
3 DIS 12 ..\..\model\ImportedP1_Casela\ImportedP1_Casela.dis OLD
4 BAS6 13 ..\..\model\ImportedP1_Casela\ImportedP1_Casela.bas OLD
5 OC 39 ..\..\model\ImportedP1_Casela\ImportedP1_Casela.oc OLD
6 DATA 37 ..\..\output\ImportedP1_Casela\ImportedP1_Casela.fhd REPLACE
7 DATA 38 ..\..\output\ImportedP1_Casela\ImportedP1_Casela.fdn REPLACE
8 PCG 18 ..\..\model\ImportedP1_Casela\ImportedP1_Casela.pcg OLD
9 BCF6 64 ..\..\model\ImportedP1_Casela\ImportedP1_Casela.bcf OLD
10 LMT6 135 ..\..\model\ImportedP1_Casela\ImportedP1_Casela.lmt OLD
11 DATA (BINARY) 136 ..\..\output\ImportedP1_Casela\ImportedP1_Casela.ftl REPLACE
12 CHD 17 ..\..\model\ImportedP1_Casela\ImportedP1_Casela.chd OLD
13

```

Figure 9. Screenshot showing a MODFLOW name file modified to generate output files in a directory separate from the input files.

MODPATH Name File

In the original MODPATH name file (fig. 10), all the input and output files are in the same directory. In the modified MODPATH name file (fig. 11), the DIS file is an input file for the MODFLOW model, so it is read from a directory different from that of the other MODPATH input files. In addition, some of the other input files, such as the BUDGET file are output files of a MODFLOW model and are read from the output subdirectory of that MODFLOW model. Finally, the MODPATH output files are directed to the appropriate output subdirectory.

```

1 LIST 11 MP6_Test1.mplst
2 MAIN 12 MP6_Test1.mpm
3 DIS 13 MP6_Test1.dis
4 CBF 14 MP6_Test1.cbf
5 ENDPOINT 15 MP6_Test1.end
6 LOCATIONS 19 MP6_Test1.strt
7 BUDGET 20 MP6_Test1.cbc
8 HEAD(BINARY) 22 MP6_Test1.bhd
9 DRAWDOWN 23 MP6_Test1.fdn
10

```

Figure 10. Screenshot showing an unmodified MODPATH name file.

```

1 LIST 11 ..\..\output\MP6_Test1_Modpath\MP6_Test1.mplst
2 MAIN 12 MP6_Test1.mpm
3 DIS 13 ..\..\model\MP6_Test1\MP6_Test1.dis
4 CBF 14 ..\..\output\MP6_Test1_Modpath\MP6_Test1.cbf
5 ENDPOINT 15 ..\..\output\MP6_Test1_Modpath\MP6_Test1.end
6 LOCATIONS 19 MP6_Test1.strt
7 BUDGET 20 ..\..\output\MP6_Test1\MP6_Test1.cbc
8 HEAD(BINARY) 22 ..\..\output\MP6_Test1\MP6_Test1.bhd
9 DRAWDOWN 23 ..\..\output\MP6_Test1\MP6_Test1.fdn

```

Figure 11. Screenshot showing a MODPATH name file with directory paths modified.

Modifications to ModelMuse to Support ModelArchiver

A number of modifications have been made to ModelMuse (Winston, 2009, 2014) to better support ModelArchiver. First, ModelMuse, version 3.9, or later creates archive list files (fig. 12) that can be used by ModelArchiver. Archive list files are created automatically whenever ModelMuse creates the input files for a model. Archive list files also can be created by clicking the Save archive list button in the Files|Files to Archive dialog box. This dialog box can be used to add or remove files from an archive list. The user can delete files from the archive list by selecting them in the dialog box, right clicking, and selecting Delete from the pop-up menu. Files can be added by selecting a category in the Files|Files to Archive dialog box, right clicking, selecting Add Files from the pop-up menu, and then selecting the files.

Another modification is that when exporting the model input files, ModelMuse creates files with the .archive extension for use in the model archive. Modified versions of the following file types are created in this way:

- MODFLOW name files,
- MODFLOW-LGR LGR files,
- MODPATH name files,
- MODPATH response files,
- MODPATH simulation files,
- ZONEBUDGET response files,
- MT3DMS and MT3D-USGS name files,
- SUTRA.FIL files, and
- Batch files used to run models.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <Model_Archive_List>
3  <bin Model_Name="MP6 Test1" FileType="bin">
4    <FileName Model_Name="MP6 Test1" FileType="bin">C:\WRDAPP\MP2005.1_11\bin\mf2005.exe</FileName>
5    <FileName Model_Name="MP6 Test1" FileType="bin">C:\WRDAPP\modpath.6_0\bin\mp6.exe</FileName>
6  </bin>
7  <ancillary Model_Name="MP6 Test1" FileType="ancillary">
8    <FileName Model_Name="MP6 Test1" FileType="ancillary">MP6 Test1.gpt</FileName>
9  </ancillary>
10 <model Model_Name="MP6 Test1" FileType="Model Input">
11   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.bas</FileName>
12   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.dis</FileName>
13   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.gsfc</FileName>
14   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.lpf</FileName>
15   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.nam.archive</FileName>
16   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.oc</FileName>
17   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.pcg</FileName>
18   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.rch</FileName>
19   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.riv</FileName>
20   <FileName Model_Name="MP6 Test1" FileType="Model Input">MP6_Test1.wel</FileName>
21   <FileName Model_Name="MP6 Test1" FileType="Model Input">RunModflow.Bat.archive</FileName>
22   <FileName Model_Name="MP6 Test1" FileType="Model Input">usgs.MP6_Test1.reference</FileName>
23 </model>
24 <output Model_Name="MP6 Test1" FileType="Model Output">
25   <FileName Model_Name="MP6 Test1" FileType="Model Output">MP6_Test1.bhd</FileName>
26   <FileName Model_Name="MP6 Test1" FileType="Model Output">MP6_Test1.cbc</FileName>
27   <FileName Model_Name="MP6 Test1" FileType="Model Output">MP6_Test1.fdn</FileName>
28   <FileName Model_Name="MP6 Test1" FileType="Model Output">MP6_Test1.lst</FileName>
29 </output>
30 <Modpath_Input_Files Model_Name="MP6 Test1" FileType="Modpath Input">
31   <FileName Model_Name="MP6 Test1" FileType="Modpath Input">Mp.bat.archive</FileName>
32   <FileName Model_Name="MP6 Test1" FileType="Modpath Input">MP6_Test1.mpbas</FileName>
33   <FileName Model_Name="MP6 Test1" FileType="Modpath Input">MP6_Test1.mpn.archive</FileName>
34   <FileName Model_Name="MP6 Test1" FileType="Modpath Input">MP6_Test1.mpsim.archive</FileName>
35   <FileName Model_Name="MP6 Test1" FileType="Modpath Input">MP6_Test1.strt</FileName>
36   <FileName Model_Name="MP6 Test1" FileType="Modpath Input">RunModpath.Bat.archive</FileName>
37   <FileName Model_Name="MP6 Test1" FileType="Modpath Input">usgs.MP6_Test1_Modpath.reference</FileName>
38 </Modpath_Input_Files>
39 <Modpath_Output_Files Model_Name="MP6 Test1" FileType="Modpath Output">
40   <FileName Model_Name="MP6 Test1" FileType="Modpath Output">MP6_Test1.end</FileName>
41   <FileName Model_Name="MP6 Test1" FileType="Modpath Output">MP6_Test1.mplst</FileName>
42   <FileName Model_Name="MP6 Test1" FileType="Modpath Output">MPATH6.LOG</FileName>
43 </Modpath_Output_Files>
44 </Model_Archive_List>

```

Figure 12. Screenshot showing an example Archive List file

Finally, ModelMuse can create a shapefile that outlines the active and inactive areas of the model. To create it, select File|Export|Shapefile|Model Outline to Shapefile. The shapefile will not include projection or metadata information. These can be created in a geographic information system program, if desired.

Summary

ModelArchiver is a tool for creating archives of groundwater models in the structure mandated by the U.S. Geological Survey (USGS) for groundwater models. The program version 1.0 leads the user step-by-step through the process of creating the archive, whereby it can ensure the completeness of the archive. It also can automatically generate portions of the readme.txt and metadata files on the basis of the content of the archive. The program allows the user to edit the metadata for the archive and can help the user to ensure that the metadata are complete. It provides links to other tools related to model archive creation, such as the USGS Digital Object Identifier Creation Tool. Archive lists generated by ModelMuse and potentially other programs can be used with ModelArchiver to identify many of the files that it would be appropriate to include in the archive.

ModelArchiver is able to use Archive Lists to identify files to include in model archives. The structure of Archive Lists is described in the appendix. ModelMuse (Winston, 2009, 2014) can create Archive Lists.

FgdcMetaEditor is a program for creating metadata for geographic data using the format described by the Federal Geographic Data Committee (1998, 2000). Its version 1.0 functionality has been incorporated into the built-in metadata editor in ModelArchiver. FgdcMetaEditor displays the structure of the metadata in a tree control allowing the user to easily move to whichever section of the metadata is desired. It uses color in the tree control to highlight missing data or other problems in the metadata. Each item in the metadata has an associated description describing what it should contain. The background color of each description is used to indicate whether the item is always required, required if applicable, or optional.

Acknowledgments

I would like to thank Paul Misut and Glen Carleton, both of the U.S. Geological Survey, for testing and providing feedback on earlier versions of ModelArchiver and for their reviews of this manuscript.

References Cited

- Ahlfeld, D.P., Barlow, P.M., and Mulligan, A.E., 2005, GWM—A ground-water management process for the U.S. Geological Survey modular ground-water model (MODFLOW-2000): U.S. Geological Survey Open-File Report 2005-1072, 124 p.
- Federal Geographic Data Committee, 1998, FGDC-STD-001-1998, Content standard for digital geospatial metadata (revised June 1998): Washington, D.C., Federal Geographic Data Committee, accessed October 16, 2017, at https://www.fgdc.gov/standards/projects/metadata/base-metadata/v2_0698.pdf.
- Federal Geographic Data Committee, 2000, Content standard for digital geospatial metadata workbook, Version 2.0: Washington, D.C., Federal Geographic Data Committee, accessed October 16, 2017, at https://www.fgdc.gov/metadata/documents/workbook_0501_bmk.pdf.
- Harbaugh, A.W., 1990, A computer program for calculating subregional water budgets using results from the U.S. Geological Survey modular three-dimensional ground-water flow model: U.S. Geological Survey Open-File Report 90-392, 46 p., <https://water.usgs.gov/nrp/gwsoftware/zonebud3/zonebudget3.html>.
- Pollock, D.W., 2012, User guide for MODPATH version 6—A particle-tracking model for MODFLOW: U.S. Geological Survey Techniques and Methods, book 6, chap. A41, 58 p., <https://pubs.er.usgs.gov/publication/tm6A41>.
- Voss, C.I., and Provost, A.M., 2002 (Version of September 22, 2010), SUTRA, A model for saturated-unsaturated variable-density ground-water flow with solute or energy transport: U.S. Geological Survey Water-Resources Investigations Report 02-4231, 291 p., <https://pubs.er.usgs.gov/publication/wri024231>.
- Winston, R.B., 2009, ModelMuse—A graphical user interface for MODFLOW-2005 and PHAST: U.S. Geological Survey Techniques and Methods, book 6, chap. A29, 52 p., <https://pubs.usgs.gov/tm/tm6A29/>.
- Winston, R.B., 2014, Modifications made to ModelMuse to add support for the Saturated-Unsaturated Transport model (SUTRA): U.S. Geological Survey Techniques and Methods, book 6, chap. A49, 6 p., <https://doi.org/10.3133/tm6A49>.
- Winston, R.B., 2018, Software release—ModelArchiver and FgdcMetaEditor: U.S. Geological Survey software release, <https://doi.org/10.5066/F73X85M1>.

Appendix 1. Structure of Archive List Files and ModelMuse Integration

Archive lists are XML (Extensible Markup Language) files that specify the source locations of files to be included in the archive (see fig. 12). The archive file identifies input files, output files, ancillary files, and model program files. ModelMuse 3.9 or later (Winston, 2009, 2014) creates an Archive List file with the extension .axml each time it generates input files for a model.

XML files have a single root node, which in this case is Model_Archive_List (see fig. 12). Below it in the structure are several additional nodes: bin for programs, ancillary for ancillary data, model for model input files and a georeference file, output for model output files, Modpath_Input_Files for input files for MODPATH, and Modpath_Output_Files for output files from MODPATH. These nodes each have child nodes that identify individual files. Except for the root node, each node has two attributes: Model_Name and FileType. For example, for the file MP6_Test1.bas, the Model_Name is MP6 Test1 and the FileType is Model Input. For model input and output files, the FileType must always end with Input or Output, respectively. In this example, the archive list relates to two separate models, a MODFLOW groundwater flow model and a MODPATH particle tracking model. For groundwater flow models, such as MODFLOW and SUTRA (Voss and Provost, 2002), the FileType will be Model Input or Model Output. For other types of models, the FileType will indicate the type of model in the FileType before Input or Output. For models other than groundwater flow models, ModelArchiver will use whatever text precedes Input or Output and add it to the value in Model_Name to determine the name of the folder in the model archive. For example, the folder for the MODPATH input files would be model.MP6_Test1_Modpath.

In the example Archive List file (see fig. 12), some of the files have the extension .archive added. These .archive extensions will be stripped off the file names when the archive is created. The purpose of the .archive extension is to allow two different versions of a file to be created. One will be used when the model is originally run, and the other will be used when the model is run from the archive where the file structure may be different from the file structure used when the models were originally created. For example, ModelMuse created two different versions of the file RunModflow.bat. One version lacks the .archive extension (fig. 1–1), and one has it (fig. 1–2).

```

1 call D:\Colab\GWModelTools\ModelMuse\Release\Win64\ModelMonitor.exe -m
  C:\WRDAPP\MF2005.1_11\bin\mf2005.exe -n D:\Colab\GWModelTools\ModelMuse\MP6_Test1.nam
2 Start Notepad.exe D:\Colab\GWModelTools\ModelMuse\MP6_Test1.lst
3 call RunModpath.bat
4 pause
5

```

Figure 1–1. Screenshot showing an unmodified RunModflow.bat file.

```

1 if not exist "..\..\output\NUL" mkdir "..\..\output"
2 if not exist "..\..\output\output.MP6_Test1\NUL" mkdir "..\..\output\output.MP6_Test1"
3 ..\..\bin\mf2005.exe MP6_Test1.nam /wait
4 pause
5

```

Figure 1–2. Screenshot showing a RunModflow.bat.archive file.

In RunModflow.bat, the program ModelMonitor runs MODFLOW rather than MODFLOW being run in a command-line window. When ModelMonitor is closed, the batch files open the listing file with Notepad and then uses a batch file to run MODPATH.

In RunModflow.bat.archive, the batch file first checks that the proper output directory exists, and if it does not, the batch file creates it. Then it runs MODFLOW in a command-line window. It does not run MODPATH.

Other files that use the .archive extension in this example are MP6_Test1.nam.archive, Mp.bat.archive, MP6_Test1.mpn.archive, MP6_Test1.mpsim.archive, and RunModpath.Bat.archive. These files are all slightly different from the corresponding versions without the .archive extension so that the model can run properly from within the archive.

References Cited in Appendix 1

- Voss, C.I., and Provost, A.M., 2002 (Version of September 22, 2010), SUTRA, A model for saturated-unsaturated variable-density ground-water flow with solute or energy transport: U.S. Geological Survey Water-Resources Investigations Report 02-4231, 291 p., <https://pubs.er.usgs.gov/publication/wri024231>.
- Winston, R.B., 2009, ModelMuse—A graphical user interface for MODFLOW-2005 and PHAST: U.S. Geological Survey Techniques and Methods, book 6, chap. A29, 52 p., <https://pubs.usgs.gov/tm/tm6A29/>.
- Winston, R.B., 2014, Modifications made to ModelMuse to add support for the Saturated-Unsaturated Transport model (SUTRA): U.S. Geological Survey Techniques and Methods, book 6, chap. A49, 6 p., <https://doi.org/10.3133/tm6A49>.

Manuscript was approved on November 13, 2017.

Published by the USGS Science Publishing Network,
West Trenton and Reston Publishing Service Centers

