

# Modflow User Tools (MUT) Version 2025.1

## User's Guide



October 2024

Rob McLaren, Young-jin Park, Sorab Panday

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Conventions Used in this Manual . . . . .	5
<b>2</b>	<b>Software Installation and Useage</b>	<b>8</b>
2.1	Model End User . . . . .	8
2.2	Model Developer . . . . .	13
<b>3</b>	<b>Model Build</b>	<b>17</b>
3.1	Defining the Units of Length and Time . . . . .	21
3.2	Defining the Template Mesh . . . . .	22
3.3	Groundwater Flow(GWF) Domain . . . . .	30
3.3.1	Generating a Layered GWF Domain . . . . .	30
3.3.1.1	Defining the Top Elevation . . . . .	31
3.3.1.2	Adding Layers . . . . .	35
3.3.1.3	Cell Connection Properties . . . . .	37
3.3.1.4	Material Properties . . . . .	39
3.3.1.5	Material Zones . . . . .	45
3.3.1.6	Initial Conditions . . . . .	50
3.3.1.7	Boundary Conditions . . . . .	54
3.4	Surface Water Flow(SWF) Domain . . . . .	56
3.4.1	Generating a SWF Domain . . . . .	56
3.4.1.1	Cell Connection Properties . . . . .	56
3.4.1.2	Material Properties . . . . .	57
3.4.1.3	Initial Conditions . . . . .	59
3.4.1.4	Boundary Conditions . . . . .	60

3.5	Connected Linear Network(CLN) Domain . . . . .	62
3.5.1	Generating a CLN Domain . . . . .	63
3.5.1.1	Cell Connection Properties . . . . .	65
3.5.1.2	Material Properties . . . . .	65
3.5.1.3	Initial Conditions . . . . .	66
3.5.1.4	Boundary Conditions . . . . .	67
3.6	Inactive MODFLOW-USG <sup><i>Swf</i></sup> Cells . . . . .	67
3.7	Observation Points . . . . .	68
3.8	Stress Periods . . . . .	69
3.9	Output Control . . . . .	72
3.10	Solver Parameters . . . . .	74
3.11	3D Model Build Visualization . . . . .	76
3.11.1	GWF Domain . . . . .	78
3.11.2	SWF Domain . . . . .	85
3.11.3	CLN Domain . . . . .	92
<b>4</b>	<b>Model Simulation and Post-Processing</b> . . . . .	<b>95</b>
4.1	MODFLOW-USG <sup><i>Swf</i></sup> Simulation . . . . .	95
4.2	MUT Post-processing . . . . .	96
4.3	Volumetric Water Budget Plots . . . . .	97
4.4	3D Visualization of Model Results . . . . .	100
4.4.1	Solution Times and Animation . . . . .	100
4.4.2	Data Set Infomation . . . . .	101
4.4.3	Inactive Cells and Value Blanking . . . . .	102
4.4.4	Slices and Fence Diagrams . . . . .	103
4.4.5	Defining New Variables . . . . .	104
4.4.6	Water Table Isosurface Plot . . . . .	106
4.4.7	Infiltration Plot . . . . .	107
<b>5</b>	<b>Demonstration Models</b> . . . . .	<b>110</b>
5.1	1D Variably-saturated Flow in a Column . . . . .	110
5.2	1D Surface Flow . . . . .	113

5.3	2D Surface/subsurface Flow . . . . .	116
5.4	3D Surface/subsurface Flow: Field Study of Abdul . . . . .	119
<b>6</b>	<b>References</b>	<b>122</b>
<b>A</b>	<b>QGIS Useage</b>	<b>123</b>
A.1	QGIS Set-up . . . . .	123
A.2	Setting the Coordinate Reference System (CRS) . . . . .	125
A.3	Layers . . . . .	127
A.3.1	Layer Properties . . . . .	127
A.3.2	Layer Appearance (symbology) . . . . .	128
A.3.3	Layer Clipping . . . . .	129
A.3.4	Vector Layers . . . . .	130
A.3.4.1	Loading Vector Layers . . . . .	130
A.3.4.2	Selecting Features . . . . .	131
A.3.4.3	Extract Vertices . . . . .	132
A.3.4.4	Add Geometry Attributes . . . . .	134
A.3.4.5	Georeference an Image . . . . .	137
A.3.4.6	Add a new shapefile . . . . .	140
A.3.4.7	Digitize a new shapefile . . . . .	142
A.3.4.8	Point Sampling From a Raster Layer . . . . .	143
A.3.4.9	Export to a CSV File . . . . .	145
A.3.4.10	Import from a CSV File . . . . .	147
A.3.5	Raster Layers . . . . .	150
A.3.5.1	Loading Raster Layers . . . . .	150
A.3.5.2	Smoothing a Raster Layer . . . . .	151
<b>B</b>	<b>Microsoft Excel Database Files</b>	<b>154</b>
<b>C</b>	<b>Microsoft Excel Modifications</b>	<b>159</b>

# Chapter 1

## Introduction

This document describes a new MODFLOW-USG<sup>1</sup> development environment which has these features:

- We refer to it as Modflow User Tools, or MUT for short.
- MUT is designed to work with a modified version of MODFLOW-USG, where a new surface water flow package, called SWF, has been added. Like the Connected Linear Network (CLN) package, the SWF package represents a new domain type that is fully-coupled to the 3D groundwater flow (GWF) domain. There can also be cell-to-cell flows between the SWF and CLN domains. The SWF domain uses the diffusion-wave approach to simulate 2D surface-water flow. We will refer to this new version of MODFLOW-USG as MODFLOW-USG<sup>Swf</sup> in this manual.
- We currently develop and run it on a MICROSOFT WINDOWS 10-based computing platform, writing the software using the INTEL FORTRAN compiler running inside the MICROSOFT VISUAL STUDIO interactive development environment, which includes software version control tools through GITHUB.
- A text-based approach is used for the MUT interface, in which we first develop an input file of instructions that define our MODFLOW-USG<sup>Swf</sup> project, then run MUT to read it and write a complete MODFLOW-USG<sup>Swf</sup> data set. MUT also writes output files for TECPLOT, a third-party visualization software package, which provides a 3D graphical visualization tool to review the model numerical mesh and material properties in the data set. In future, MUT could be extended to support other third-party visualization packages, for example the open source program Paraview.
- We will introduce the use of QGIS<sup>2</sup> software to view and process project data consisting of physiography (e.g. DEM's, often provided as raster-formatted TIF files) and model boundaries (e.g. often provide as shapefiles).
- MUT can post-process a MODFLOW-USG<sup>Swf</sup> simulation to provide a TECPLOT visualization of temporal model results, including hydraulic heads, saturations, water depths and flow budget data. *If applied to output files which were produced by an earlier version of Modflow, results may*

---

<sup>1</sup><https://www.gsienv.com/software/modflow-usg/modflow-usg/>

<sup>2</sup>QGIS is a geographic information system (GIS) software that is free and open-source. It supports viewing, editing, printing, and analysis of geospatial data in a range of data formats.

be mixed. It is not our intent here to support all existing Modflow packages, many of which have been superceded.

Changes made to MUT since version 1.34 are listed in Appendix C

## 1.1 Conventions Used in this Manual

A small-caps font is used when referring to software names e.g.: MUT, MODFLOW-USG<sup>Swf</sup>, TECPLOT and MICROSOFT WINDOWS.

A sans-serif font is used when referring to model domain types (e.g. GWF for the groundwater flow domain), menu and dialogue box options (e.g. the TECPLOT option Specify equations) and variables presented during on-screen visualization (e.g. MUT variables GWF Head and GWF z Cell in TECPLOT).

A typewriter font is used when referring to typed input, file contents and names and path names e.g.: `ctrl-C`, `c:\MUT\MUT_Examples`.

Blue-highlighted items are active hyper-links. If you left-click on these you can navigate to other relevant parts of the manual. These include:

- Table of contents entries.
- Page and section numbers in the text and index.
- Active links to internet web sites (i.e. URL's) e.g.: <https://www.gsienv.com/software/modflow-usg/modflow-usg/>. If you left-click on this it will open a browser window at the site.
- Footnote numbers in the text.

Your PDF reader should have a way to quickly jump back through previously viewed pages e.g.: the `backspace` key (SUMATRA PDF Reader) or the `alt-left arrow` key combination (ADOBE ACROBAT READER).

This manual uses the following formatting conventions when introducing new MUT instructions. Documentation for an instruction begins with a bold upper corner line:

- 
- The instruction name is presented in a large sans-serif font e.g.:  
**read some data**
  - Data inputs, if required, are presented in a numbered list consisting of a variable name, dimensions (unless dimensionless) and description e.g.:
    1. **A<sub>real</sub>[L], B<sub>int</sub>, C<sub>str</sub>** This input line requires a real number A<sub>real</sub> with dimensions of length [L], an integer number B<sub>int</sub> and a text string C<sub>str</sub>.

Variable names are rendered in an underlined bold font, followed by one of the following data type requirement subscripts:

real A real number, e.g. "24.32" or "1.5E-08".

int An integer number, e.g. "25".

str A text string, e.g. "centimeters".

Variable dimension indicators are  $L$  for length and  $T$  for time (future implementations of MUT that support mass transport will use  $M$  for mass). Some examples of variables with more complex dimensionality are:

**Hydraulic conductivity** [ $L T^{-1}$ ], length over time.

**Manning's coefficient of friction** [ $T L^{-1/3}$ ], time over the cube root of length.

**Concentration** [ $M L^{-3}$ ], mass per unit volume.

- General notes about instruction usage are presented if necessary, e.g.:

MODFLOW-USG<sup>Swf</sup> currently supports units of feet, meters or centimeters.

- Some instructions are subtasks, which are procedures that process a local set of instructions until an **end** instruction is encountered. Documentation for a subtask suggests appending the subtask name to the **end** instruction, e.g.:

**end read some data**

This makes the input file more readable, and debugging easier if subtasks are nested.

Documentation for an instruction ends with a bold lower corner line:



Here is an example of an instruction which requires two lines of input, with each line requiring two input variables:



## generate uniform rectangles

1. **LengthX**<sub>real</sub>[ $L$ ], **nRectX**<sub>int</sub> Domain length and number of rectangles in the  $x$ -direction
2. **LengthY**<sub>real</sub>[ $L$ ], **nRectY**<sub>int</sub> Domain length and number of rectangles in the  $y$ -direction



In this case the first line of input consists of the real variable **LengthX**<sub>real</sub> of dimension length [ $L$ ], and a second integer variable **nRectX**<sub>int</sub> which is dimensionless. A second line of input requires similar variables but for the  $y$ -direction.

Here is an example showing possible input file contents for this instruction:

```
generate uniform rectangles
1000.0, 1000
1.0, 1
```

This input would generate a strip of 1000 rectangular elements with a total length of 1000 length units in the  $x$ -direction and 1 element with a width of 1 length unit in the  $y$ -direction.

This rest of this document is subdivided into these sections:

**Chapter 2 Software Installation and Useage:** How to install MUT, MODFLOW-USG<sup>Swf</sup> and TEC-PLOT and define MICROSOFT WINDOWS environment variables.

**Chapter 3 Model Build** How to build a MUT input file, produce a MODFLOW-USG<sup>Swf</sup> compatible data set and TECPLOT compatible output files with MUT, then review the results of the model build with TECPLOT.

**Chapter 4 Model Simulation and Post-Processing** How to run MODFLOW-USG<sup>Swf</sup>, convert the output to TECPLOT-compatible files with MUT, then visualize them with TECPLOT.

**Chapter 5 Demonstration Models** These models verify the accuracy of MODFLOW-USG<sup>Swf</sup> models built using MUT and demonstrate it's ability to capture a range of conceptual situations, from simple to complex.

**Appendix B Microsoft Excel Database Files** Details about using the provided MICROSOFT EXCEL database files, which are currently used to store MODFLOW-USG<sup>Swf</sup> model material property and solver parameter data sets.

# Chapter 2

## Software Installation and Useage

This chapter is subdivided into the following two sections:

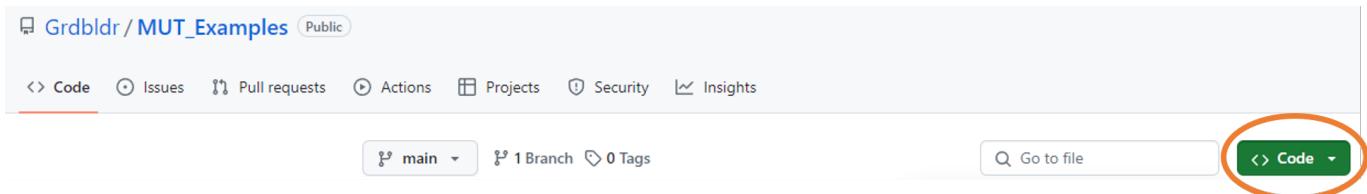
**Model End User** If you are only interested in downloading and running the MUT and MODFLOW-USG<sup>Swf</sup> executables you can skip the second section of this chapter.

**Model Developer** If you want to view and possibly modify and re-compile the source code for MUT and MODFLOW-USG<sup>Swf</sup> you should complete both sections of this chapter.

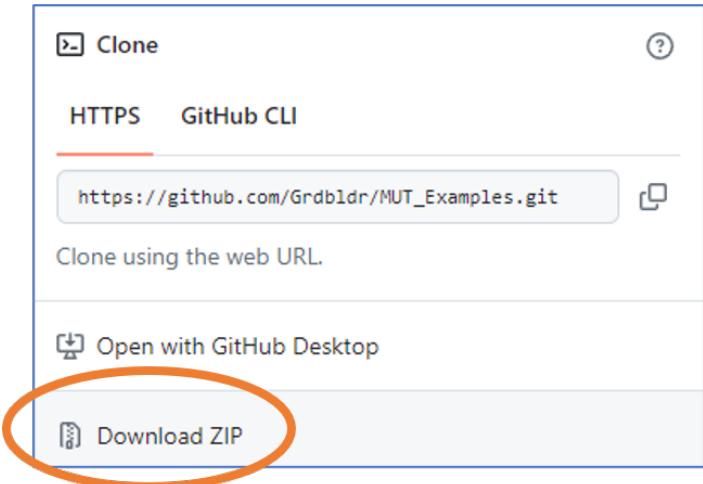
### 2.1 Model End User

The first step in the software installation process is to obtain the MUT examples, executables and database files from GITHUB. To do this:

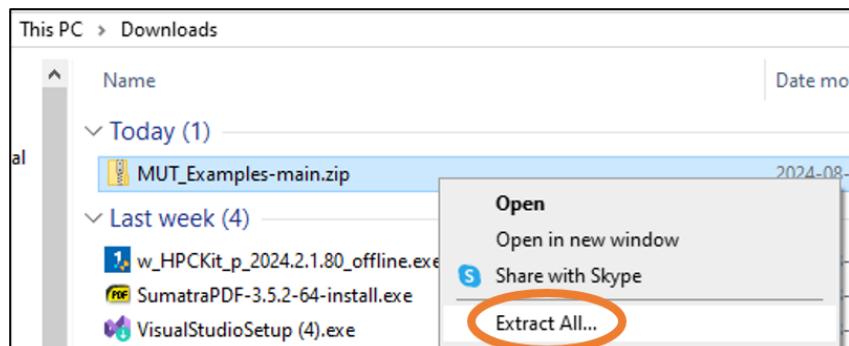
- Click on this link, [https://github.com/Grdbldr/MUT\\_Examples.git](https://github.com/Grdbldr/MUT_Examples.git), which will take you the MUT\_Examples GITHUB page.
- Click on the green Code button.



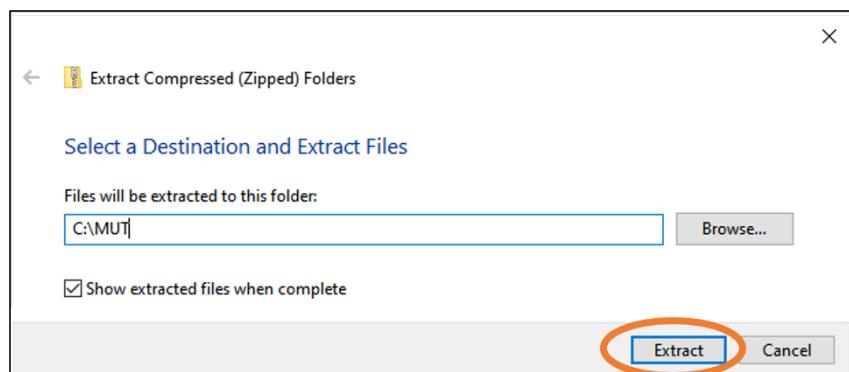
- Choose Download ZIP from the drop-down menu.



Once the download is complete, the zip file can be found in the MICROSOFT WINDOWS Downloads folder. The contents need to be extracted to a local directory by right-clicking on the download file MUT-Examples-Main.zip and choosing Extract All... from the drop-down menu:



This opens the Extract Compressed(Zipped) Folders dialogue, where you are free to choose a different drive and folder to store the extracted files. Here we changed the destination folder to C:\MUT. Left-click the Extract button:



The extracted contents can now be found in the specified destination folder:

	Name	Date modified	Type
	.vs	2024-11-18 1:35 PM	File folder
	_MUT_USERBIN	2024-11-18 1:36 PM	File folder
	1_VSF_Column	2024-11-18 1:35 PM	File folder
	1_VSF_Column_Brooks	2024-11-18 1:35 PM	File folder
	2_VSF_Hillslope	2024-11-18 1:35 PM	File folder
	3_0_SWF_CHD	2024-11-18 1:35 PM	File folder
	3_1_CLN_for_SWF	2024-11-18 1:35 PM	File folder
	3_SWF	2024-11-18 1:35 PM	File folder
	4_SWF_RCH_CRD	2024-11-18 1:35 PM	File folder
	6_Abdul_MODHMS	2024-11-18 1:36 PM	File folder
	6_Abdul_Prism_Cell	2024-11-18 1:36 PM	File folder
	6_Abdul_Prism_Cell_nc	2024-11-18 1:36 PM	File folder

This folder contains a subfolder called \_MUT\_USERBIN, which contains the following files:

	Name	Date modified	Type	Size
	mut.exe	2024-11-18 1:36 PM	Application	6,785 KB
	USGS_1.exe	2024-11-18 1:36 PM	Application	4,002 KB
	tecio.dll	2024-11-18 1:36 PM	Application exten...	1,732 KB
	CLN.csv	2024-11-18 1:36 PM	Microsoft Excel C...	1 KB
	ET.csv	2024-11-18 1:36 PM	Microsoft Excel C...	1 KB
	GWF.csv	2024-11-18 1:36 PM	Microsoft Excel C...	3 KB
	LAI.csv	2024-11-18 1:36 PM	Microsoft Excel C...	1 KB
	LAI_default.csv	2024-11-18 1:36 PM	Microsoft Excel C...	1 KB
	SMS.csv	2024-11-18 1:36 PM	Microsoft Excel C...	2 KB
	SWF.csv	2024-11-18 1:36 PM	Microsoft Excel C...	1 KB
	CLN.xlsx	2024-11-18 1:36 PM	Microsoft Excel W...	13 KB
	ET.xlsx	2024-11-18 1:36 PM	Microsoft Excel W...	13 KB
	GWF.xlsx	2024-11-18 1:36 PM	Microsoft Excel W...	14 KB
	SMS.xlsx	2024-11-18 1:36 PM	Microsoft Excel W...	13 KB
	SWF.xlsx	2024-11-18 1:36 PM	Microsoft Excel W...	13 KB

These include the executable and supporting files for MUT and the executable for MODFLOW-USG<sup>Swf</sup>.

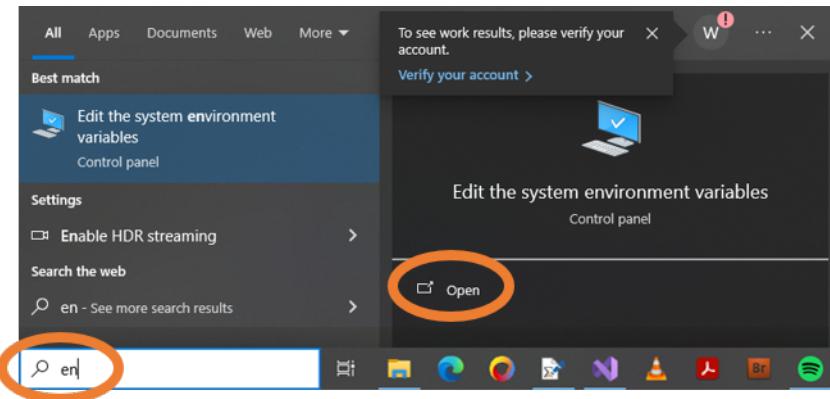
Before you run MUT for the first time, you need to define a windows environment variable called USERBIN, which contains the path to the \_MUT\_USERBIN folder, and modify the existing PATH variable.

First, highlight the path by clicking on it in the File Explorer window, then copy it by pressing CTRL-C or by right-clicking and choosing Copy from the drop-down menu, as shown here:

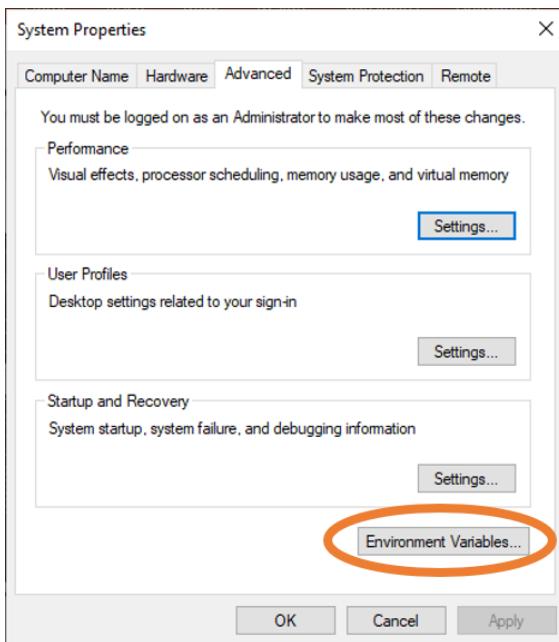


To define the environment variables:

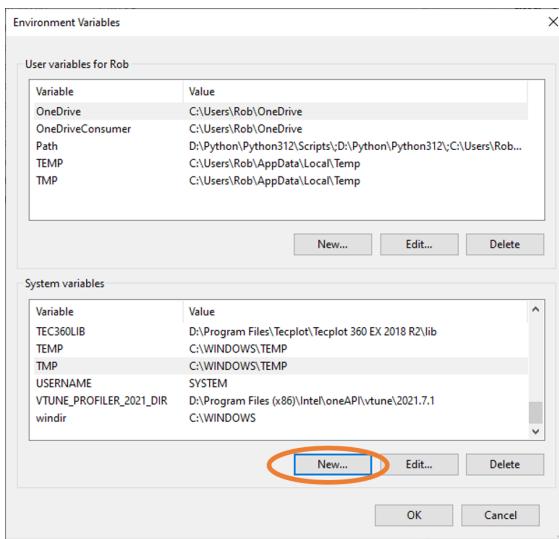
- Type the string en in the windows taskbar search field and open the Edit the system environment variables dialogue:



- Click on the Environment variables... button at the bottom of the dialogue:

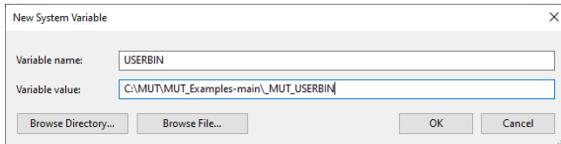


- Click on the New button:

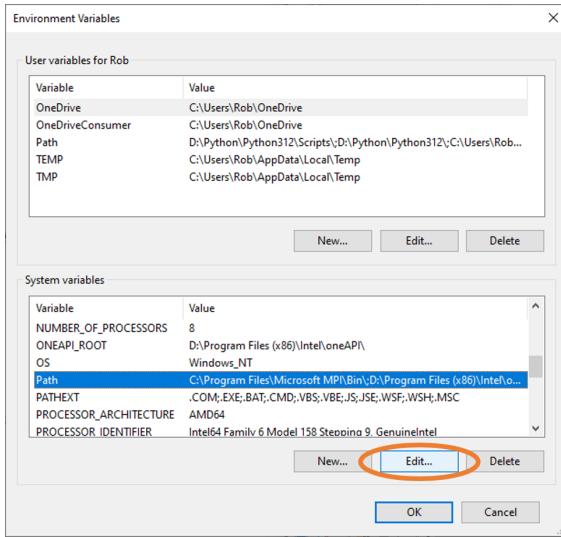


- Add a new variable named **USERBIN** and define the variable value by pasting in the path copied

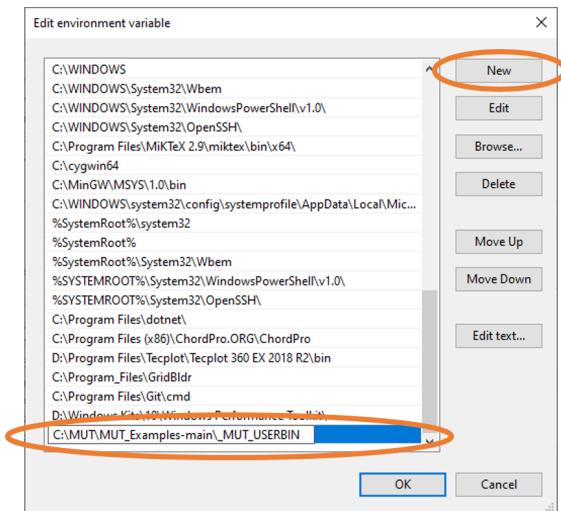
earlier, then click the OK button:



- Add the path to USERBIN to the existing Path variable. First, choose Path, then click the Edit... button:



- Click the New button and paste in the path copied earlier, then click the OK button:



You should now be able to run MUT and MODFLOW-USG<sup>Swf</sup> from the command prompt. To test this, start a new command prompt, then type `mut`, you should see the MUT header:

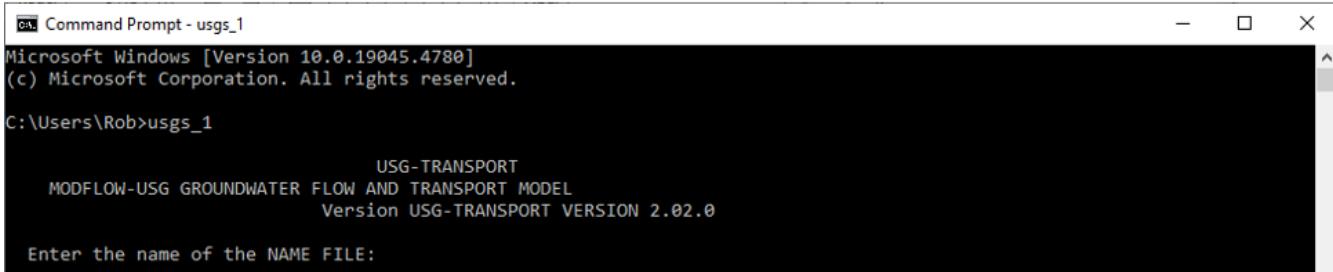


```
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Rob>mut
MUT version 1.25
No command line prefix
No file: _mut.pfx
Checking for default file: a.mut
No file: a.mut
Enter a prefix for a mut file:
```

Type **ctrl-C** to stop the program.

Run MODFLOW-USG<sup>Swf</sup> by typing `usgs_1`. You should see the MODFLOW-USG<sup>Swf</sup> header:



```
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Rob>usgs_1
USG-TRANSPORT
MODFLOW-USG GROUNDWATER FLOW AND TRANSPORT MODEL
Version USG-TRANSPORT VERSION 2.02.0

Enter the name of the NAME FILE:
```

Type **ctrl-C** to stop the program.

If this is not the case, check the definitions of the `USERBIN` and `PATH` variables. If they are correct, you may need to re-boot your computer and try again.

A licensed version of TECPLLOT can be obtained from <https://tecplot.com/products/tecplot-360/>. They provide a free trial for those who want to assess the software before purchase. They also offer educational discounts.

Instructions on how to install QGIS software can be found in Appendix A.1.

Those of you who are just interested in running the MUT and MODFLOW-USG<sup>Swf</sup> programs have completed the required software installation tasks and can proceed to Chapter 3, Model Build.

## 2.2 Model Developer

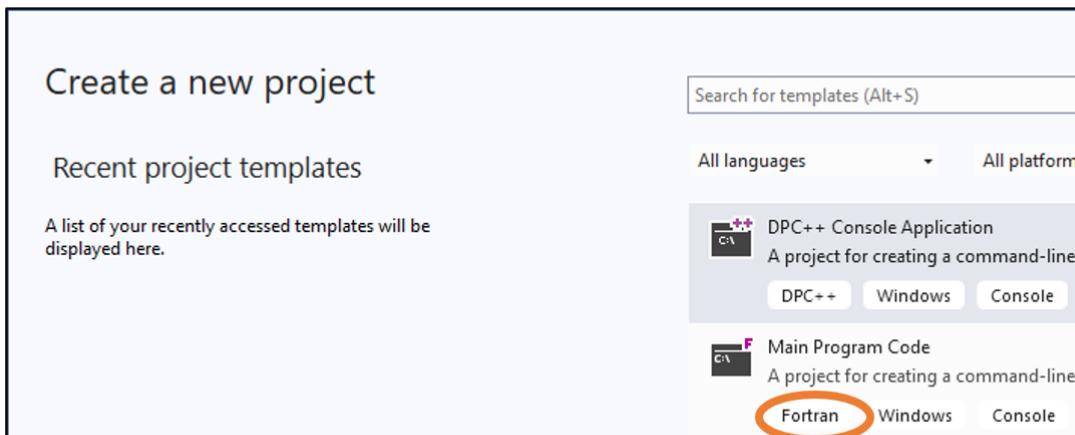
As stated earlier, we use and recommend MICROSOFT VISUAL STUDIO and INTEL FORTRAN. You should install MICROSOFT VISUAL STUDIO before INTEL FORTRAN, which will then be automatically integrated into MICROSOFT VISUAL STUDIO.

A free version of the latest MICROSOFT VISUAL STUDIO (currently 2022) can be obtained from <https://visualstudio.microsoft.com/vs/community/>. Once you are on the site just click the Download button. This will download a file (e.g. `VisualStudioSetup.exe`) which can be run to install MICROSOFT VISUAL STUDIO. If you already have a version of MICROSOFT VISUAL STUDIO, you can choose to keep your old version and add the latest version. When you come to the installation options `Workloads` page, be sure to check the option for `Desktop development with C++`, shown here:

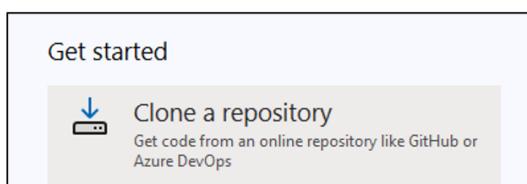


A free version of the latest INTEL FORTRAN compiler can be obtained from <https://www.intel.com/content/www/us/en/developer/tools/oneapi/hpc-toolkit.html>. Once you are on the site just click the Get It Now button to download the Intel® HPC Toolkit, which includes INTEL FORTRAN. Choose the Windows option then the Offline Installer option. Now you can either fill in the required information and start the download or choose to Continue as guest(download starts immediately). This will download a file (e.g. `w_HPCKit_p_2024.2.1.80_offline.exe`) which can be run to install INTEL FORTRAN.

You can check the installation of MICROSOFT VISUAL STUDIO and INTEL FORTRAN by starting MICROSOFT VISUAL STUDIO and choosing Create a new project. The window that appears should have links for creating Fortran projects, as shown here:

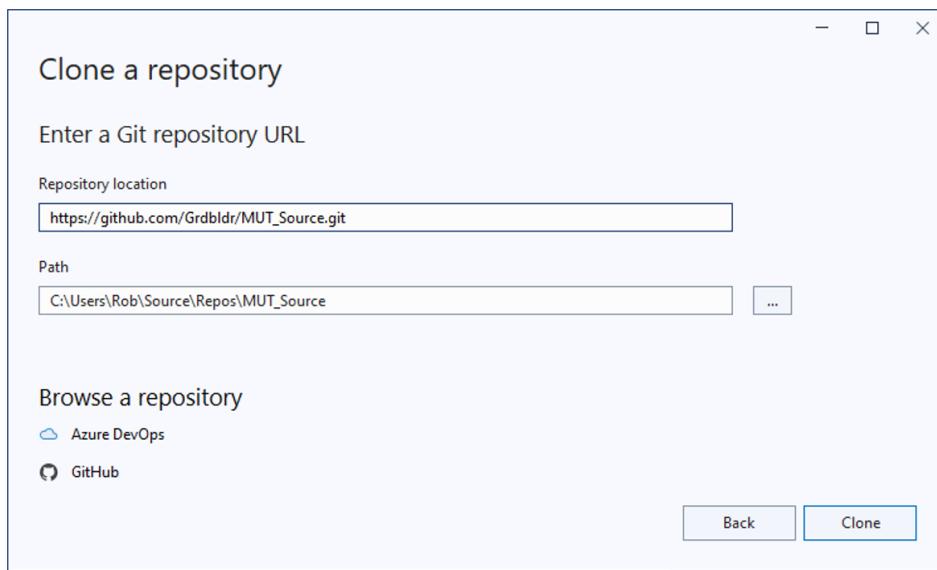


The MUT source files can be obtained from a GITHUB repository at [https://github.com/Grdbldr/MUT\\_Source.git](https://github.com/Grdbldr/MUT_Source.git). Since GITHUB has been integrated into MICROSOFT VISUAL STUDIO we will use it to download the MUT repository. When you start MICROSOFT VISUAL STUDIO choose this option:

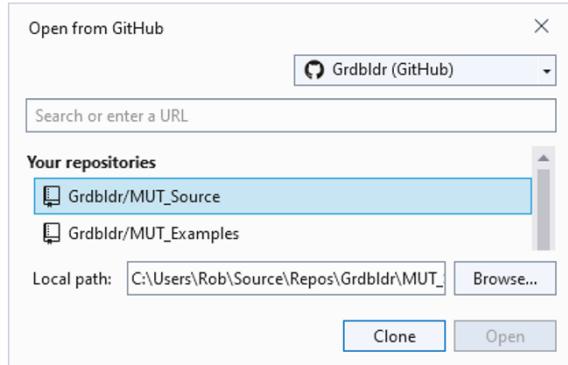


This opens the dialogue box shown below, where you can define the repository location on GITHUB and

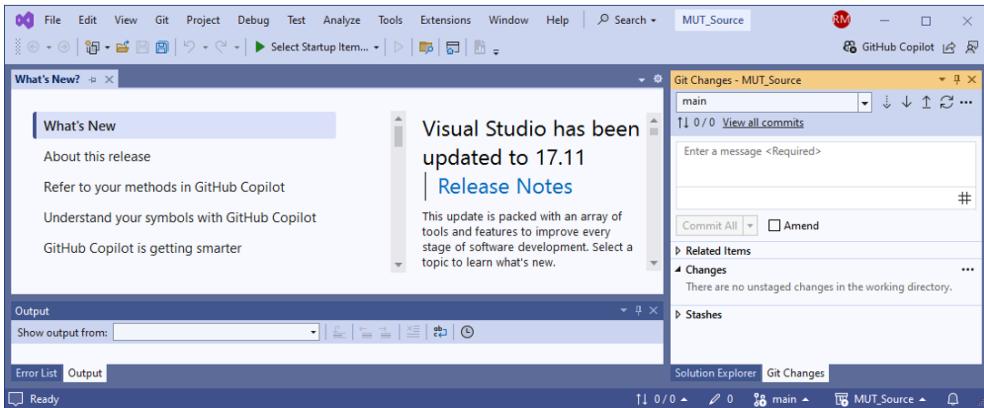
the path to the local repository. First copy the link above in this PDF file (i.e. [https://github.com/Grdbldr/MUT\\_Source.git](https://github.com/Grdbldr/MUT_Source.git)) by right-clicking on it and choosing **Copy Link Address**, then paste it in as the **Repository location**.



Now choose the GitHub option under **Browse a Repository** and you will see this dialogue shown below, Choose Grdbldr/MUT\_Source from the list of repositories then click the **Clone** button.



This shows the MICROSOFT VISUAL STUDIO window after Grdbldr/MUT\_Source has been cloned. Note the GITHUB window on the right side, and information along the bottom about the project:



The software has been developed and tested under:

- Windows 10
- TECPLT360 EX 2018 R2
- Microsoft Visual Studio Community 2022, Version 17.11.1
- Intel® Fortran Compiler 2024.1

# Chapter 3

## Model Build

The first step in any model build is to develop a conceptual model, which defines the extent, inflows and outflows, material distributions and physical properties of a hydrogeologic flow system, real or imaginary. The intent of MUT is to then facilitate the production of a set of MODFLOW-USG<sup>Swf</sup> input files by minimizing the amount of time we spend building and testing it. This chapter describes our current model build workflow, which can provide a sound basis for developing your own personal workflow.

The steps in our model build workflow are:

1. Create a new working folder or copy an existing MUT project folder.
2. Modify the MUT input file (and other input files if necessary) to reflect the new Modflow project.
3. Run MUT to build the new Modflow project, which also produces TECPLOT output files for the various Modflow domains (i.e. GWF, SWF and/or CLN) created during the build process.
4. Run TECPLOT and examine the build output files.
5. Repeat steps 2-4 until the new project is defined to your satisfaction.

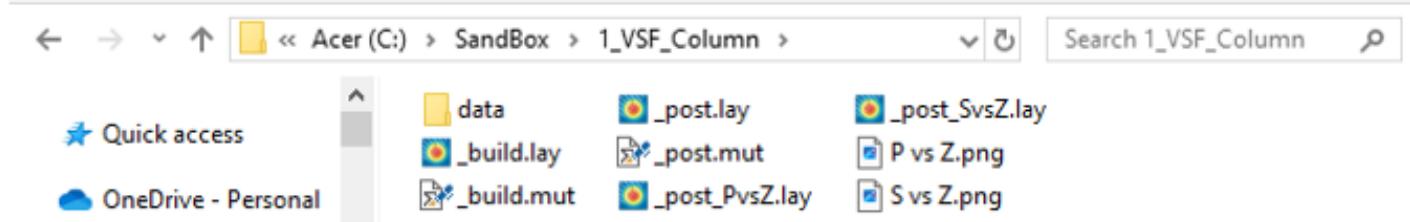
A MUT input file is a plain ascii text file that you can edit with your preferred editor (e.g. Windows Notepad<sup>1</sup>). The MUT input file name can have a prefix of your choice, followed by the extension .mut. Examples of valid MUT input file names are \_build.mut or good.mut. Most often, the easiest approach is to copy an existing input file and modify it as required. This helps reduce set-up time and avoid potential errors that are introduced when creating input files from scratch.

To illustrate our model build workflow, we will refer to the various conceptual models developed for our existing suite of examples described in Chapter 5. As you read along, we urge you to carry out the steps we describe as we move through the workflow. We recommend copying the contents of an existing model to a new location (e.g. copy the folder 1\_VSF\_Column to C:\SandBox). If you did so, your working

---

<sup>1</sup>Our personal favourite editor is WinEdt (<https://www.winedt.com/snap.html>), which also provides a nice L<sup>A</sup>T<sub>E</sub>X document development environment when coupled with the T<sub>E</sub>X software package MiK<sup>T</sup>eX. This manual was produced using these word processing tools.

directory would look something like this:



In the `1_VSF_Column` example, there is a MUT input file for the model build called `_build.mut` and a TECPLOT layout file called `_build.lay` used to visualize the model build results. The rest of the files are related to post-processing MODFLOW-USG<sup>Swf</sup> model results and will be discussed later in chapter 4.

In our preferred workflow we would first start a command prompt in the folder which contains the MUT input file by:

1. Navigating to the folder in File Explorer (e.g. `C:\SandBox\1_VSF_Column`).
2. Highlighting the path in File Explorer:

`C:\SandBox\1_VSF_Column`

3. Replacing the existing path with the string cmd:

`cmd`

4. Pressing Enter/Return.

A command prompt window rooted at the input folder should appear:

```
cmd C:\WINDOWS\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

C:\SandBox\1_VSF_Column>
```

When you run MUT it will try to obtain a prefix in the following order:

1. **From a command line argument:** At the command prompt, MUT checks for the presence of a command line argument. For example, typing this:

```
mut Good
```

would cause MUT to process the input file `Good.mut`.

2. **From a prefix file:** If there is no command line argument, MUT checks for the presence of the file `_mut.pfx` in the folder. If present, MUT will read the prefix from it. For example, if the mut file was called `Good.mut` then the file `_mut.pfx` would contain the single line:

```
good
```

3. **From the default input file:** If there is no command line argument or prefix file in the folder, MUT checks for the presence of the file `a.mut`. If present in the folder, MUT will process it.
4. **From the keyboard:** If none of these methods are successful, MUT will prompt for a prefix as shown here:

```
C:\WINDOWS\System32\cmd.exe - mut
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

C:\SandBox>mut
MUT version 1.25
No command line prefix
No file: _mut.pfx
Checking for default file: a.mut
No file: a.mut
Enter a prefix for a mut file:
```

The user would type the prefix e.g.: good and press Enter.

To build the `1_VSF_Column` example, we would run MUT using the input file `_build.mut` by typing:

```
mut _build
```

which uses the first method to supply the prefix.

If you open the file `_build.mut` in a text editor you will see the first couple of lines are comments (which begin with an exclamation point character: '!'') describing the problem:

```
! Examples\1_VSF_Column:
!   A modflow project of a 1D column generated from a simple 2d rectangular mesh
```

MUT first creates a clean copy of the input file called `_buildo.input` by removing all comment lines.

As MUT processes the input file, output is written to both the screen and to the file `_buildo.eco`. If you open `_buildo.eco`, you will see the first thing written is the MUT header, which contains the version number and build date.

These are followed by the stripped comments, which can provide a synopsis of the input file contents. The rest of the cleaned input file contains MUT instructions, which may require data in the form of numbers (e.g. parameter values) or alphanumeric strings (e.g. file names).

The first instruction in the cleaned input file begins the model build:

---

## **build modflow usg**

This is a *subtask* that defines the characteristics of the MODFLOW-USG<sup>Swf</sup> model including:

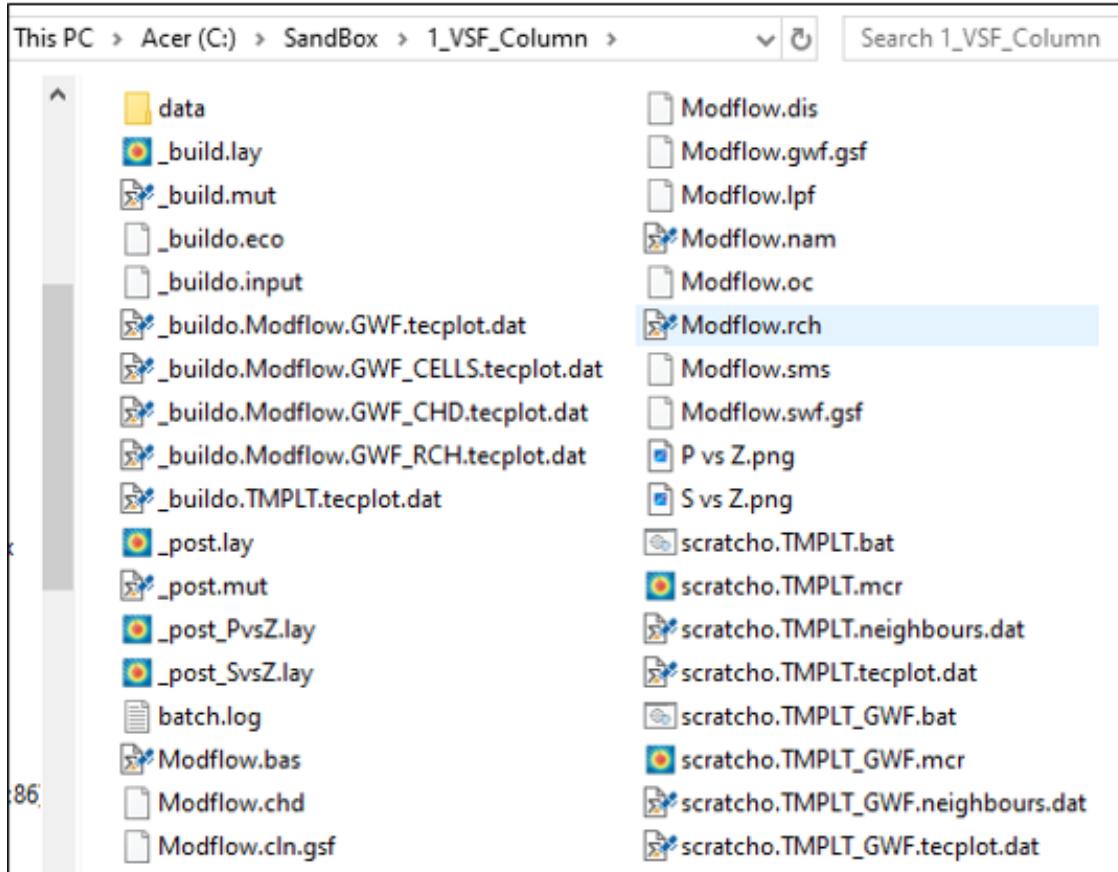
- Units of length and time.
- Numerical model meshes.
- Material properties.

- Boundary conditions.
- Time-stepping, stress period and output control parameters.
- Solver parameters.

An end instruction is required to stop the subtask e.g.:

```
end build modflow usg
```

After MUT finishes, the working folder should look something like this:



Several new output files have been created, of which it may be noted:

- Build output files, which have the prefix `_buildo`, appear near the start of the list if sorted by name. MUT deletes previously generated build output files and writes a fresh set each time it is run. This can prevent confusion that might arise if out-of-date output files were present.<sup>2</sup>
- TECPLOT output files are indicated by the suffix `.tecplot.dat`.

---

<sup>2</sup>For example, if we define a recharge boundary condition, MUT will create the file `prefixo.Modflow.SWF_RCH.Tecplot.dat` which shows the locations and recharge values assigned to Modflow cells. If we then removed the recharge condition from the input file, but did not delete this output file, we may assume the recharge condition still applies.

By default, MUT writes TECPLOT file output but this can be prevented by the following instruction:

---

### disable tecplot output

TECPLOT-formatted output files will not be written.

---

- Modflow model input files are written using the default prefix `Modflow`, (e.g. `Modflow.nam`, `Modflow.bas` etc.) The prefix can be customized if desired using this instruction:

---

### modflow prefix

1. NewPrefix<sub>str</sub> MODFLOW-USG<sup>Swf</sup> prefix NewPrefix<sub>str</sub>.
- 

There may be advantages to using a standard prefix naming convention, such as portability of post-processing scripts or TECPLOT layout files.

- Several scratch files (with prefix `scratcho`) are written. These are used for debugging during code development and can be ignored in most cases.
- If the run is successful the last line written will be `Normal exit`, otherwise an error message will be given.

## 3.1 Defining the Units of Length and Time

The input files written by MUT define the units of length and time that are applied during the MODFLOW-USG<sup>Swf</sup> model simulation.

By default, MUT assigns meters as the units of length and seconds as the units of time. If desired, the following instructions can be used to define a different unit system:

---

### units of length

1. LengthUnits<sub>str</sub> The desired units of length.

MODFLOW-USG<sup>Swf</sup> currently supports length units of feet, meters or centimeters.

---

---

### units of time

1. TimeUnits<sub>str</sub> The desired units of time.

MODFLOW-USG<sup>Swf</sup> currently supports time units of seconds, minutes, hours, days or years.

---

These instructions would define units of centimeters and days instead of meters and seconds:

units of length  
centimeters

units of time  
days

MUT converts the string variables **LengthUnits<sub>str</sub>** and **StringUnits<sub>str</sub>** to their numeric equivalents and passes that to MODFLOW-USG<sup>Swf</sup> through the variables **LENUNI** and **ITMUNI** respectively.

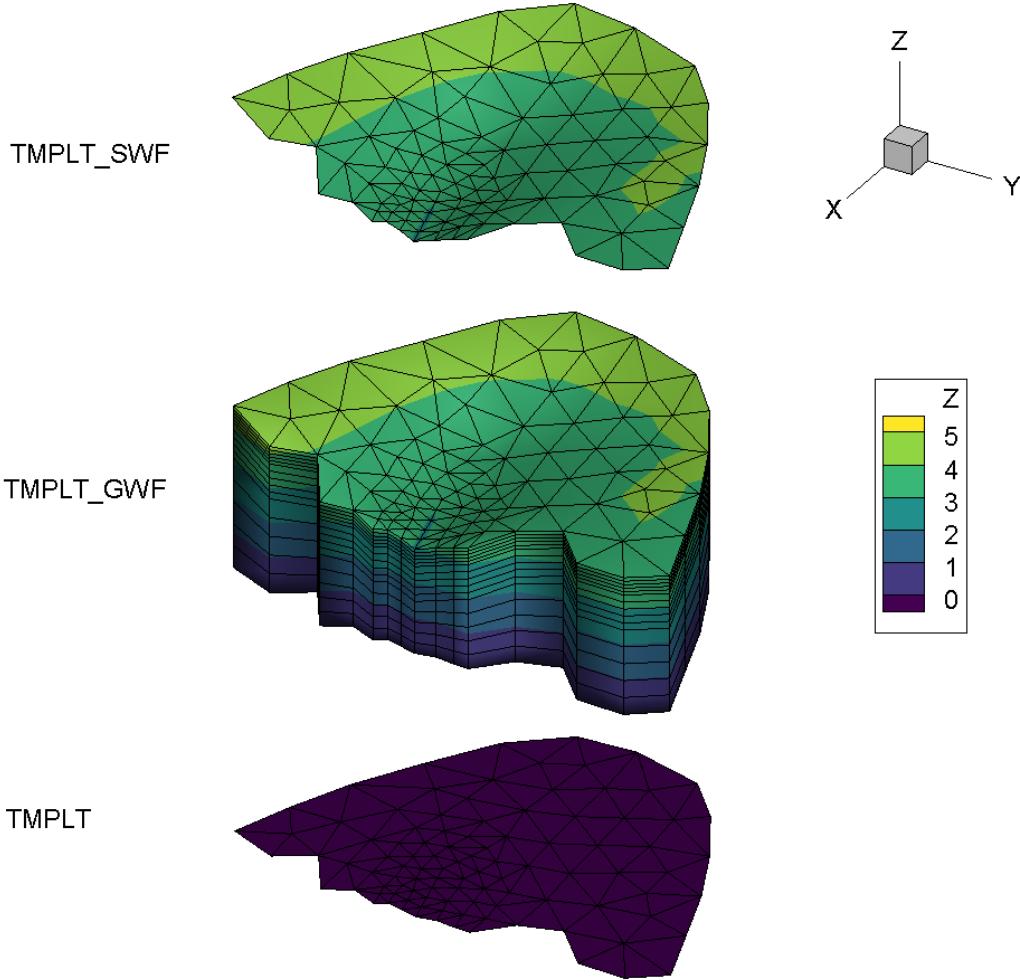
## 3.2 Defining the Template Mesh

The next step in the model build workflow is to define a template mesh, which is a 2D finite-element mesh that is used to generate a 3D GWF (and possibly a 2D SWF) finite-element mesh.

Below is an example<sup>3</sup> showing an exploded view of a template mesh (**TMPLT**, bottom image) that was used as a basis for generating finite-element meshes for the GWF (**TMPLT\_GWF**, middle image) and SWF (**TMPLT\_SWF**, upper image) domains:

---

<sup>3</sup>This example was generated using the TECPLOT layout file `MUT_Examples\6_Abdul_Prism_Cell\FIG Template Abdul.lay`.



Some key features of this example are:

- The template mesh is assigned an elevation of zero, and only the  $xy$  coordinate data are used to define the other domains.
- The GWF domain has been assigned a base elevation of zero, and a variable top elevation.
- The SWF domain has been assigned the same elevation as the GWF domain i.e. they are coincident.

In this example, the template mesh was defined using these instructions:

```
2d mesh from gb
.\gb\grid
```

The instruction `2d mesh from gb`, which requires a single line of input (i.e.: `.\gb\grid`), is documented as shown here:

---

## 2d mesh from gb

1. **Prefix<sub>str</sub>** The GRID BUILDER<sup>4</sup> dataset prefix, including the path to it.

---

<sup>4</sup>GRID BUILDER is a legacy 2D triangular finite-element grid generator.

Given **Prefix<sub>str</sub>**, this instruction reads the 2D finite-element grid data and uses it to define the 2D template mesh. **Prefix<sub>str</sub>** should contain a relative path<sup>5</sup> to the dataset.

---

Most input instructions, including **2d mesh from gb**, do not include length or time unit information, and MUT assumes the current units (i.e. either the default units of meters and seconds or other explicitly defined units) apply to the given input data. When supplying input data *you must be careful to supply the values in the unit system defined for the model*. As a reminder, MUT will echo the assumed units to the screen and **o.eco** file, for example:

```
2d mesh from gb
Number of nodes:          1372
Number of elements:        2651
Assumed length Units:     METERS
```

When databases are used to define material parameter values (see page 41), fields defining the length and time units of each record are included. In this case MUT will automatically convert the parameter values to the unit system that has been defined for the MODFLOW-USG<sup>Swf</sup> model.

---

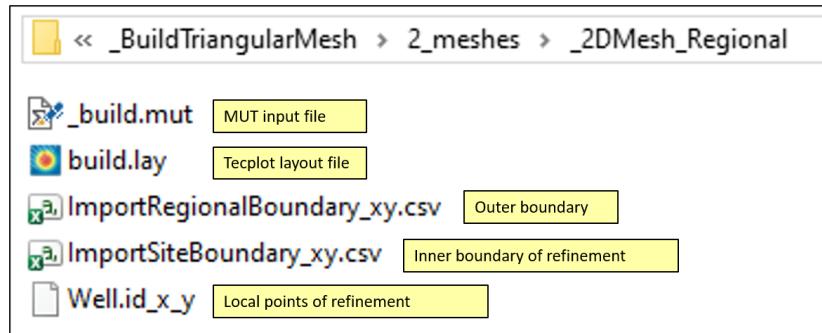
<sup>5</sup>Examples of relative paths are:

**\gb\grid** The MUT input folder contains a local folder **gb** with the data set prefix **grid**.

**..\gb\grid** The parent folder to the MUT input folder contains a folder **gb** with the data set prefix **grid**.

**C:\gb\grid** Absolute path to a drive C: folder **gb** with the data set prefix **grid**. Absolute paths are not recommended as they may lead to portability issues.

MUT can now generate 2D-triangular-element meshes given a list of *xy* coordinates that define the outer boundary of the model domain and a target element size for the mesh triangles. There are also optional instructions for refining portions of the mesh inside a polygon or at a point location. The initial contents of an example<sup>6</sup> mesh generation folder are shown below:



QGIS was used to define the following files:

**ImportRegionalBoundary\_xy.csv** The regional boundary that forms the outer boundary of the 2D mesh.

**ImportSiteBoundary\_xy.csv** The site boundary that defines a zone of refinement.

**Well.id\_x\_y** The well location that defines a point of refinement.

This MUT input file `_build.mut` generates a triangular mesh. The file contents are shown below:

```
! This example builds a 2D mesh from an outer boundary defined in QGIS
build triangular mesh
    ! Tecplot output mesh prefix
        Regional
            ! name of outer boundary file
            ImportRegionalBoundary_xy.csv
                Regional boundary forms
                mesh outer boundary
            ! Target element length
            300.
                Rough element size inside outer
                boundary will be 300 m
            ! zone of refinement
            refine inside polygon
            ImportSiteBoundary_xy.csv
                Refine the region inside the site
                boundary to roughly 150 m (half
                original target size)
            ! refinement at well
            wells from id_x_y file
            well.id_x_y
            10.
end build triangular mesh
```

<sup>6</sup>This example can be found in folder `MUT_Examples\BuildTriangularMesh\2DMesh_Regional`

This instruction starts the grid generation procedure:

---

## build triangular mesh

This subtask currently requires the following input:

- MeshName<sub>str</sub> Mesh name, in this case `Regional`
- FName<sub>str</sub> Name of the file containing the *xy* coordinate list around the outer boundary, in this case `ImportRegionalBoundary_xy.csv`
- TargetLength<sub>real</sub> [*L*] Target element length, in this case 300 m.

*Optional grid modification instructions may be inserted here...*

An end instruction is required to stop the subtask e.g.:

**end build triangular mesh**

---

There are currently 2 *Optional grid modification instructions*:

---

## refine inside polygon

- FName<sub>str</sub> Name of the file containing the *xy* coordinate list around the zone of refinement, in this case `ImportSiteBoundary_xy.csv`.

---

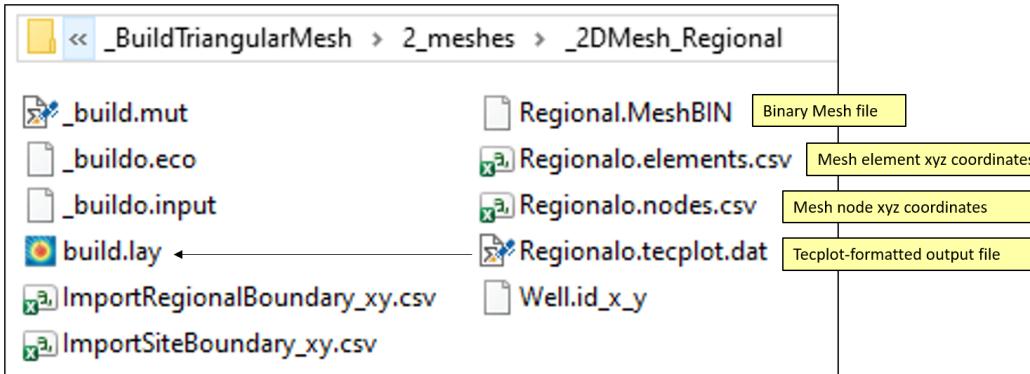
## wells from id\_x\_y file

- FName<sub>str</sub> File name of the file containing the point *xy* coordinate list.
- WellTargetLength<sub>real</sub> [*L*] Target element length near the well, in this case 10 m.

These optional instructions must follow the `TargetLength` input line but can be repeated to define, for example:

- Different or nested areas of mesh refinement.
- Multiple wells with different target lengths.

The contents of the mesh generation folder are shown below after grid generation:



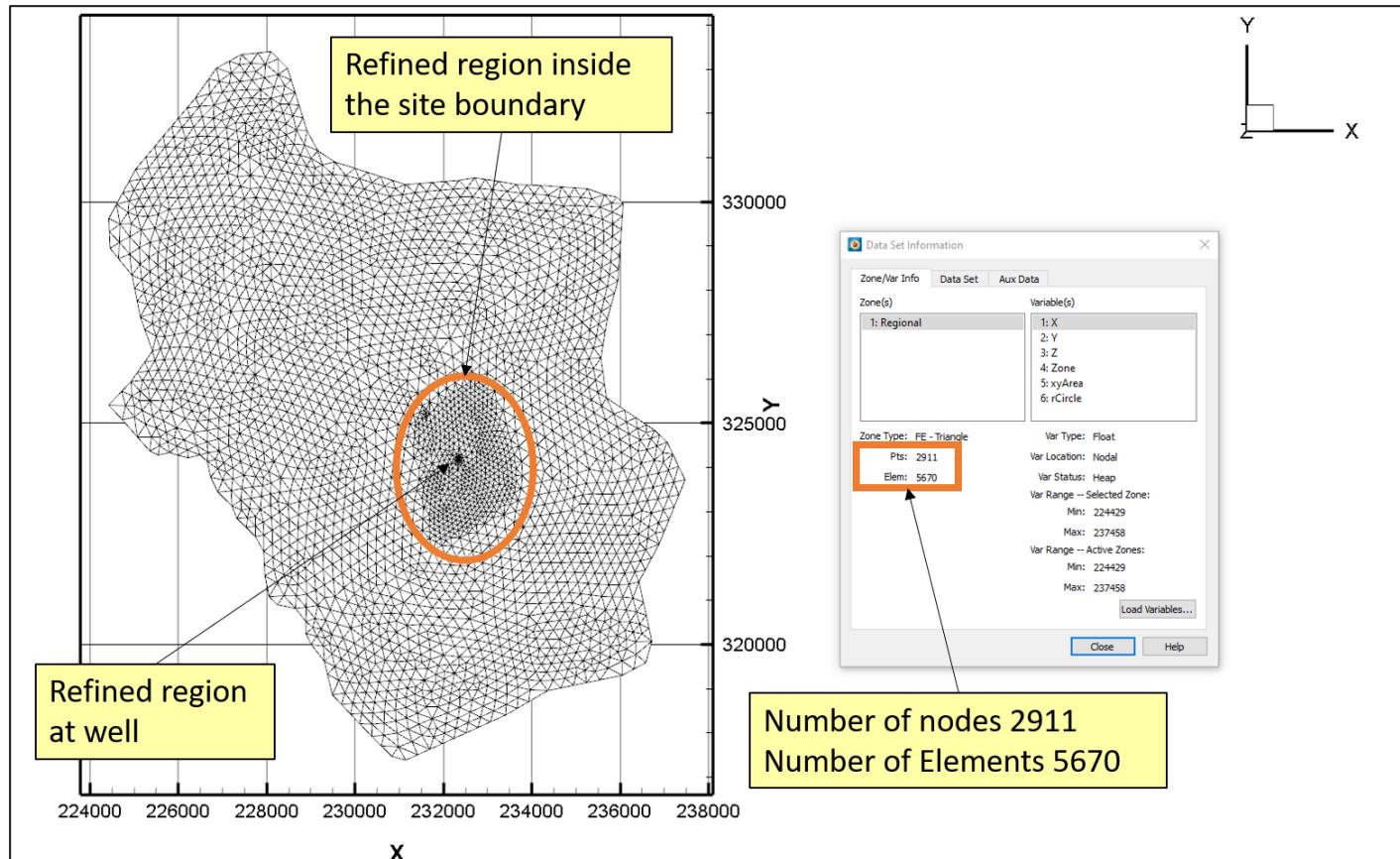
Note the following files that have been created:

**Regional.MeshBIN** This binary file contains the mesh structure and can be read by MUT using the instruction `read mesh`, described below.

**Regionalo.nodes.csv** This csv file contains node *xyz* coordinates that can be imported by QGIS to, for example, visualize with other map data or to sample DEM elevation data (initially, all *z* coordinates are set to zero by the grid generator) at node coordinates.

**Regionalo.elements.csv** As above but for triangular-element centroid coordinates.

**Regionalo.tecplot.dat** A TECPLOT-formatted output file of the generated mesh. Shown below, the TECPLOT layout file `_build.lay` has been loaded into TECPLOT and the Data Set Information dialogue has been opened:



The layout file is configured to load the data file `Regionalo.tecplot.dat` by default.

By default, MUT writes `csv` files for QGIS but this can be prevented by the following instruction:

---

## disable qgis output

QGIS-formatted output files will not be written.

---

To generate uniform 2D rectangular element template meshes<sup>7</sup> use this instruction:

---

## generate uniform rectangles

1.  $\underline{xLength}_{real}[L]$ ,  $\underline{nRectX}_{int}$  Domain length and number of rectangles in the  $x$ -direction
2.  $\underline{yLength}_{real}[L]$ ,  $\underline{nRectY}_{int}$  Domain length and number of rectangles in the  $y$ -direction

A 2D finite-element mesh composed of uniform rectangular elements will be generated. In this case, the grid is formed by subdividing the domain in the  $x$ -direction into  $\underline{nRectX}_{int}$  rectangles, each of length  $\underline{xLength}_{real}/\underline{nRectX}_{int}$ . The domain is subdivided in a similar fashion in the  $y$ -direction, using the second set of input parameters.

---

To generate variable 2D rectangular element template meshes use this instruction:

---

## generate variable rectangles

1.  $\underline{nx}_{int}$  Desired number of cells  $\underline{nx}_{int}$  in the  $x$ -direction +1.
2.  $\underline{xi(i)}_{real}$ ,  $i=1,\underline{nx}_{int}$  Array of  $\underline{nx}_{int}$   $x$ -coordinates.
3.  $\underline{ny}_{int}$  Desired number of cells  $\underline{ny}_{int}$  in the  $y$ -direction +1.
4.  $\underline{yi(i)}_{real}$ ,  $i=1,\underline{ny}_{int}$  Array of  $\underline{ny}_{int}$   $y$ -coordinates.

A 2D finite-element mesh composed of variably-sized rectangular elements will be generated. In this case, the grid is formed by subdividing the domain in the  $x$ -direction into  $\underline{nx}_{int}-1$  rectangles, each of variable length. The domain is subdivided in a similar fashion in the  $y$ -direction, using the second set of input parameters.

---

Once a template mesh has been defined by any of the mesh generation options described above, it is automatically saved to a `MeshBIN` file, which can later be loaded into memory using the instruction `read mesh`, which requires the following input:

---

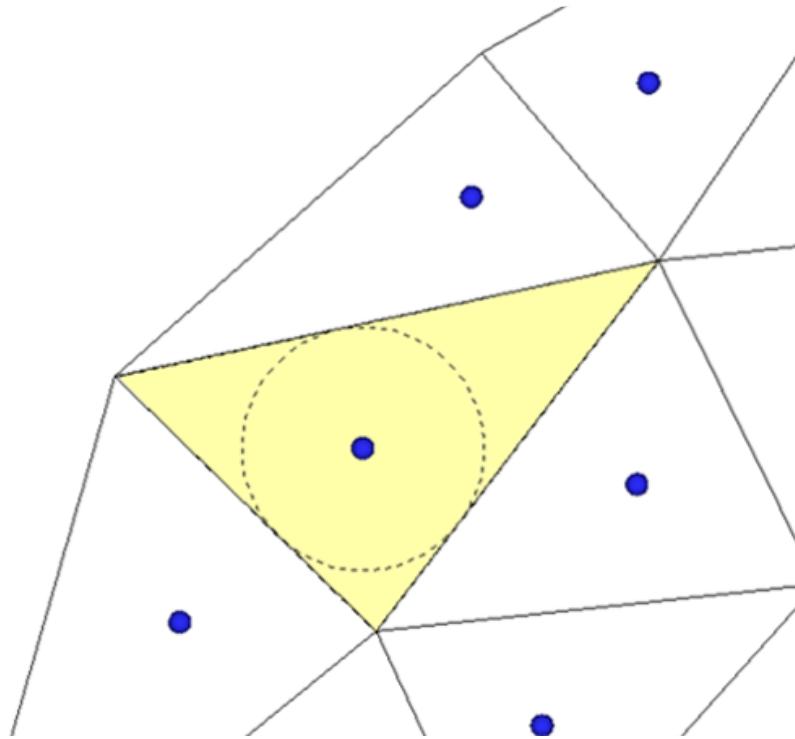
## read mesh

1.  $\underline{\text{Prefix}}_{str}$  The `MeshBIN` file prefix, including the path to it (see Footnote 5).

---

<sup>7</sup>See for example the verification cases `1_VSF_Column` or `6_Abdul_MODHMS`

There are two ways that MODFLOW cell control volumes can be defined from the template mesh. By default, MUT uses a mesh-centred approach as shown here for a triangular-element template mesh:



Some key features to note are:

- Inner circles (dashed circle in yellow triangle), which are tangent to all three element sides, are defined for each triangular element.
- The blue-filled circles show the locations of the defined MODFLOW cell control volumes.
- The vertical connection area of the cell is defined by the triangular element area (yellow-shaded triangle).
- The horizontal connection length of the cell is defined by the triangular element side length between neighbouring elements.

The mesh-centred approach is similar when using a rectangular-element template mesh, with the rectangular element area and side lengths defining the vertical connection area and horizontal connection length respectively.

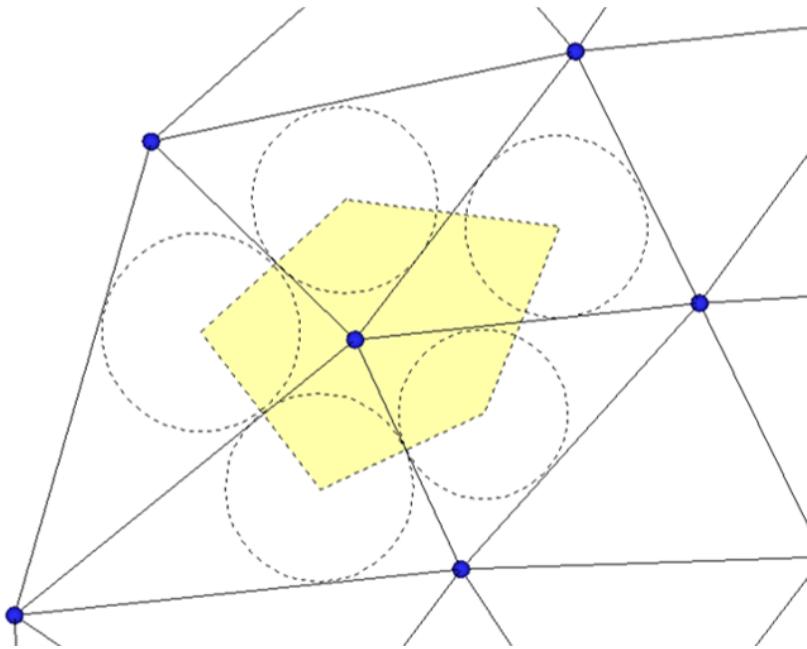
To use a node-centred control volume approach, add this instruction *before* defining any GWF or SWF model domains:

---

### nodal control volumes

The node-centered approach will be used to define MODFLOW cell centres instead of the default mesh-centered approach.

The result of using a node-centred approach is shown here for a triangular-element template mesh:



Some key features to note are: .

- The blue-filled circles show that the locations of the defined MODFLOW cell control volumes are now located at template mesh node locations.
- The vertical connection area of the cell (yellow-shaded polygon) is defined by the contributing area formed by joining the inner circle centres of each element containing the template mesh node.
- The horizontal connection length of the cell is defined by the distance to a neighbouring node.

### 3.3 Groundwater Flow(GWF) Domain

Currently, every MODFLOW-USG<sup>Swf</sup> model must contain a GWF domain, which may be reduced to a single layer of very low hydraulic conductivity in cases where GWF flow and interaction with other model domains is to be neglected.

#### 3.3.1 Generating a Layered GWF Domain

A MODFLOW-USG<sup>Swf</sup> 3D groundwater flow (GWF) domain can be generated from the template using this instruction:

---

**generate layered gwf domain**

This subtask has instructions that are used to define:

- Element zone numbering scheme
- Top elevation of domain (e.g. ground surface)
- New mesh layers, vertical discretization (sublayering) and base elevation

An end instruction is required to stop the subtask e.g.:

**end generate layered gwf domain**

*NOTE: The term layers used here should not be confused with the MODFLOW term of the same name. A MODFLOW layer is one cell thick, while a MUT layer can be one or more elements thick.*

The construction of the 3D GWF finite-element mesh proceeds from top to bottom. First, we define the top elevation, then add layers one at a time, defining the base elevation and vertical discretization of each new layer, until we reach the base of the domain. By default, element zone numbering corresponds with layer numbering. Alternatively, if the template mesh is divided into horizontal patches with unique zone numbers, these can be assigned instead to the 3D GWF mesh using this instruction<sup>8</sup>:

### **Zone by template**

Causes MUT to assign the template mesh element zone number to the corresponding 3D GWF element. This results in zones that are vertical columns of elements from the top to the bottom of the GWF domain.

*This instruction should appear in the input file at the beginning of the generate layered gwf domain subtask before new layers are added.*

#### **3.3.1.1 Defining the Top Elevation**

To assign an elevation to the top layer of template nodes use this instruction:

### **top elevation**

This subtask defines the elevation (i.e. z-coordinate) of the top layer of nodes in the GWF finite-element template mesh in one of these ways:

- By assigning a given elevation to all nodes
- By reading variable elevation data from a file
- By interpolating elevation data from a function  $z(x)$  where the elevation  $z$  varies by the nodal  $x$  coordinate.

An end instruction is required to stop the subtask e.g.:

**end top elevation**

The top elevation can be defined by one of these instructions:

---

<sup>8</sup>The example 6\_Abdul\_Prism\_Cell uses this option to define SWF domain zones.

## elevation constant

1. Elev<sub>real</sub> [L] The elevation Elev<sub>real</sub> will be assigned to all top layer nodes.

## elevation from gb file

1. FName<sub>str</sub> The elevation data in the GRID BUILDER nodal property file FName<sub>str</sub> will be assigned to the top layer nodes.

The GRID BUILDER nodal property file uses a legacy binary file format. You can develop your own ascii input files and read elevation data from them using this instruction:

## elevation from list file

1. FName<sub>str</sub> The elevation data in the ascii file FName<sub>str</sub> will be assigned to the top layer nodes.

Part of a sample list file<sup>9</sup> is shown here:

```
Kriged cell top elevation for layer 1
4.414571762E+000
4.415914536E+000
...
4.415914536E+000
```

Some key features of this example are:

- The first line of the file is discarded, and in this case contains a string describing the data.
- You must supply a value for each node in the template finite-element mesh.
- The data is read in free format so there can be more than one value entered per line.
- Only the start and end of the file are shown here, with the string '...' replacing the middle portion.

To define the top elevation as a function of  $x$  (usually used for 2D cross-sectional models) use this instruction:

## elevation from xz pairs

<sup>9</sup>The example 6\_Abdul\_MODHMS uses an ascii file input to define nodal elevations.

1.  $x(1)_{real}$  [L],  $z(1)_{real}$  [L] First  $x, z$  coordinate pair.
2. ...
- n.  $x(n)_{real}$  [L],  $z(n)_{real}$  [L] nth  $x, z$  coordinate pair.

An elevation is calculated for each chosen cell, based on it's  $x$ -coordinate location, by interpolating an elevation from the given list of  $xz$ -coordinate pairs.

An end instruction is required to stop the subtask e.g.:

### **end elevation from xz pairs**

---

Here is an example showing the use of this instruction<sup>10</sup>:

```
elevation from xz pairs
    0.0,    0.0
    1000.0, 100.0
end elevation from xz pairs
```

Some key features of this example are:

- The two given  $xz$  pairs define a line that slopes from  $z = 0$  at  $x = 0$  to  $z = 100.0$  at  $x = 1000$ . You may supply as many pairs as needed to define the top of your cross-section.
- $x$  coordinates must increase continuously from the top of the list to the bottom.
- the  $x$ -range of the supplied pairs should cover the entire  $x$ -range of the template mesh.
- For each node in the template mesh, the  $x$  coordinate is used to interpolate an elevation (i.e.  $z$  value) using the appropriate  $xz$  pair.

To define the top elevation as a function of  $x$  and  $y$  (for 3D models) use this instruction:

---

### **elevation from bilinear function in xy**

1.  $xfrom_{real}$  [L],  $xto_{real}$  [L],  $yfrom_{real}$  [L],  $yto_{real}$  [L]  $x$  and  $y$  coordinates ranges.
2.  $a1_{real}$ ,  $a2_{real}$ ,  $a3_{real}$ ,  $a4_{real}$ ,  $a5_{real}$  Constants for the bilinear function.

For nodes falling within the given  $x$  and  $y$  range, the  $z$ -coordinate is computed according to the following function:

$$z = a1 + a2(x - xfrom) + a3 * (x - xfrom)^2 + a4(y - yfrom) + a5(y - yfrom)^2$$

---

<sup>10</sup>The example `1_VSF_Hillslope` uses the `elevation from xz pairs` instruction to define the top elevation of the cross-sectional domain.

Here is an example showing the use of this instruction<sup>11</sup>:

```
elevation from bilinear function in xy  
0.0 800.0 0.0 1000.0  
41.0 -0.05 0.0 0.02 0.0  
  
elevation from bilinear function in xy  
800.01 819.99 0.0 1000.0  
1.0 0.0 0.0 0.02 0.0  
  
elevation from bilinear function in xy  
820.0 1620.01 0.0 1000.0  
1.0 0.05 0.0 0.02 0.0
```

Some key features of this example are:

- Three sets of `elevation from bilinear function in xy` instructions define a tilted catchment with 3 sections.
- If a node in the template mesh falls within a bilinear function *xy*-range, then that function is used to interpolate an elevation (i.e. *z* value) for the node.
- The *xy*-ranges supplied should cover the entire top of the template mesh.

---

<sup>11</sup>The example `8_V_catchment` uses the `elevation from bilinear function in xy` instruction to define the top elevation of a portion of the 3D domain.

### 3.3.1.2 Adding Layers

A MODFLOW-USG<sup>Swf</sup> model must contain at least 1 layer, and each layer is defined using this instruction:

---

#### new layer

This subtask adds a new layer to the GWF domain by defining the layer:

- Base elevation
- Vertical discretization

An end instruction is required to stop the subtask e.g.:

---

#### end new layer

The base elevation is defined using the elevation instructions described on page [31](#) that are given for the **top elevation** instruction.

By default, MUT will stop and issue a warning message if the computed layer base elevation is greater than or equal to the current layer top elevation. This instruction forces the base to be below the top by a set amount:

---

#### Minimum layer thickness

1. MinThick<sub>real</sub>[L] Minimum thickness value.

This instruction causes MUT to enforce a minimum thickness constraint for the current layer. At nodes where the computed layer base elevation is greater than or equal to the current top elevation, MinThick<sub>real</sub> will be subtracted from the current top elevation to get the base elevation.

By default, a new layer will be assigned the name 'Layer *n*' where *n* is the current layer number. If you want to assign your own layer name use this instruction:

---

#### Layer name

1. LayerName<sub>str</sub> Layer name.

By default, a new layer will not be subdivided vertically unless one the following two instructions is issued. The first creates a uniform subdivision:

---

#### Uniform sublayering

1. nsublayer<sub>int</sub> Number of sublayers.

This instruction divides the layer vertically into  $\underline{\text{nsublayer}}_{int}$  elements, which will each have the same element height, equal to the top elevation minus the current base elevation divided by  $\underline{\text{nsublayer}}_{int}$ .

This instruction creates a non-uniform subdivision<sup>12</sup>:

## Proportional sublayering

1.  $\underline{\text{nSublayer}}_{int}$  Number of proportional sublayers.
2.  $\underline{\text{SubThick(i)real}}$ ,  $i=1,\underline{\text{nSublayer}}_{int}$  Proportional thicknesses in order from top to bottom.

This instruction can be used if you want to refine the GWF domain mesh vertically, for example, in the active zone near the SWF domain at ground surface.

It is important to understand that the variable  $\underline{\text{SubThick}}_{real}$  is not a true thickness, but is instead a dimensionless relative thickness, which is used along with the layer thickness to determine the element heights in the current column.

For example, these instructions:

```
Proportional sublayering
 3
 0.1
 1.0
 10.0
end
```

would subdivide the current layer vertically into three elements, between the current base and top elevation, with the middle element being ten times as thick as the top element, and 1/10th as thick as the bottom element.

This instruction can be used to create a basal layer surface a given distance below, and parallel to, its top surface:

## Offset base

1.  $\underline{\text{Offset}}_{real}$  [L] Height of offset.

This instruction causes the elevation of layer base to be offset below the top by the value  $\underline{\text{Offset}}_{real}$ .

<sup>12</sup>The example 6\_Abdul\_MODHMS uses the Proportional sublayering instruction to match the cell top elevations of the original MODHMS mesh.

For example, these instructions:

```
top elevation
    elevation from list file
    elev.list
end top elevation

new layer
    uniform sublayering
    3

    elevation from list file
    elev.list

    offset base
    -1.0

end new layer

end generate layered gwf domain
```

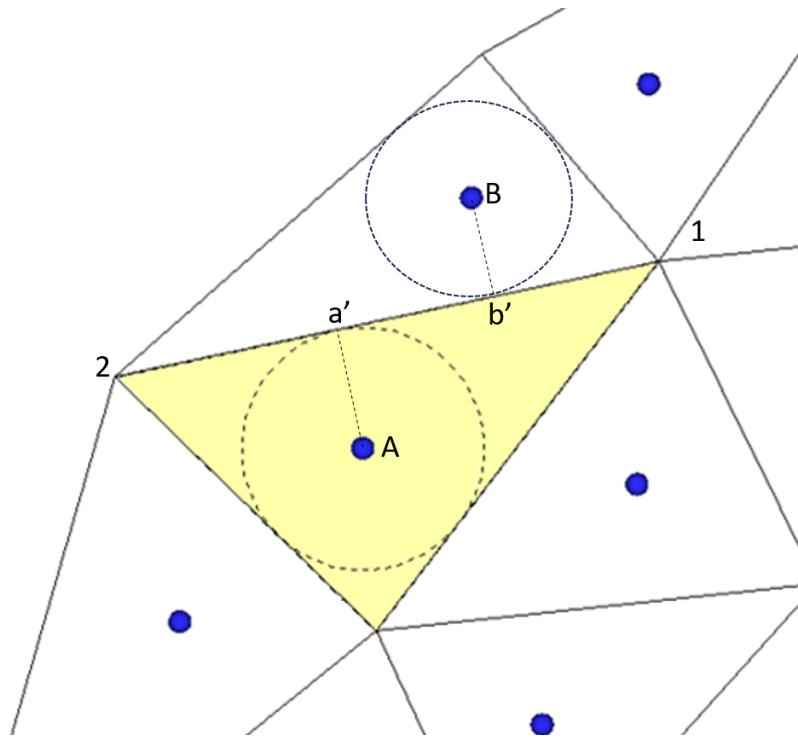
create a layer with a top elevation 1 length unit (e.g. 1 meter) below the elevation defined in the raster file `elev.list`:

### 3.3.1.3 Cell Connection Properties

Currently, MUT defines lateral connections between neighbouring cells (i.e. that share a common side) in the template 2D finite element mesh, and vertical connections between neighbouring cells (i.e. that share a common face) in the stacked 3D GWF finite-element mesh.

The GWF-GWF cell connection length (distance from cell to cell) and area (cross-sectional area perpendicular to the direction of flow) vary on a cell-by-cell basis, and depend on control volume approach (mesh- or node-centred) and element shape (triangle or rectangle).

Consider the mesh-centred case, where cell control volumes are defined at the triangular element centroid:

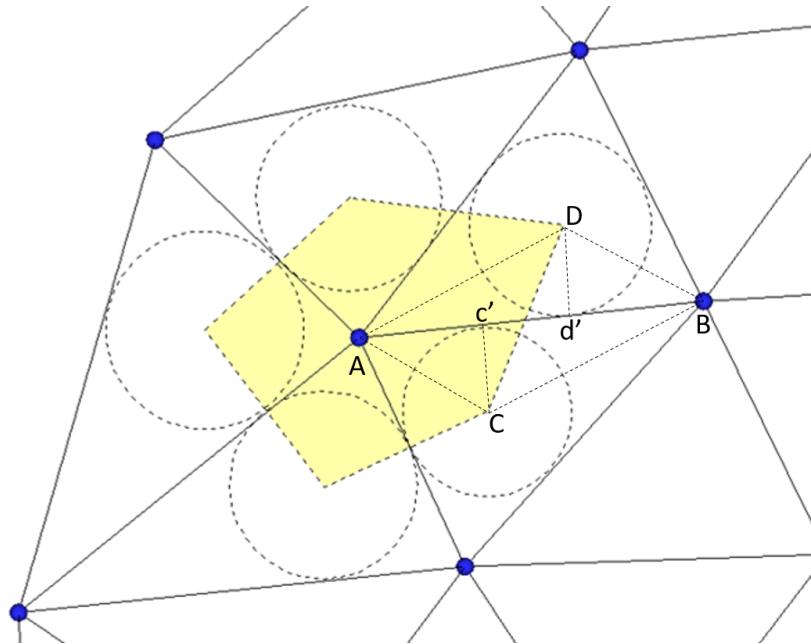


The yellow triangle shows the footprint of a MODFLOW-USG<sup>Swf</sup> cell.

Cell B is a lateral neighbour of cell A and shares side 1-2. The lateral connection length between A and B is the sum of the lengths of the inner circle radii for each cell:  $A-a' + B-b'$ . The lateral connection area is the product of the length of the shared side 1-2 and the cell thickness.

In the vertical direction, cells above and below cell A are its neighbours. The vertical connection length is the sum of the two half-cell thicknesses. The vertical connection area equals the area of the template mesh element (yellow triangle).

Here is the node-centred case, where cell control volumes are defined at the triangular element node:



The yellow polygon shows the footprint of a MODFLOW-USG<sup>Swf</sup> cell.

Cell B is a lateral neighbour of cell A. The lateral connection length between A and B is the distance from node A to node B. The lateral connection area is defined by the sum of the inner circle radii for each neighbouring triangle: C-c' + D-d' and the cell thickness.

In the vertical direction, cells above and below cell A are its neighbours. The vertical connection length is the sum of the two half-cell thicknesses. The vertical connection area is the area of the polygon formed by joining the inner circle centres of the template mesh elements common to cell A (yellow polygon).

Cell connection data are defined in a similar fashion for model domains that are generated from a rectangular element template mesh. In this case, element side lengths are used to define connection lengths and areas instead of inner circle radii.

### 3.3.1.4 Material Properties

GWF domain material properties may vary on a cell-by-cell basis. MUT has instructions for selecting subsets of cells. A cell is selected when a true/false attribute (referred to as the cells **Chosen** attribute) is set to true. Selected cells can be assigned material property values. This instruction selects all cells in the *active* model domain:

#### choose all cells

Select all cells in the active model domain.

This is an example of what we refer to as a *generic* instruction, which means it will be applied to the currently active model domain: GWF, SWF or CLN. To choose all cells in the GWF domain, we first need to activate it using this instruction:

#### active domain

1. **Domain<sub>str</sub>** The name of the domain to be activated: GWF, SWF or CLN

This instruction activates the given domain named in **Domain<sub>str</sub>** so that it will be used with generic instructions such as **choose all cells**.

So to activate the GWF domain, we would insert these instructions in the input file:

```
active domain
gwf
```

These instructions can now be used to choose GWF cells in various ways:

#### choose cell at xyz

1. **x1<sub>real</sub>** [L], **y1<sub>real</sub>** [L], **z1<sub>real</sub>** [L] An xyz coordinate triplet.

The cell closest to the given *xyz* coordinate triplet will be chosen.

## choose cells by layer

1. **Layer**<sub>int</sub> The number of the layer to be chosen.

The cells in Modflow layer number **Layer**<sub>int</sub> will be chosen. Remember that Modflow layers are one cell high and are numbered from the top to the bottom of the model domain.<sup>13</sup>

## choose cells from xyz list

This subtask reads *xyz* coordinates from a list.

1. **x1**<sub>real</sub> [L], **y1**<sub>real</sub> [L], **z1**<sub>real</sub> [L] An *xyz* coordinate triplet.

An end instruction is required to stop the subtask e.g.:

end choose cells from xyz list

## choose cells from file

1. **FName**<sub>str</sub> The file **FName**<sub>str</sub> containing a list of cell numbers.

The cells listed in the file **FName**<sub>str</sub> will be chosen.<sup>14</sup>

## choose cells from gb elements

1. **FName**<sub>str</sub> The GRID BUILDER chosen element file **FName**<sub>str</sub> containing information concerning the status, chosen or not chosen, of each element in the GRID BUILDER model domain.

If an element is flagged as chosen in the GRID BUILDER model domain then the corresponding cell will be chosen in the MODFLOW-USG<sup>Swf</sup> model domain.<sup>15</sup>

## choose cells from gb nodes

1. **FName**<sub>str</sub> The GRID BUILDER chosen node file **FName**<sub>str</sub> containing information concerning the status, chosen or not chosen, of each node in the GRID BUILDER model domain.

<sup>13</sup>See the example 1\_VSF\_Column which uses the previous two instructions to define a constant head at the base and a recharge boundary condition at the top of the 1D column.

<sup>14</sup>See the example 1\_Abdul\_MODHMS which uses this instruction to assign some cells as inactive.

<sup>15</sup>See the example 1\_Abdul\_Prism\_Cell which uses this instruction to assign some cells as inactive.

If a node is flagged as chosen in the GRID BUILDER model domain then the corresponding cell will be chosen in the MODFLOW-USG<sup>*Swf*</sup> model domain.<sup>16</sup>

The previous two instructions are used to choose cells for the mesh-centered and node-centered approaches respectively.

Cell selection instructions are cumulative. For example, you can modify the current selection by repeating instructions like **choose cell at xyz** or **choose cells by layer** with different inputs and then assign properties to the current selection. This instruction clears the selection before beginning new cell selection(s):

---

### **clear chosen cells**

Clears the current cell selection.

It is good practice to clear the selection before starting a new selection. MUT echoes the results of the selection instructions to the screen and .eco file as shown in this example:

```
clear chosen cells
GWF Cells chosen:      0

choose all cells
GWF Cells chosen:      39765
```

If a cell selection instruction has unexpected results it is good practice to check the .eco file for these outputs.

The recommended way to assign domain material properties is through the use of a lookup table. Lookup tables are provided for each domain in the **USERBIN** directory, as outlined on page [10](#). In this case, the file **GWF.csv** contains the lookup table for the **GWF** domain.

In order for MUT to access the lookup table, you first need to provide a link to this file using the instruction:

---

### **gwf materials database**

1. **FName<sub>str</sub>** GWF material properties lookup table file name.

MUT uses the file **FName<sub>str</sub>** to look up GWF material properties.

In the case of the **GWF** domain, the instructions:

```
gwf materials database
GWF.csv
```

---

<sup>16</sup>See the example **1\_Abdul\_Prism\_Cell\_nc** which uses this instruction to assign some cells as inactive.

would be used to link to the lookup table.

You can now assign a set of GWF material properties to the current cell selection using this instruction:

## chosen cells use gwf material number

1. MaterialID<sub>int</sub> GWF material ID number.

The unique set of GWF material properties with ID number MaterialID<sub>int</sub> is retrieved from the lookup table and assigned to the chosen cells.

The example 1\_VSF\_Column uses the following instructions to assign units of length and time for the MODFLOW-USG<sup>Swf</sup> model and the properties of the material with ID number 1 as shown here:

```
build modflow usg
    units of length
    centimeters

    units of time
    days

    ... etc

    gwf materials database
    GWF.csv

    active domain
    gwf
        choose all cells

            chosen cells use gwf material number
            1
```

The line '... etc' indicates a section of the input file that did not need to be shown for the purposes of this example.

The first few lines of the GWF.csv file are shown here:

GWF.csv - Excel																
File Home Insert Page Layout Formulas Data Review View Help Q Tell me what you want to do																
K15																
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Material ID	Material name	Porosity	Kh (Kx)	Kv (Kz)	Ky	Specific storage	Specific Yield	Function Type	Alpha	Beta	Sr	Brooks-Corey Exponent	Length Unit	Time Unit	Notes
2	1	1D Column	0.3	10	10	0	0.000001	0.43	Van Genuchten	0.036	1.56	0.18	6.5714	CENTIMETERS	DAYs	Rob copied these p
3	2	1D Column Brooks	0.3	10	10	0	0.000001	0.43	Brooks-Corey	0.036	1.56	0.18	6.5714	CENTIMETERS	DAYs	-

Some key features to note are:

- CSV files can be loaded and examined using MICROSOFT EXCEL.

- The first line is a header with contains field (i.e. column) names.
- The first column contains the material ID number, followed by the material parameters (one per column), unit definitions and notes.
- Material 1 defines length units of centimeters and time units of days. If these are the same as the defined MODFLOW-USG<sup>Swf</sup> unit system then no unit conversion is required, otherwise parameter values will be converted into the defined MODFLOW-USG<sup>Swf</sup> unit system according to their dimensionality.

Shown below is the output echoed to the screen and `_builde.eco` file for the example `1_VSF_Column`:

```

gwf materials database
  Materials file C:\MUT\MUT_Examples-main\_MUT_USERBIN\GWF.csv

active domain
  gwf

choose all cells
  GWF Cells chosen:      100

chosen cells use gwf material number
  Assigning all chosen GWF cells properties of material      1, 1D Column
  Kh_Kx:           10.000    CENTIMETERS  DAYS^(-1)
  Kv_Kz:           10.000    CENTIMETERS  DAYS^(-1)
  Specific Storage: 1.00000E-07  CENTIMETERS^(-1)
  Specific Yield:   0.43000    DIMENSIONLESS
  Alpha:            3.60000E-02  CENTIMETERS^(-1)
  Beta:             1.5600     DIMENSIONLESS
  Sr:              0.18140     DIMENSIONLESS
  Unsaturated Function Type:  Van Genuchten

```

Some key features to note are:

- The location and name of the materials database file used are shown.
- The material name (**1D Column**) associated with material ID number 1 is shown.
- The units defined in the database for each property are given. For example, property **Kh\_Kx** has units of **CENTIMETERS DAYS<sup>-1</sup>**, where the string <sup>-1</sup> means 'raised to the power of minus 1', giving units of **CENTIMETERS/DAYS**.

Editing the **csv** files directly is not recommended. To modify existing or define new database files, please refer to the guidelines given in Appendix [B](#).

The following section describes instructions used to assign values to individual materials properties for the current cell selection. When using these instruction, *you must be careful to supply the values in the unit system defined for the model*.

By default, MODFLOW-USG<sup>Swf</sup> simulates variably-saturated flow in the GWF domain. This instruction can be used to switch to a confined, saturated flow solution:

---

## saturated flow

---

This instruction is used to define the horizontal hydraulic conductivity, Kh:

---

### gwf kh

---

1.  $\underline{Kh}_{real}$  [ $L\ T^{-1}$ ] Horizontal hydraulic conductivity.

A horizontal hydraulic conductivity of  $\underline{Kh}_{real}$  is assigned to the chosen cells.

---

As a reminder to be aware of units, MUT will echo the assumed units to the screen and o.eco file e.g.:

```
gwf kh
Assigning all chosen GWF cells a Kh of 0.10000E-04 CENTIMETERS DAYS^(-1)
```

---

### gwf kv

---

1.  $\underline{Kv}_{real}$  [ $L\ T^{-1}$ ] Vertical hydraulic conductivity.

A vertical hydraulic conductivity of  $\underline{Kv}_{real}$  is assigned to the chosen cells.

---

### gwf ss

---

1.  $\underline{Ss}_{real}$  [ $L^{-1}$ ] Specific storage.

A specific storage of  $\underline{Ss}_{real}$  is assigned to the chosen cells.

---

### gwf sy

---

1.  $\underline{Sy}_{real}$  Specific yield.

A specific yield of  $\underline{Sy}_{real}$  is assigned to the chosen cells.

---

### gwf alpha

---

1.  $\underline{\text{Alpha}}_{real}$  [ $L^{-1}$ ] Van Genuchten/Brooks-Corey Alpha.

A Van Genuchten/Brooks-Corey Alpha of  $\underline{\text{Alpha}}_{real}$  is assigned to the chosen cells.

---

## gwf beta

1. Beta<sub>real</sub> Van Genuchten/Brooks-Corey Beta.

A Van Genuchten/Brooks-Corey Beta of Beta<sub>real</sub> is assigned to the chosen cells.

## gwf sr

1. Sr<sub>real</sub> Residual saturation.

A residual saturation of Sr<sub>real</sub> is assigned to the chosen cells.

## gwf brooks

1. Brooks<sub>real</sub> Brooks-Corey exponent.

A Brooks-Corey exponent of Brooks<sub>real</sub> is assigned to the chosen cells.

### 3.3.1.5 Material Zones

During the model build, each GWF cell is assigned a zone number from either the layer number or 2D template zone number. Just like individual cells, zones can be selected using these instructions:

#### choose all zones

Select all zones in the active model domain.

#### choose zone number

1. value<sub>int</sub> The number of the zone to be chosen.

#### clear chosen zones

Clears the current zone selection.

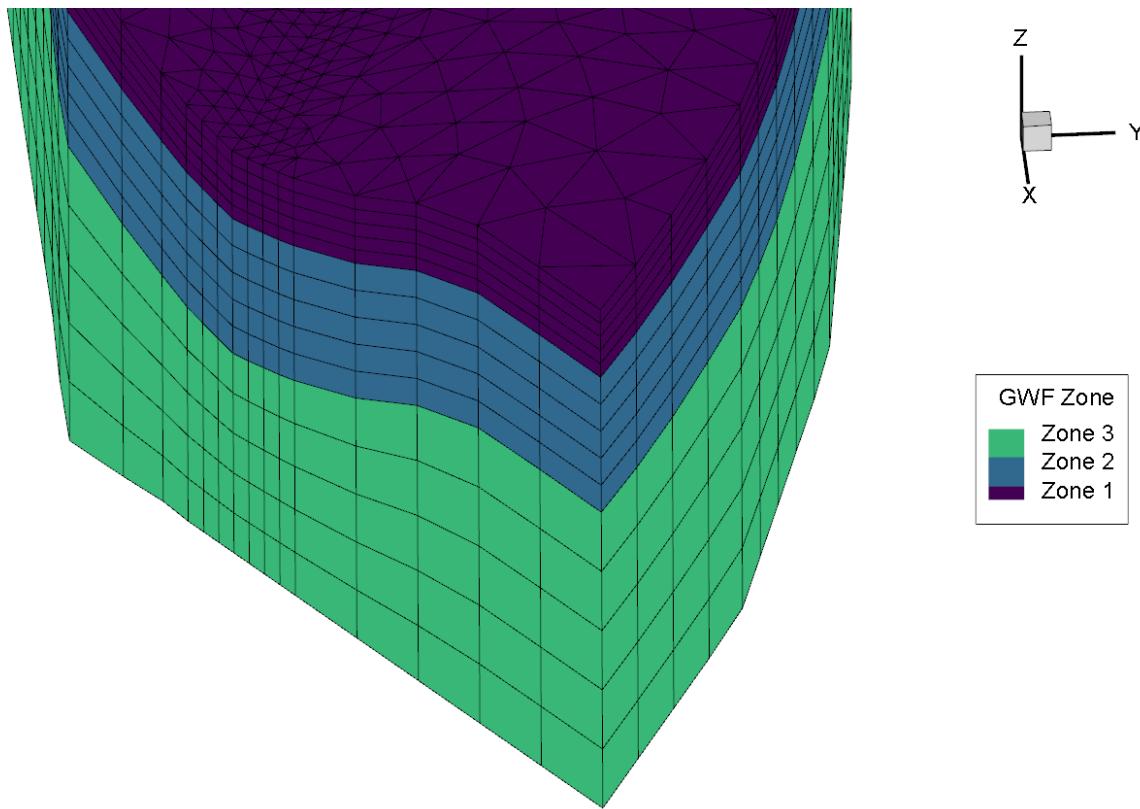
The zone selection can be converted into a cell selection using this instruction:

## choose cells by chosen zones

If a zone is currently chosen, any cell which has that zone number will be chosen.

This cell selection can now be used to assign material properties.

Below is an example<sup>17</sup> of a case in which the element zone numbers have been assigned by layer number for a 3-layer case:



Some key features to note are:

- There are 3 zones, corresponding to the layers 1 to 3.
- 'Zone 1', coloured dark blue, corresponds to layer 1. Recall that the MODFLOW-USG<sup>Swf</sup> mesh is generated from the top down, so layer 1 is at the top of the model domain.
- Each layer is composed of multiple MODFLOW layers, which are each one cell thick.

The following section from the input file shows how material properties were assigned for the 3-layer case:

```
gwf materials database  
GWF.csv
```

```
active domain
```

<sup>17</sup>See example 6\_Abdul\_Prism\_Cell

```
gwf
```

```
    clear chosen zones
    choose zone number
    1
    choose zone number
    2
    choose zone number
    3

    clear chosen cells
    choose cells by chosen zones

    chosen cells use gwf material number
    5
```

The following section from the `eco` file shows zone selection output for the 3-layer case:

```
clear chosen zones
GWF Zones chosen:          0

choose zone number
Adding zone number:         1
GWF zone numbers currently chosen:
                           1

choose zone number
Adding zone number:         2
GWF zone numbers currently chosen:
                           1
                           2

choose zone number
Adding zone number:         3
GWF zone numbers currently chosen:
                           1
                           2
                           3

clear chosen cells
GWF Cells chosen:          0

choose cells by chosen zones
GWF Cells chosen:          39765
```

Some key features to note are:

- As we choose zone numbers, the list of currently chosen zones grows accordingly.
- The final number of GWF cells chosen is equal to the total number of cells in the domain, since we had selected all 3 layers prior to converting the zone selection to a cell selection.

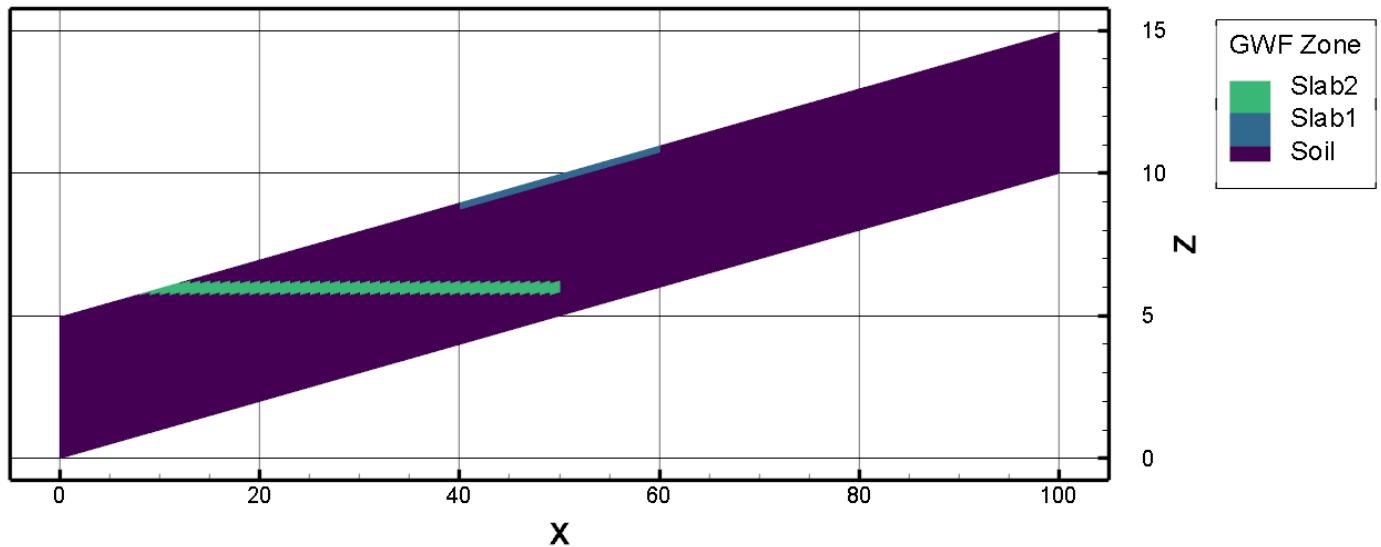
Since we are assigning uniform properties to the entire model domain, this example could be simplified by simply choosing all cells then assigning properties. However, if you wanted to assign different material properties to each zone, you can easily do so by modifying the example to assign properties to one zone at a time, being careful to clear the zone and cell selections for each layer.

In some cases, we may want to assign material properties to selected subregions of the model domain. In such situations, we first select a subset of cells, then assign them a unique zone number using the instruction:

### **new zone**

Increments the total number of zones in the currently active domain by 1 and assigns that as the zone number of each selected cell.

Below is an example<sup>18</sup> in which a GWF model domain which initially had a single zone called **Soil** has had two new zones added called **Slab1** and **Slab2**:



In this example, we introduce the following instruction for selecting arbitrary ranges of cells:

### **choose cells by xyz layer range**

1. xfrom<sub>real</sub> [L], xto<sub>real</sub>[L] x coordinate range.
2. yfrom<sub>real</sub> [L], yto<sub>real</sub>[L] y coordinate range.
3. zfrom<sub>real</sub> [L], zto<sub>real</sub>[L] z coordinate range.

<sup>18</sup>See example 7\_SuperSlab

#### 4. LayerFrom<sub>int</sub>, LayerTo<sub>int</sub> Layer range.

Cells whose centroid falls within the given  $x$ ,  $y$ ,  $z$  and layer ranges are selected.

The model zones were defined using these instructions:

```
active domain
gwf

choose all cells

chosen cells use gwf material number
11

clear chosen zones
clear chosen cells
choose cells by xyz layer range
40.0 60.0
-1e30 1e30
-1e30 1e30
1 5

new zone

choose zone number
2

chosen cells use gwf material number
12

clear chosen zones
clear chosen cells
choose cells by xyz layer range
8.0 50.0
-1e30 1e30
5.8 6.2
0 1000

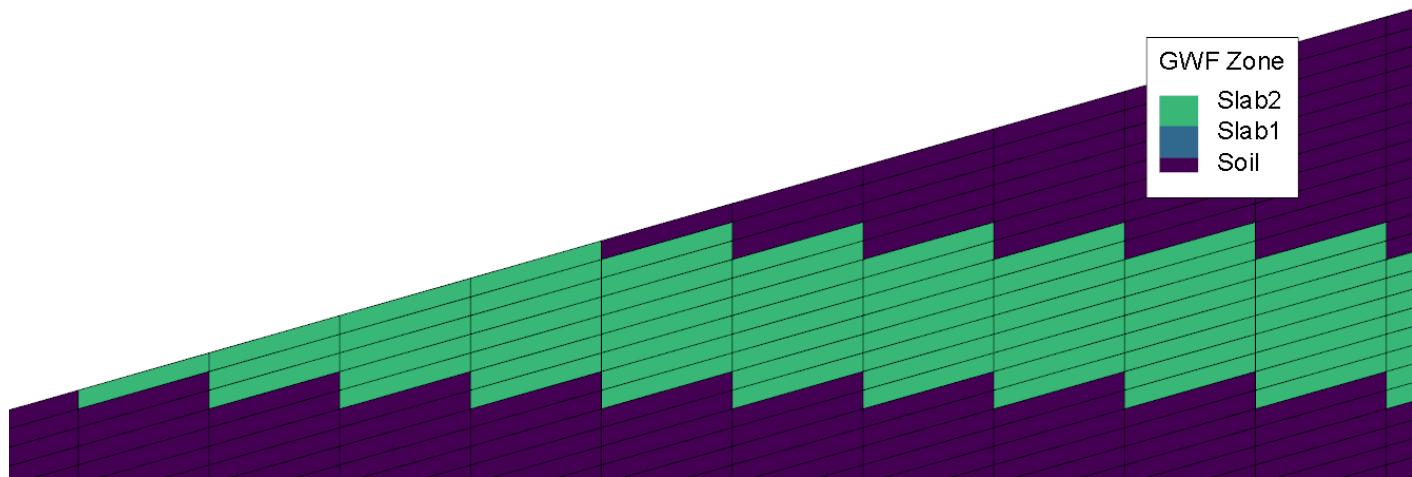
new zone

choose zone number
3

chosen cells use gwf material number
13
```

Some key features to note are:

- The model has only one zone initially, which is first assigned properties of material 11 (Soil) after choosing all cells in the model domain.
- Cells in the range from  $x=40$  to  $x=60$  and from layers 1 to 5 (inclusive) are chosen and assigned properties of material 12 (Slab1). Specifying a range between -1e30 and 1e30 means that no restrictions will be applied based on the  $y$  and  $z$  cell coordinates. Recall that the MODFLOW-USG<sup>Swf</sup> mesh is generated from the top down, so layers 1 to 5 are the top 5 layers of the model domain.
- The first **new zone** instruction increments the total number of zones from 1 to 2 and assigns zone number 2 to the chosen cells.
- Cells in the range from  $x=8$  to  $x=50$  and from  $z=5.8$  to  $z=6.2$  are chosen and assigned properties of material 13 (Slab2). Specifying a range between -1e30 and 1e30 for means that no restrictions will be applied based on the  $y$  cell coordinates, while a range of 0 to 1000 means that no layer restrictions are being applied.
- Because of the sloping nature of the mesh layers, the cells chosen for zone 3 (Slab2) are depicted by TECPLOT as a 'jagged' boundary, as shown here:



- The second **new zone** instruction increments the total number of zones from 2 to 3 and assigns zone number 3 to the chosen cells.
- In this example, zone and cell selections are cleared before defining new zones and assigning material properties, but note that multiple cell selections can be made without clearing to define more complex zones.

### 3.3.1.6 Initial Conditions

An initial (or starting) head should be assigned to each cell in the GWF domain. This could be an initial guess at the beginning of a transient stress period or a set of hydraulic heads from a previous run.

To assign a uniform hydraulic head to the GWF model domain, you must first make a cell selection as described on page [39](#), then use this instruction:

## gwf initial head

1. **IHead<sub>real</sub>** [L] Initial (or starting) hydraulic head.

An initial hydraulic head of **IHead<sub>real</sub>** is assigned to the chosen cells.

As a reminder, MUT will echo the assumed units to the screen and o.eco file:

```
gwf initial head
Assigning all chosen GWF cells starting heads of      2.7800      METERS
```

By default, the instruction **postprocess existing modflow model** writes hydraulic heads at the last output time to a csv file called **GWF\_FinalHeads.csv**. This file can be used to start a subsequent MODFLOW-USG<sup>Swf</sup> model simulation by first activating the **GWF** domain then using this instruction:

### initial head from csv file

1. **FName<sub>str</sub>** The hydraulic head solution in the file **FName<sub>str</sub>** will be assigned as the initial head in the currently active domain.

The following example uses the instruction to start a simulation with GWF heads from a previous run:

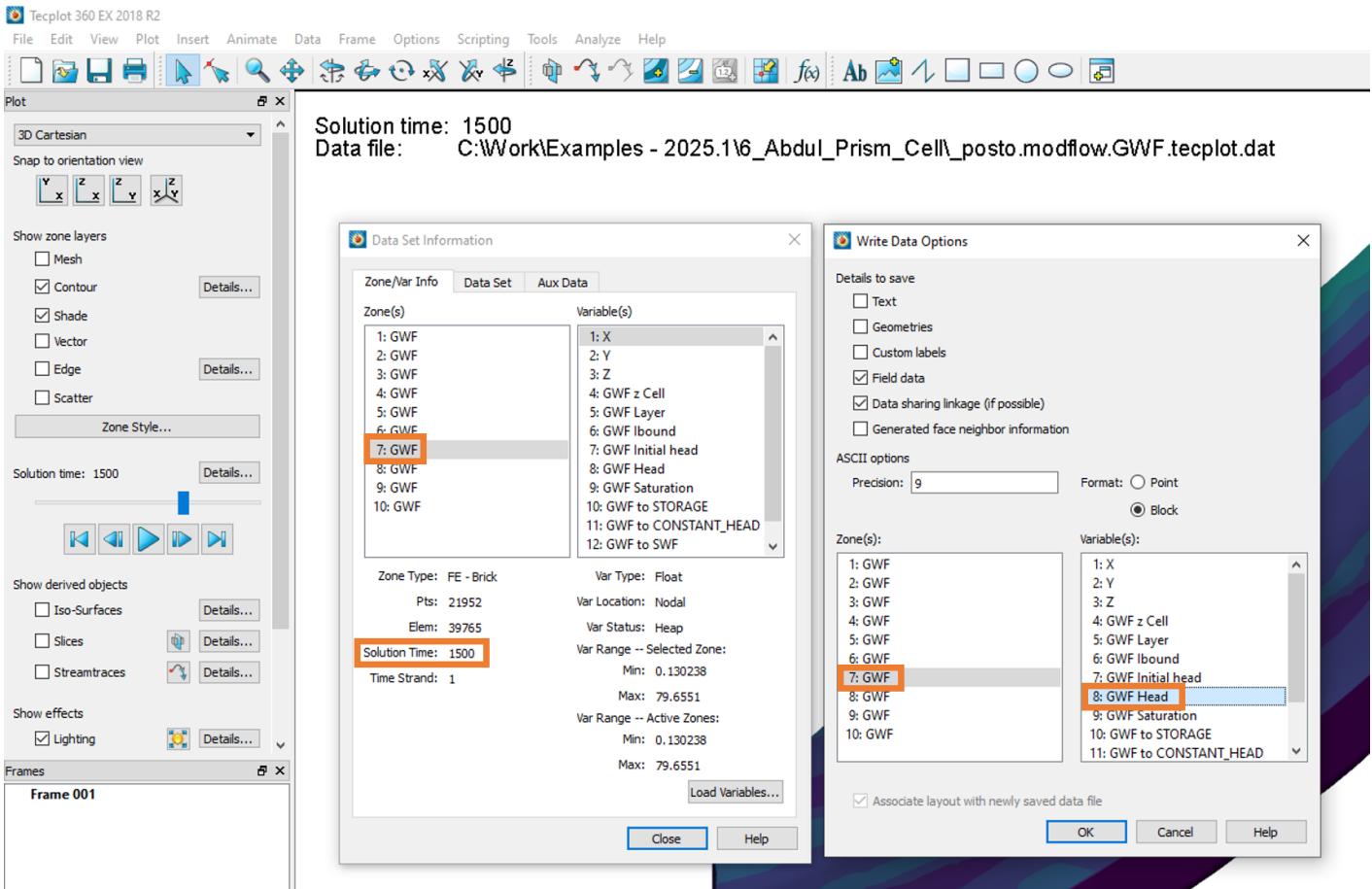
```
active domain
gwf
    initial head from csv file
    ..\1_Regional_Boundary_Average_Rainfall\GWF_FinalHeads.csv
```

The instruction **postprocess existing modflow model** also writes heads at all output times to a TECPLOT-formatted output file, and earlier times can also be used to start a subsequent MODFLOW-USG<sup>Swf</sup> model simulation by first activating the **GWF** domain then using this instruction:

### gwf initial head from tecplot file

1. **FName<sub>str</sub>** The hydraulic head solution in the file **FName<sub>str</sub>** will be assigned as the initial head in the currently active domain.

This file can be created using TECPLOT, as shown in the example below:



Note the following:

- The post-processed data file `_posto.modflow.GWF.tecplot.dat` for the example `6_Abdul_Prism_Cell` has been loaded into TECPLOT.
- The Data Set Information dialogue is open showing Zone 7:GWF corresponds to Solution Time 1500.
- The Write Data dialogue is open and configured to write Field data using Format: Block from Zone(s): 7:GWF, Variable(s): 8: GWF Heads.
- The data is being written to a file called `Head_1500.GWF.tecplot.dat`, shown below.

```

TITLE      = "Modflow Project: modflow"
VARIABLES = "GWF Head"           ← Variable name
ZONE T="GWF"
STRANDID=1, SOLUTIONTIME=1500   ← Solution time
Nodes=21952, Elements=39765, ZONETYPE=FEBrick
DATAPACKING=BLOCK
VARLOCATION=([1]=CELLCENTERED)
AUXDATA LengthUnits="METERS"    ← Solution length units
AUXDATA MUTVersion=" 2025.1 RELEASE" ← MUT Version
AUXDATA TimeUnits="SECONDS"     ← Solution time units
DT=(SINGLE )
3.546358109E+000 3.516053200E+000 3.570491791E+000 3.750024319E+000 3.726
3.815949440E+000 3.401170254E+000 3.469510317E+000 3.436817646E+000 3.350
3.372439623E+000 3.413620949E+000 3.237037182E+000 3.212256432E+000 3.385

```

The code above is a portion of the `Head_1500.GWF.tecplot.dat` file. Annotations highlight specific parameters: 'Variable name' points to 'VARIABLES = "GWF Head"', 'Solution time' points to 'SOLUTIONTIME=1500', 'Solution length units' points to 'LengthUnits="METERS"', 'MUT Version' points to 'MUTVersion=" 2025.1 RELEASE"', 'Solution time units' points to 'TimeUnits="SECONDS"', and 'Start of block formatted head data' points to the first data row.

For earlier output times, the post-processed heads can be By default, the instruction `postprocess` existing `modflow` model writes hydraulic heads at the last output time to a `csv` file called `GWF_FinalHeads.csv`. This file can be used to start a subsequent MODFLOW-USG<sup>Swf</sup> model simulation by first activating the GWF domain then using this instruction:

To assign an initial head equal to the local ground surface elevation use this instruction:

### **gwf initial head equals surface elevation**

For each column of cells in the GWF domain, each cell in the column is assigned an initial head equal to the top elevation (i.e. ground surface elevation) of the cell at the top of the column.

This is commonly used as an initial guess when starting a simulation with no previous head solution.

To assign a linearly varying head that is a function of  $z$  (i.e. depth or elevation), use this instruction:

### **gwf initial head function of z**

1. `z(1)real` [ $L$ ], `Head(1)real` [ $L$ ] First  $z, head$  pair.
2. `z(2)real` [ $L$ ], `Head(2)real` [ $L$ ] Second  $z, head$  pair.
3. ...
- n. `z(n)real` [ $L$ ], `Head(n)real` [ $L$ ] nth  $z, head$  pair.

An initial head is calculated for each chosen cell, based on it's  $z$ -coordinate location, by interpolating a head from the given list of  $z, head$  pairs.

An end instruction is required to stop the subtask e.g.:

### **end gwf initial head function of z**

This is commonly used to generate an initial head for a simple column model<sup>19</sup> as shown here:

```
gwf initial head function of z
! z      head
 0.0  -100.0
100.0     0.0
```

Some key features of this example are:

- The two given  $z, head$  pairs define an initial head that varies from  $head = -100.0$  at  $z = 0$  to  $head = 0.0$  at  $z = 100.0$ . You may supply as many pairs as needed to define the initial head.
- $z$ -coordinates must increase continuously from the top of the list to the bottom.
- the  $z$ -range of the supplied pairs should cover the entire  $z$ -range of the model domain.

<sup>19</sup>The example `1_VSF_Column` uses the `gwf initial head function of z` instruction to define the initial head of the model domain.

- For each node in the model domain mesh, the  $z$  coordinate is used to interpolate an initial head (i.e. *head* value) using the appropriate  $z, \text{head}$  pair.

To assign a linearly varying head that is a function of saturation versus depth, use this instruction:

---

### gwf initial head from depth-saturation table

1. dpth(1)<sub>real</sub> [L], sat(1)<sub>real</sub> [L] First depth dpth(1)<sub>real</sub>, saturation sat(1)<sub>real</sub> pair.
2. dpth(2)<sub>real</sub> [L], sat(2)<sub>real</sub> [L] Second depth dpth(2)<sub>real</sub>, saturation sat(2)<sub>real</sub> pair.
3. ...
- n. dpth(n)<sub>real</sub> [L], sat(n)<sub>real</sub> [L] nth depth dpth(n)<sub>real</sub>, saturation sat(n)<sub>real</sub> pair.

For each column of cells in the GWF domain, each local cell in the column is assigned a depth calculated from the top elevation (i.e. ground surface elevation) of the cell at the top of the column minus the local cell elevation. A saturation is then looked up from the depth-saturation table. A pressure head is than interpolated from the pressure-saturation table defined for the GWF material.

An end instruction is required to stop the subtask e.g.:

---

### end gwf initial head from depth-saturation table

---

This was used to generate an initial head for the example simulation 9\_Smith\_Woolhiser as shown here:

```
active domain
gwf
    gwf initial head from depth-saturation table
        0.0          0.156997082747331
        0.14166834633879  0.156997082747331
        1.13525629270851  0.155244035878502
        ...
        45.7047727441498  0.195564113861573
        49.96300680002   0.197317160730402
        1000.            0.197317160730402
    end initial head from depth-saturation table
```

#### 3.3.1.7 Boundary Conditions

To assign boundary conditions to the GWF model domain, you must first make a cell selection as described on page 39.

A constant head boundary condition fixes the head at a GWF cell at a given value, allowing water to flow into or out of the GWF model domain depending on surrounding conditions. To assign a uniform constant head to the GWF model domain use this instruction:

---

### gwf constant head

1. **CHead<sub>real</sub>** [ $L$ ] Constant hydraulic head.

An constant hydraulic head of **CHead<sub>real</sub>** is assigned to the chosen cells.

A drain boundary condition allows water to flow out of the GWF model domain if the hydraulic head of the cell is higher than the drain elevation. To add a drain to the GWF model domain use this instruction:

## gwf drain

1. **DrainK<sub>real</sub>** [ $L T^{-1}$ ] Drain conductance.

A drain conductance of **DrainK<sub>real</sub>** is assigned to the chosen cells. The top elevation of the cell is assigned automatically as the drain elevation

A recharge boundary condition forces water to flow in to the GWF model domain at a specified rate. To add recharge to the GWF model domain use this instruction:

## gwf recharge

1. **RechRate<sub>real</sub>** [ $L T^{-1}$ ] Recharge rate.
2. **RechOpt<sub>int</sub>** Recharge option.

A recharge rate of **RechRate<sub>real</sub>** is assigned to the chosen cells.

The recharge option **RechOpt<sub>int</sub>** is used to define where the recharge is to be applied and can have one of the following values:

1. To top layer
2. To one specified node in each vertical column
3. To highest active node in each vertical column
4. To the swf domain on top of each vertical column

A well boundary condition forces water to flow in or out of the GWF model domain at a specified rate. To add a well to the GWF model domain use this instruction:

## gwf well

1. **PumpRate<sub>real</sub>** [ $L^3 T^{-1}$ ] Pumping rate.

A pumping rate of **PumpRate<sub>real</sub>** is assigned to the chosen cells. Positive pumping rates add water to the domain, negative pumping rates remove water from the domain.

## 3.4 Surface Water Flow(SWF) Domain

The SWF domain is a 2D network of cells which is usually, but not necessarily, coincident with the top of the GWF domain.

MODFLOW-USG<sup>Swf</sup> allows individual (i.e. GWF, SWF and CLN) processes to add to the global conductance matrix in order to represent fluxes between cells within a process as well as with cells of other processes. MODFLOW-USG<sup>Swf</sup> provides a framework for tightly coupling multiple hydrologic processes. The tight coupling, in contrast to a sequential or iterative coupling approach, occurs through the formulation of a global conductance matrix that includes the cells for all processes.

The flows between SWF cells are governed by the diffusion-wave equations, which ultimately provide a pressure head (i.e. surface water depth) at each cell.

### 3.4.1 Generating a SWF Domain

A SWF domain can be easily added to the MODFLOW-USG<sup>Swf</sup> model using the same template mesh that was used to define the GWF mesh, as described in Section 3.2.

The SWF domain is generated using this instruction:

---

#### generate swf domain

This subtask currently has only one instruction that is used to define:

- Top elevation (e.g. ground surface elevation)

An end instruction is required to stop the subtask e.g.:

---

#### end generate swf domain

---

Here, the top elevation instruction has the same options as described in Section 3.3.1.1 for the GWF domain.

Currently, the element zone numbering for the SWF domain is determined by the instruction used to generate the template mesh:

**2d mesh from gb** The GRID BUILDER element area numbers are used to define the MODFLOW-USG<sup>Swf</sup> element zone numbers.

**generate uniform rectangles** The element zone numbers default to 1.

#### 3.4.1.1 Cell Connection Properties

The SWF-SWF lateral cell connection lengths and areas are defined in the same way as they are for the GWF domain (see Section 3.3.1.3), except lateral connection area uses surface water depth instead of cell thickness.

SWF-GWF cell connection properties vary on a cell-by-cell basis:

- By default, the SWF-GWF connection length is assigned a value of 0.001.
- The vertical SWF-GWF connection area is defined in the same way as it is for vertical GWF-GWF cell connections (see Section 3.3.1.3).

You can use this instruction to change the SWF-GWF cell connection length after making a cell selection (see the instructions described on page 39 for the GWF domain):

### **swf to gwf connection length**

1. Sgcl<sub>real</sub> [L] SWF cell to GWF cell connection length.

A SWF cell to GWF cell connection length Sgcl<sub>real</sub> is assigned to the chosen cells.

#### **3.4.1.2 Material Properties**

Unlike the GWF domain, SWF domain material properties vary on a zone-by-zone basis, which means assigning material property values are done using zone selections instead of cell selections.

Prior to making zone selections and assigning properties, we need to activate the SWF domain using these instructions:

```
active domain
swf
```

Zone selections must first be made using the instructions described on page 45 for the GWF domain, then these instructions can be used to assign material properties to the current zone selection:

### **swf manning**

1. Manning<sub>real</sub> [ $L^{-1/3} T$ ]. Manning's coefficient of friction

A Manning's coefficient of Manning<sub>real</sub> is assigned to the chosen cells.

### **swf depression storage height**

1. DHeight<sub>real</sub> [L] Depression storage height.

A depression storage height of DHeight<sub>real</sub> is assigned to the chosen zones.

### **swf obstruction storage height**

1. OHeight<sub>real</sub> [L] Obstruction storage height.

An obstruction storage height of  $\text{OHeight}_{real}$  is assigned to the chosen zones.

## swf depth for smoothing

1.  $\text{Depth1}_{real}$  [L] Depth for smoothing height 1.
2.  $\text{Depth2}_{real}$  [L] Depth for smoothing height 2.

Two depth for smoothing heights are read in  $\text{Depth1}_{real}$  and  $\text{Depth2}_{real}$  and assigned to the chosen zones.

A lookup table of SWF material properties is provided in the file **SWF.csv**, located in the **USERBIN** directory as outlined on page [10](#).

In order for MUT to access the lookup table, you first need to provide a link to this file using the instruction:

## swf materials database

1.  $\text{FName}_{str}$  SWF material properties lookup table file name.

MUT uses the file  $\text{FName}_{str}$  to look up SWF material properties.

You can now assign a full set of SWF material properties to the current zone selection, as described on page [45](#), using this instruction:

## chosen zones use swf material number

1.  $\text{MaterialID}_{int}$  SWF material ID number.

The unique set of SWF material properties with ID number  $\text{MaterialID}_{int}$  is retrieved from the lookup table and assigned to the chosen cells.

The assigned SWF material properties are written to the screen and **.eco** file e.g.:

```
chosen zones use swf material number
Assigning all chosen SWF zones properties of material      4, Grass
Manning's Coefficient:          0.30000      METERS^(-1/3)  SECONDS
Depression Storage Height:    0.10000      METERS
Obstruction Storage Height:   0.0000      METERS
SWF Smoothing Depth 1:        1.00000E-06      METERS
SWF Smoothing Depth 2:        1.00000E-06      METERS
```

You can find detailed information about how to use MICROSOFT EXCEL to modify or define your own lookup tables in Appendix [B](#).

### 3.4.1.3 Initial Conditions

An initial (or starting) head should be assigned to each cell in the SWF domain. This could be an initial guess at the beginning of a transient stress period or a set of hydraulic heads from a previous run.

To assign an initial head to the SWF model domain, you must first make a cell selection as described on page [39](#), then this instruction can be used to calculate an initial (or starting) head for the flow solution given an initial surface water depth:

#### swf initial depth

1. **IDepth<sub>real</sub>** [*L*] Initial depth.

An initial depth of **IDepth<sub>real</sub>** is used to calculate an initial head at each of the chosen cells.

By default, the instruction **postprocess existing modflow model** writes hydraulic heads at the last output time to a csv file called **SWF\_FinalHeads.csv**. This file can be used to start a subsequent MODFLOW-USG<sup>Swf</sup> model simulation by first activating the SWF domain then using this instruction:

#### initial head from csv file

1. **FName<sub>str</sub>** The hydraulic head solution in the file **FName<sub>str</sub>** will be assigned as the initial head in the currently active domain.

The following example uses the instruction to start a simulation with SWF heads from a previous run:

```
active domain
swf
    initial head from csv file
    ...\\1_Regional_Boundary_Average_Rainfall\\SWF_FinalHeads.csv
```

The instruction **postprocess existing modflow model** also writes heads at all output times to a TECPLOT-formatted output file, and earlier times can also be used to start a subsequent MODFLOW-USG<sup>Swf</sup> model simulation by first activating the SWF domain then using this instruction:

#### swf initial head from tecplot file

1. **FName<sub>str</sub>** The hydraulic head solution in the file **FName<sub>str</sub>** will be assigned as the initial head in the currently active domain.

This file can be created using TECPLOT, as discussed for the GWF domain on Page [51](#).

### 3.4.1.4 Boundary Conditions

To assign boundary conditions to the SWF model domain, you must first make a cell selection as described on page [39](#).

A constant head boundary condition fixes the head at a SWF cell at a given value, allowing water to flow into or out of the SWF model domain depending on surrounding conditions. To assign a uniform constant head to the SWF model domain use this instruction:

---

#### swf constant head

1.  $\underline{\text{CHead}_{real}}$  [ $L$ ] Constant hydraulic head.

An constant hydraulic head of  $\underline{\text{CHead}_{real}}$  is assigned to the chosen cells.

---

A recharge boundary condition forces water to flow in to the SWF model domain at a specified rate. To add recharge to the SWF model domain use this instruction:

---

#### swf recharge

1.  $\underline{\text{RechRate}_{real}}$  [ $L T^{-1}$ ] Recharge rate.
2.  $\underline{\text{RechOpt}_{int}}$  Recharge option.

A recharge rate of  $\underline{\text{RechRate}_{real}}$  is assigned to the chosen cells.

The recharge option  $\underline{\text{RechOpt}_{int}}$  is used to define where the recharge is to be applied and in this case should be set to a value of 4, which applies the recharge to the SWF domain.

---

A well boundary condition forces water to flow in or out of the SWF model domain at a specified rate. To add a well to the SWF model domain use this instruction:

---

#### swf well

1.  $\underline{\text{PumpRate}_{real}}$  [ $L^3 T^{-1}$ ] Pumping rate.

A pumping rate of  $\underline{\text{PumpRate}_{real}}$  is assigned to the chosen cells. Positive pumping rates add water to the domain, negative pumping rates remove water from the domain.

---

A critical depth boundary condition assigned to a SWF cell allows water to flow out of the SWF model domain at a rate that depends on the surface water depth and a contributing length (i.e. representing the length of the cell side over which the outflow occurs).

One of the following two instructions can be used to assign a critical depth outflow boundary condition to the SWF model domain:

---

### swf critical depth with sidelength1

A critical depth outflow boundary condition is assigned to the chosen cells.

It is assumed that an accurate estimate of the contributing length of a cell can be based on the square root of the cells horizontal area.

---

---

### swf critical depth

A critical depth outflow boundary condition is assigned to the chosen cells.

The contributing length of the cell outflow boundary is calculated from the SWF mesh outer boundary nodes, with each outer boundary node connected to a chosen cell contributing a half-element side length in both directions along the outer boundary.

---

Although `swf critical depth` is less convenient than `swf critical depth with sidelength1`, it does calculate a contributing length that matches the actual length along the SWF mesh outer boundary.

The example `6_Abdul_Prism_Cell` uses the second approach to define the critical depth outflow boundary condition:

```
clear chosen nodes
choose gb nodes
./gb/grid.nchos.Outer boundary nodes
flag chosen nodes as outer boundary

clear chosen cells
clear chosen nodes
choose cells from gb elements
./gb/grid.echoes.Critical depth outlet
swf critical depth
```

Some key features of this example are:

- SWF *nodes* are chosen and flagged to be on the outer boundary with the instruction `flag chosen nodes as outer boundary`.
- SWF *cells* are chosen using a GRID BUILDER chosen *elements* file with the instruction `choose cells from gb elements`. Because this example was generated using the mesh-centred control volume approach, there is a 1-to-1 correspondence between template mesh elements and SWF cells.

In the example `6_Abdul_Prism_Cell.nc`, the node-centred control volume approach is used and the instruction `choose cells from gb nodes` is used instead, because in this case there is a 1-to-1 correspondence between template mesh *nodes* and SWF cells.

The **SWF** and **GWF** meshes that MUT generates inherit node and element information from the template mesh. Currently, you can define node selections using these instructions:

#### choose all nodes

Select all nodes in the active model domain.

#### choose node at xyz

1.  $\underline{x1}_{real}$  [L],  $\underline{y1}_{real}$  [L],  $\underline{z1}_{real}$  [L] An *xyz* coordinate triplet.

The node closest to the given *xyz* coordinate triplet will be chosen.

#### choose gb nodes

1.  $\underline{\text{FName}}_{str}$  The GRID BUILDER chosen node  $\underline{\text{FName}}_{str}$  containing information concerning the status, chosen or not chosen, of each node in the GRID BUILDER model domain.

If a node is flagged as chosen in the GRID BUILDER model domain then the corresponding node will be chosen in the MODFLOW-USG<sup>*Swf*</sup> model domain.

#### clear chosen nodes

Clears the current node selection.

## 3.5 Connected Linear Network(**CLN**) Domain

A CLN domain is a quasi-3D network of modflow cells which are each defined by individual line segments. The flows between CLN cells are governed by either open- or closed-channel flow equations, depending on surrounding conditions, which ultimately provide a pressure head or water depth at each cell.

MUT adds the CLN process equations to the global conductance matrix in order to represent fluxes between cells within the process as well as with cells of other processes.

The current version of MUT has the following limitations for the definition of the CLN domain:

1. General CLN domains, where the cell geometry is independent of the GWF mesh and cell-to-cell connection can be one-to-many or many-to-one are not yet implemented. MUT assumes a 1-to-1 connection exists between CLN cells and GWF layer 1 cells (i.e. top layer).
2. CLN flows to the GWF domain are implemented, but not CLN flows to the SWF domain.

This is sufficient for the short-term purpose of solving the example `3_1_CLN_for_SWF`, which compares the use of a CLN versus an SWF domain for simulating flow in a surface water domain coupled to a GWF domain, but not for solving more general problems of interest.

### 3.5.1 Generating a CLN Domain

A CLN domain can be added to the MODFLOW-USG<sup>Swf</sup> model using this instruction:

---

#### generate cln domain

This subtask is currently limited to defining the CLN domain from a single pair of *xyz* coordinates and a specified number of cells.

An end instruction is required to stop the subtask e.g.:

---

#### end generate cln domain

This instruction can be used to define a simple CLN domain:

---

#### cln from xyz pair

1. x1<sub>real</sub> [L], y1<sub>real</sub> [L], z1<sub>real</sub> [L] First *xyz* coordinate triplet.
2. x2<sub>real</sub> [L], y2<sub>real</sub> [L], z2<sub>real</sub> [L] Second *xyz* coordinate triplet.
3. nCells<sub>int</sub> Number of cells in the CLN.

The 2 given *xyz* coordinates define the endpoints of a line defining the CLN. The CLN is subdivided into nCells<sub>int</sub> individual cells.

In the example 3\_1\_CLN\_for\_SWF, the inputs are defined so the CLN cells coincide exactly with the top layer of GWF cells as shown below:

```
generate uniform rectangles
101.0, 101, -0.5 ! Mesh length in X, nRectElements in X, X Offset
1.0, 1, -0.5   ! Mesh length in Y, nRectElements in Y, Y Offset
```

```
generate layered gwf domain
```

```
    top elevation
        elevation from xz pairs
            -100.0, 3.0
            200.0, 0.0
        end elevation from xz pairs
    end top elevation
```

```
    new layer
        layer name
        Top layer
```

```
    uniform sublayering
```

```

1

elevation from xz pairs
-100.0, 2.0
200.0, -1.0
end elevation from xz pairs
end new layer

end generate layered gwf domain

generate cln domain
cln from xyz pair
-.5    0.0    2.005
100.5   0.0    0.995
101 ! number of new CLN cells

end generate cln domain

```

Some key features of this example are:

- A template mesh of length 101.0 and with 101 elements is used to define the **GWF** domain.
- The mesh is offset in  $x$  by -0.5 m so that **GWF** and **CLN** cell locations start at  $x = 0.0$  and end at  $x = 100$  m.
- The top of the **GWF** domain slopes from  $z = 3.0$  at  $x = -100.0$  to  $z = 0.0$  at  $x = 200.0$ . These are extended beyond the limits of the mesh so that the correct elevations are generated at the limits of the domain. Recall that the  $x$ -range must cover the entire template mesh.
- Because **CLN** domains are not necessarily dependent on **GWF** meshes, it does not use the template mesh, but instead generates a **CLN** domain using the **cln from xyz pair** instruction. The instruction is set up to generate 101 **CLN** cells that slope from  $z = 2.005$  at  $x = -.5$  to  $z = 0.995$  at  $x = 100.5$

### **cln from xyz list**

This subtask creates a **CLN** domain from a list of  $xyz$  coordinates.

1. **x1<sub>real</sub> [L], y1<sub>real</sub> [L], z1<sub>real</sub> [L]** An  $xyz$  coordinate triplet.

An end instruction is required to stop the subtask e.g.:

**end cln from xyz list**

### 3.5.1.1 Cell Connection Properties

CLN domain cells are currently limited to have a direct one-to-one correspondence with cells in the top layer of the GWF domain, as is the case for the simple example `3_1_CLN_for_SWF`. A more general approach, in which CLN cells do not have to conform to the GWF or SWF meshes will be developed in a subsequent version of MUT.

CLN-CLN cells are connected through neighbouring nodes and connection properties are defined by the channel geometry inputs.

CLN-GWF cell vertical connections are defined by element areas.

### 3.5.1.2 Material Properties

CLN domain material properties vary on a zone-by-zone basis. In the current version of MUT the assignment of CLN material properties is very rudimentary.

Prior to assigning properties, we need to activate the CLN domain using these instructions:

```
active domain  
cln
```

A lookup table of CLN material properties is provided in the file `CLN.csv`, located in the `USERBIN` directory as outlined on page [10](#).

In order for MUT to access the lookup table, you first need to provide a link to this file using the instruction:

---

#### cln materials database

1. FName<sub>str</sub> CLN material properties lookup table file name.

MUT uses the file FName<sub>str</sub> to look up CLN material properties.

---

Zone selections must first be made using the instructions described on page [45](#) for the GWF domain.

You can now assign a full set of CLN material properties to the current zone selection, as described on page [45](#), using this instruction:

---

#### chosen zones use cln material number

1. MaterialID<sub>int</sub> CLN material ID number.

The unique set of CLN material properties with ID number MaterialID<sub>int</sub> is retrieved from the lookup table and assigned to the chosen cells.

---

The assigned CLN material properties are written to the screen and `.eco` file:

```

chosen zones use cln material number
Assigning all chosen CLN zones properties of material 1, 2D Hillslope 100 m length
Geometry:           Rectangular
Rectangular Width:    1.0000      METERS
Rectangular Height:   1.0000      METERS
Direction:          Horizontal
Flow Treatment:     Unconfined/Mannings
Longitudinal K:      5.48000E-02    METERS    SECONDS^(-1)

```

You can find detailed information about how to use MICROSOFT ACCESS to modify or define your own lookup tables in Tutorial [B](#).

### 3.5.1.3 Initial Conditions

An initial (or starting) head should be assigned to each cell in the CLN domain. This could be an initial guess at the beginning of a transient stress period or a set of hydraulic heads from a previous run.

To assign an initial head to the CLN model domain, you must first make a cell selection as described on page [39](#), then this instruction can be used to calculate an initial (or starting) head for the flow solution given an initial water depth:

---

#### cln initial depth

1. Depth<sub>real</sub> [L] Initial depth.

An initial depth of Depth<sub>real</sub> is used to calculate an initial head at each of the chosen cells.

---

By default, the instruction `postprocess existing modflow model` writes hydraulic heads at the last output time to a `.csv` file called `CLN_FinalHeads.csv`. This file can be used to start a subsequent MODFLOW-USG<sup>Swf</sup> model simulation by first activating the CLN domain then using this instruction:

---

#### initial head from csv file

1. FName<sub>str</sub> The hydraulic head solution in the file FName<sub>str</sub> will be assigned as the initial head in the currently active domain.

---

The following example uses the instruction to start a simulation with CLN heads from a previous run:

```

active domain
cln
    initial head from csv file
    ..\1_Regional_Boundary_Average_Rainfall\CLN_FinalHeads.csv

```

### 3.5.1.4 Boundary Conditions

To assign boundary conditions to the CLN model domain, you must first make a cell selection as described on page 39.

A constant head boundary condition fixes the head at a CLN cell at a given value, allowing water to flow into or out of the CLN model domain depending on surrounding conditions. To assign a uniform constant head to the CLN model domain use this instruction:

---

#### cln constant head

1. CHead<sub>real</sub> [ $L$ ] Constant hydraulic head.

An constant hydraulic head of CHead<sub>real</sub> is assigned to the chosen cells.

---

A well boundary condition forces water to flow in or out of the CLN model domain at a specified rate. To add a well to the CLN model domain use this instruction:

---

#### cln well

1. PumpRate<sub>real</sub> [ $L^3 T^{-1}$ ] Pumping rate.

A pumping rate of PumpRate<sub>real</sub> is assigned to the chosen cells. Positive pumping rates add water to the domain, negative pumping rates remove water from the domain.

---

## 3.6 Inactive Modflow-Usg<sup>Swf</sup> Cells

In MODFLOW-USG<sup>Swf</sup>, inactive cells are grid cells that are excluded from the groundwater flow simulation. These cells are marked as inactive to keep them out of the computational domain, meaning no calculations or flow processes are performed in those cells. This is useful for representing areas outside the modeled region or areas where groundwater flow is not relevant, such as dry land or impermeable regions. Using inactive cells helps improve model performance and accuracy by focusing computations only on the active parts of the model domain.

The following instruction can be used to make a selected set of cells inactive:

---

#### flag chosen cells inactive

All chosen cells will be flagged as inactive.

---

## 3.7 Observation Points

Observation points are used to output cell properties such as simulated hydraulic head or surface water depth during a numerical simulation. The following instructions can be used to define new observation points:

### observation point

1. Name<sub>str</sub> Observation point name Name<sub>str</sub>.
2. x<sub>real</sub> [L], y<sub>real</sub> [L], z<sub>real</sub> [L]. Observation point location as xyz triplet.

The example below<sup>20</sup> shows observation points being defined for both the GWF and SWF domains:

```
active domain
gwf
  observation point
    Outlet
    69.651031494140625  3.1029424667358398  2.941847562789917

  observation point
    Hilltop
    15.270270347595215  0.45934233069419861  4.0170412063598633

active domain
swf
  observation point
    Outlet
    69.651031494140625  3.1029424667358398  2.9918475151062012
```

These coordinates are mesh cell locations that were determined using the TECPLOT Probe Tool.

The following instruction reads observation point data from a `csv` file:

### observation points from csv file

1. FName<sub>str</sub> Observvtion points `csv` file name.

MUT uses the file FName<sub>str</sub> to read the names and *xyz* coordinates of the observation points.

The contents of a `csv` file containing observation point data that was created in QGIS are shown below:

<sup>20</sup>See example `6_Abdul_Prism_Cell`

```

id,name,xcoord,ycoord,Hillshade
1,Mountain_Top,230457.7389,325604.0043,350.26743
2,Western_Outlet,224741.7577,329260.7132,21.24521
4,Central_Basin,231680.3169,321567.5367,50.60401
5,Shaft,232327.1318,324233.1978,109.77948
6,Steep_Slope,230308.2854,325193.6199,263.45529
7,Eastern_Basin,235203.9606,324292.4197,40.30713
8,Eastern_Outlet,235923.3672,330050.4323,30.1562

```

Note the following:

- The first line of the file, which contains the list of field names generated in QGIS, is read and ignored.
- The remainder of the file contains observation point data records (id's, names and *xyz* coordinates) which are added until the end of file is reached.

## 3.8 Stress Periods

A MODFLOW-USG<sup>*Swf*</sup> simulation can be broken up into separate periods of time called "stress periods". Boundary conditions can be defined at the beginning of each stress period and changed in subsequent stress periods.

At least one stress period must be defined using this instruction:

---

### stress period

This subtask has several instructions that can be used to define the duration, type and timestepping parameters of the stress period.

An end instruction is required to stop the subtask e.g.:

---

### end stress period

---

These instructions can be used to define the stress period parameters:

---

### type

1. type<sub>str</sub> Stress period type.

The stress period type is defined by the string type<sub>str</sub>. It can be one of the following:

- SS A steady-state stress period in which the simulation is carried out until it reaches a state of equilibrium with the defined boundary conditions.
- TR A transient stress period in which the simulation is carried out for a specified duration with the defined boundary conditions.

## duration

1. **Duration<sub>real</sub>** [T]. Stress period duration.

The stress period duration, is defined by the string **Duration<sub>real</sub>**. It should be entered using the correct units of time as outlined in Section 3.1.

## number of timesteps

1. **nSteps<sub>int</sub>** Number of timesteps to be used for this stress period.

You can change the default starting time step size of  $1 \times 10^{-3}$  time units with this instruction:

## deltat

1. **StartTStep<sub>real</sub>** [T]. Starting time step size.

The starting time step size used for the stress period is defined by the string **StartTStep<sub>real</sub>**. It should be entered using the correct units of time as outlined in Section 3.1.

You can change the default minimum time step size of  $1 \times 10^{-5}$  time units with this instruction:

## tminat

1. **MinTStep<sub>real</sub>** [T]. Minimum time step size.

The minimum time step size to allow for the stress period is defined by the string **MinTStep<sub>real</sub>**. It should be entered using the correct units of time as outlined in Section 3.1.

You can change the default maximum time step size of 60.0 time units with this instruction:

## tmaxat

1. **MaxTStep<sub>real</sub>** [T]. Maximum time step size.

The maximum time step size to allow for the stress period is defined by the string **MaxTStep<sub>real</sub>**. It should be entered using the correct units of time as outlined in Section 3.1.

You can change the default multiplier for time step size of 1.1 with this instruction:

## tadjat

1. **TStepMult<sub>real</sub>**. Multiplier for time step size.

The multiplier for adjusting time step size when using adaptive time-stepping is defined by **TStepMult<sub>real</sub>**.

You can change the default divider for time step size of 2.0 with this instruction:

## tcutat

1. **TstepCut<sub>real</sub>**. Divider for time step size.

The divider for adjusting time step size when using adaptive time-stepping is defined by **TstepCut<sub>real</sub>**.

To add more stress periods, repeat the **stress period** subtask instructions and boundary condition definitions as many times as required. Stress periods are numbered automatically as they are added.

Here is an example which could be used to define two stress periods:

```
! stress period 1
stress period
    type
    TR

    duration
    3000.0d0
end stress period

active domain
swf
    choose all cells
    swf recharge
    5.56d-6
    4

    clear chosen nodes
    choose cell at xyz
    0.0 0.0 0.0
    swf constant head
    1.0

! stress period 2
stress period
    type
    TR
```

```

duration
3000.0d0
end stress period

active domain
swf
choose all cells
swf recharge
0.0d0
4

```

Some key features of this example are:

- Both stress periods are transient (type TR) with a duration of 3000 time units.
- The recharge applied to the SWF domain (recharge option 4) is 5.5e-6 for the first stress period, then is reduced to 0.0 in the second stress period.
- The constant head applied to the SWF domain in stress period 1 is maintained for the entire simulation. By default, a boundary condition is maintained through subsequent stress periods unless it is redefined.
- Any boundary conditions given after an `end stress period` instruction apply to that stress period until another `stress period` instruction is encountered.

## 3.9 Output Control

This instruction can be used to generate a MODFLOW-USG<sup>*Swf*</sup> output control file:

---

### generate output control file

1. t(1)<sub>real</sub> [T]. First output time.
2. ...
- n. t(n)<sub>real</sub> [T]. nth output time.

An end instruction is required to stop the subtask e.g.:

---

### end generate output control file

---

The example `1_Abdul_prism_cell` generates an output control file with 10 output times using these instructions:

```

! -----Output Control
generate output control file

```

```

1e-4
60.
300.0
600.0
900.0
1200.0
1500.0
3000.0
4500.0
6000.0
end generate output control file

```

The output control file looks like this:

```

# MODFLOW-USG OC file written by Modflow-User-Tools version 1.28
ATSA NPTIMES    10
 9.99999747378752E-005    60.00000000000000    300.000000000000
 600.00000000000000    900.00000000000000    1200.000000000000
 1500.00000000000000    3000.00000000000000    4500.000000000000
 6000.000000000000
HEAD SAVE UNIT    114
HEAD PRINT FORMAT 0
DRAWDOWN SAVE UNIT    115
DRAWDOWN PRINT FORMAT 0
PERIOD    1
  DELTAT    1.0000E-03
  TMINAT    1.0000E-05
  TMAXAT    60.00
  TADJAT    1.100
  TCUTAT    2.000
    SAVE HEAD
    PRINT HEAD
    SAVE DRAWDOWN
    SAVE BUDGET
    PRINT BUDGET
PERIOD    2
  DELTAT    1.0000E-03
  TMINAT    1.0000E-05
  TMAXAT    60.00
  TADJAT    1.100
  TCUTAT    2.000
    SAVE HEAD
    PRINT HEAD
    SAVE DRAWDOWN
    SAVE BUDGET
    PRINT BUDGET

```

Some key features of this example are:

- MUT automatically inserts the adaptive time-stepping option (ATSA) in the file, defines the number of print times in the simulation (NPTIMES 10) and the list of print (i.e. output) times.
- Two stress periods were defined and the listed parameters are using the default values.

## 3.10 Solver Parameters

A lookup table of MODFLOW-USG<sup>Swf</sup> solver parameters is provided in the file **SMS.csv**, located in the **USERBIN** directory as outlined on page [10](#).

In order for MUT to access the lookup table, you first need to provide a link to this file using the instruction:

---

## sms database

1. FName<sub>str</sub> Solver parameters lookup table file name.

MUT uses the file FName<sub>str</sub> to look up the solver parameter values.

---

You can now assign the full set of solver parameters using this instruction:

---

## sms parameter set number

1. ParameterSetID<sub>int</sub> Solver parameter set ID number.

The unique set of solver parameter values with ID number ParameterSetID<sub>int</sub> is retrieved from a lookup table.

---

You can find detailed information about how to use MICROSOFT ACCESS to modify or define your own lookup tables in Tutorial [B](#).

Currently, all of the examples use default solver parameters, which are defined in the input file as shown in this example:

```
sms database
SMS.csv
sms parameter set number
1
```

The solver parameter values are written to the screen and .eco file:

```
sms parameter set number
Using SMS parameter set 1, Default
OUTER ITERATION CONVERGENCE CRITERION (HCLOSE) 1.00000E-03 METERS
INNER ITERATION CONVERGENCE CRITERION (HICLOSE) 1.00000E-05 METERS
MAXIMUM NUMBER OF OUTER ITERATIONS (MXITER) 250
MAXIMUM NUMBER OF INNER ITERATIONS (ITER1) 600
SOLVER PRINTOUT INDEX (IPRSMS) 1
NONLINEAR ITERATION METHOD (NONLINMETH) 1
LINEAR SOLUTION METHOD (LINMETH) 1
D-B-D WEIGHT REDUCTION FACTOR (THETA) 0.70000
D-B-D WEIGHT INCREASE INCREMENT (KAPPA) 0.10000
D-B-D PREVIOUS HISTORY FACTOR (GAMMA) 0.10000
MOMENTUM TERM (AMOMENTUM) 0.0000
MAXIMUM NUMBER OF BACKTRACKS (NUMTRACK) 200
BACKTRACKING TOLERANCE FACTOR (BTOL) 1.0000
```

BACKTRACKING REDUCTION FACTOR	(BREDUC)	0.20000
BACKTRACKING RESIDUAL LIMIT	(RES_LIM)	1.0000
TRUNCATED NEWTON FLAG	(ITRUNCNEWTON)	0
Options	SOLVEACTIVE	DAMPBOT
ACCELERATION METHOD	(IACL)	1
EQUATION ORDERING FLAG	(NORDER)	0
LEVEL OF FILL	(LEVEL)	7
MAXIMUM NUMBER OF ORTHOGONALIZATIONS	(NORTH)	14
INDEX FOR USING REDUCED SYSTEM	(IREDSYS)	0
RESIDUAL REDUCTION CONVERGE CRITERION	(RRCTOL)	0.0000
INDEX FOR USING DROP TOLERANCE	(IDROPTOL)	1
DROP TOLERANCE VALUE	(EPSRN)	1.00000E-03

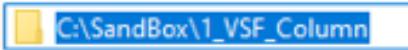
## 3.11 3D Model Build Visualization

The current version of MUT writes Tecplot-compatible output files that can be visualized to check the following attributes for GWF, SWF and CLN model domains:

- Finite-element mesh and MODFLOW-USG<sup>Swf</sup> cell locations derived from it
- Material properties
- Initial conditions
- Boundary conditions (if specified)

A TECPLOT layout file, `_build.1ay`, has been created for each example and provides a quick way to view the results of the model build. We will demonstrate some basic concepts using the example `1_VSF_Column`, which has a GWF domain defined by a simple 1D column mesh with boundary conditions assigned to the top and bottom cells.

To load `_build.1ay` in TECPLOT first navigate to the folder in File Explorer (e.g. C:\Sandbox\1\_VSF\_Column) then highlight the path in File Explorer:



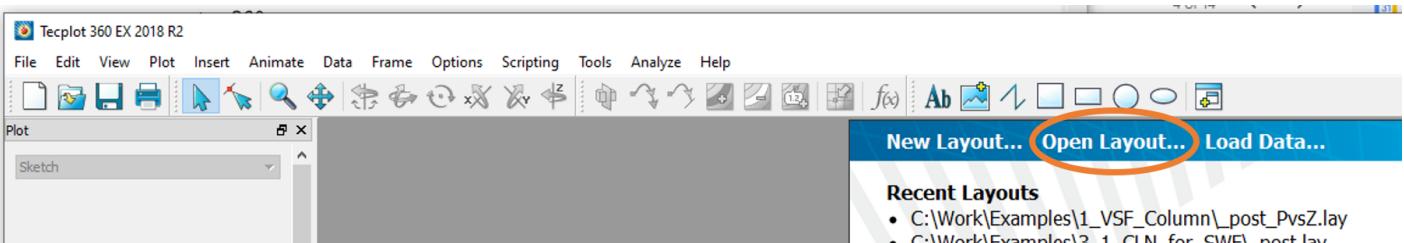
Replace the existing path with the string `cmd`:



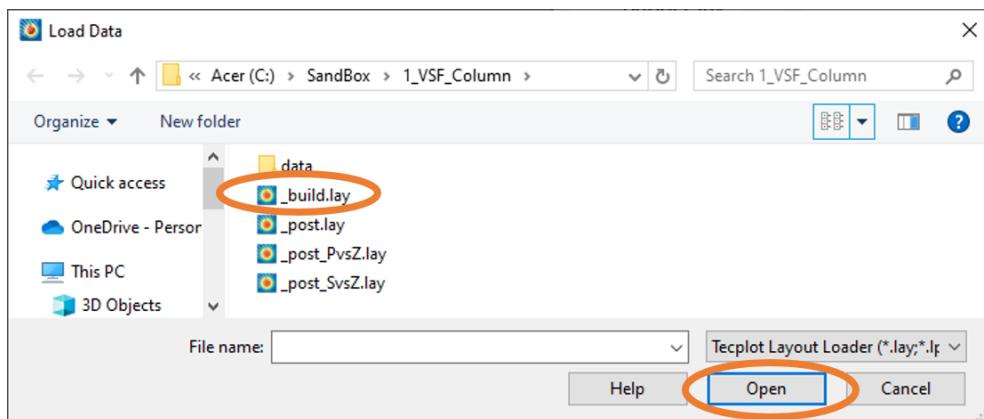
Press `Enter/Return`. A command prompt window rooted at the input folder should appear. To start TECPLOT type:

```
tec360
```

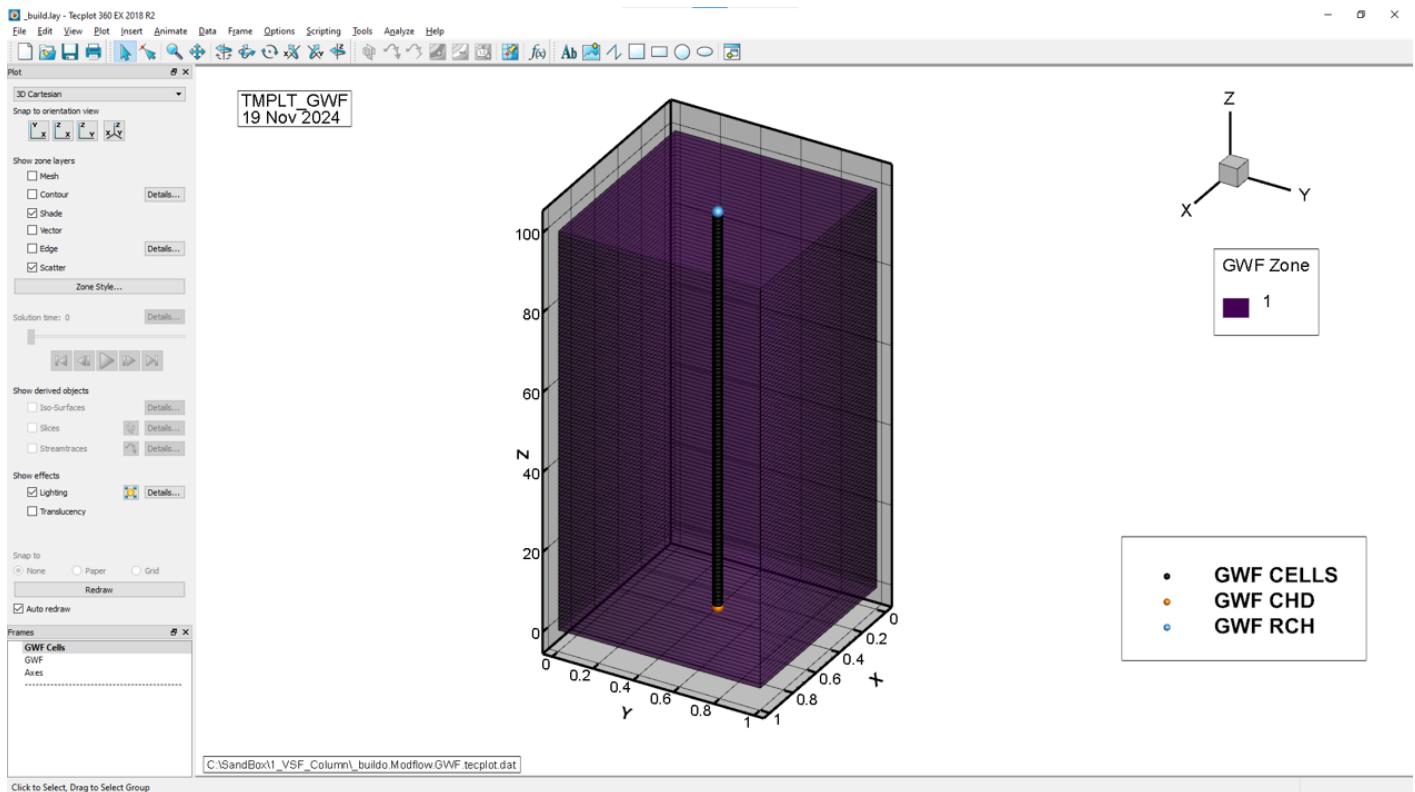
Choose Open Layout to open a file selection dialogue:



Select and open the file \_build.lay:

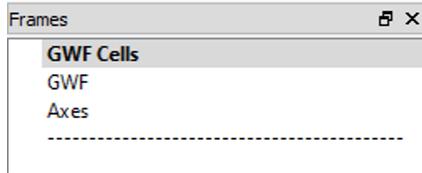


You should now see the following 3D visualization of the example:



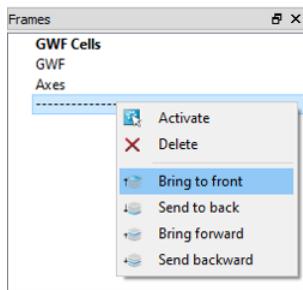
There are 4 Tecplot Frames that make up this image. Each frame can house it's own data for plotting, and have unique settings for visualization. The Frames window at the bottom left corner shows the

frame names and plotting order:



The frame at the front, called **GWF Cells**, is at the top of the list, and the bold font indicates it is the currently active frame.

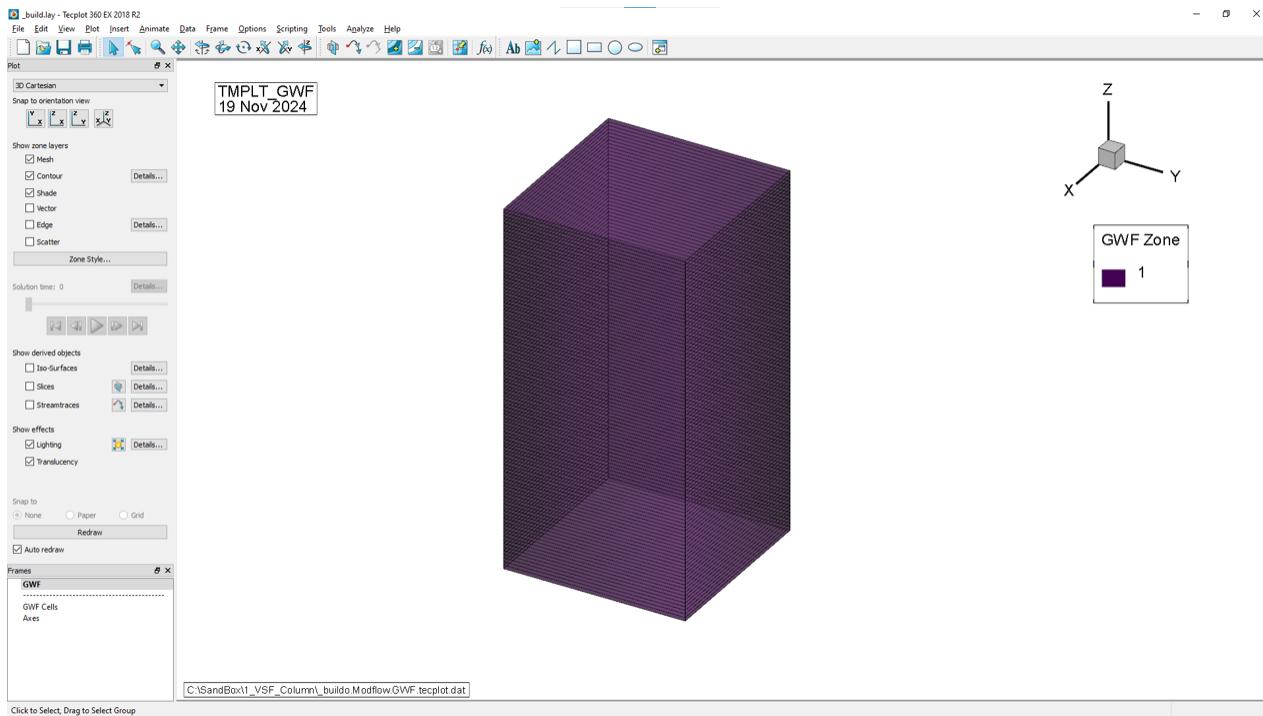
The frame at the bottom, indicated by the dashed line, is a special frame we will refer to as the **background**. The contents of any frame above it may be partly or completely visible, depending on which other frames are in front of it. Move the **background** to the front by right-clicking on the name and selecting 'Bring to front'.



You should now see an empty white TECPLT image.

### 3.11.1 GWF Domain

Right-click on the GWF frame and bring it to the front to see it in isolation.

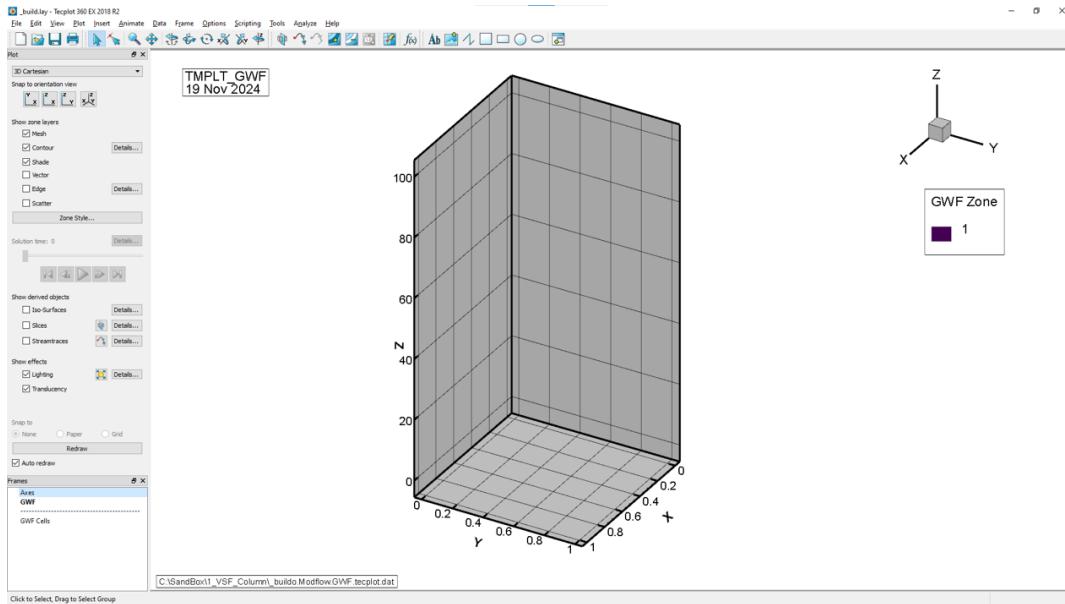


Every frame below the **background** frame is now invisible.

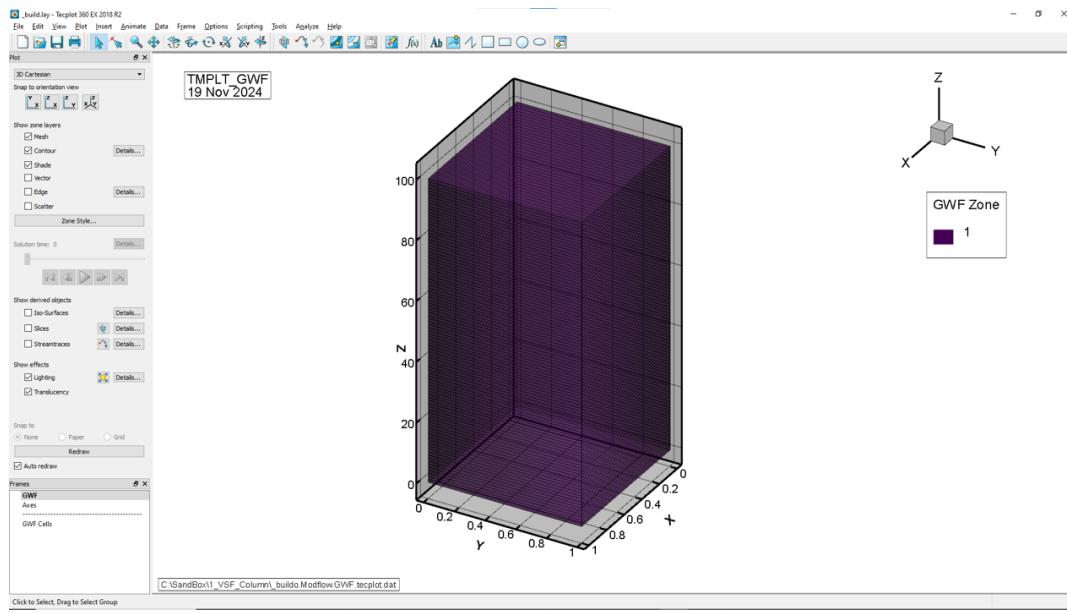
The **GWF** frame has the following contents:

- The finite-element mesh is shown by the translucent blue-shaded volume and wireframe block elements.
- The names of the data files loaded into the frame are shown at the bottom left corner.
- The data set title and current date (on the day the file was loaded) are shown at the top left corner. The data set title **TMPLT\_GWF** indicates that this is the **GWF** domain mesh that was created from the template mesh.
- The contouring legend, showing there is one **GWF** zone, is shown at the middle right side.
- The 3D orientation axis is shown at the top right corner.

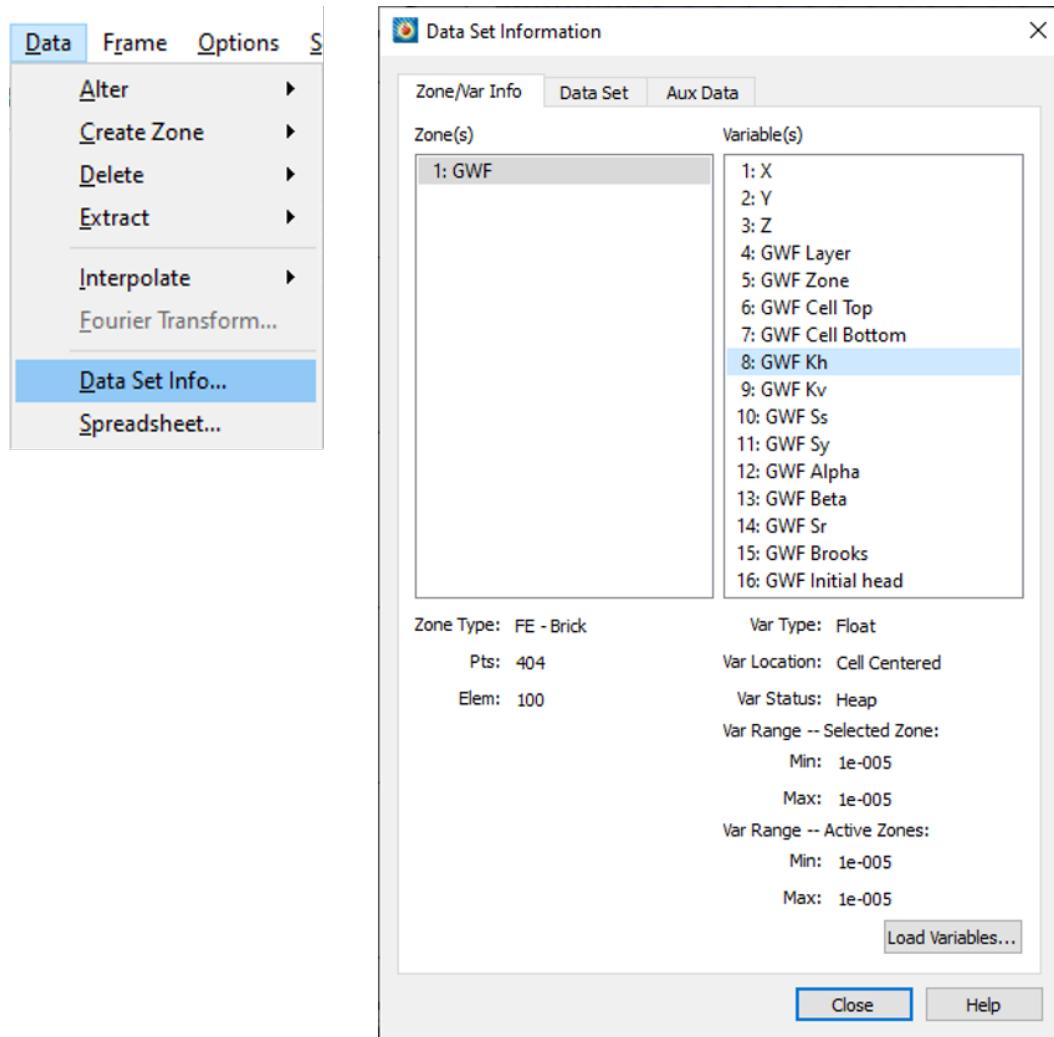
Right-click on the **Axes** frame and bring it to the front.



Some of the contents of the **GWF** frame are still visible but the finite-element mesh is obscured by the axes. Right-click on the **GWF** frame and bring it to the front.



The menu option Data\DataSet Info...(shown below left), brings up the Data Set Information dialogue(shown below right):



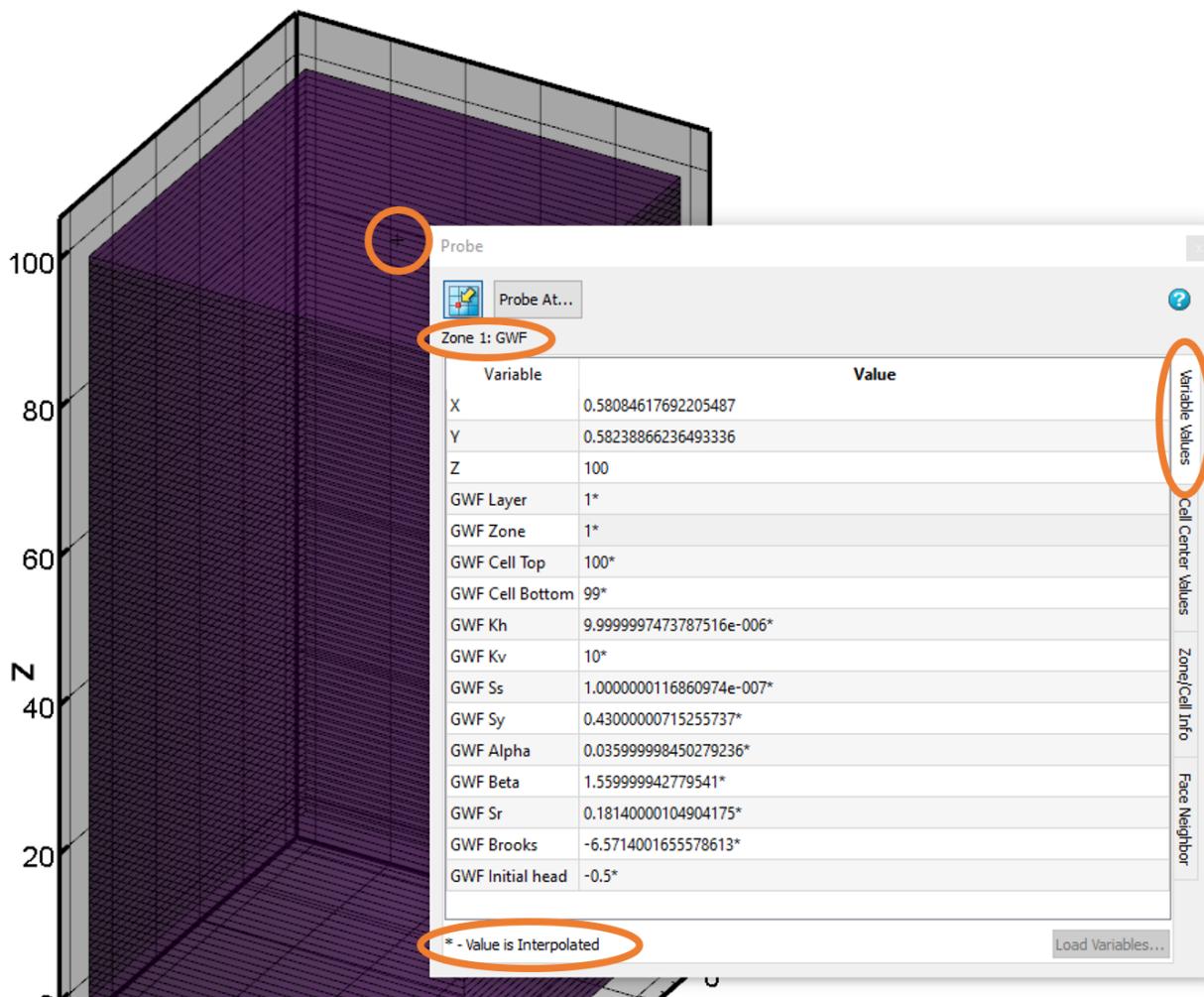
The currently active frame title GWF is shown in the Zone(s) field, while the data set variables are listed in the Variable(s) field. The variables include the *xyz* coordinates of the TMPLT\_GWF mesh nodes, followed by the Layer and Zone numbers assigned to the TMPLT\_GWF mesh elements. The remainder of the list shows the MODFLOW-USG<sup>*swf*</sup> cell properties that were defined during the model build. The Var–Range Selected Zone: area shows the variable Min: and Max: values for the currently chosen (highlighted) zone and variable (currently GWF and GWF Kh respectively).

### TECPLOT Probe Tool



The TECPLOT Probe Tool is used to probe for values of the dataset's variables at a particular point. When selected, the mouse cursor changes to a modified cross-hair which indicates the Probe Tool is active. To obtain *interpolated* values of the dataset variables at the specified location, click at any point in the data region.

Shown below are the results of probing the top cell of the GWF domain:



The probe location is indicated by the small '+' sign in the top cell (upper left orange circle). This opens the Probe dialogue which shows the zone probed (Zone 1: GWF), and a table of Variable names and values.

It is important to understand that in TECPLOT nomenclature, the Variable Values tab refers to values assigned to TMPLT\_GWF mesh nodes, while the Cell Centred Values tab (*Not to be confused with*

`MODFLOW cells!`) refers to values assigned to `TMPLT_GWF` mesh elements.

If the mesh-centred control volume approach is used, as is the case for this example, then MODFLOW cell locations align with `TMPLT_GWF` mesh element centroids, and all values except the *XYZ* coordinates (which are defined at `TMPLT_GWF` mesh nodes) are interpolated (as indicated by an asterisk \* appended to the value). The *XYZ* coordinates show the exact location of the small '+' sign.

If we select the **Cell Centred Values** tab, the Probe output looks like this:

Variable	Value
X	0.5*
Y	0.5*
Z	99.5*
GWF Layer	1
GWF Zone	1
GWF Cell Top	100
GWF Cell Bottom	99
GWF Kh	9.9999997473787516e-006
GWF Kv	10
GWF Ss	1.0000000116860974e-007
GWF Sy	0.43000000715255737
GWF Alpha	0.035999998450279236
GWF Beta	1.559999942779541
GWF Sr	0.18140000104904175
GWF Brooks	-6.5714001655578613
GWF Initial head	-0.5

\* - Value is Interpolated      Load Variables...

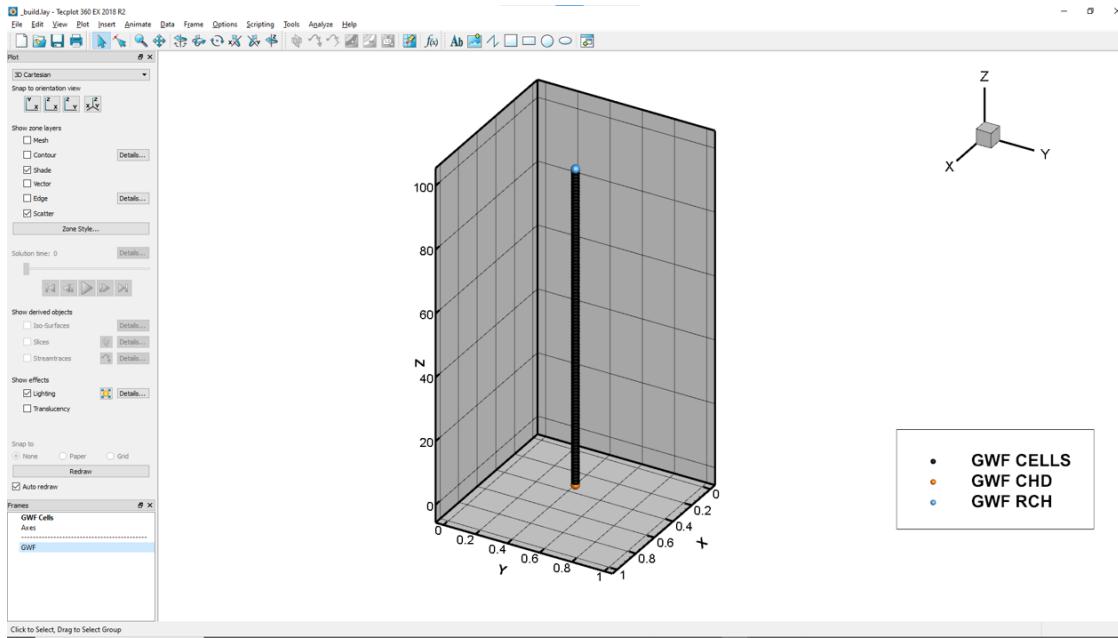
Now the *XYZ* coordinates are interpolated and show the approximate `TMPLT_GWF` mesh element centroid location, while for all other variables, exact values (i.e. what was input) are shown.

If the node-centred control volume approach is used, as is the case for the `6_Abdul_Prism_Cell_nc` example, then MODFLOW cell locations align with `TMPLT_GWF` mesh nodes, and no variable values are interpolated when the **Cell Centred Values** tab is selected:

Variable	Value
X	47.937981085299597
Y	11.83543514715265
Z	3.2684222284199946
GWF Layer	1
GWF Zone	1
GWF Cell Top	3.2684222284199946
GWF Cell Bottom	3.21842227610371
GWF Kh	9.9999997473787516e-006
GWF Kv	9.9999997473787516e-006
GWF Ss	1.199999957179898e-007
GWF Sy	0.34000000357627869
GWF Alpha	1.8999999761581421
GWF Beta	6
GWF Sr	0.18000000715255737
GWF Brooks	-1
GWF Initial head	2.7799999713897705

\* - Value is Interpolated      Load Variables...

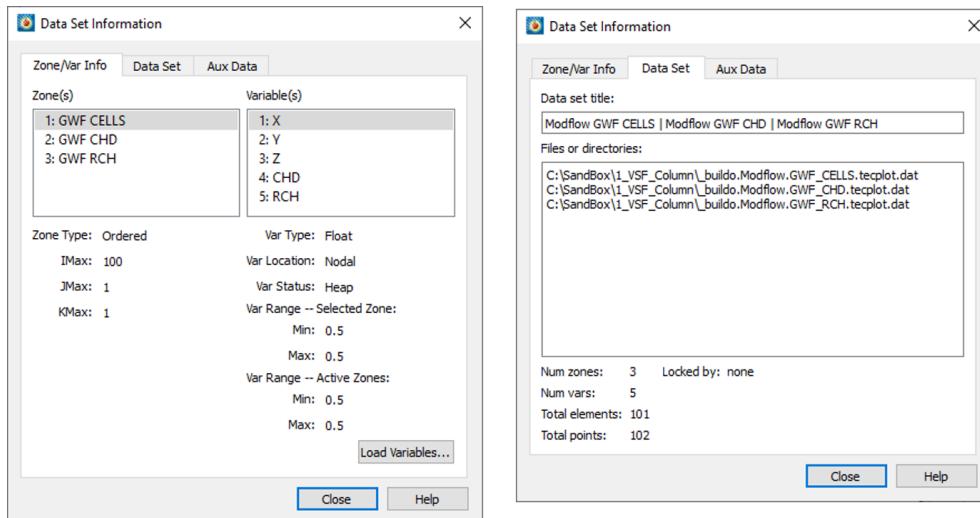
Bring the GWF Cells frame to the front, and send the GWF frame to the back.



The GWF Cells frame has the following contents:

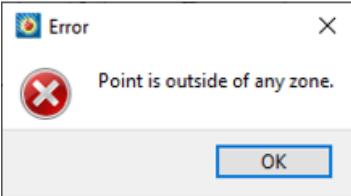
- The legend, which shows three types of scatter points called GWF Cells, GWF CHD and GWF RCH, is shown at the middle right side.
- The MODFLOW-USG<sup>Swf</sup> cell locations are shown by the black spheres.
- The cell assigned recharge is shown by the large blue sphere.
- The cell assigned a constant head is shown by the large green sphere.

If boundary conditions are assigned to the GWF domain, as they are in this case, then Tecplot output files will be produced for each type. The `_build.1ay` file has been configured to load these files:

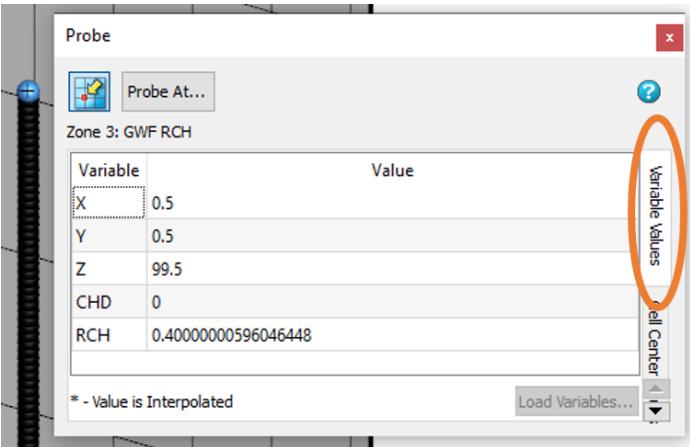


In the Zone/Var Info tab, there are 3 zones, and 5 variables. In the data Set tab, we can see that 3 files have been loaded.

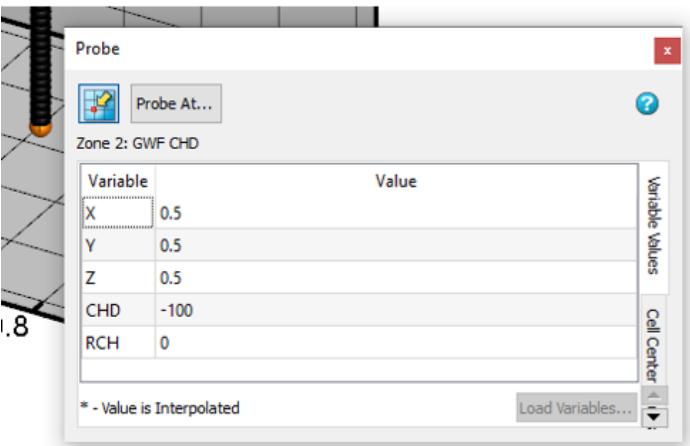
Try to probe the blue sphere, and you will likely get the following warning:



This is because the cell is located at an exact  $XYZ$  point, and the chance of clicking the mouse right there is very small. In this case, to obtain *exact* values for the data point nearest the specified location, hold down the **ctrl** key while clicking the mouse at the desired location.



You must make sure to have the **Variable Values** tab selected to see values. Here, we see the  $XYZ$  location of the nearest cell, and the recharge (RCH) value 0.4. Note that the CHD value of 0 is shown by default at non-constant head nodes, and does not mean a constant head of zero has been assigned. If we probe the green sphere we see this.

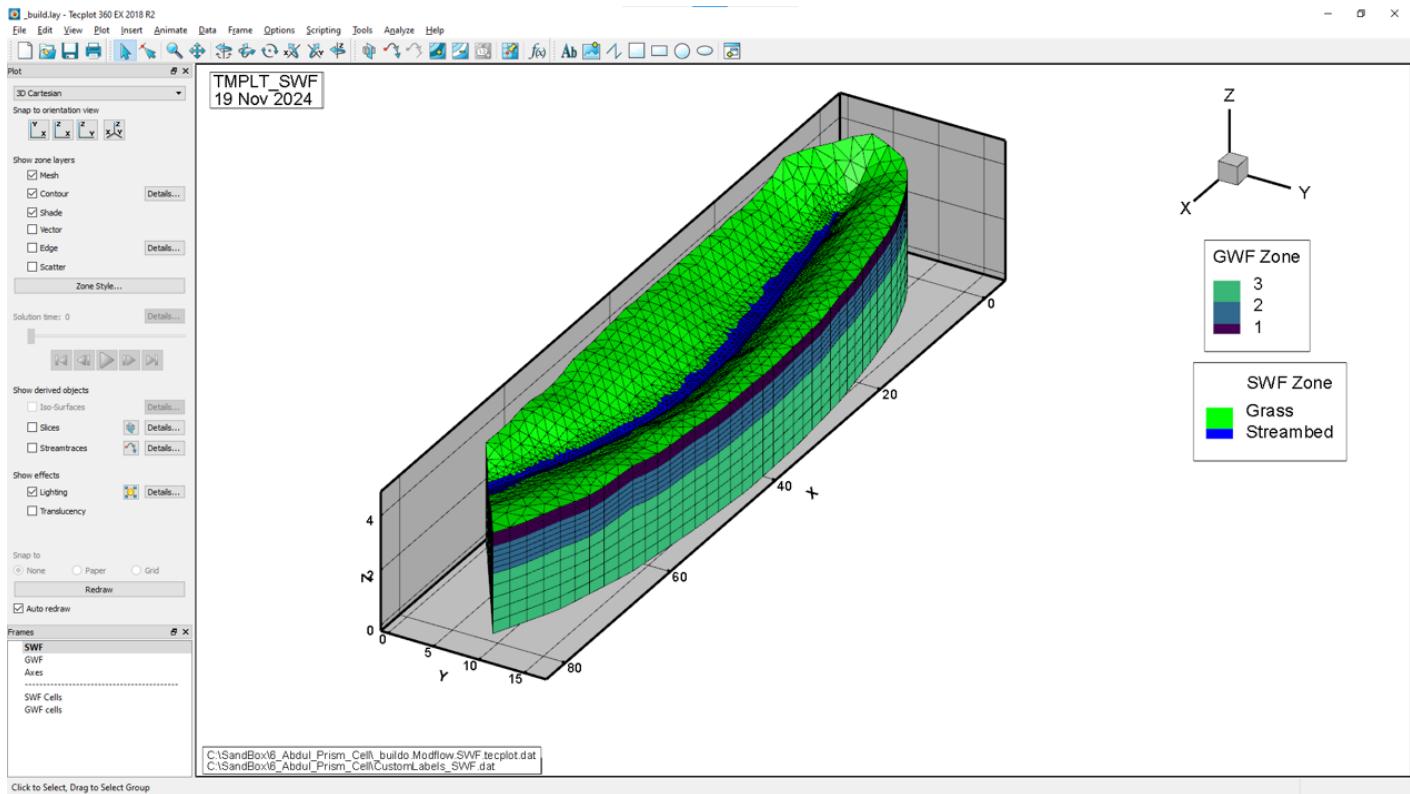


Now the constant head (CHD) value -100 is shown while the (RCH) value is 0, because this is not an assigned recharge cell.

### 3.11.2 SWF Domain

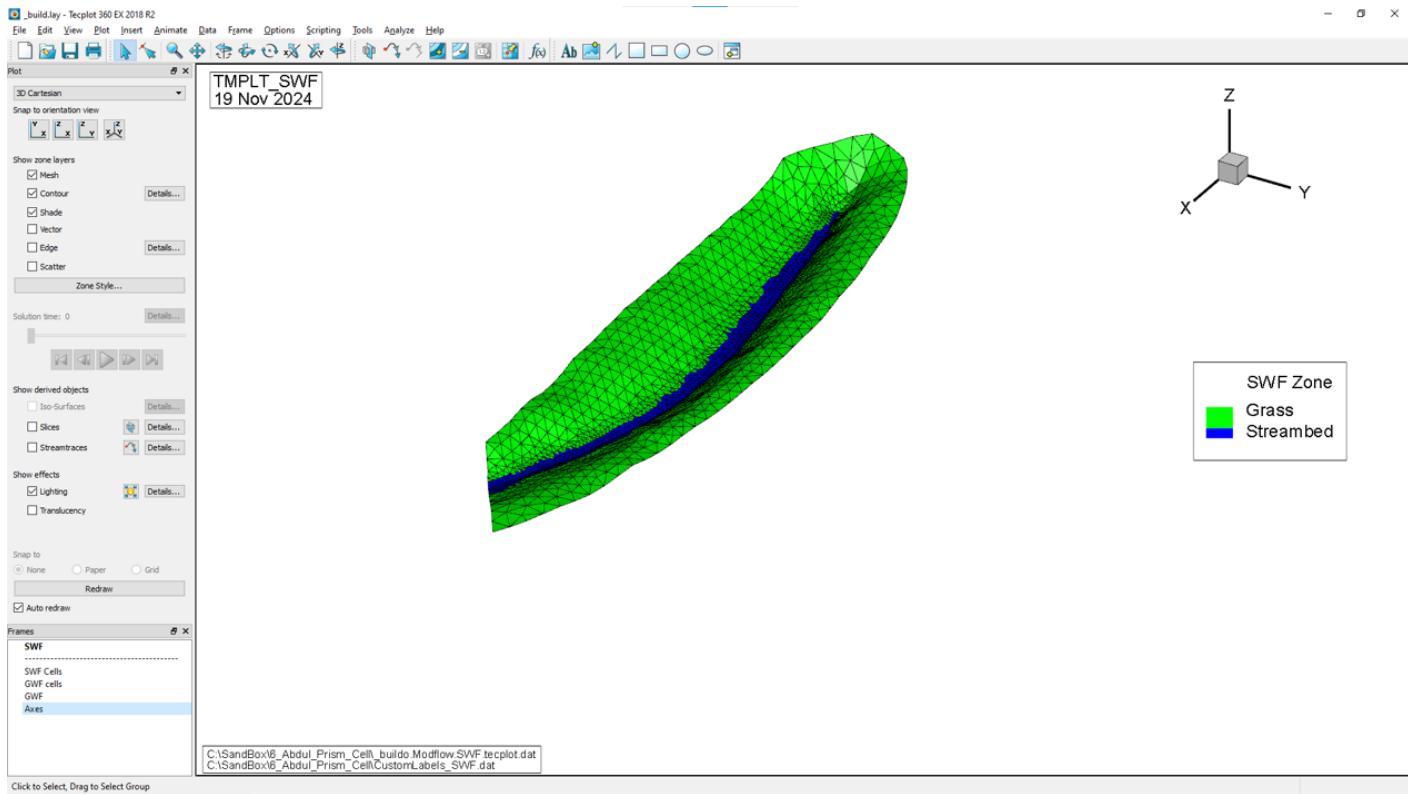
The example 6\_Abdul\_Prism\_Cell has a SWF domain defined by an irregular 2D surface with a recharge boundary condition assigned to the entire domain and a critical depth outflow boundary condition assigned at the downstream outlet. Since it also has a GWF domain the \_build.lay file is a bit more complicated, but has many similarities to the previous example.

Start a new command prompt, run TECPLOT and load the \_build.lay file.



Here we can see the **SWF**, **GWF** and **Axes** frames are visible (i.e. placed above the **background** frame in the list). The **SWF** is currently active (i.e. the name is bolded) and placed at the front of the image (i.e. at the top of the list). It uses a different colormap to make it easier to distinguish the **GWF** domain below.

Send the **GWF** and **Axes** frames to the back to see only the contents of the **SWF** frame.



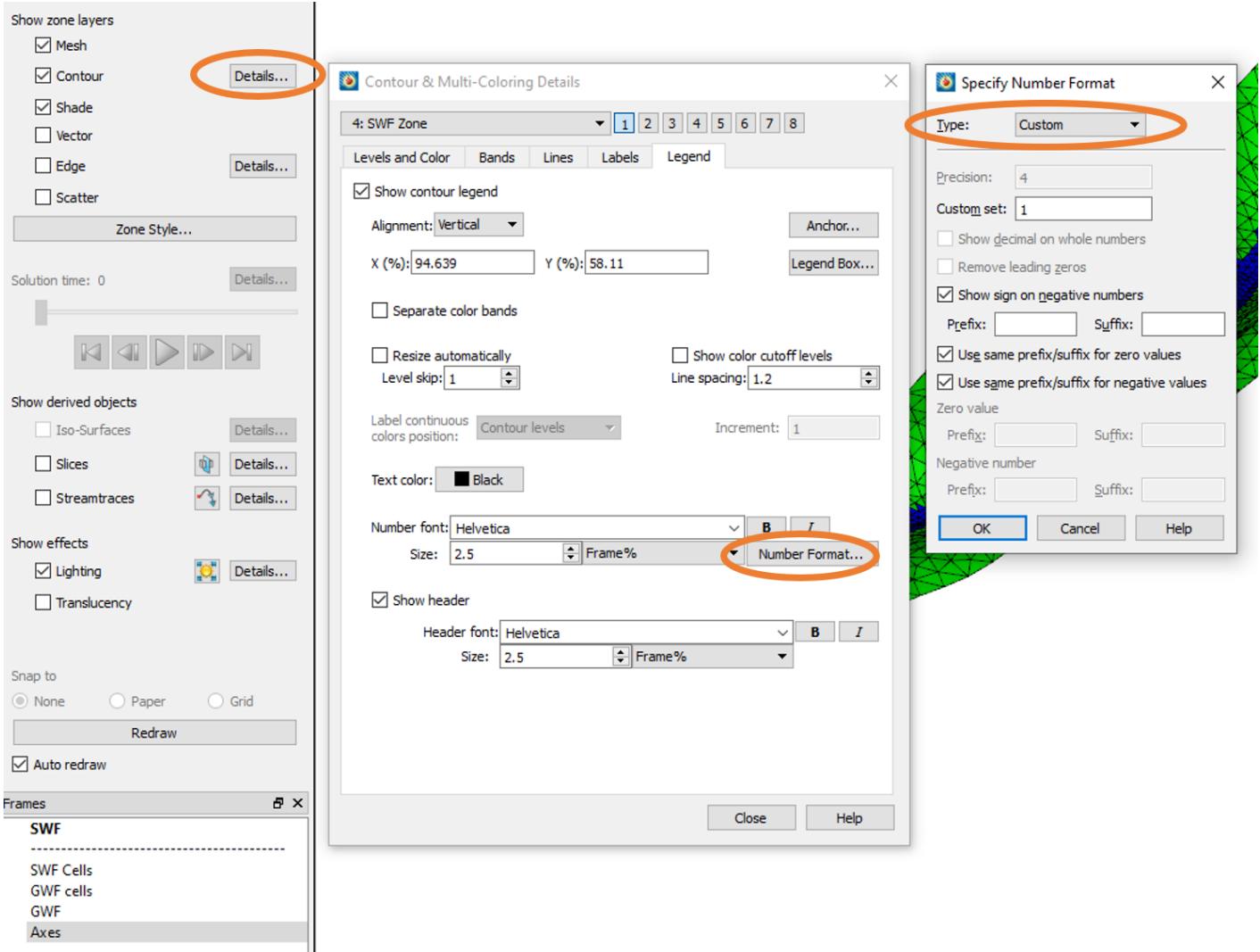
The SWF frame has very similar contents to the GWF frame described earlier, but note that:

- The data set title TMPLT\_SWF indicates that this is the SWF domain mesh that was created from the template mesh.
- The names of the data files loaded into the frame are for the SWF domain, and there is an extra data set that has been loaded called CustomLabels\_SWF.dat.
- The contouring legend, showing there are two SWF zones called Grass and Streambed is a bit more descriptive than a list of zone numbers. These are referred to as custom labels in TECPLLOT.

To define custom labels as shown for the SWF legend, you must create a tecplot file that contains the custom label set. This is in fact the CustomLabels\_SWF.dat file loaded in the SWF frame, which has the following contents:

```
CUSTOMLABELS
"Streambed",
"Grass",
"Zone 3",
"Zone 4",
"Zone 5",
"Zone 6",
```

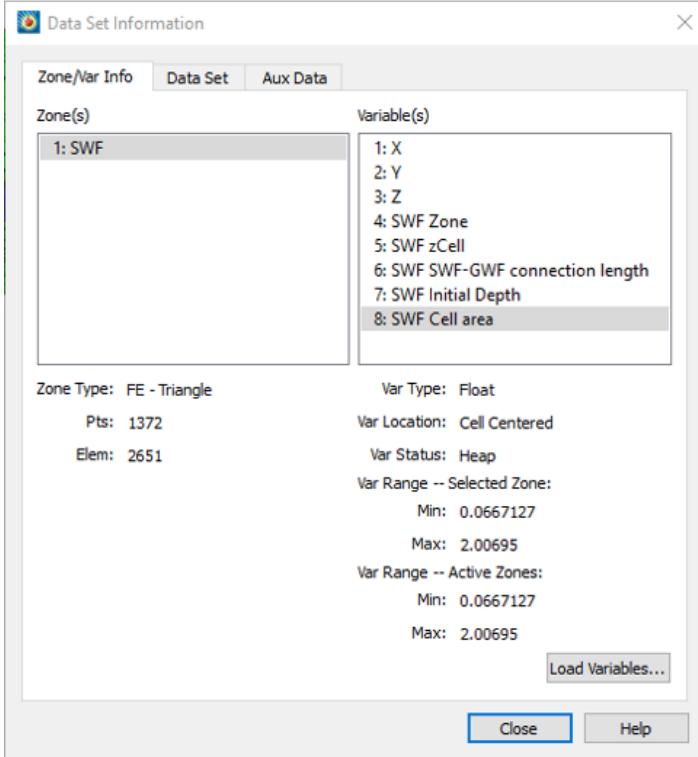
Although the SWF domain only has 2 zones, the set contains enough entries for 6 zones. The extra entries are ignored.



As shown above, the use of custom labels is configured by first choosing the contouring **Details...** button, which opens the **Contour & Multi-Coloring Details** dialogue. Choose the **Number Format...** button to open the **Specify Number Format** dialogue, then choose **Custom** from the **Type:** drop-down menu.

Custom labels can also be defined for the GWF and CLN domains if desired.

Data Set Information defined for the SWF domain is shown below:



The currently active frame title SWF is shown in the Zone(s) field, while the data set variables are listed in the Variable(s) field. The variables include the *xyz* coordinates of the TMPLT\_SWF mesh nodes, followed by the SWF Zone numbers assigned to the TMPLT\_SWF mesh elements. The remainder of the list shows the MODFLOW-USG<sup>*swf*</sup> cell properties that were defined during the model build. The Var-Range Selected Zone: area shows the variable Min: and Max: values for the currently chosen (highlighted) zone and variable (currently SWF and SWF Cell Area respectively).

For zoned variables, the SWF Zone number can be used to check the assigned properties by referring back to the `_buildo.eco` file output. For the `6_Abdul_Prism_Cell` example, the properties of the first zone are shown as being for the material Streambed:

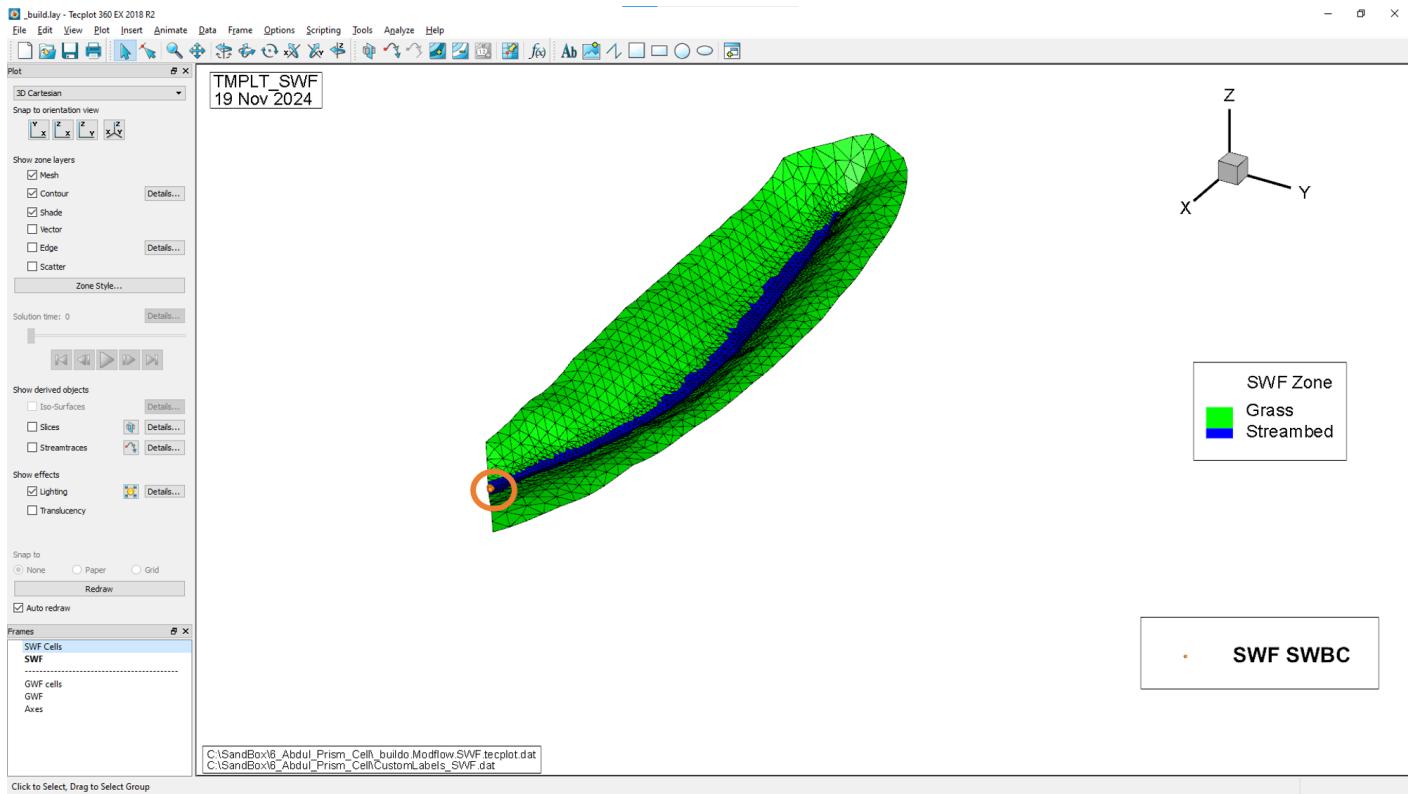
```

choose zone number
Adding zone number:      1
SWF zone numbers currently chosen:
  1

chosen zones use swf material number
Assigning all chosen SWF zones properties of material    3, Streambed
Manning's Coefficient:      3.00000E-02    METERS^(-1/3)  SECONDS
Depression Storage Height:  0.10000    METERS
Obstruction Storage Height: 0.0000    METERS
SWF Smoothing Depth 1:     1.00000E-06    METERS
SWF Smoothing Depth 2:     1.00000E-06    METERS

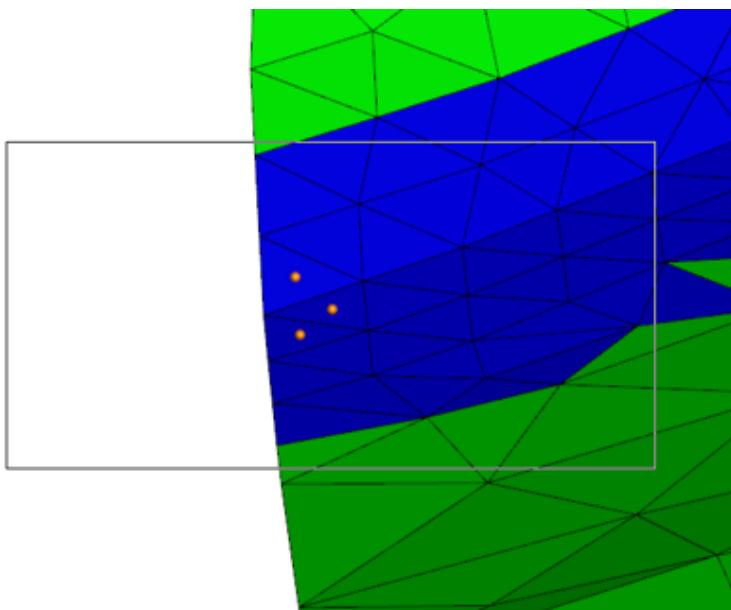
```

Bring the SWF Cells frame to the front.

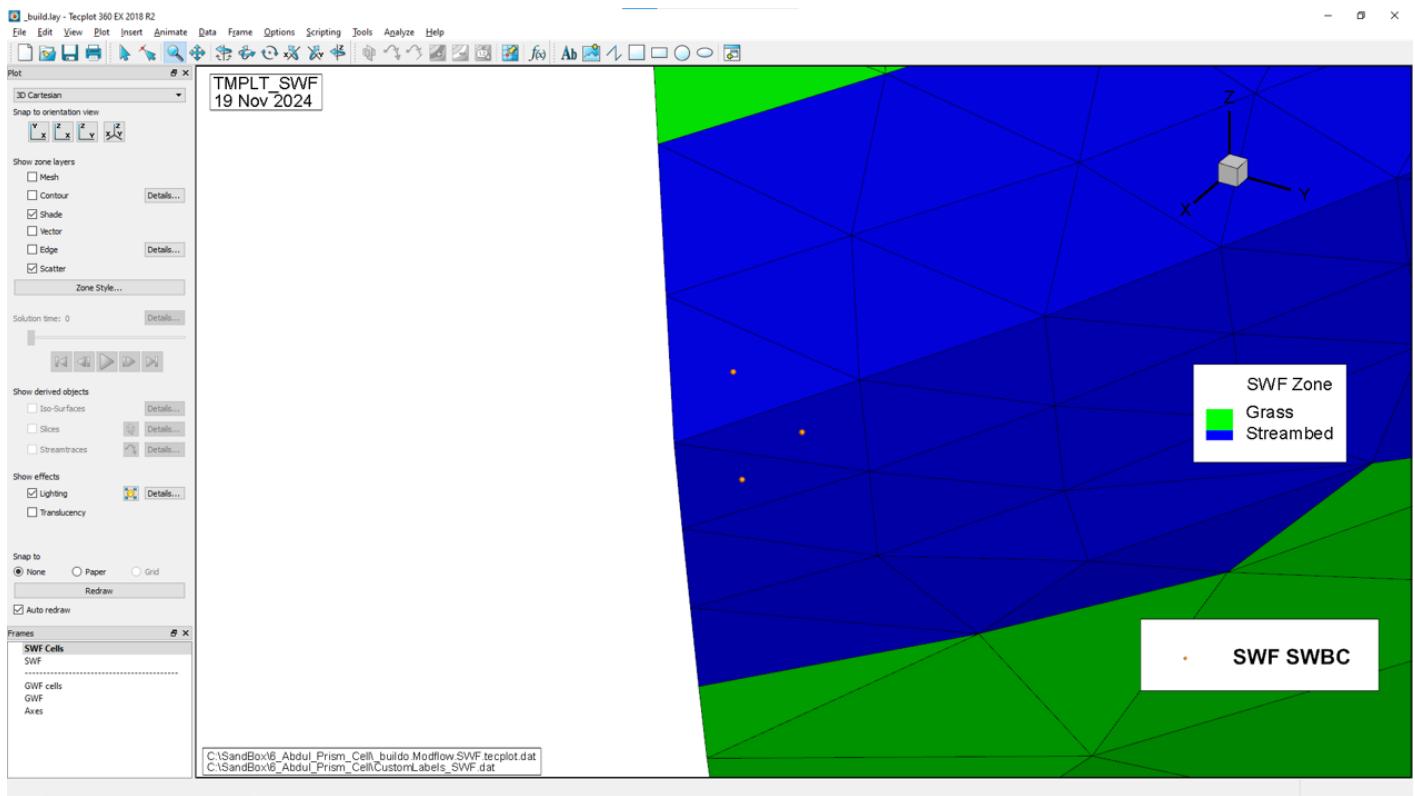


By default, only one scatter point set called **SWF SWBC** is being displayed. **SWBC** stands for Surface Water Boundary Condition, and contains values for the assigned critical depth boundary condition, shown as 3 orange spheres at the downstream end of the streambed, as indicated by the orange oval.

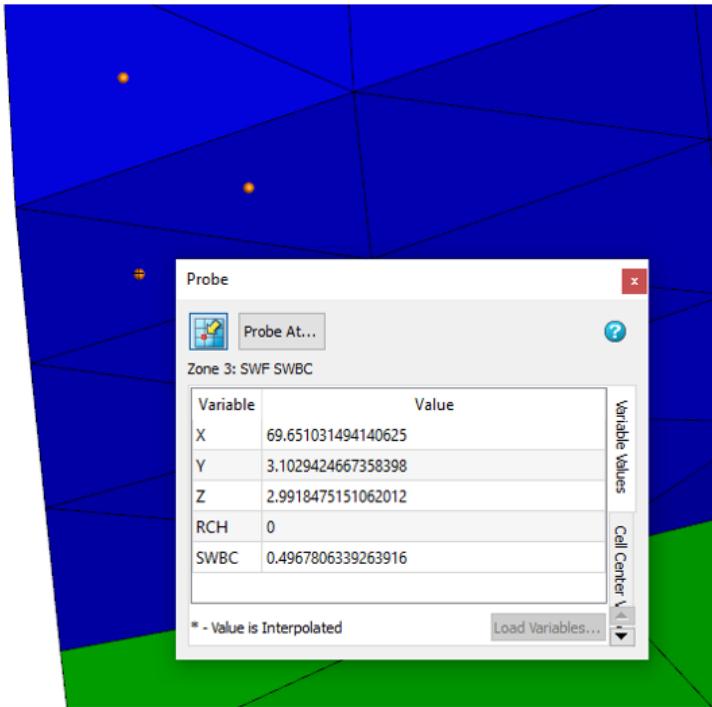
The Zoom Tool can be used to expand the scale around the outlet. Drag the magnifying glass cursor to draw a box around the region that you want to fit into the frame.



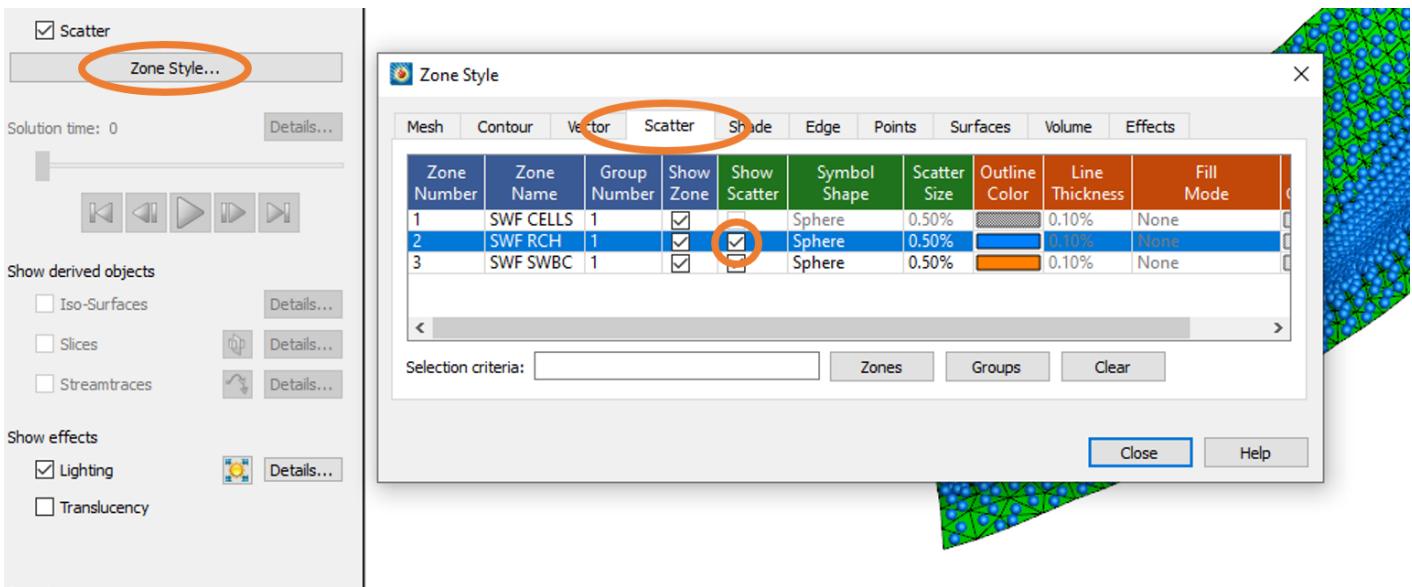
The region within the view box will be resized to fit into the frame.



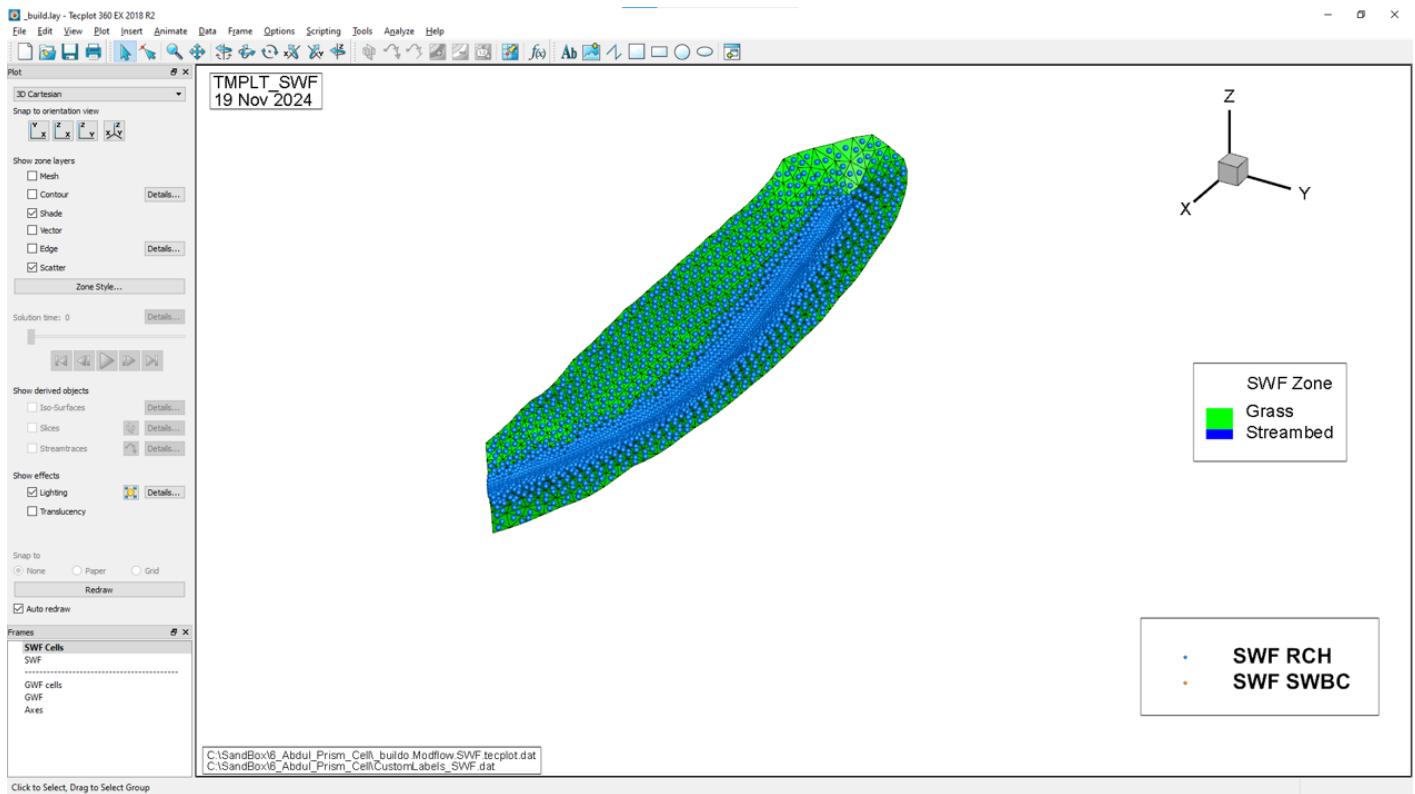
Using the **Probe Tool**, hold down the **Control** key and click the mouse on one of the orange spheres.



Although the critical depth boundary condition does not require any input, it does use a contributing length over which it is applied, which is the value shown above for SWBC. The presence of the variable RCH indicates it has been defined for the SWF domain. To show the RCH cells:



Click the Zone Style... button, which opens the Zone Style dialogue. Select the Scatter tab, then click in the empty box in the Show Scatter column for the SWF RCH zone.

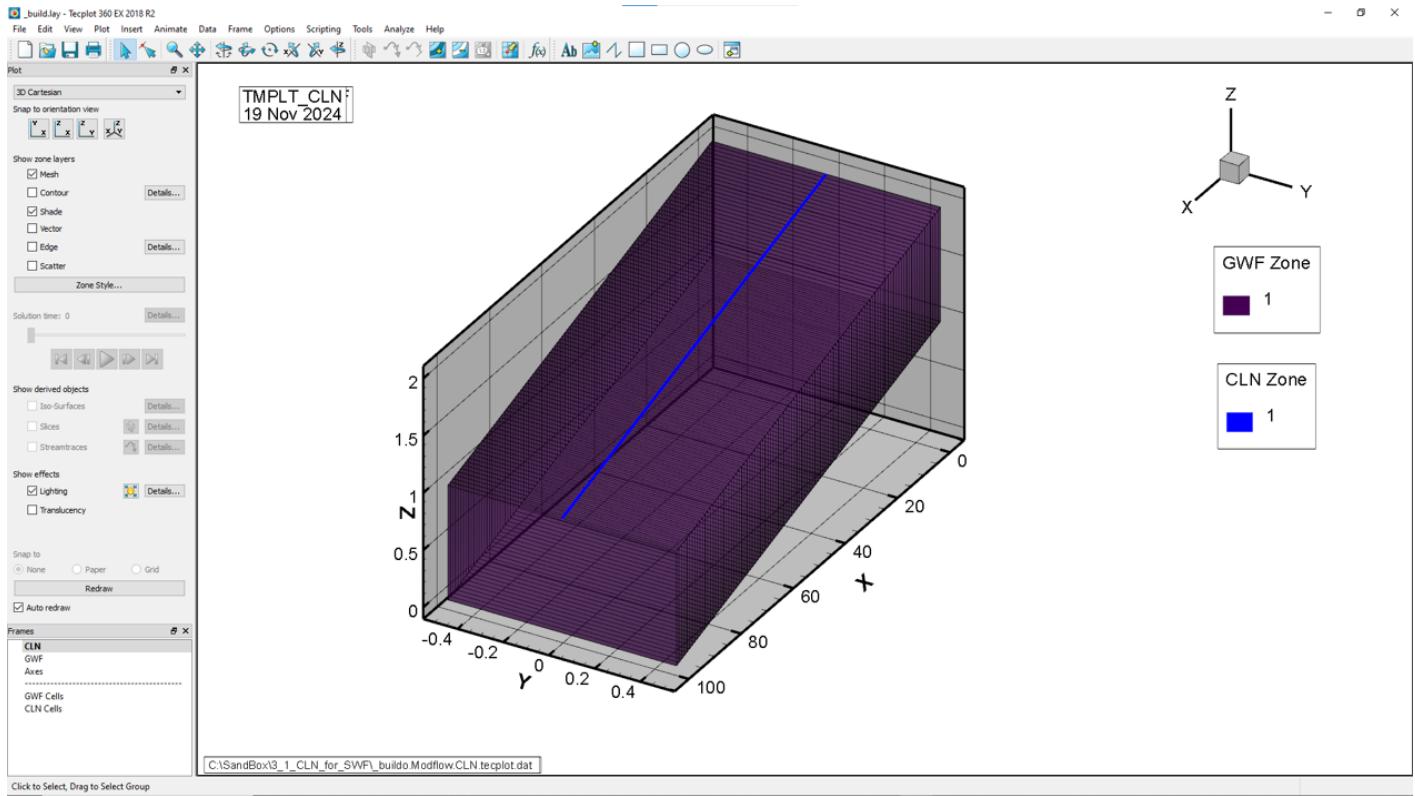


If you probe the recharge cells, you will see values of zero everywhere. This is because the values shown are for the last defined stress period, which for this example is the second one, where the recharge (rainfall) was stopped (i.e. set to zero).

### 3.11.3 CLN Domain

The example 3\_1\_CLN\_for\_SWF has a CLN domain defined by the end coordinates of a sloping straight line with constant head boundary conditions defined at each end of the GWF domain.

Start a new command prompt, run TECPLOT and load the \_build.lay file.

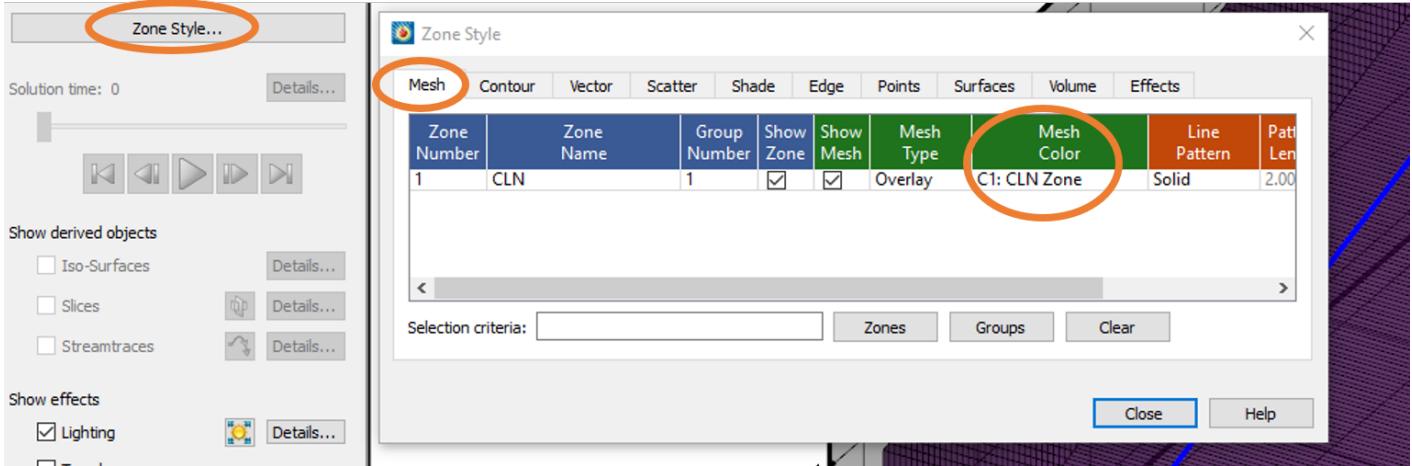


Here we can see the **CLN**, **GWF** and **Axes** frames are visible (i.e. placed above the **background** frame in the list). The **CLN** frame is currently active (i.e. the name is bolded) and placed at the front of the image (i.e. at the top of the list). It uses a different colormap to make it easier to distinguish the **GWF** domain below.

The **CLN** frame has very similar contents to the **GWF** frame described earlier, but note that:

- The names of the data files loaded into the frame are for the **CLN** domain.
- The data set title **TMPLT\_CLN** indicates that this is the **CLN** domain mesh (which was created from the **cln from xyz pair** instruction).
- The contouring legend, showing there is one **CLN** zone.

**CLN** domain meshes cannot be contoured like **GWF** or **SWF** meshes because the **CLN** element is a 1D straight line and contouring can only be shown for 2D or 3D elements. In this case, the finite-element mesh has been coloured by the **CLN Zone** number.



For zoned variables, the CLN Zone number can be used to check the assigned properties by referring back to the `_buildo.eco` file output. For the `3_1_CLN_for_SWF` example, the properties of all zones are shown as being for material 1, 2D Hillslope 100 m length:

choose all zones

chosen zones use cln material number

Assigning all chosen CLN zones properties of material 1, 2D Hillslope 100 m length

Geometry: Rectangular

Rectangular Width: 1.0000 METERS

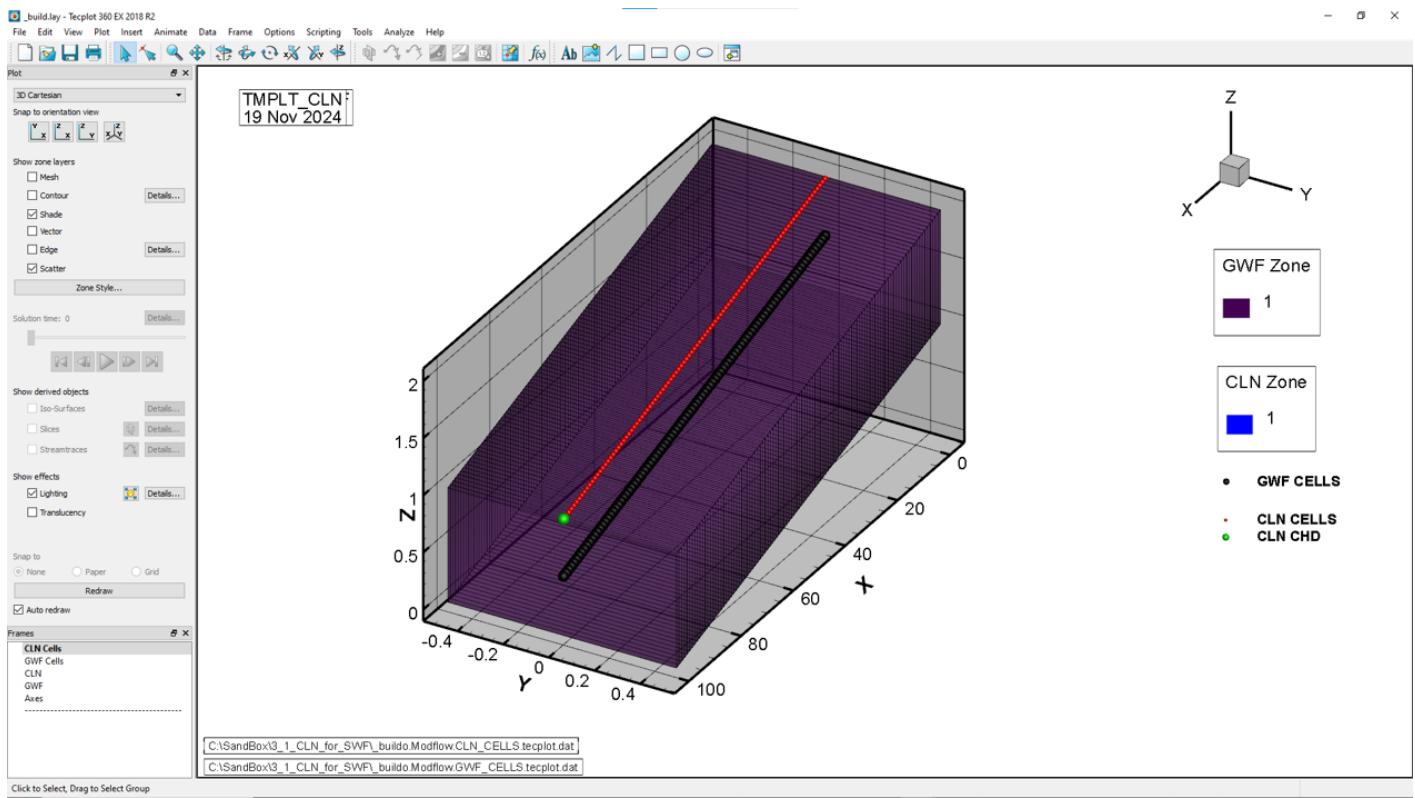
Rectangular Height: 1.0000 METERS

Direction: Horizontal

Flow Treatment: Unconfined/Mannings

Longitudinal K: 5.48000E-02 METERS SECONDS<sup>-1</sup>

Bring the CLN Cells and GWF Cells frames to the front.



The CLN cells (red spheres) show that it is located along the top surface of the GWF domain, and the constant head cell (light green sphere) is located at the downstream end of the CLN domain. The GWF cells (black spheres) are located at the centroid of the GWF elements (i.e. mesh-centred control volumes).

# Chapter 4

## Model Simulation and Post-Processing

The steps in our model execution workflow are:

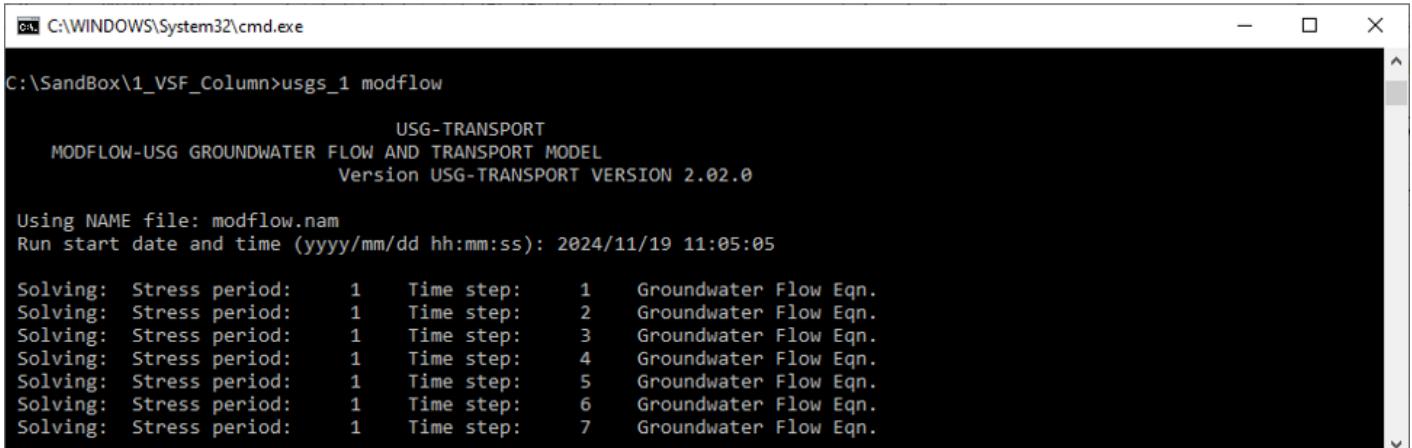
1. Run MODFLOW-USG<sup>Swf</sup> to create the new project output files (e.g. time-varying hydraulic head, drawdown etc).
2. Run MUT to post-process the MODFLOW-USG<sup>Swf</sup> simulation, which produces TECPLOT output files for the various model domains (i.e. GWF, SWF and/or CLN) created during the simulation.
3. Run TECPLOT and examine the MODFLOW-USG<sup>Swf</sup> output files.

### 4.1 Modflow-Usg<sup>Swf</sup> Simulation

To run MODFLOW-USG<sup>Swf</sup> on the 1\_VSF\_Column example, and assuming we have a command prompt open at the appropriate directory, we can type:

```
usgs_1 modflow
```

which obtains the prefix for the MODFLOW-USG<sup>Swf</sup> input files, in this case the default prefix **modflow**. As MODFLOW-USG<sup>Swf</sup> processes the input file, output is written to the screen:



```
C:\WINDOWS\System32\cmd.exe
C:\SandBox\1_VSF_Column>usgs_1 modflow

          USG-TRANSPORT
MODFLOW-USG GROUNDWATER FLOW AND TRANSPORT MODEL
          Version USG-TRANSPORT VERSION 2.02.0

Using NAME file: modflow.nam
Run start date and time (yyyy/mm/dd hh:mm:ss): 2024/11/19 11:05:05

Solving: Stress period:    1    Time step:    1    Groundwater Flow Eqn.
Solving: Stress period:    1    Time step:    2    Groundwater Flow Eqn.
Solving: Stress period:    1    Time step:    3    Groundwater Flow Eqn.
Solving: Stress period:    1    Time step:    4    Groundwater Flow Eqn.
Solving: Stress period:    1    Time step:    5    Groundwater Flow Eqn.
Solving: Stress period:    1    Time step:    6    Groundwater Flow Eqn.
Solving: Stress period:    1    Time step:    7    Groundwater Flow Eqn.
```

If execution is successful you will see the Normal termination of simulation message:

```
Solving: Stress period: 1 Time step: 840 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 841 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 842 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 843 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 844 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 845 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 846 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 847 Groundwater Flow Eqn.  
Solving: Stress period: 1 Time step: 848 Groundwater Flow Eqn.  
Run end date and time (yyyy/mm/dd hh:mm:ss): 2024/11/19 11:05:06  
Elapsed run time: 0.658 Seconds  
Normal termination of simulation  
C:\SandBox\1_VSF_Column>
```

Every MODFLOW-USG<sup>Swf</sup> simulation generates a run-time listing file, in this case called `modflow.lst`, that consists of the input data for the simulation; the solver and nonlinear outputs at user-requested detail; head and drawdown solutions, if requested; mass-balance information; and time-step information for the simulation. It also produces binary files of head, drawdown, saturation and cell-by-cell flows for each model domain.

To post-process the output produced by MODFLOW-USG<sup>Swf</sup> for the `1_VSF_Column` example, we would run MUT using the input file `_post.mut` by typing:

```
mut _post
```

If you open the file `_post.mut` in a text editor you will see the first line is a comment followed by one instruction and input:

```
! This example reads a modflow project and postprocesses it  
postprocess existing modflow model  
modflow
```

As in the model build, MUT first creates a clean copy of the input file called `_posto.input` by removing all comment lines. As it processes the input file, output is written to both the screen and to the file `_posto.eco`.

## 4.2 Mut Post-processing

The instruction to post-process the MODFLOW-USG<sup>Swf</sup> model after execution is:

---

**postprocess existing modflow model**

1. **Prefix<sub>str</sub>** The MODFLOW-USG<sup>Swf</sup> model prefix.

Given **Prefix<sub>str</sub>**, this instruction:

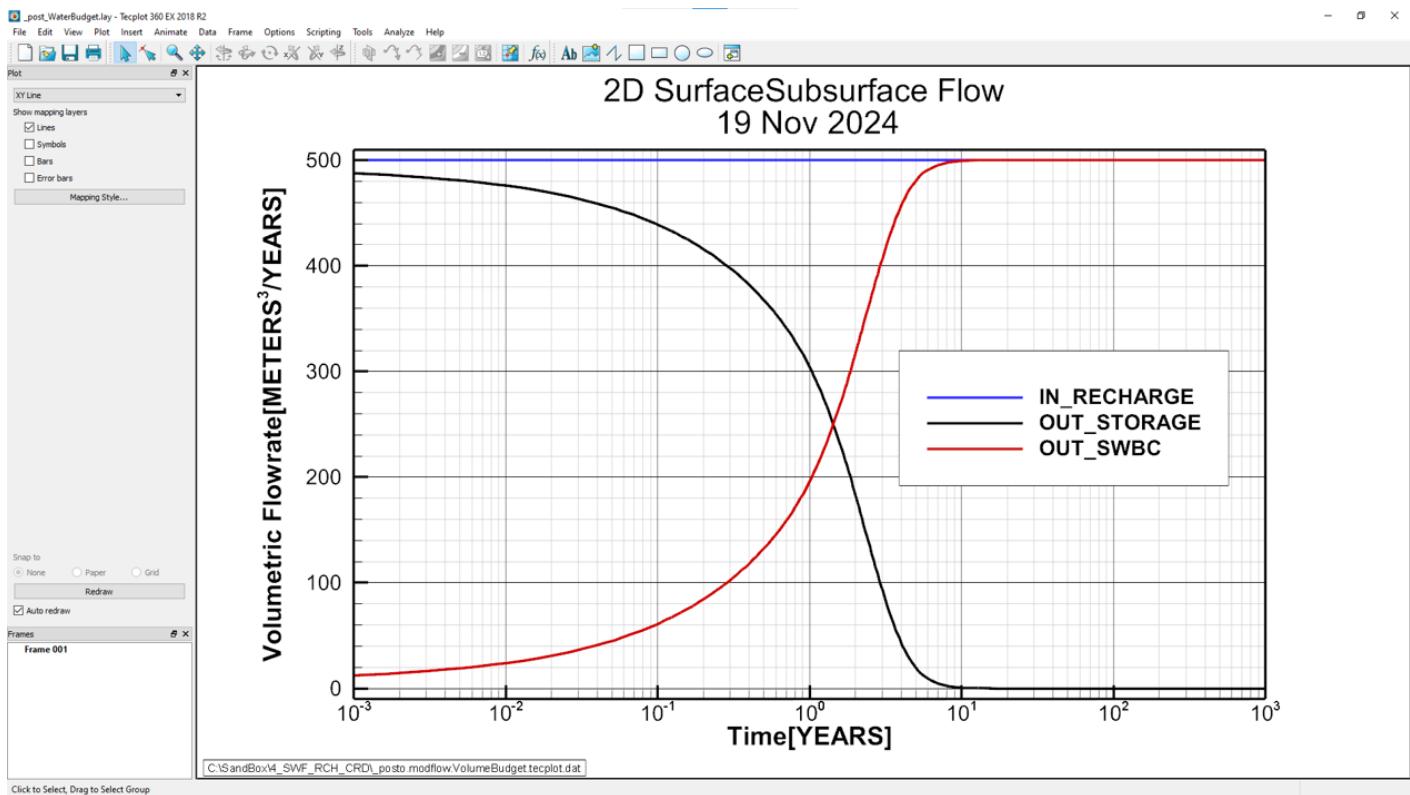
- Reads head, drawdown and cell-by-cell flow binary output files for each output time and writes the results to the file `_posto.modflow.GWF.tecplot.dat`.
- scans the MODFLOW-USG<sup>Swf</sup> listing file, extracts volumetric budget data at each time step and writes the results to the file `_posto.modflow.VolumeBudget.tecplot.dat`.

4.3 Volumetric Water Budget Plots

Volumetric water budget data is useful for checking the fluid mass balance of the model run. The example 4\_SWF\_RCH\_CRD has a TECPLOT layout file `_post_WaterBudget.lay` which you can load directly into TECPLOT from the command prompt by typing:

```
tec360 VolumetricBudget.lay
```

You should see the following TECPLOT window:

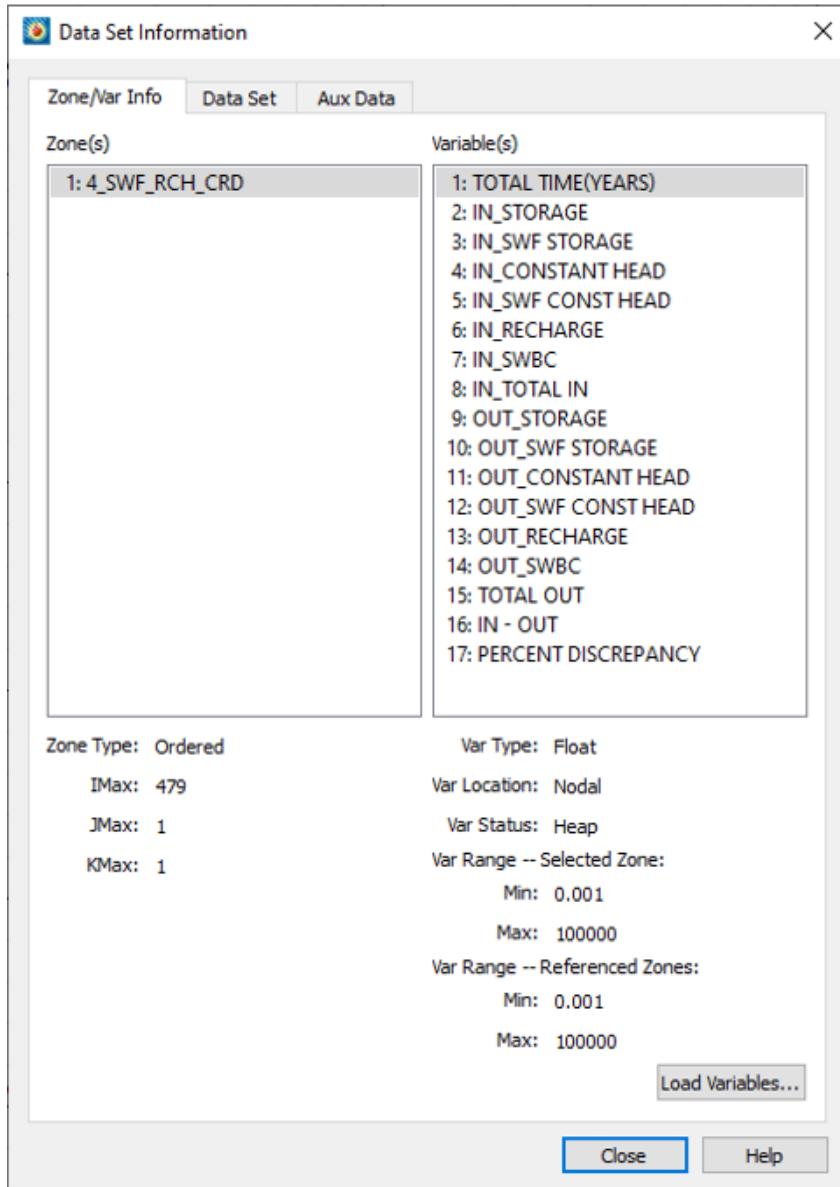


This TECPLOT frame has the following features and contents:

- It is an XY Line plot showing the volumetric flowrate versus time for selected components of the model.
- The name of the data file loaded into the frame is shown at the bottom left corner.

- The plot title and current date (on the day the file was loaded) are shown centred above the plot.
- The line legend is shown on the right side of the plot.
- The X-axis uses a log time scale.

The TECPLLOT data set information dialogue shows all of the variables available for plotting:



Variable names are derived from the `modflow.lst` file:

```
VOLUMETRIC BUDGET FOR ENTIRE MODEL AT END OF TIME STEP 479 IN STRESS PERIOD 1
```

---

CUMULATIVE VOLUMES	L**3	RATES FOR THIS TIME STEP
--------------------	------	--------------------------

---

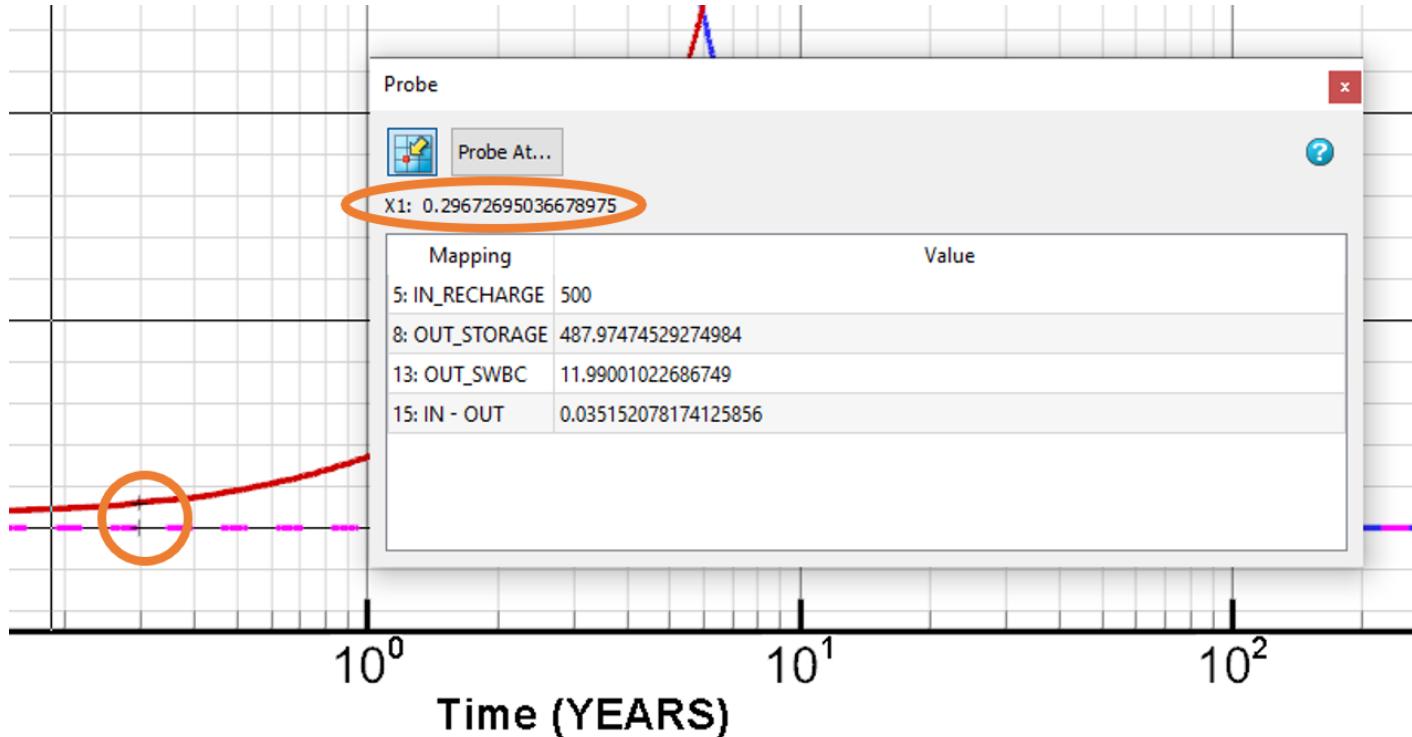
IN:	---
STORAGE =	3.0881E-04
SWF STORAGE =	9.9743E-03
CONSTANT HEAD =	0.0000
SWF CONST HEAD =	0.0000
RECHARGE =	50000000.0000
SWBC =	0.0000
 TOTAL IN =	50000000.0103
IN:	---
STORAGE =	5.6023E-09
SWF STORAGE =	2.1841E-13
CONSTANT HEAD =	0.0000
SWF CONST HEAD =	0.0000
RECHARGE =	500.0000
SWBC =	0.0000
 TOTAL IN =	500.0000

... etc

PERCENT DISCREPANCY = 0.01 PERCENT DISCREPANCY = 0.01

This example has a uniform recharge rate of 0.5 *meters/year* which results in a total recharge of 500 *meters<sup>3</sup>/year* when multiplied by the 1000 *meter* length of the cross-section. Initially, water comes out of storage then but this declines to zero at equilibrium. Water exiting the surface water outflow critical depth boundary OUT\_SWBC is initially zero then rises to equal the total recharge at equilibrium. Fluid balance error IN-OUT is essentially zero throughout the simulation.

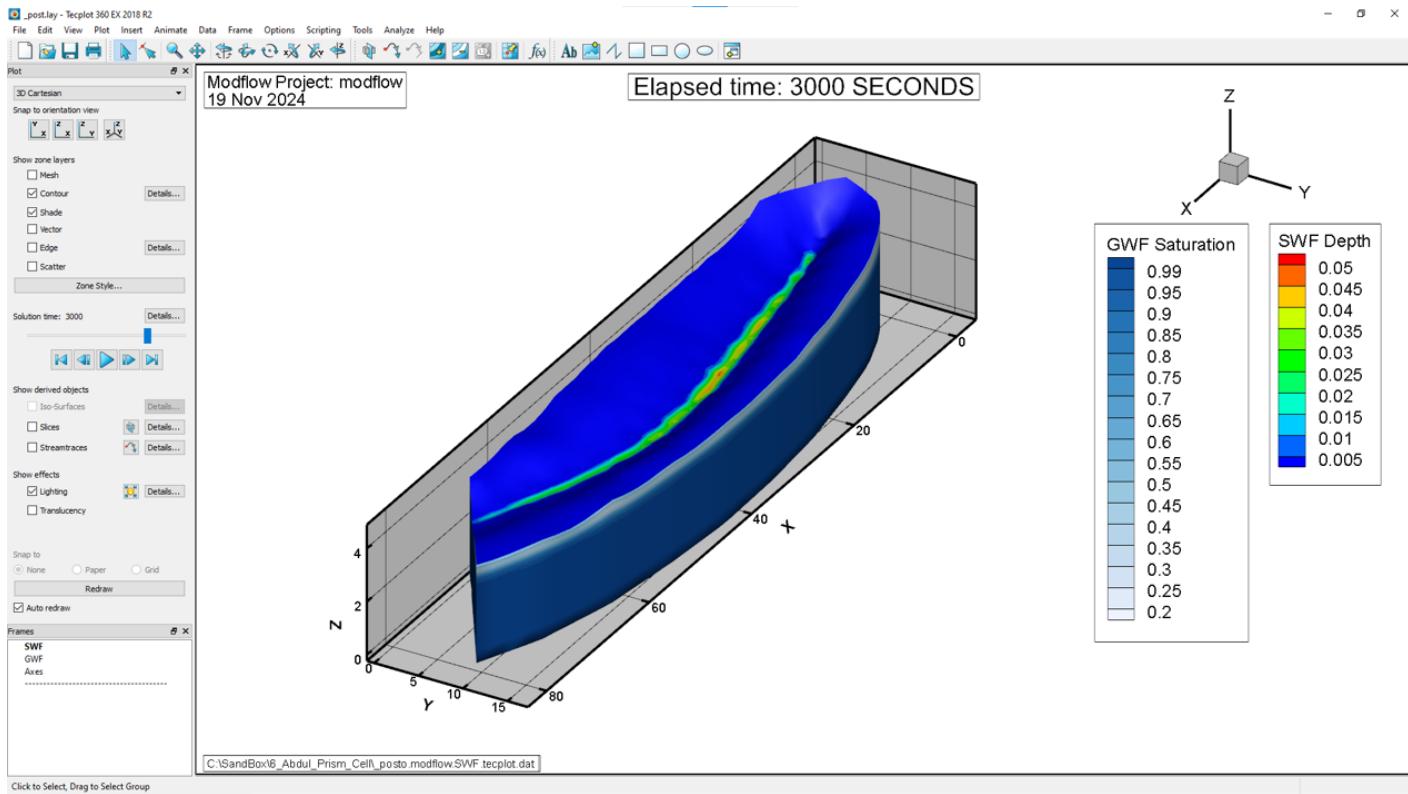
The probe tool can be used in XY Line plots to get exact variable values at a chosen location along the X or Y axis. The cursor is shown as a vertical line if probing values on the X-axis (i.e. over time):



Here, the location of the probe is shown by small vertical lines placed where the probe crossed the plotted lines (lower left orange circle). The exact coordinate is given as X1: 0.2967... years. At this early time, the OUT\_STORAGE value is still near its initial value of 500 and the OUT\_SWBC has just started rising.

## 4.4 3D Visualization of Model Results

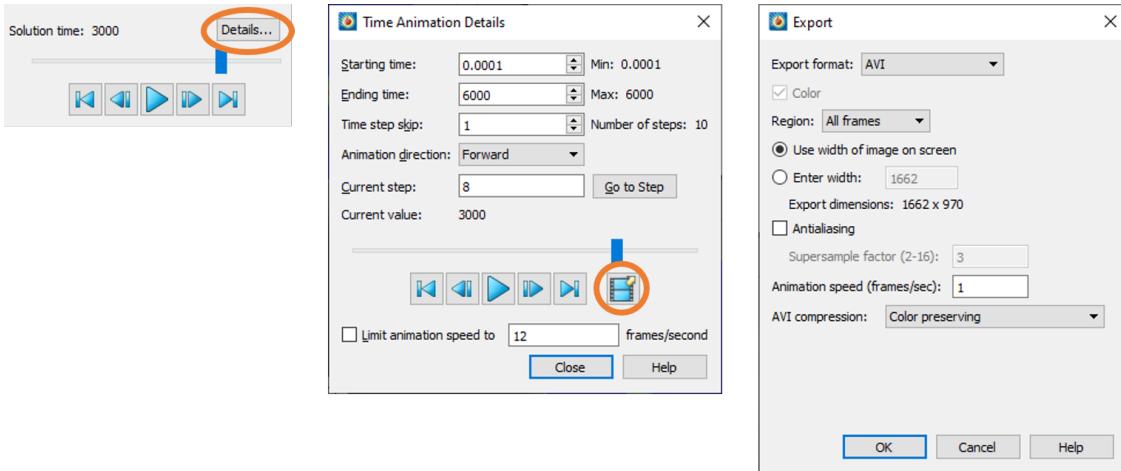
A TECPLLOT layout file, `_post.lay`, has been created for each example and provides a quick way to view MODFLOW-USG<sup>Swf</sup> model solution results. This result is from the example `6_Abdul_Prism_Cell`:



The `_post.lay` layout file is similar to the `_build.lay` file shown earlier for the same problem: SWF, GWF and Axes frames are visible (i.e. placed above the background frame in the list). The SWF is currently active (i.e. the name is bolded) and placed at the front of the image (i.e. at the top of the list). It uses a different colormap to make it easier to distinguish the GWF domain below. In this case though, the contoured variables are SWF Depth and GWF Saturation results from the MODFLOW-USG<sup>Swf</sup> simulation.

### 4.4.1 Solution Times and Animation

The model output includes data for several output times, and the image above is showing conditions at a solution time of 3000 seconds, as indicated by the label near the top center of the plot. The solution time shown is controlled by the slider and button controls near the centre of the Plot frame on the left hand side of the image:

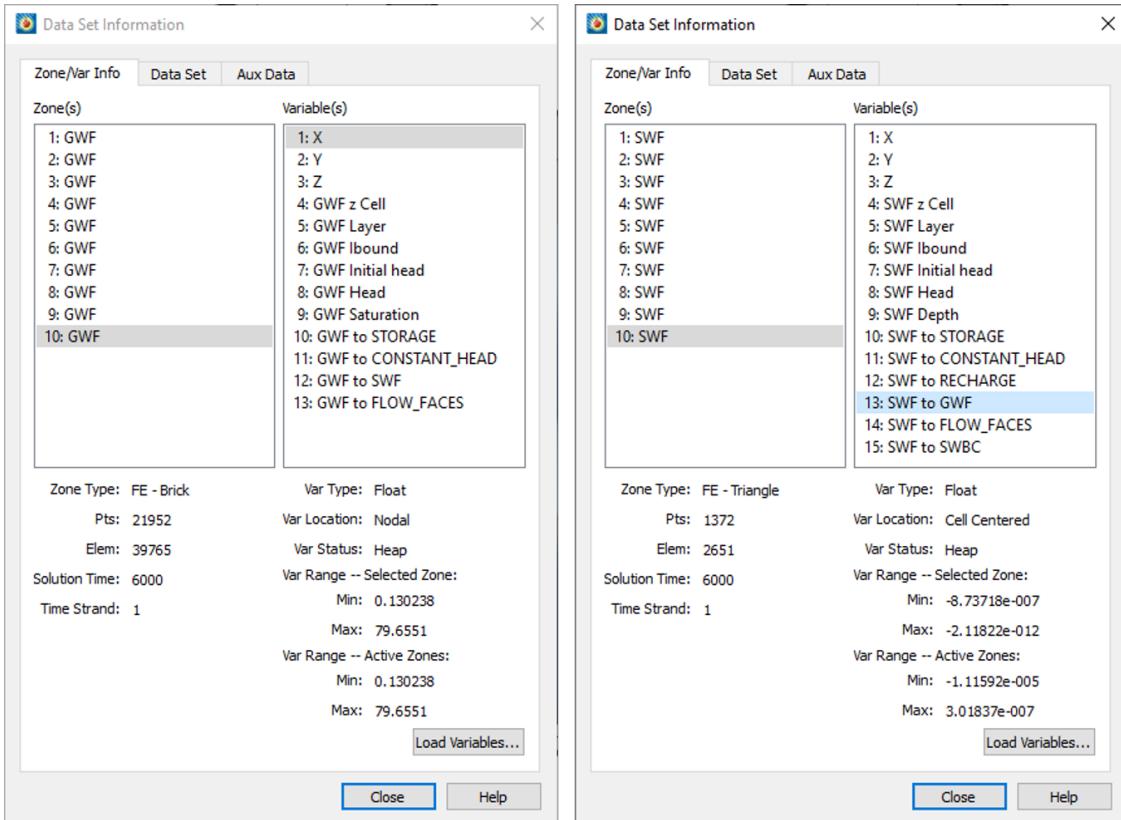


The Details... button leads to the Time animation details dialogue which allows you to control and

save animations of transient model results using the Export To File button . There is an example animation in \MUT\_Examples-main\6\_Abdul\_Prism\_Cell folder in the powerpoint file Abdul Problem Animation.pptx. It used the settings shown in the Export dialogue above to limit animation speed and write to an AVI-formatted file. This was then inserted in powerpoint where it can be viewed.

#### 4.4.2 Data Set Information

The Data Set Information dialogues for the SWF and GWF frames are shown below:



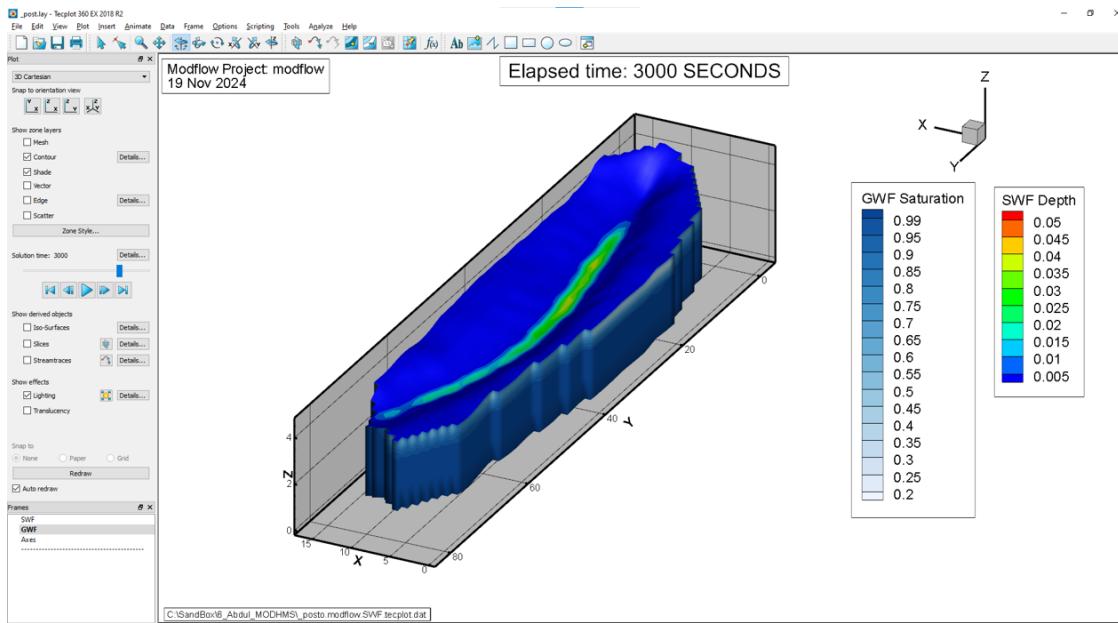
These are similar to those shown earlier during the model build except there are now multiple Zone(s), one for each output time. The Solution Time (6000) is shown for the highlighted zone, in this case zone 10.

Included are these cell properties:

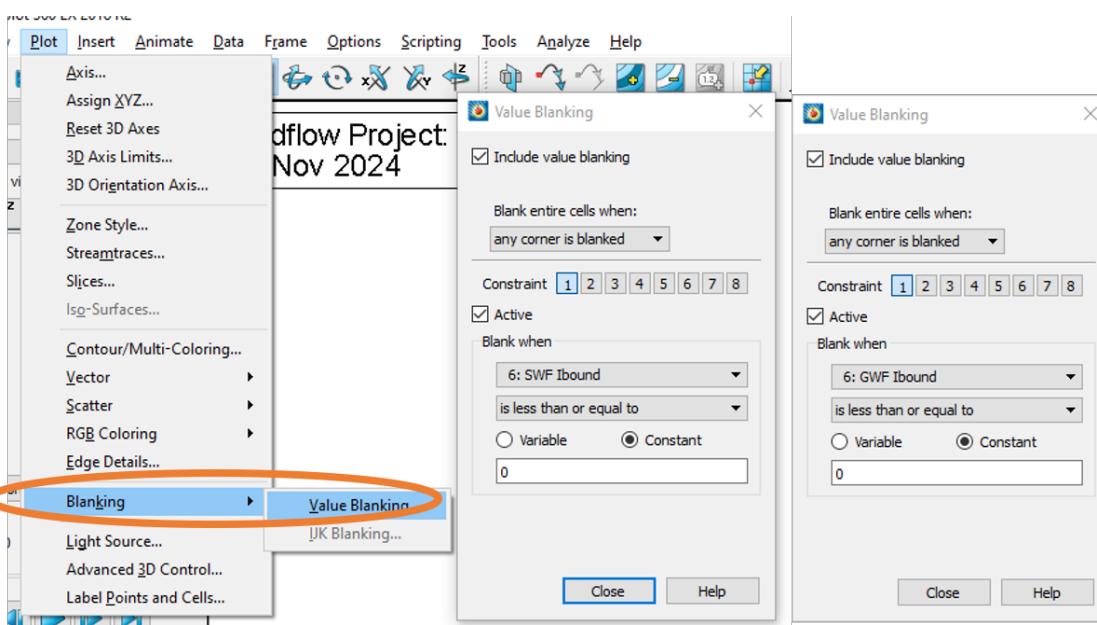
- Elevation SWF z Cell and GWF z Cell.
- MODFLOW layer number SWF layer and GWF layer.
- MODFLOW boundary number SWF lbound and GWF lbound.
- Initial head.
- Hydraulic head result.
- GWF saturation and SWF depth. These are stored in the MODFLOW DDN (drawdown) file.
- In this case, cell-by-cell flows are stored in GWF variables 10 to 13 and SWF variables 10 to 15.

#### 4.4.3 Inactive Cells and Value Blanking

The example 6\_Abdul\_MODHMS was generated from a 2D rectangular template mesh. To make it conform to the shape of the original triangular mesh used in example 6\_Abdul\_Prism\_Cell, a region of inactive cells was assigned around the outside of the domain. The layout file \_plot.lay has been configured to show only active cells:



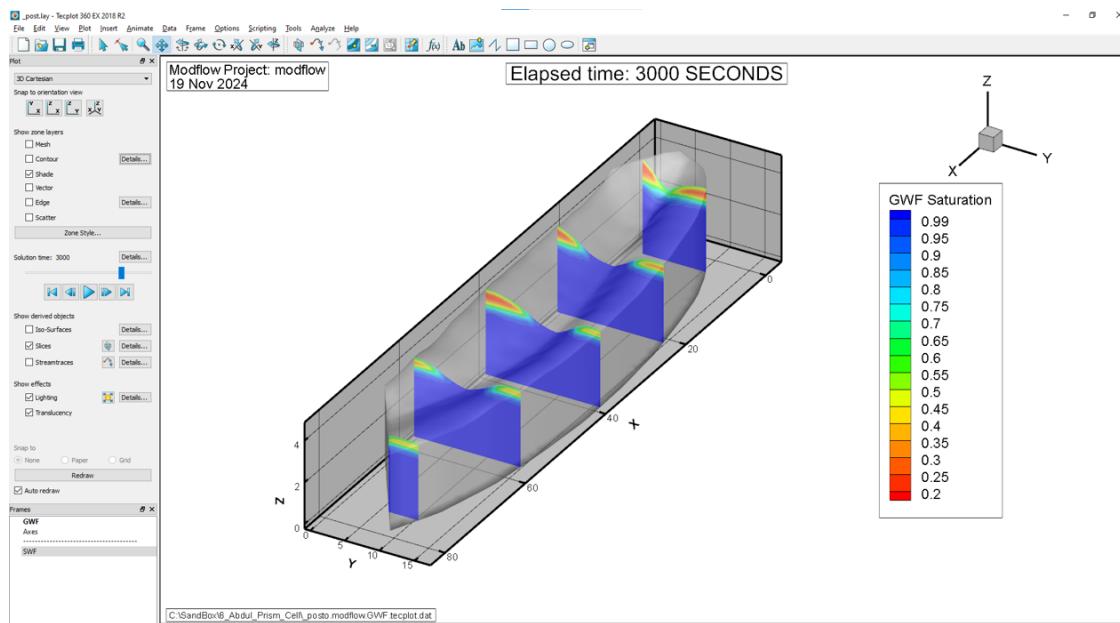
The layout uses TECPLLOT value-blanking (in this case based on the value of the blanking variables SWF lbound and GWF lbound) to prevent cells (or portions of cells) from appearing in the image. The Plot\Blanking\Value Blanking... menu option leads to the Value Blanking dialogue for the currently active frame:



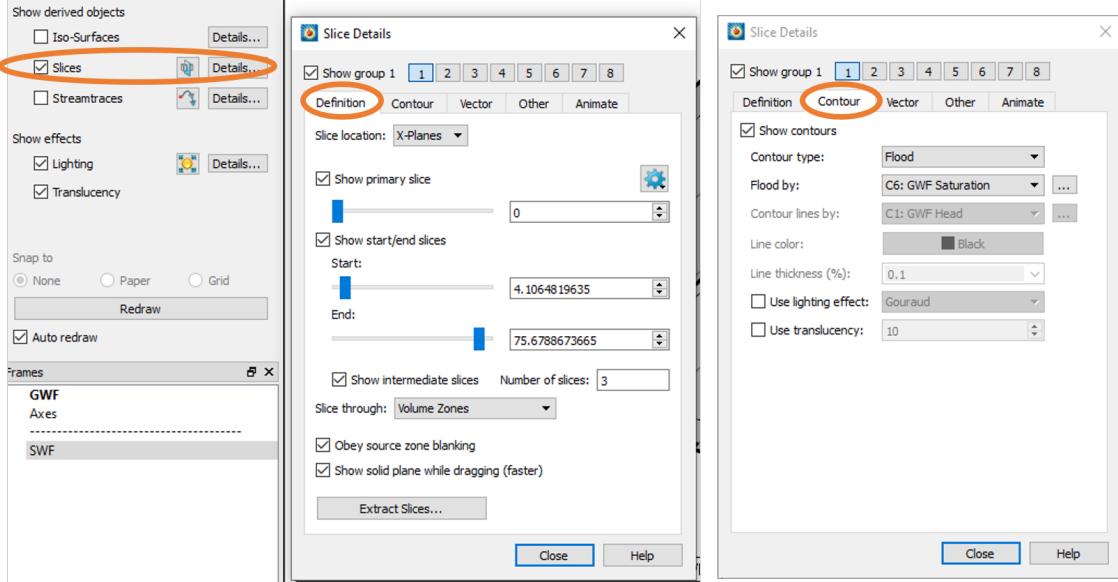
Shown here are the dialogues for both the SWF and GWF domains, where value blanking constraints are set to prevent SWF and GWF domain cells from being shown if the SWF Ibound or GWF Ibound value is less than or equal to zero.

#### 4.4.4 Slices and Fence Diagrams

The layout file 6\_Abdul\_Prism\_Cell\\_GWF\_Saturation\_Slices.lay has been configured to show a series of cross-sections in the GWF domain:



The layout uses the TECPLLOT derived object **Slices** to define the slice locations and content. The **Details...** button leads to the **Slice Details** dialogue for the currently active frame:



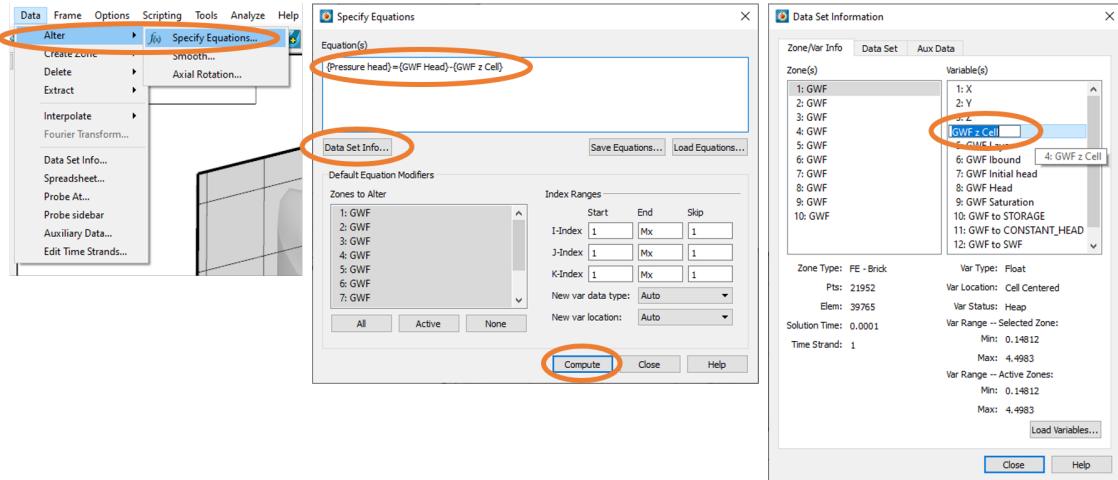
The Definition tab is configured to position the start and end slice positions and set the number of intermediate slices at 3.

The Contour tab is configured to contour-flood the slices by GWF Saturation value.

The Shade and Translucency boxes are checked to show the slice position relative to the model domain.

#### 4.4.5 Defining New Variables

New variables can be defined in TECPLLOT using the Data\Alter\Specify equations... menu option which leads to the Specify equations dialogue:



This dialogue has an Equation(s) window where we have defined a new variable, in this case Pressure Head, that is calculated from two existing variables according to the mathematical expression:

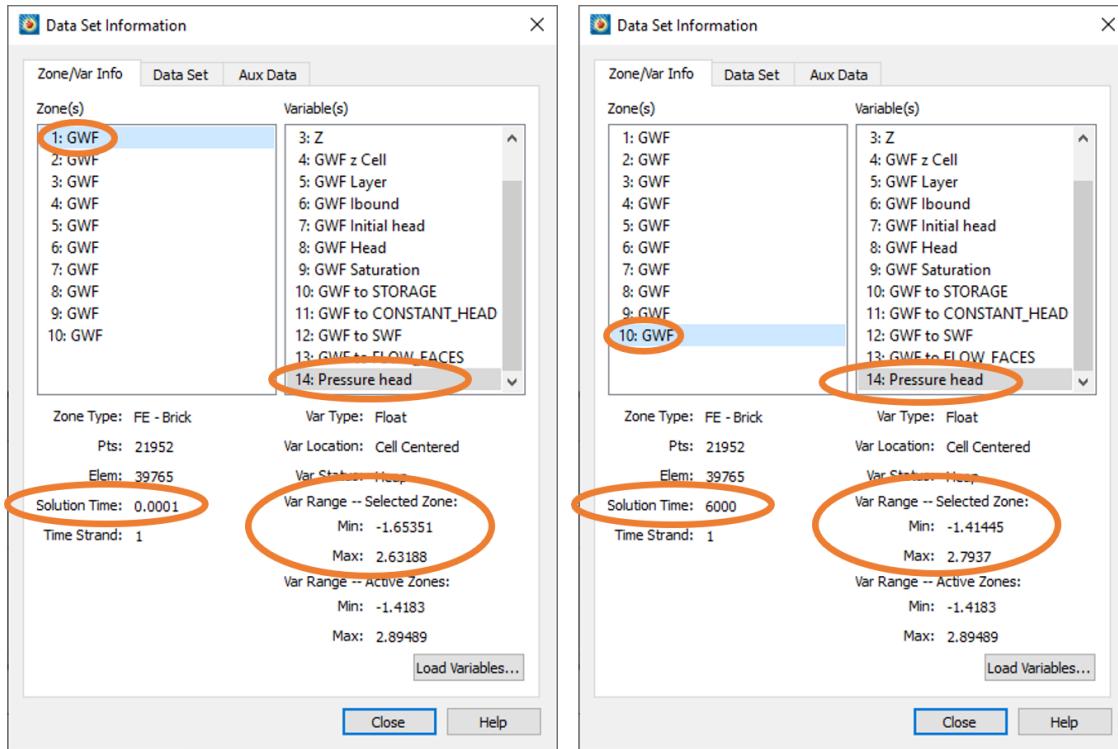
$$\{\text{Pressure Head}\} = \{\text{GWF Head}\} - \{\text{GWF z Cell}\}$$

In equation syntax, variable names must be placed within curly brackets: {}.

The Data Set Info... tab leads to the Data Set Information dialogue where variable names can be cut and

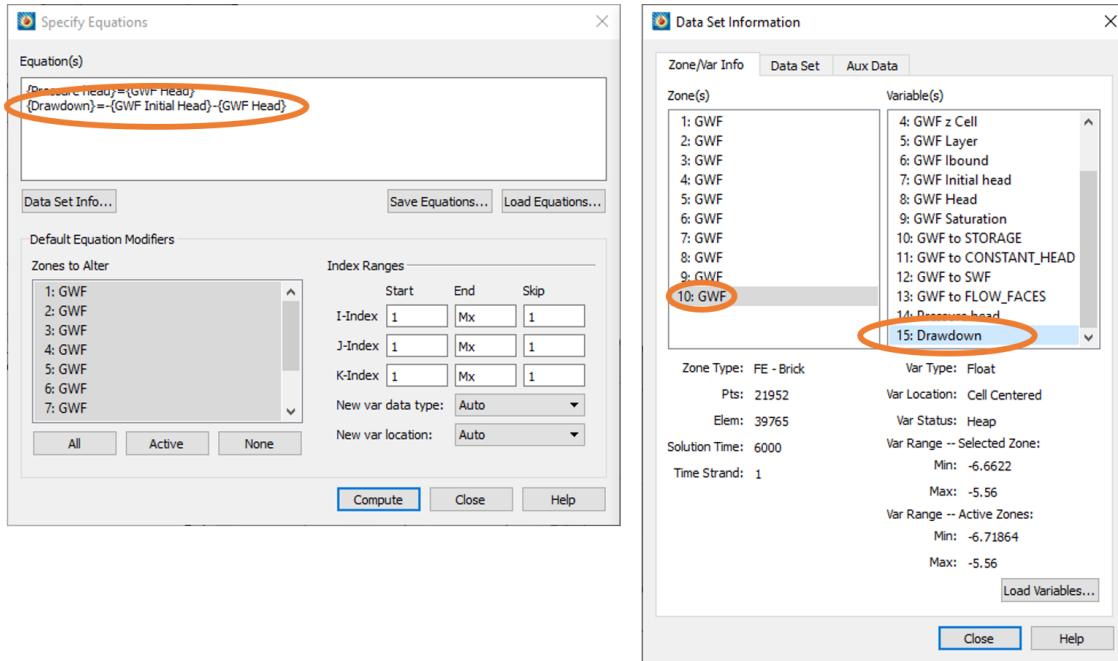
pasted into the equation window. Here the GWF z Cell variable has been selected.

Once the equation is complete, click the Compute button to generate the new variable, which will now show up in the Data Set Information dialogue:



Here we see the variable 14:Pressure Head has been added. Because the GWF Head variable changes over time and is defined at every timestep, the calculated variable does also. The two panes above show the pressure head range at solution time 0.0001 is different at time 6000.

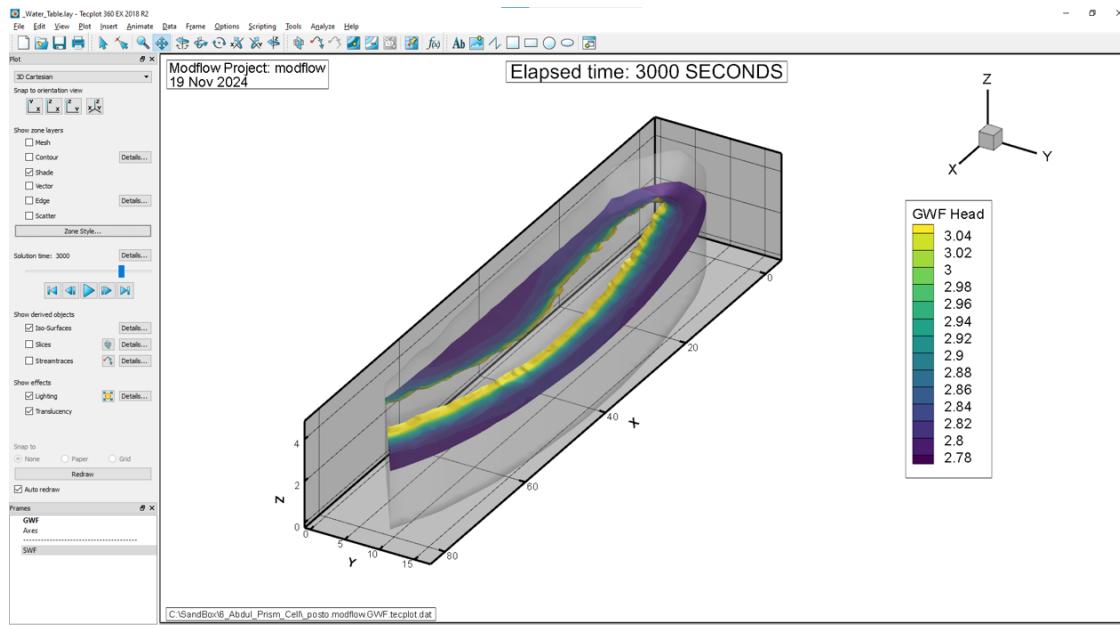
MODFLOW-USG<sup>Swf</sup> uses the MODFLOW *DDN* (i.e. drawdown) file to store saturation values for the GWF domain and surface water depth values for the SWF domain. If you want to plot drawdowns they can be calculated by adding an equation to the previous example:



Here we see the equation for calculating the variable Drawdown has been added to the Equations(s) window and the variable 15:Drawdown appear in the Data Set Information dialogue. The range shown for Drawdown values is negative because simulated heads were always higher than the initial flat water table condition.

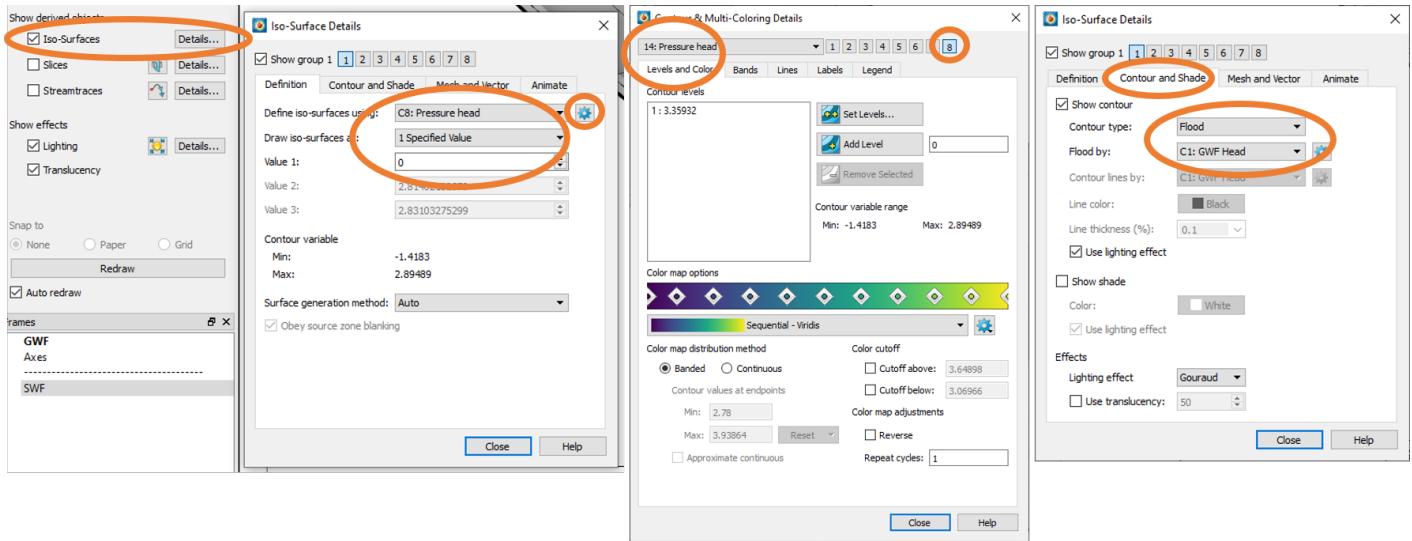
#### 4.4.6 Water Table Isosurface Plot

The layout file 6\_Abdul\_Prism\_Cell\\_Water\_Table.lay has been configured to show an isosurface of the water table in the GWF domain at a time of 3000 seconds :



The layout uses the TECPLLOT derived object Iso-surfaces to define the slice locations and content. The

Details... button leads to the Iso-Surface Details dialogue for the currently active GWF frame:

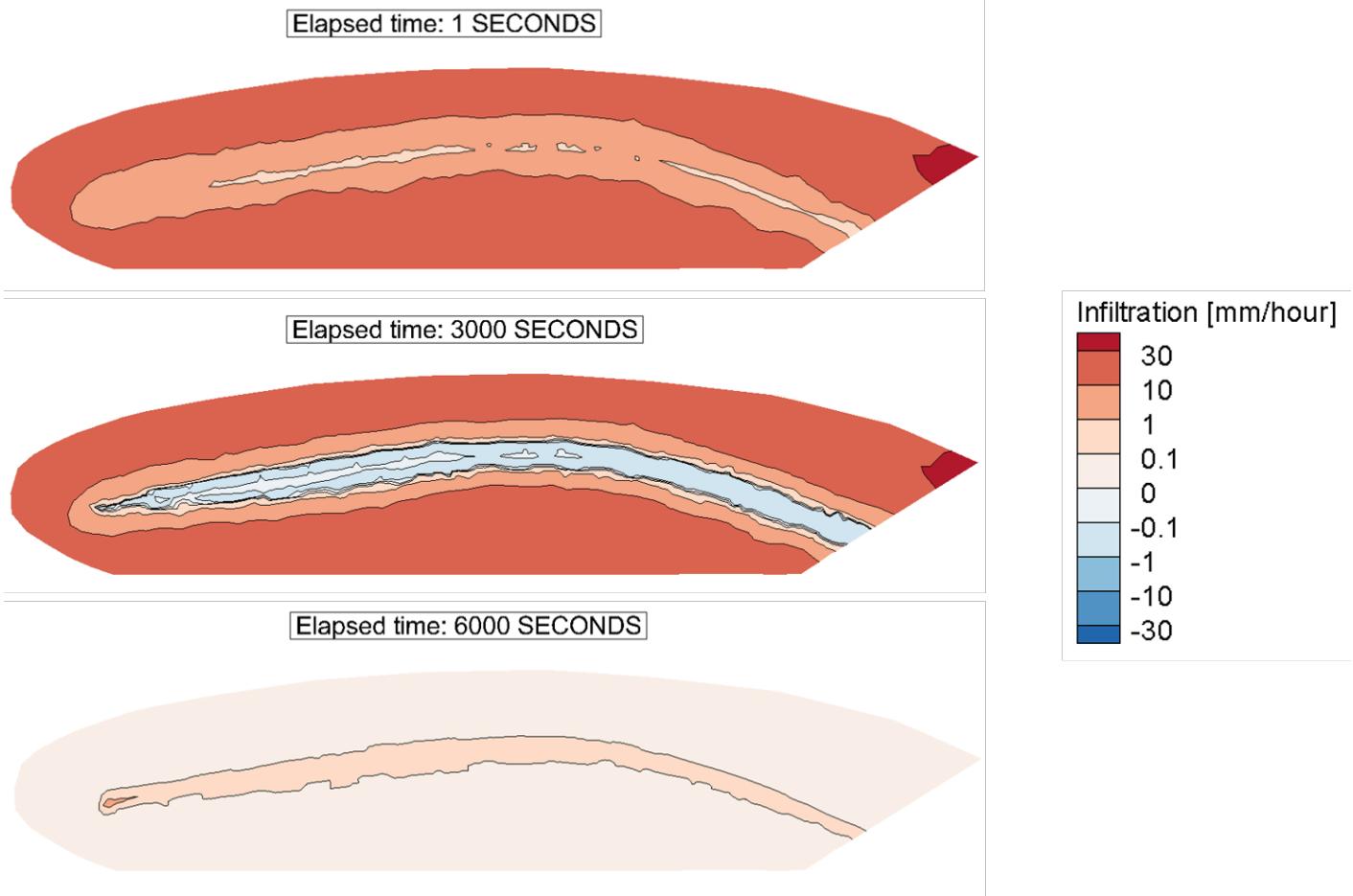


The Definition tab is configured to define iso-surfaces using the Pressure Head variable. This had to be defined as one of the eight contouring variables, and there is a short-cut to the Contour & Multi-Colouring Details dialogue using the button, where we set the eighth contouring variable (number 8 in orange circle) as 14:Pressure Head. By definition, the water table is the surface where the pressure head is equal to zero, so back on the Definitions tab we chose to draw iso-surfaces at one specified value: zero.

Finally, the Contours and shade tab is configured to contour flood the isosurface with the GWF Head variable.

#### 4.4.7 Infiltration Plot

The layout file 6\_Abdul\_Prism\_Cell\\_Infiltration defines a new variable Infiltration [mm/hour] and uses contour flood and lines to show it. The contour levels are set up with a red-blue colour map that is symmetric about the value zero. Red shading indicates zones of infiltration, while blue shading indicates zones of GWF discharge to the SWF domain.



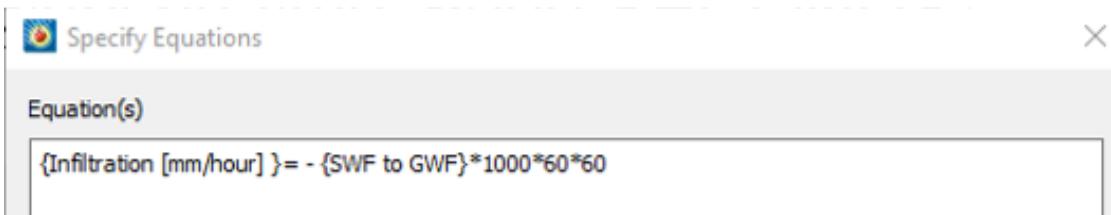
Here we can see the infiltration at 3 times:

**1 second** The application of water to the SWF domain has just started. The entire SWF domain is shaded red, showing that water is infiltrating everywhere, with the highest rate of infiltration occurring at the highest elevations in the domain (i.e. at hilltops)

**3000 seconds** Application of water stops. Infiltration is still occurring on the hills but discharge is occurring along most of the length of the swale, as shown by the blue shaded region.

**6000 seconds** End of simulation. The stream has drained and water is again infiltrating over the entire domain.

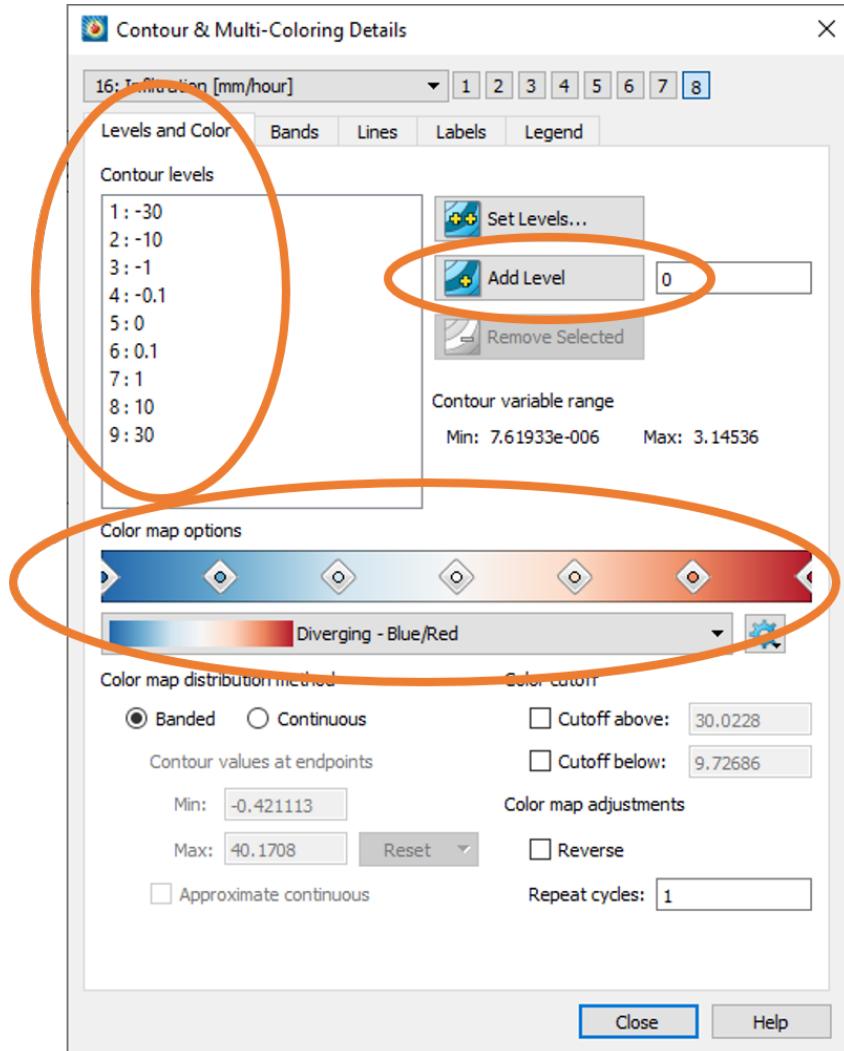
The new variable **Infiltration [mm/hour]** is calculated by the following equation in TECPLLOT:



The variable **SFW to GWF** contains the cell-by-cell flows from the SWF domain to the GWF domain, with negative values indicating a loss relative to the SWF domain (i.e. infiltration). We want infiltration to

be shown as a positive value in this case so we add a negative sign at the beginning of the equation. The final part of the equation  $*1000*60*60$  converts the value from units of meters per second to millimetres per hour.

The contour levels were set up as shown here:



The levels were entered manually using the Add Level and Remove selected buttons. The value zero is defined at the boundary between blue and red colours by using an equal number of positive and negative values with the same absolute ranges.

The color map is set to Diverging - Blue/Red using the drop-down menu.

# Chapter 5

## Demonstration Models

### 5.1 1D Variably-saturated Flow in a Column

This example `1_VSF_Column` simulates variably-saturated flow in a 1D column of homogeneous sand 100 cm thick using the following parameter values:

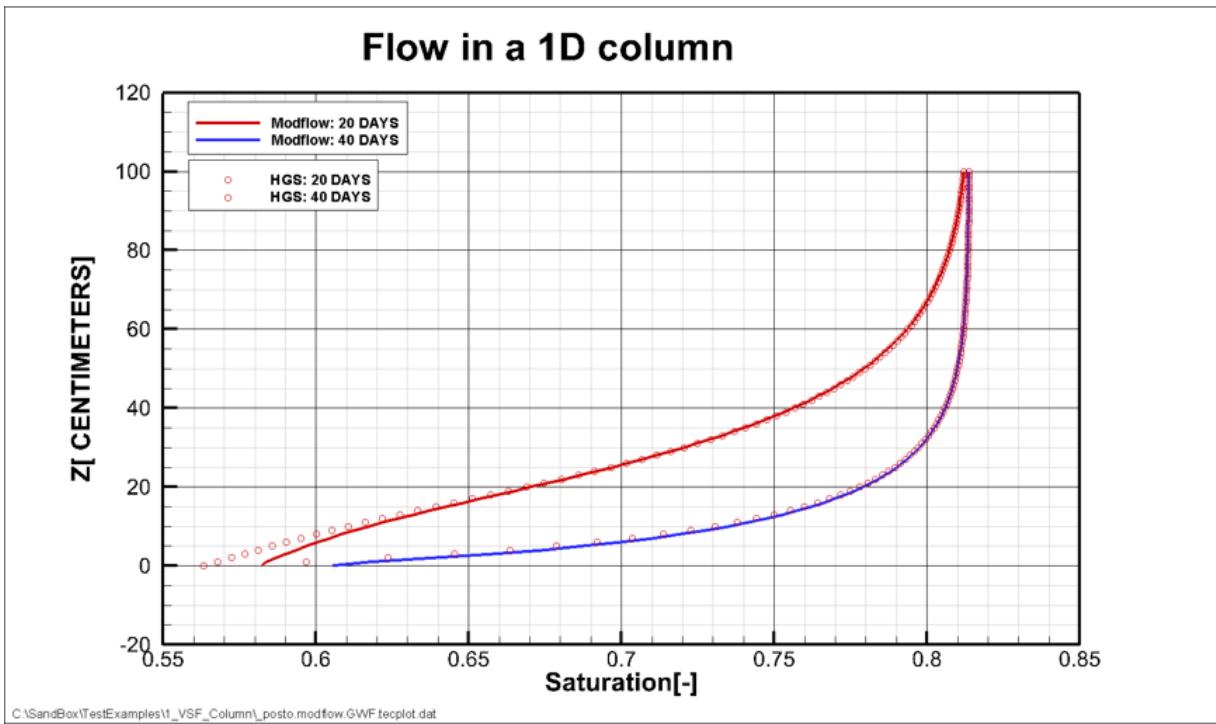
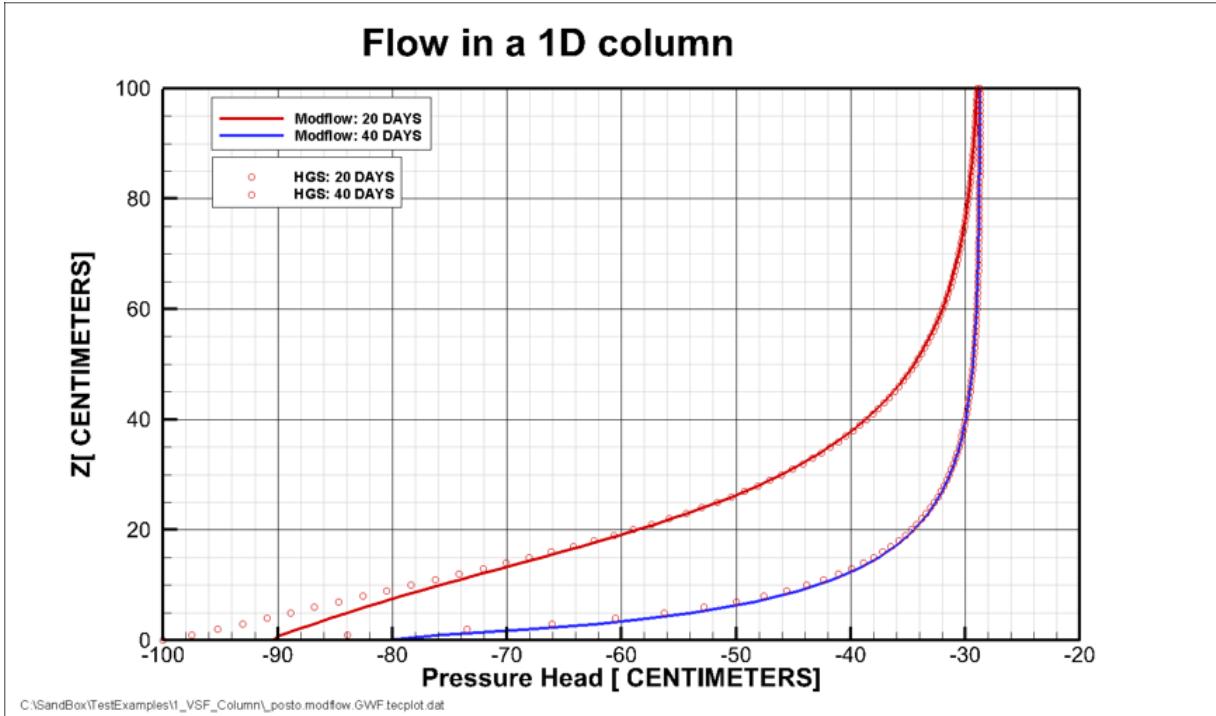
Parameter	Value	Unit
Specific yield (porosity)	0.43	
Vertical hydraulic conductivity	10	$\text{cm d}^{-1}$
Horizontal hydraulic conductivity	$1 \times 10^{-5}$	$\text{cm d}^{-1}$
Specific storage coefficient	$1 \times 10^{-7}$	$\text{cm}^{-1}$
Van Genuchten Alpha	$3.6 \times 10^{-2}$	$\text{cm}^{-1}$
Van Genuchten Beta	1.56	
Residual saturation	0.1814	

The Van Genuchten unsaturated function type was used.

A uniform rainfall of 0.4 cm/day was applied to the top of the column and the base was fixed at a pressure head of -100 cm.

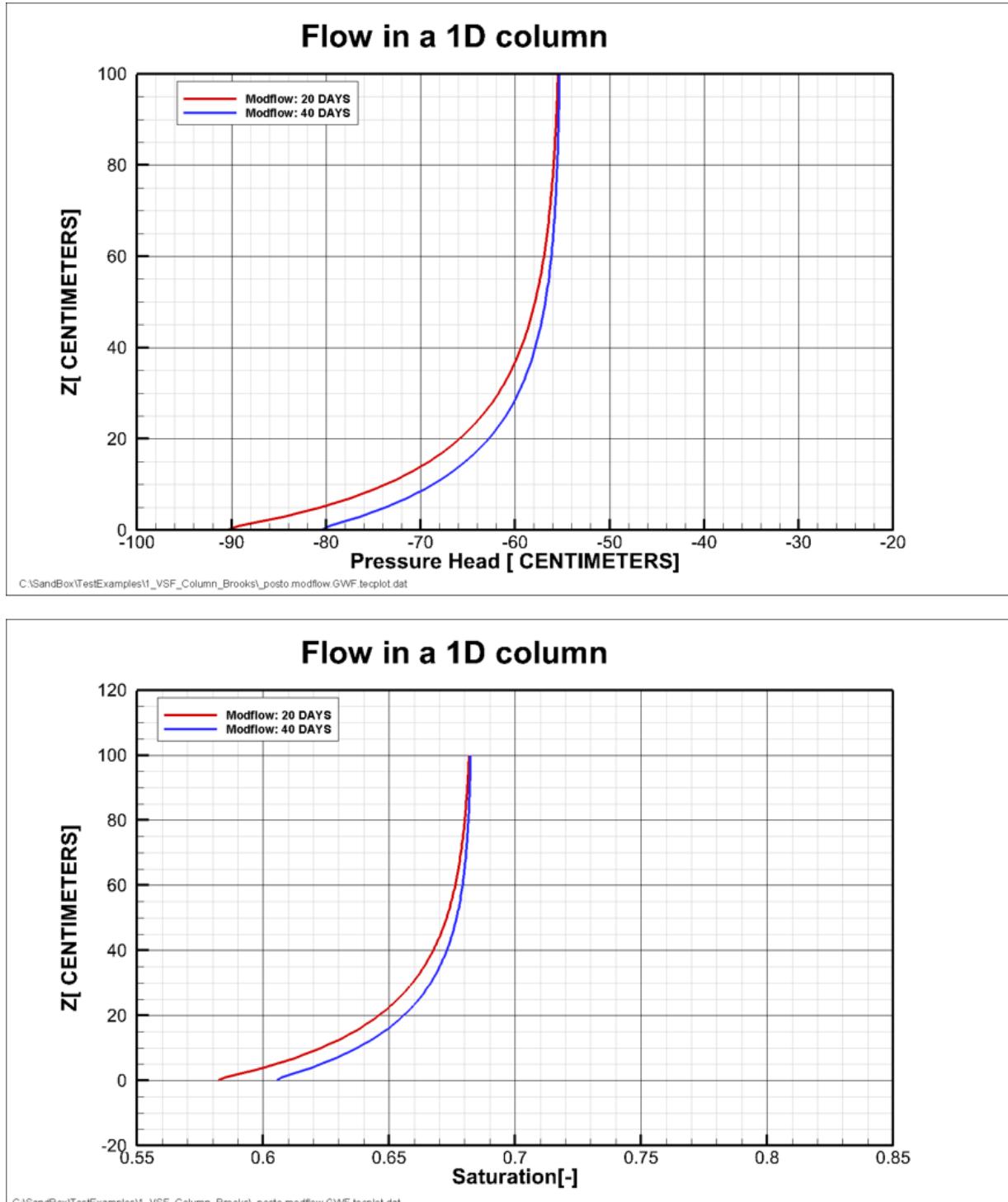
An initial head was assigned as a function of  $z$  (i.e. elevation) as: head=-100.0 cm at  $z=0.0$  cm to head=0.0 cm at  $z=100.0$  cm.

Here is a comparison of pressure head (upper plot) and saturation (lower plot) versus elevation results for MODFLOW-USG<sup>*swf*</sup> and HYDROGEO SPHERE(HGS) at 20 and 40 days:



The two models yield very similar results except near the base of the column.

Just as a demonstration, a second simulation 1\_VSF\_Column\_Brooks was carried out using the Brooks-Corey unsaturated function type, with a Brooks-Corey exponent of 6.5714. Here is a plot of pressure head (upper plot) and saturation (lower plot) versus elevation results MODFLOW-USG<sup>Swf</sup> at 20 and 40 days:



## 5.2 1D Surface Flow

The examples `3_0_SWF_CHD`, `3_1_CLN_for_SWF` and `3_SWF` simulate 1D surface flow on a 100 m long, 1 m wide sloping surface. Analytical and numerical solutions of the diffusive wave and dynamic wave equations for this problem were presented by Govindaraju et al. (1988a,b).

The first two examples compare surface flow in a **SWF** domain (`3_0_SWF_CHD`) versus an equivalent **CLN** domain (`3_1_CLN_for_SWF`). For the **SWF** domain, the following parameter values were used:

Parameter	Value	Unit
Manning's coefficient	$5.48 \times 10^{-2}$	$\text{s m}^{-1/3}$
Depression storage height	0.1	m
Obstruction storage height	0.0	m
Depth for smoothing height 1	$1 \times 10^{-6}$	m
Depth for smoothing height 2	$1 \times 10^{-6}$	m

For the equivalent **CLN** domain, the following parameter values were used:

Parameter	Value	Unit
Longitudinal hydraulic conductivity	$5.48 \times 10^{-2}$	$\text{m s}^{-1}$
Rectangular Width	1.0	m
Rectangular Height	1.0	m

The **CLN** channel was assigned a rectangular geometry, a horizontal direction and assumed an unconfined/Mannings flow treatment.

The following parameter values were used for the **GWF** domain in both cases, although interaction between the surface flow and **GWF** domains was negligible due to the short simulation time of 242 seconds:

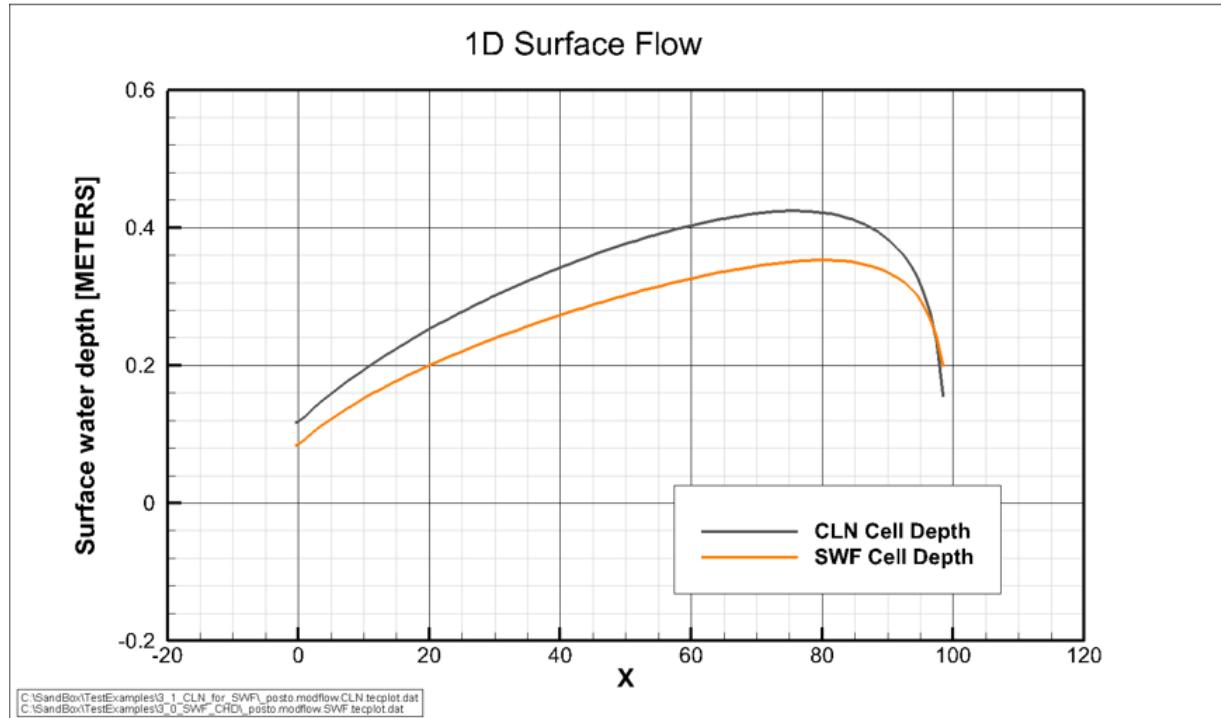
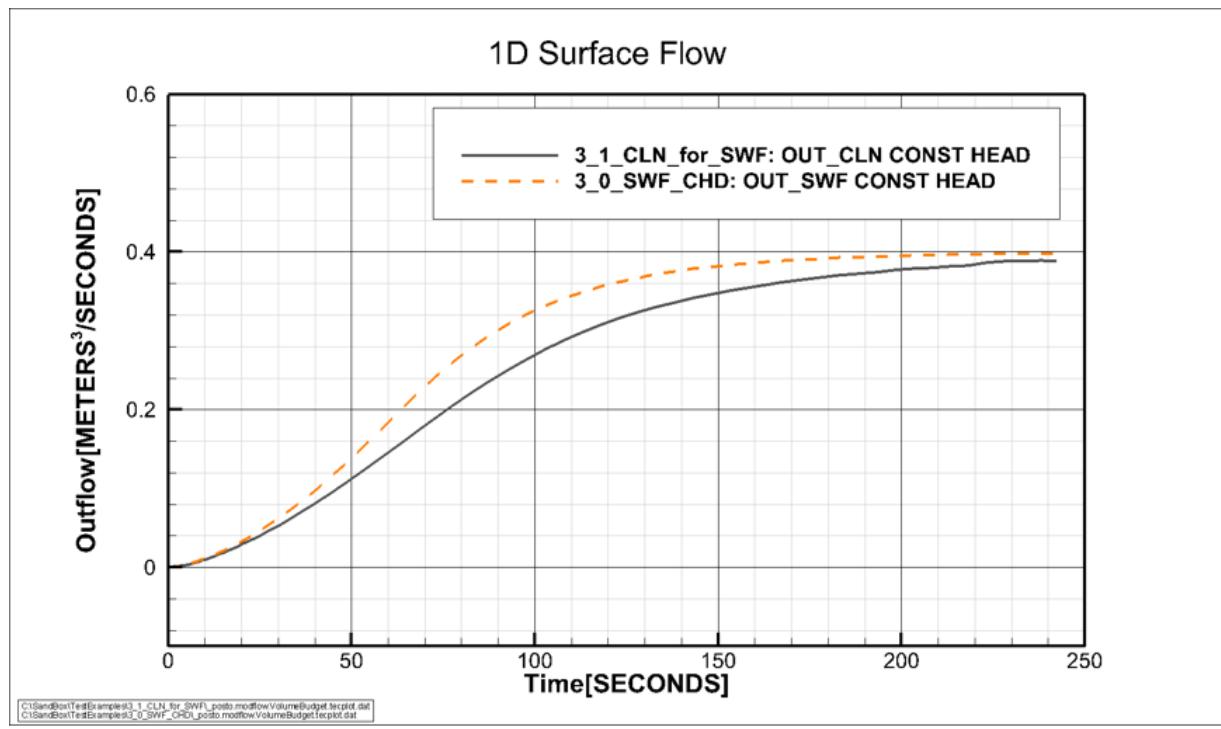
Parameter	Value	Unit
Specific yield (porosity)	0.3	
Hydraulic conductivity	$1.474 \times 10^{-4}$	$\text{m s}^{-1}$
Specific storage coefficient	$1 \times 10^{-4}$	$\text{m}^{-1}$
Van Genuchten Alpha	1.0	$\text{m}^{-1}$
Van Genuchten Beta	5	
Residual saturation	0.3	
Initial surface water depth	$1 \times 10^{-5}$	m

The Van Genuchten unsaturated function type was used.

An initial surface water depth of  $1 \times 10^{-6}$  m was assigned to the **SWF** and **CLN** surface flow domains.

Boundary conditions for the **SWF** and **CLN** surface flow domains consist of a flux of  $4 \times 10^{-3}$  m/s applied to the entire domain and a specified head slightly above ground surface applied to the cell located at the downstream outflow boundary.

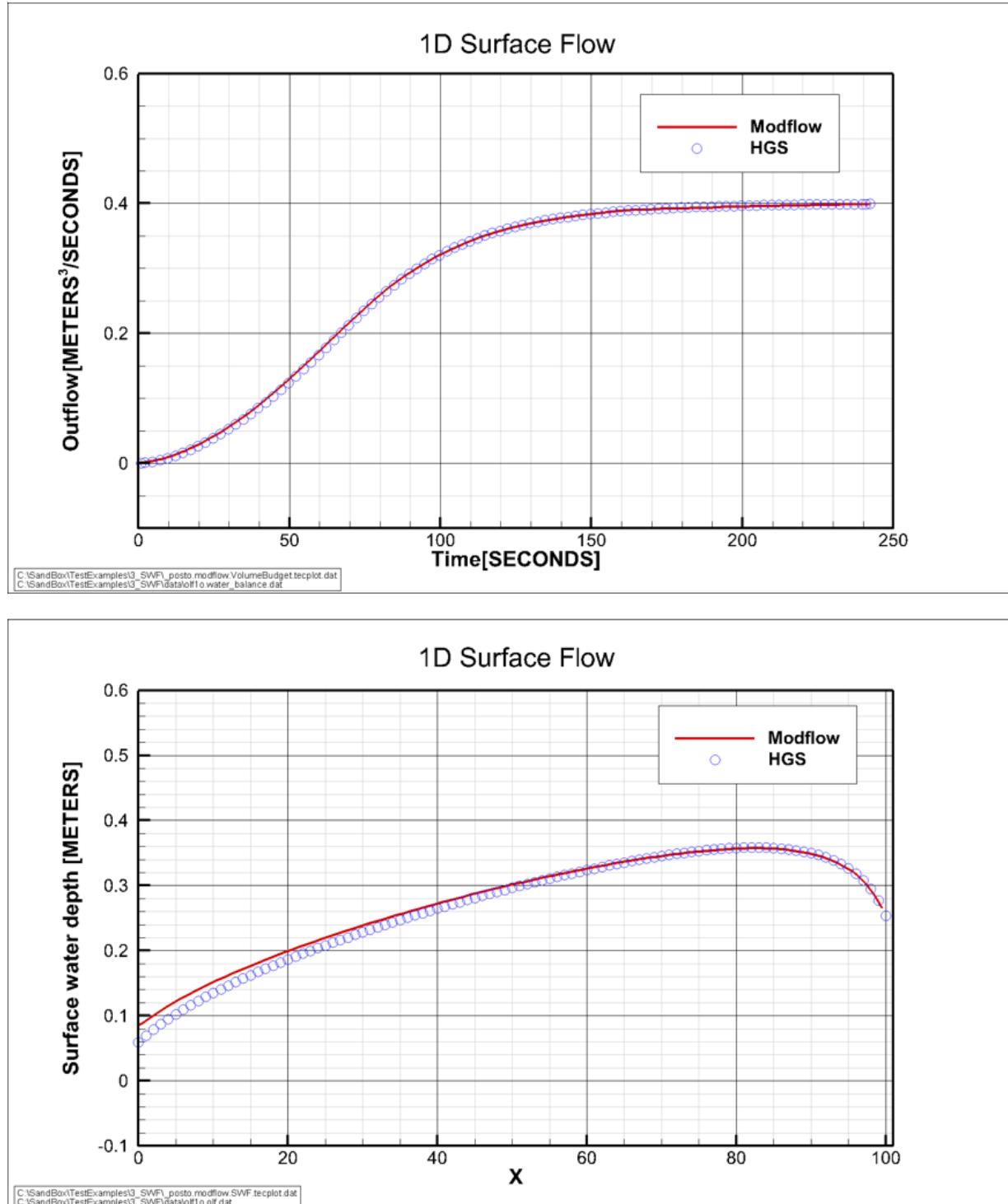
Here is a comparison of outflow versus time (upper plot) and surface water depth versus distance along the slope (lower plot) at the constant head cells for the SWF and CLN domains at the end of the simulation:



Outflow builds up during the simulation and levels off at a value of  $0.4 \text{ m}^3/\text{s}$  (i.e. recharge rate times slope length). Depths increase downstream then drop sharply approaching the constant head outlet cell.

The third example (3\_SWF) is identical to the second example, except a critical depth boundary condition is applied to the SWF domain cell located at the downstream outflow boundary.

Here is a comparison of outflow versus time (upper plot) and surface water depth versus  $x$  (i.e. distance along the slope) (lower plot) at the critical depth cells for the MODFLOW-USG<sup>*swf*</sup> and HYDROGEO-SPHERE models:



## 5.3 2D Surface/subsurface Flow

These two examples compare 2D variably-saturated flow in a hillslope:

**Example 1 (2\_VSF\_Hillslope):** A GWF domain simultaneously receives recharge to and allows drainage from its top layer of cells.

**Example 2 (4\_SWF\_RCH\_CRD):** A fully-coupled GWF-SWF system receives recharge to the SWF domain and allows drainage from the SWF cell located at the downstream outflow boundary.

The following parameter values were used for the GWF domain in both examples:

Parameter	Value	Unit
Specific yield (porosity)	0.1	
Hydraulic conductivity	31.536	$\text{m yr}^{-1}$
Specific storage coefficient	$1 \times 10^{-7}$	$\text{m}^{-1}$
Van Genuchten Alpha	$3.34 \times 10^{-2}$	$\text{m}^{-1}$
Van Genuchten Beta	1.982	
Residual saturation	0.2771	

The Van Genuchten unsaturated function type was used.

For the SWF domain of example 2, the following parameter values were used:

Parameter	Value	Unit
Manning's coefficient	$1.7 \times 10^{-9}$	$\text{yr m}^{-1/3}$
Depression storage height	$1 \times 10^{-6}$	m
Obstruction storage height	$1 \times 10^{-6}$	m
Depth for smoothing height 1	$1 \times 10^{-6}$	m
Depth for smoothing height 2	$1 \times 10^{-6}$	m

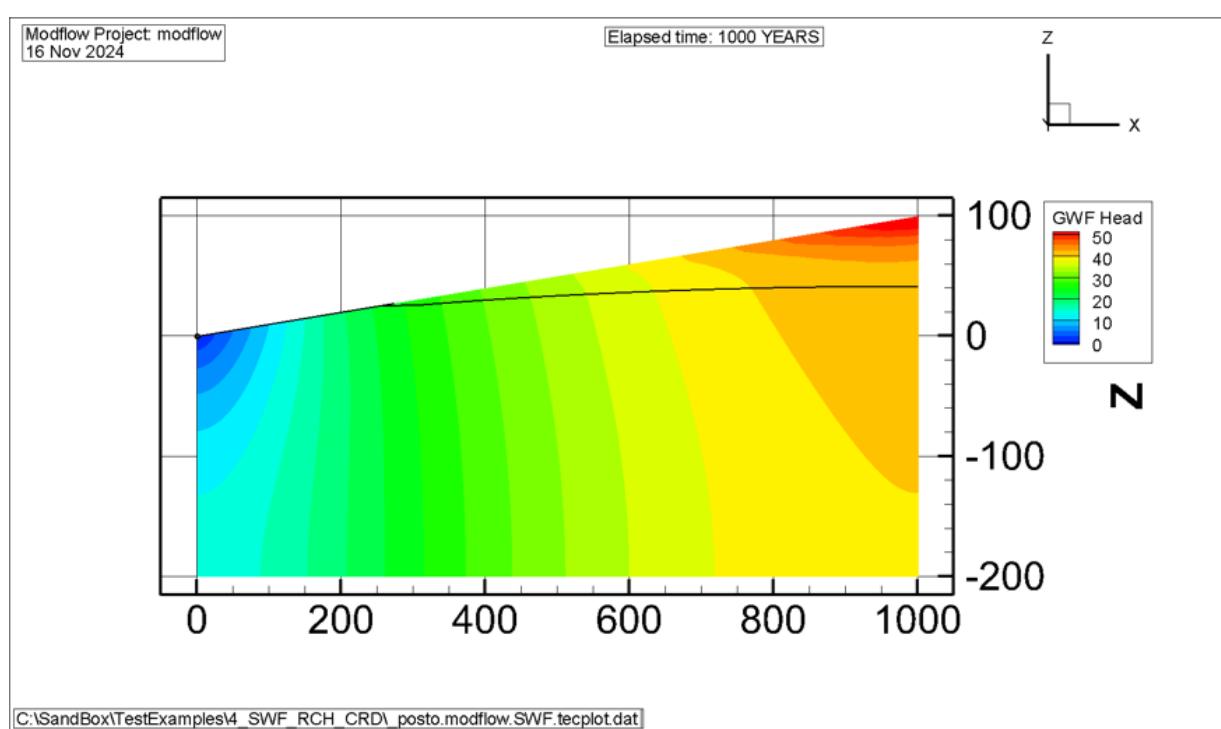
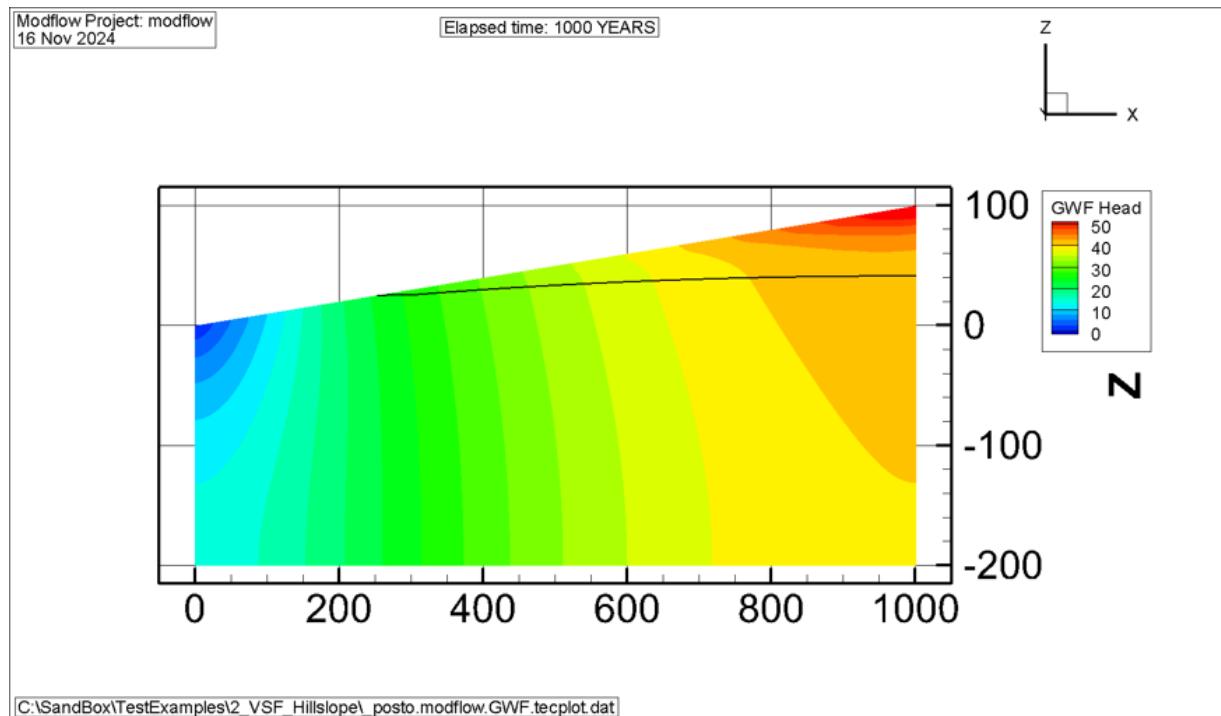
An initial head of 0.0 m was assigned to the GWF domain in both examples.

For example 1, a recharge flux of 0.5 m/yr was applied to the entire top of the GWF domain. The drain boundary condition applied to the top layer of GWF cells was assigned a drain elevation equal to the ground surface elevation and a drain conductance of 1000 m/yr.

For example 2, an initial surface water depth of  $1 \times 10^{-3}$  m was assigned to the SWF domain. A recharge flux of 0.5 m/yr was applied to the entire SWF domain. A critical depth boundary condition was assigned to the SWF cell located at the downstream outflow boundary.

The simulation was run for 1000 years during which time steady-state conditions were achieved.

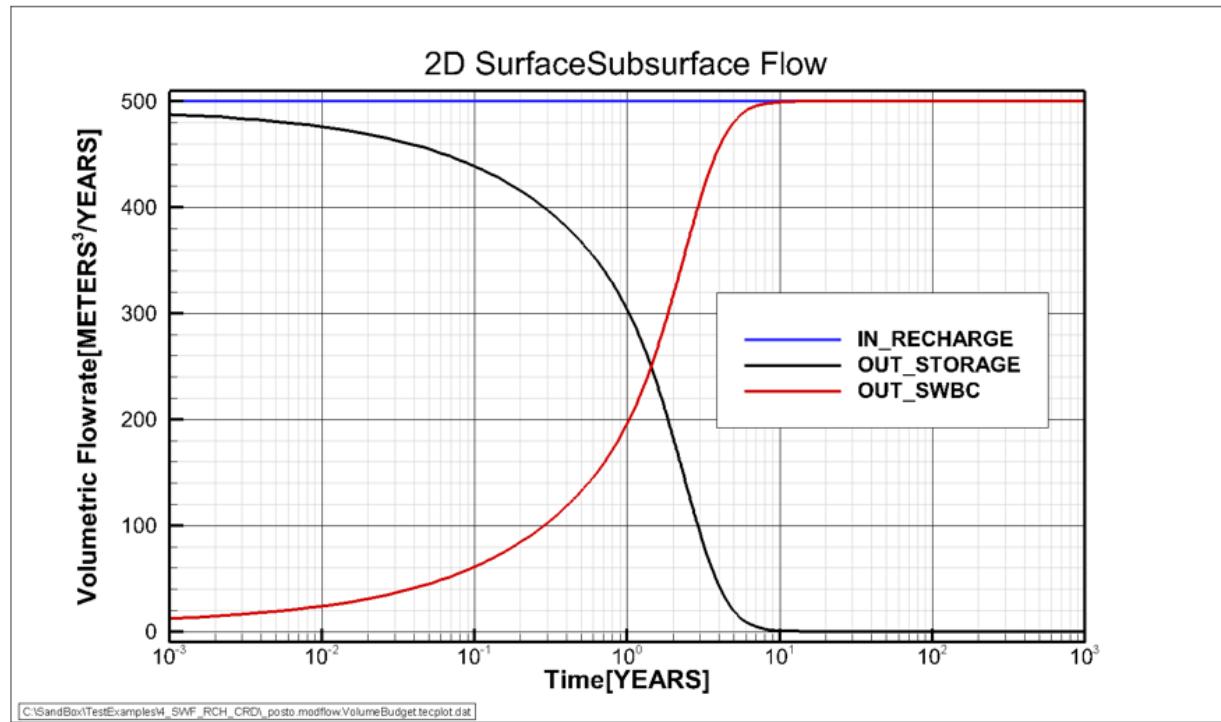
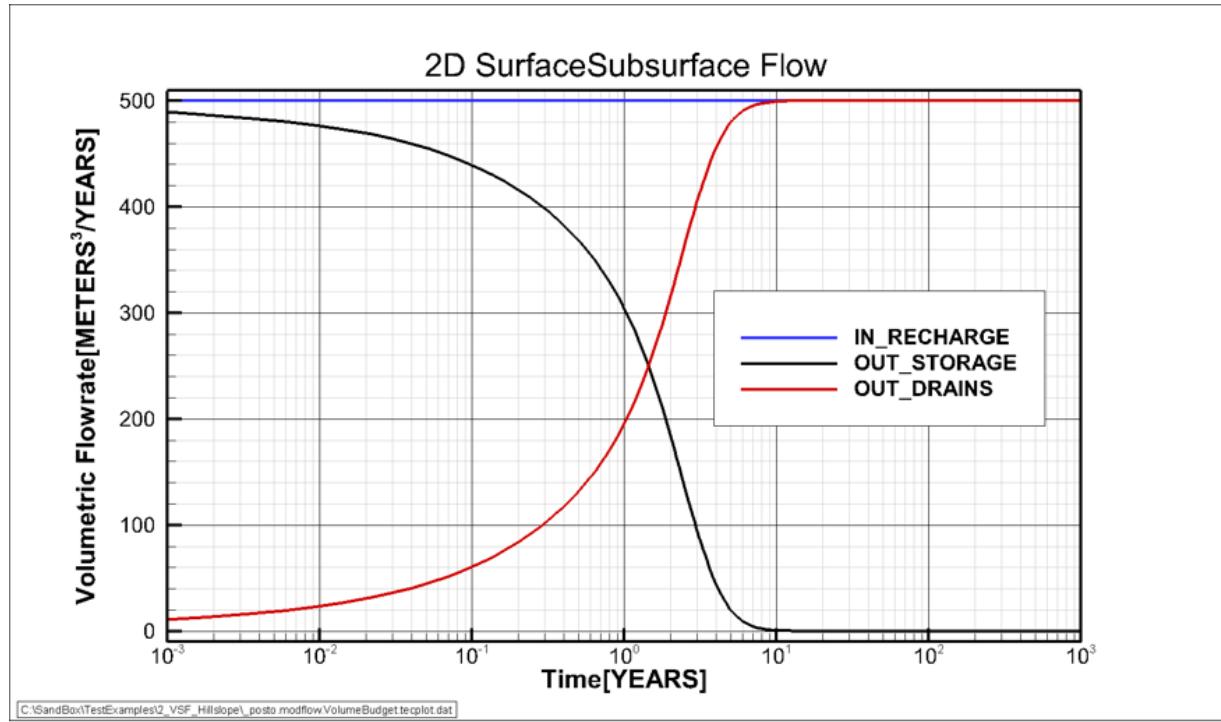
Here is a comparison of hydraulic head at steady-state for examples 1 (upper plot) and 2 (lower plot):



C:\SandBox\TestExamples\2\_VSF\_Hillslope\posto.modflow.GWF.tecplot.dat

The results are similar for both examples. The water table is shown as a heavy black line. In the lower plot, the black line extends into the SWF domain and represents surface water depths greater than  $7.7 \times 10^{-6}$  m. The critical depth boundary condition that was assigned to the SWF cell located at the downstream outflow boundary is indicated by the small black sphere.

Here is a comparison of volumetric flow rates versus time for examples 1 (upper plot) and 2 (lower plot):



The results are essentially identical for both examples, showing that the critical depth boundary outflow (OUT\_SWBC) is being calculated correctly. In both examples, water coming out of storage in the GWF domain is balanced by discharge along the top (example 1) or at the downstream outflow boundary (example 2).

## 5.4 3D Surface/subsurface Flow: Field Study of Abdul

These three examples simulate the fully three-dimensional surface/subsurface flows observed in experiments conducted at Canadian Forces Base Borden, in Ontario, Canada, by Abdul (1985):

**Example 1 (6\_Abdul\_Prism\_Cell):** A fully-coupled GWF-SWF system receives recharge to the SWF domain and allows drainage from SWF cells located at the downstream outflow boundary. The template and SWF meshes are composed of 3-node triangular elements while the GWF mesh is composed of 6-node prismatic elements. The mesh-centred control volume approach is used to define the MODFLOW-USG<sup>Swf</sup> cells.

**Example 2 (\_Abdul\_Prism\_Cell\_nc):** Similar to example 2 except the node-centred control volume approach is used to define the MODFLOW-USG<sup>Swf</sup> cells.

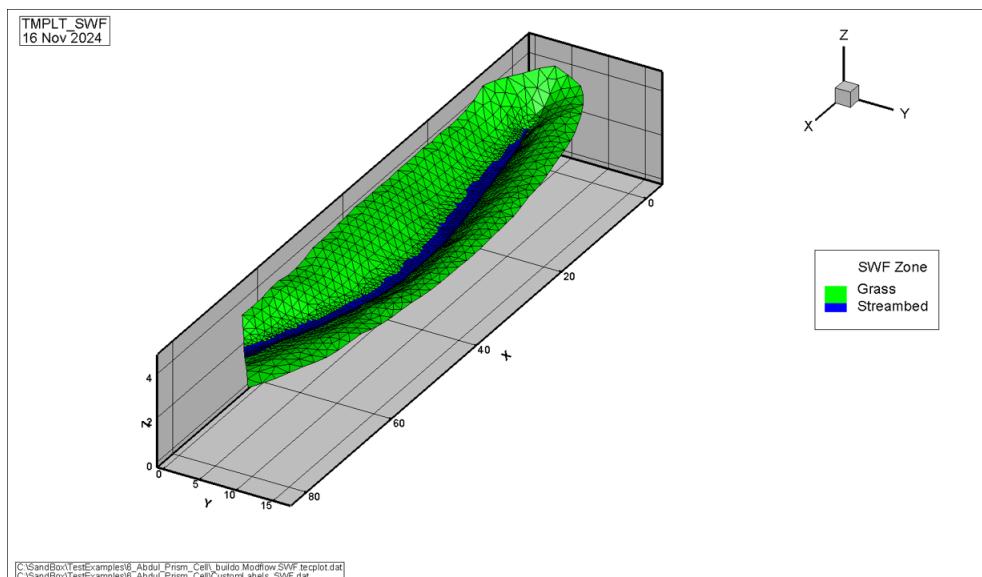
**Example 3 (6\_Abdul\_MODHMS):** Similar to example 1 except The template and SWF meshes are composed of 4-node rectangular elements while the GWF mesh is composed of 8-node hexahedral elements.

The following parameter values were used for the GWF domain in all 3 examples:

Parameter	Value	Unit
Specific yield (porosity)	0.34	
Hydraulic conductivity	$1 \times 10^{-5}$	$\text{m s}^{-1}$
Specific storage coefficient	$1.2 \times 10^{-7}$	$\text{m}^{-1}$
Van Genuchten Alpha	1.9	$\text{m}^{-1}$
Van Genuchten Beta	6	
Residual saturation	0.18	

The Van Genuchten unsaturated function type was used.

For examples 1 and 2, the SWF domain was subdivided into 2 zones called grass and streambed as shown here for example 1:



These parameter values were used for the grass zone for example 1 and 2:

Parameter	Value	Unit
Manning's coefficient	0.3	$s m^{-1/3}$
Depression storage height	0.1	m
Obstruction storage height	0.0	m
Depth for smoothing height 1	$1 \times 10^{-6}$	m
Depth for smoothing height 2	$1 \times 10^{-6}$	m

These parameter values were calibrated and used for the streambed zone in examples 1 and 2:

Parameter	Value	Unit
Manning's coefficient	0.01	$s m^{-1/3}$
Depression storage height	0.1	m
Obstruction storage height	0.0	m
Depth for smoothing height 1	$3.75334 \times 10^{-3}$	m
Depth for smoothing height 2	$1.26394 \times 10^{-3}$	m

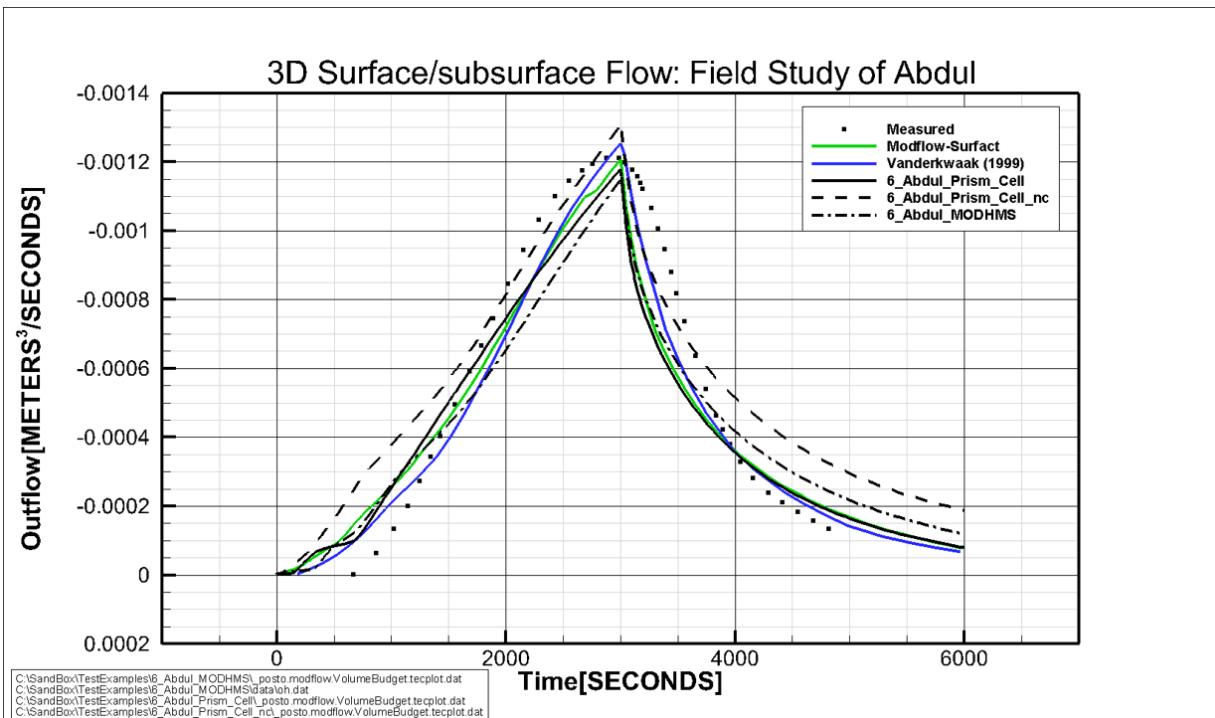
Example 3 used a single SWF zone, which used these calibrated parameter values:

Parameter	Value	Unit
Manning's coefficient	0.02	$s m^{-1/3}$
Depression storage height	0.1	m
Obstruction storage height	0.0	m
Depth for smoothing height 1	$1.19106 \times 10^{-2}$	m
Depth for smoothing height 2	$2.49907 \times 10^{-2}$	m

The 3 examples shared the following initial and boundary conditions:

- An initial head of 2.78 m was assigned to the GWF domain.
- An initial surface water depth of  $1 \times 10^{-3}$  m was assigned to the SWF domain.
- A recharge of  $5.56 \times 10^{-6}$  m/s was applied to the SWF domain for a time of 3000 s (stress period 1) then the recharge was reduced to 0.0 m/s for an additional 3000 s (stress period 2).
- A critical depth boundary was assigned to selected SWF cells located at the downstream end of the stream channel.

Here is a comparison of measured and simulated stream outflow versus time for the Abdul field study.



Simulation results are presented for the 3 examples as well as from HYDROGEO SPHERE and MODFLOW-SURFACT. The results are similar for all cases.

# Chapter 6

## References

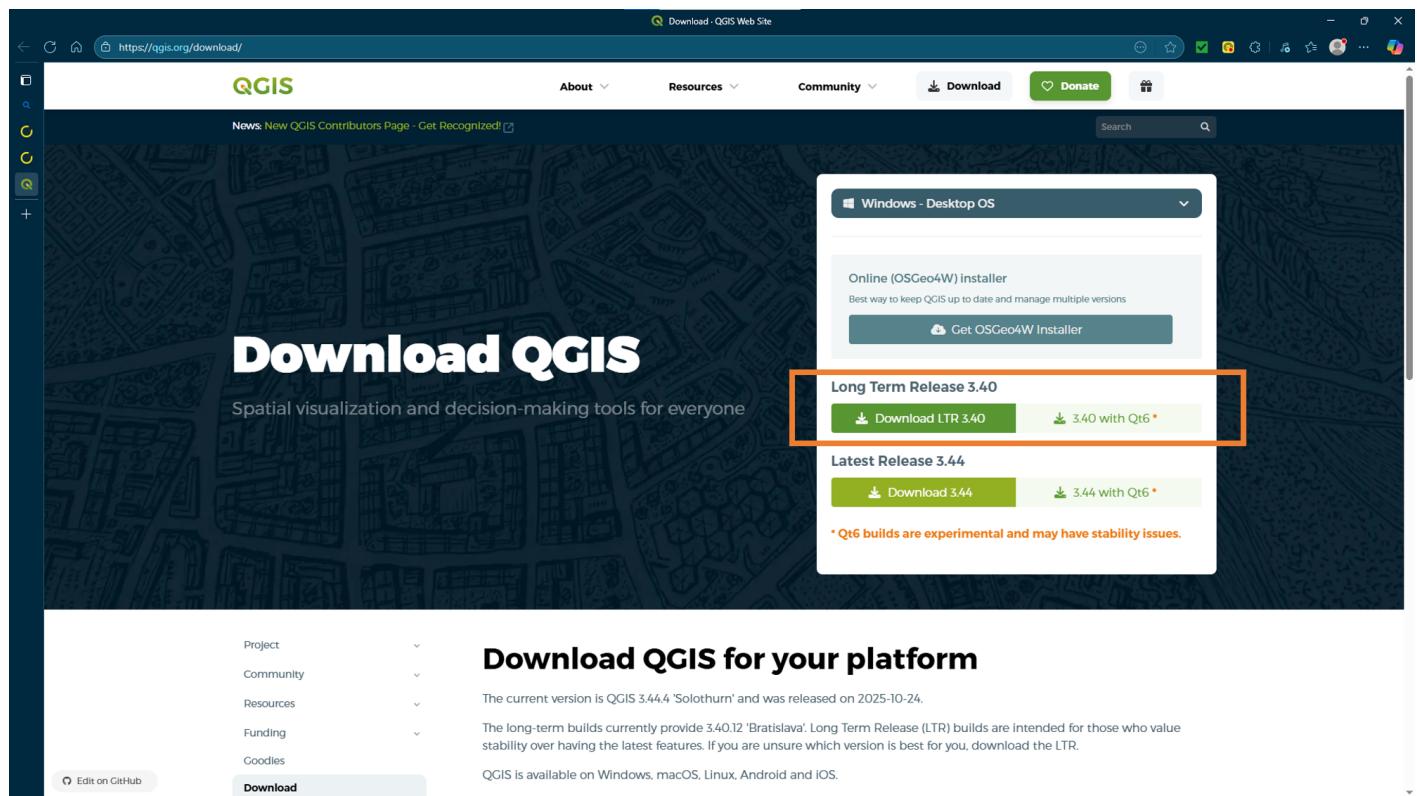
- Abdul, A.S., 1985. Experimental and Numerical studies of the effect of the capillary fringe on stream-flow generation, Ph.D. Thesis, University of Waterloo, Waterloo, Ontario, Canada, 210 pp.
- Govindaraju, R.S., S.E. Jones and M.L. Kavvas, 1988a. On the Diffusion Wave Model for Overland Flow 1. Solution for steep slopes, Water Resour. Res., 24(5), 734–744.
- Govindaraju, R.S., S.E. Jones and M.L. Kavvas, 1988b. On the Diffusion Wave Model for Overland Flow 2. Steady State Analysis, Water Resour. Res., 24(5), 745–754.

# Appendix A

## QGIS Useage

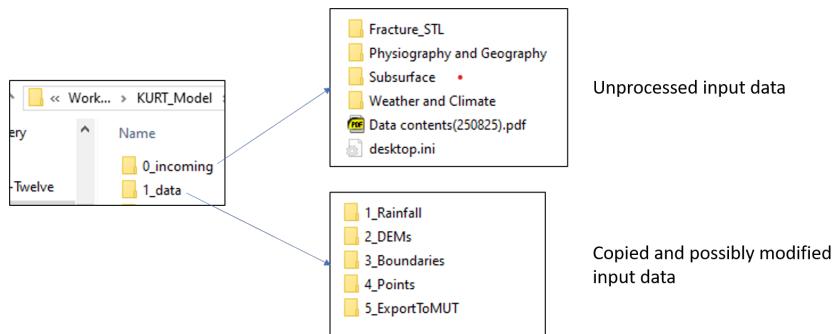
### A.1 QGIS Set-up

QGIS software can be downloaded from <https://qgis.org/download/> where you can either make a donation or skip directly to the download page shown here:



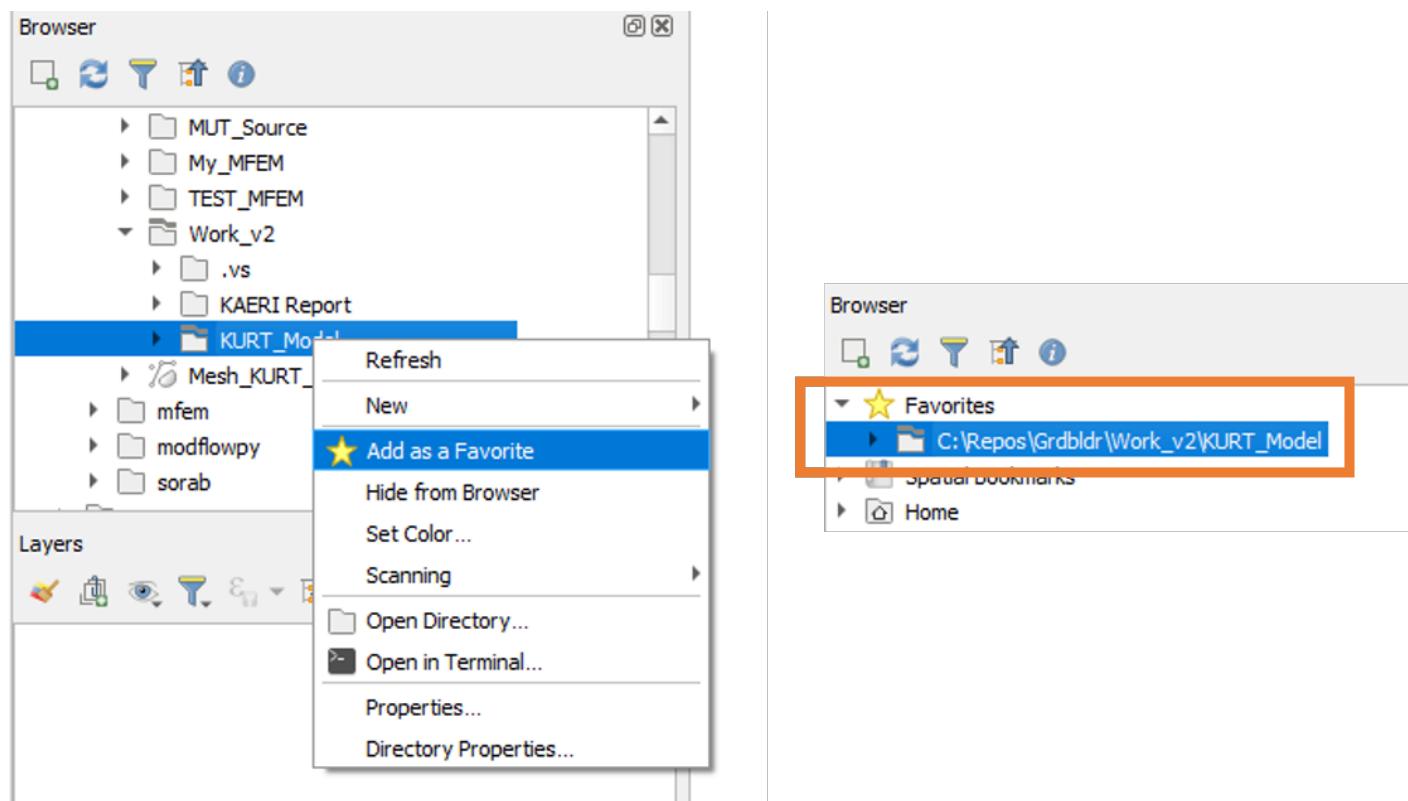
We recommend downloading the Long Term Release version, which may have a different version number than shown above.

When working on specific projects with QGIS, we recommend creating a folder inside your project directory called, for example, KURT\_Model\0\_incoming and copying any raw project conceptual data files (e.g. QGIS-compatible, raster and shapefiles, MICROSOFT EXCEL files etc.) into it, as shown here:



When using and modifying the data in QGIS we then recommend saving it in a separate working folder called, for example, KURT\_Model\1\_data

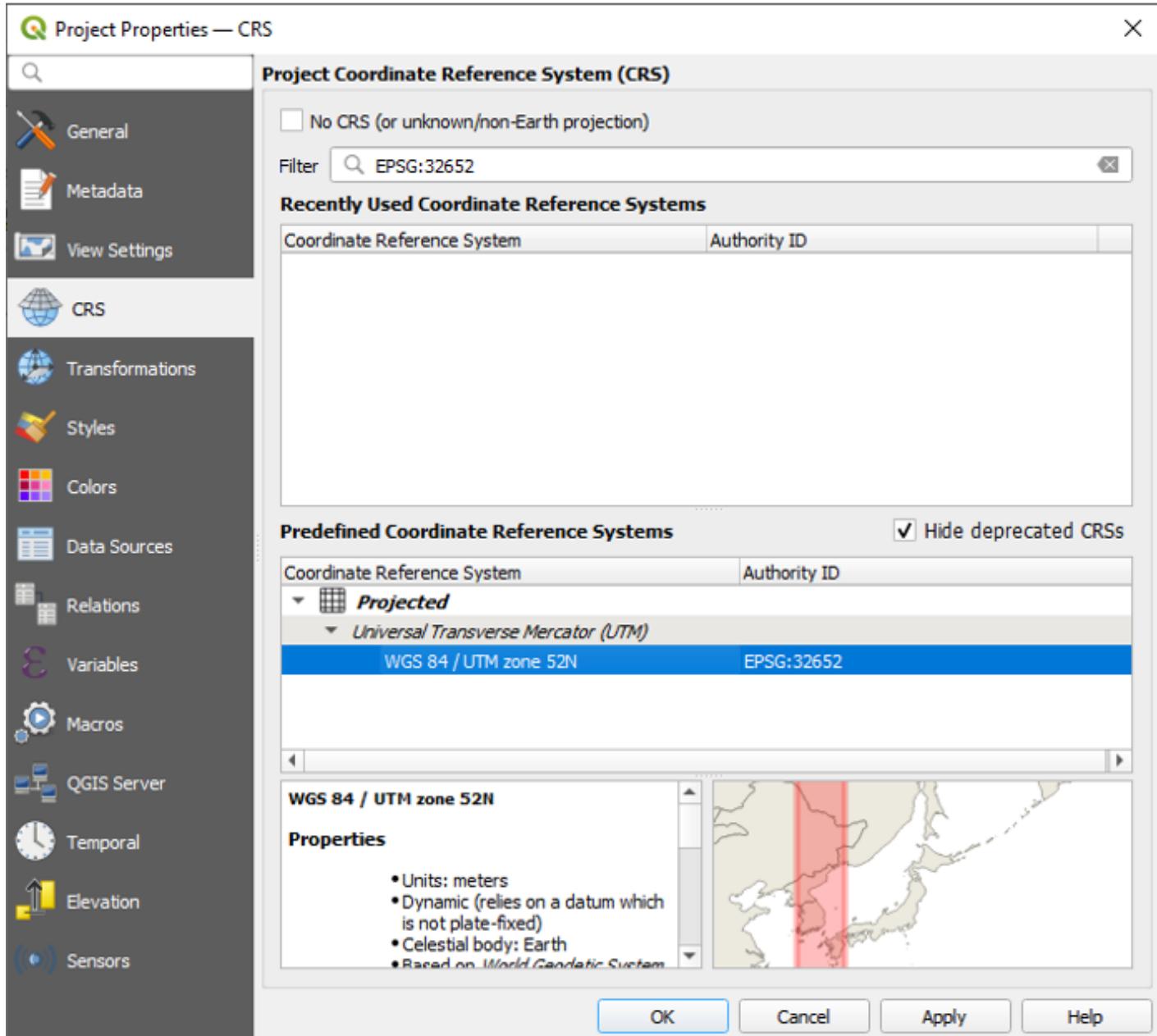
In QGIS, we can navigate to any folder and add it to the list of favourites as a convenience:



All QGIS project and working data files should be saved in this folder (or sub-folders contained in it) to facilitate the sharing of QGIS projects and their associated data.

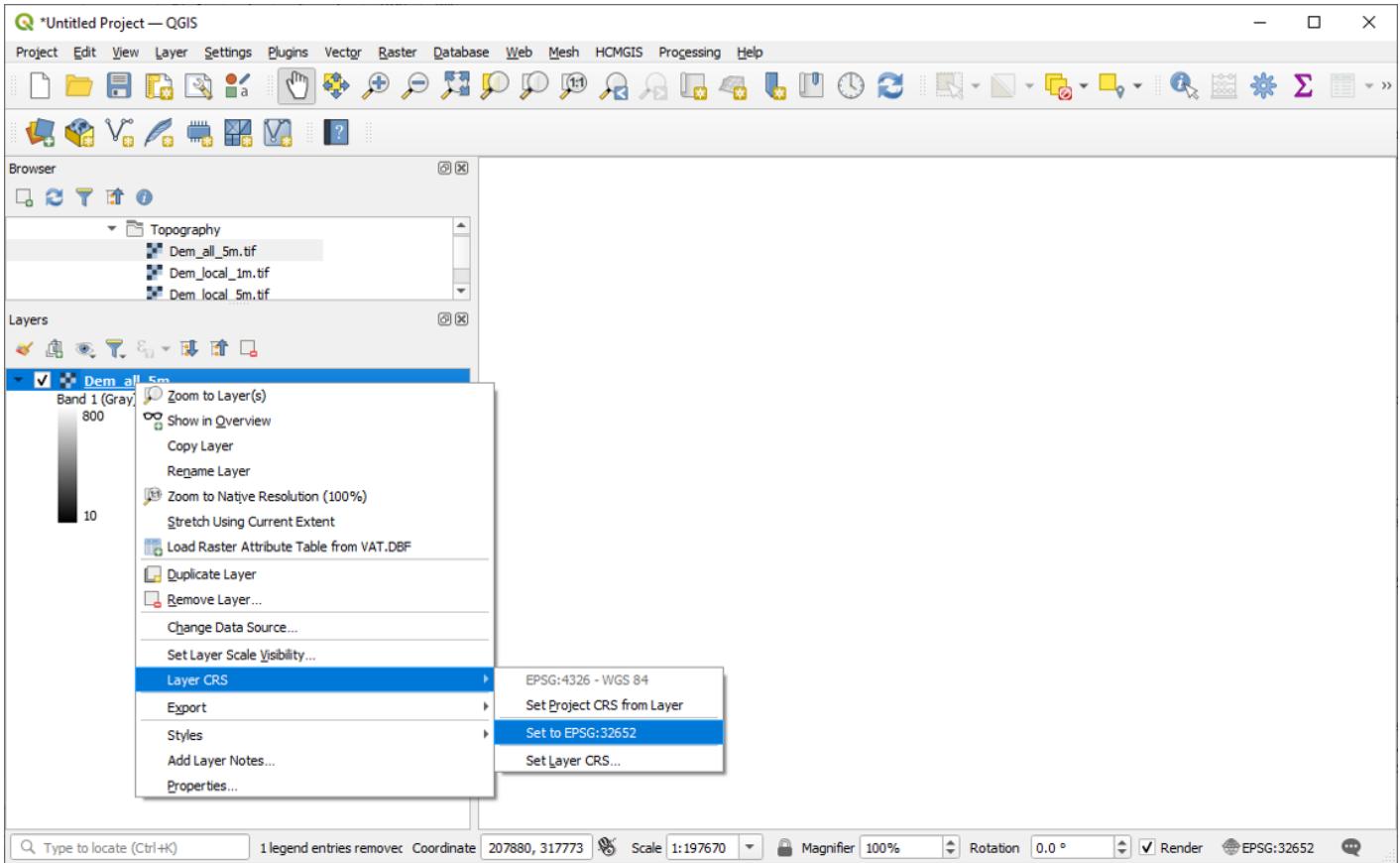
## A.2 Setting the Coordinate Reference System (CRS)

To change the displayed units from degrees latitude and longitude to metres, apply a pre-defined Universal Transverse Mercator (UTM) projection by double-clicking the Current CRS field to open the Project Properties - CRS dialogue:



In the example above the chosen projection is EPSG:32652 – WGS 84 / UTM zone 52N, which includes South Korea. Initially, there may be no Recently Used Coordinate Reference Systems to choose from so use the search field to locate it with the string, e.g. EPSG:32652. Once the desired projection is located choose it and click the Apply button.

Now project the DEM by choosing Dem\_all\_5m.tif\Layer CRS\Set Layer CRS... to open the dialogue or Dem\_all\_5m.tif\Layer CRS\Set to EPGS:32652 if it appears in the menu:

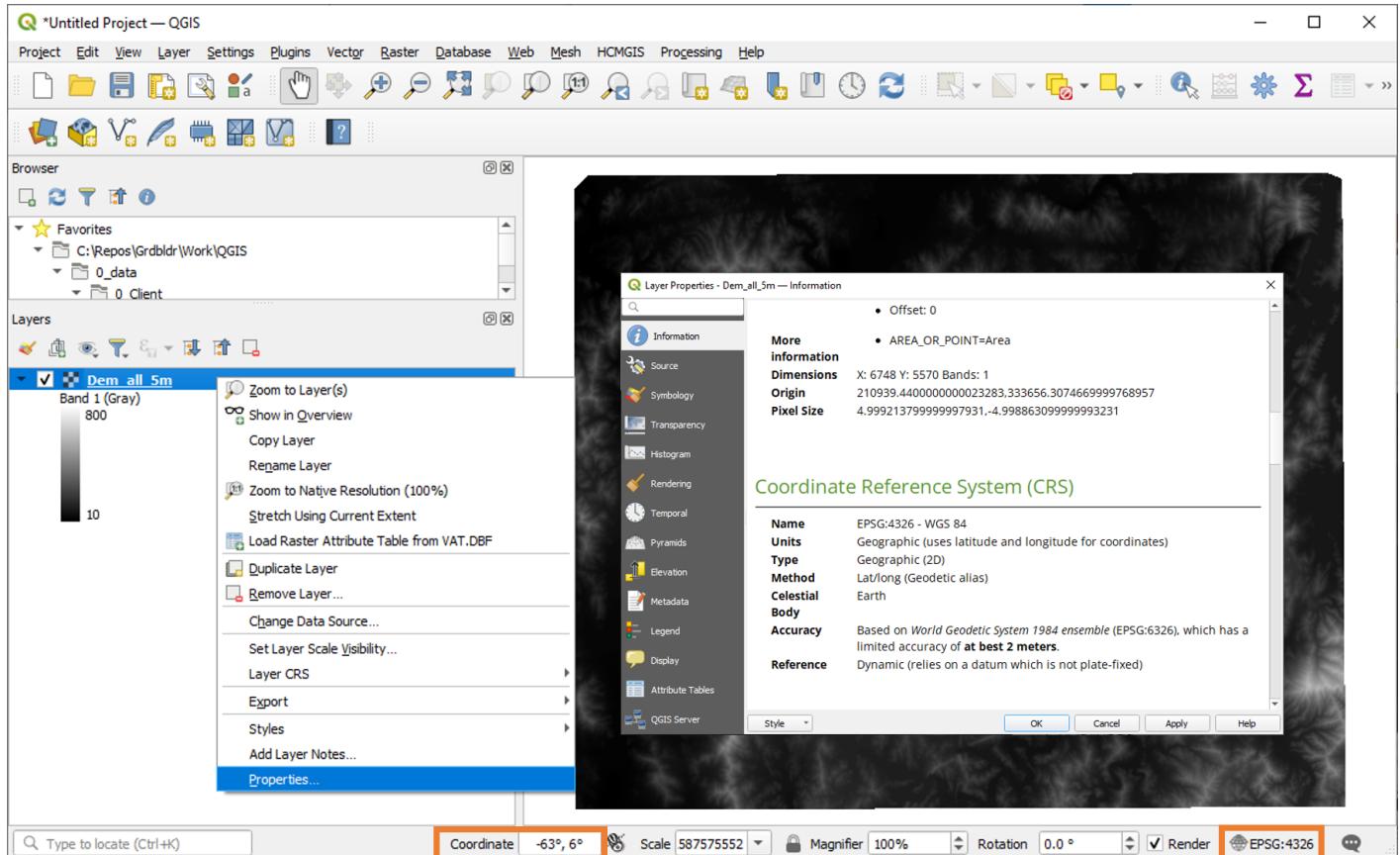


To re-centre the DEM, choose Dem\_all\_5m.tif\Zoom to Layer(s). Note that the displayed coordinates are now in metres.

## A.3 Layers

### A.3.1 Layer Properties

The layer properties dialogue can be opened by double-clicking on the layer name or by right-clicking and choosing **Properties** from the drop-down menu.<sup>1</sup>



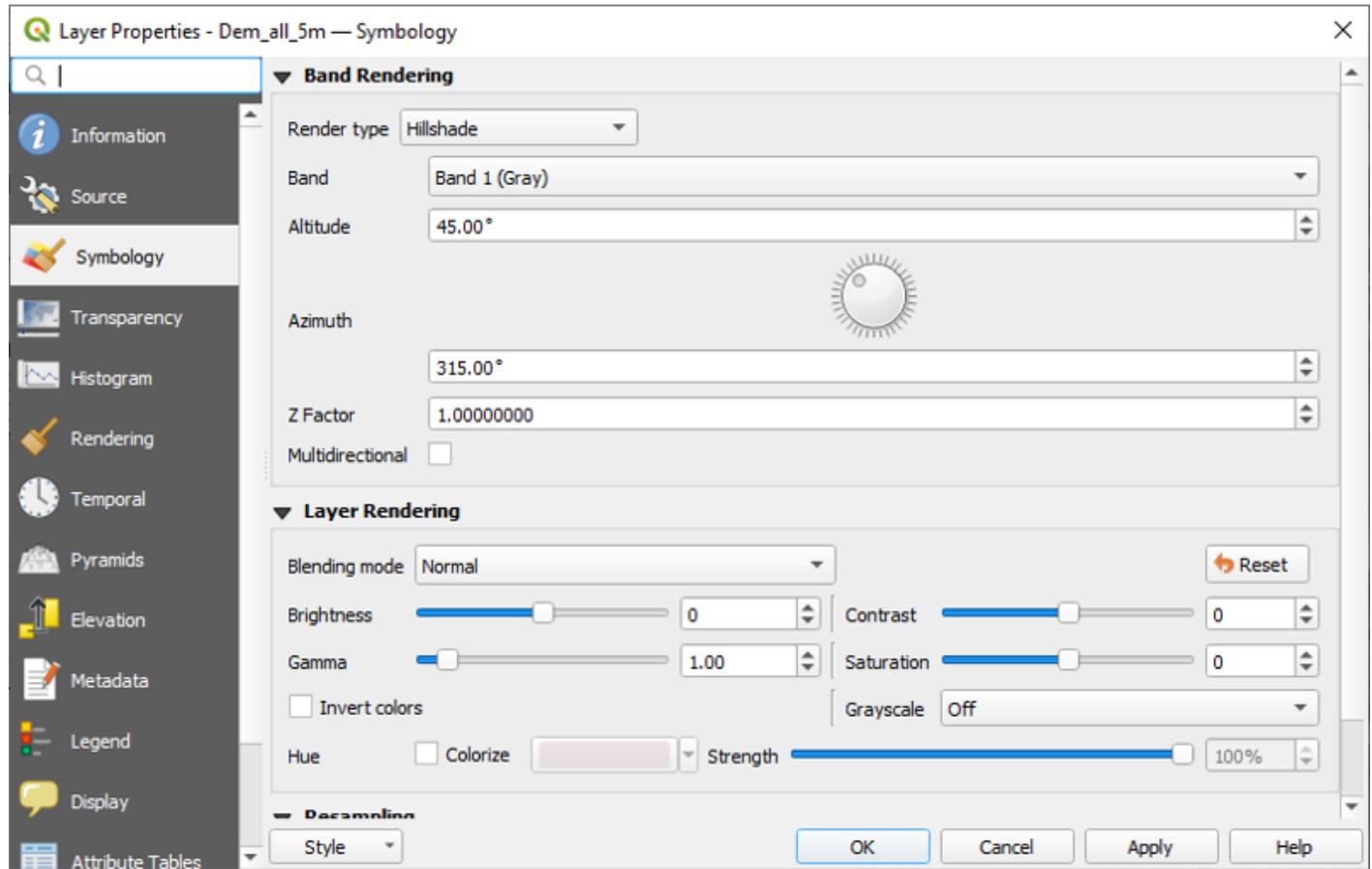
Note the following in the example shown above:

- Properties for the raster layer layer **Dem\_all\_5m** are being viewed.
- The raster has a pixel size of about 5-by-5-metres
- The Coordinate Reference System (CRS) is in units of degrees latitude and longitude.
- The Current CRS is shown in the lower right hand corner of the QGIS window.

<sup>1</sup>Note that detailed documentation for QGIS tools is often available through the **Help** button in the lower right corner of the dialogue.

### A.3.2 Layer Appearance (symbology)

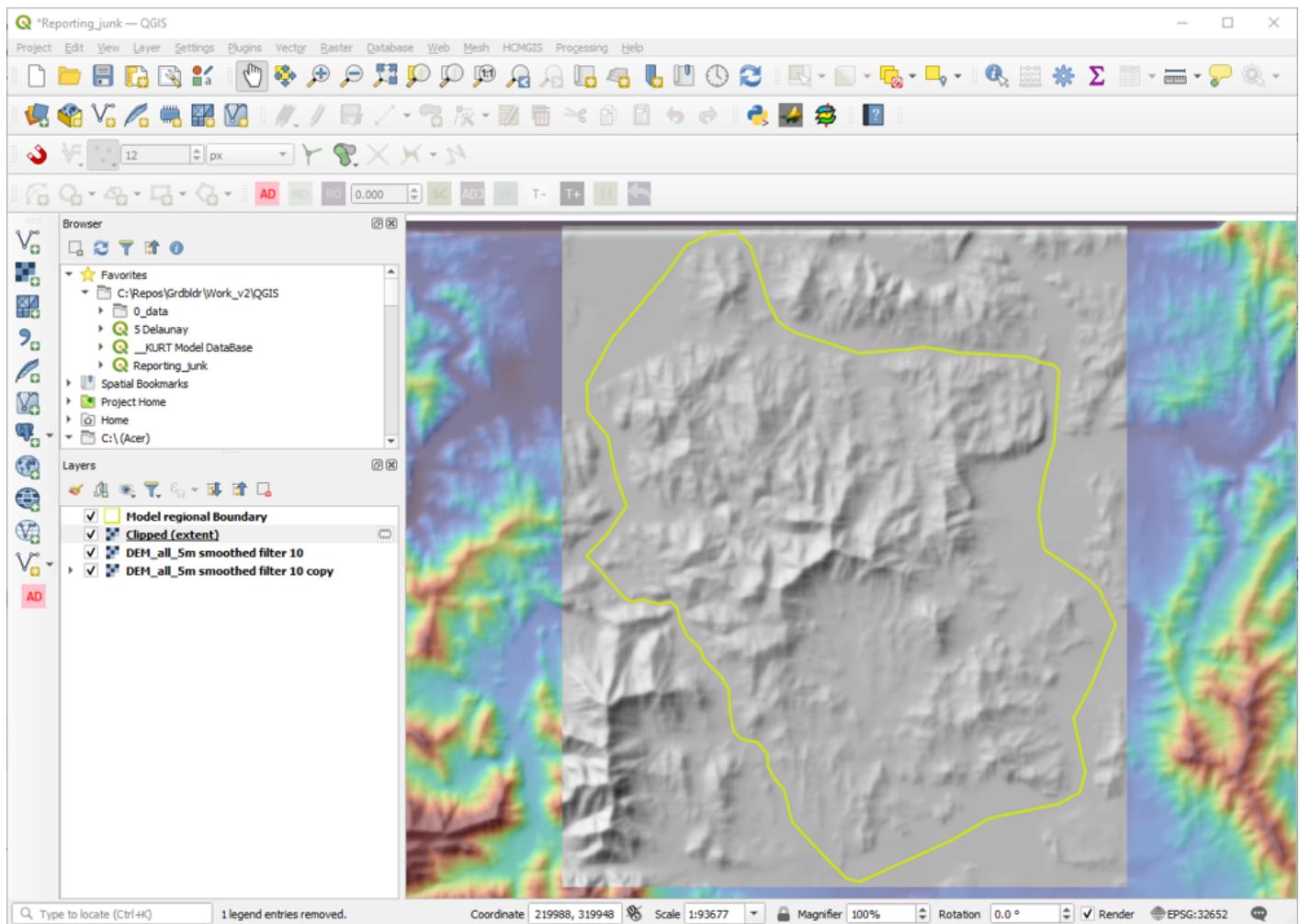
The layer symbology dialogue can be opened by double-clicking on the layer name or by right-clicking and choosing Properties from the drop-down menu:



In the example above we have chosen Render Type - Hillshade for the raster layer Dem\_all\_5m.tif\Properties\Symbology

### A.3.3 Layer Clipping

Once the extents of the model domain have been determined, the DEM can be clipped to a smaller size which saves disk space. Clip the raster by choosing (Raster\Extraction\Clip by extent) and then the Clipping extent option\Draw on Map Canvas. The mouse cursor will be shown as a 'plus' sign. Click and drag a rectangle delimiting the area to take into account, in this case enclosing the regional boundary. The resulting clipped rectangular region is shown below as a hillshade inside the original unclipped DEM.



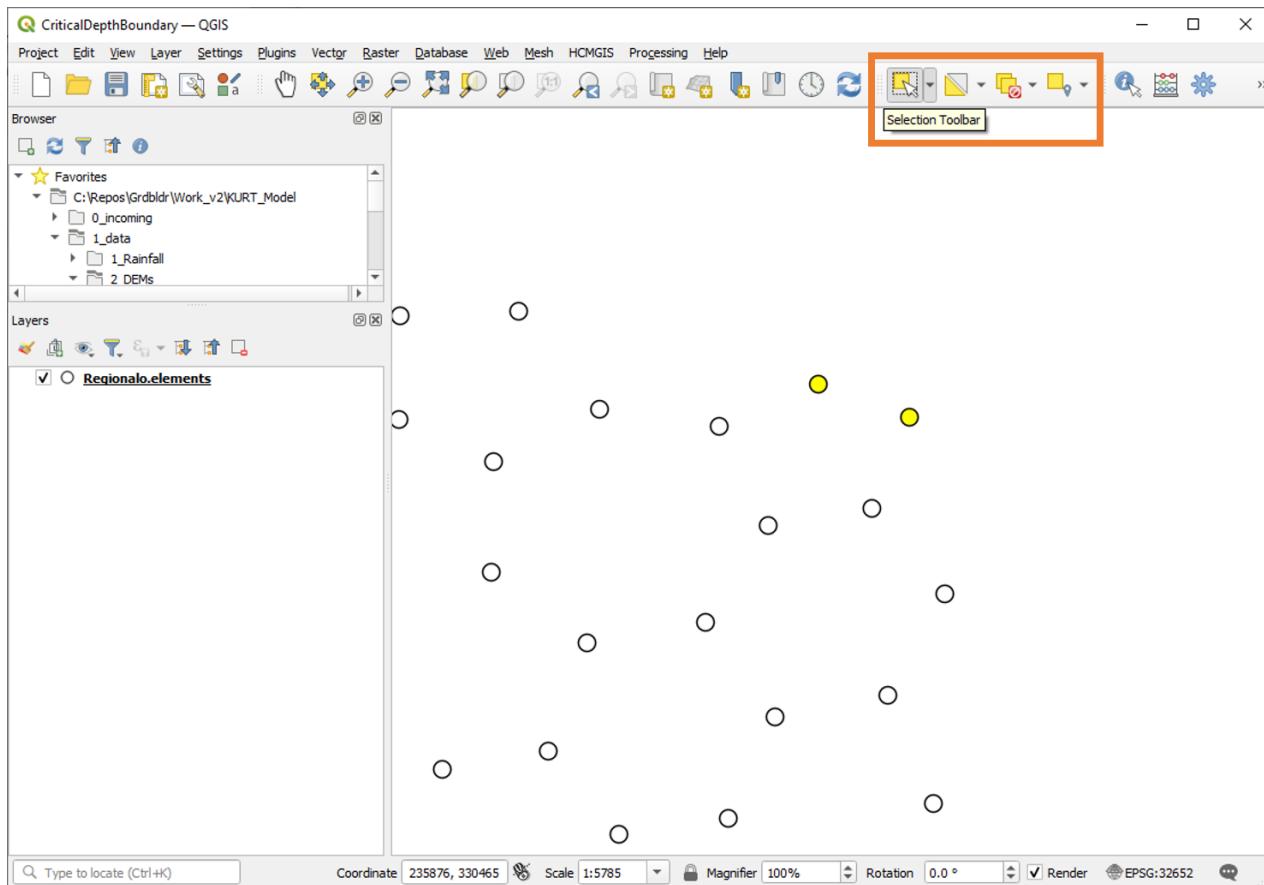
## A.3.4 Vector Layers

### A.3.4.1 Loading Vector Layers

To load a vector layer, simply drag and drop the vector file into the QGIS workspace:

#### A.3.4.2 Selecting Features

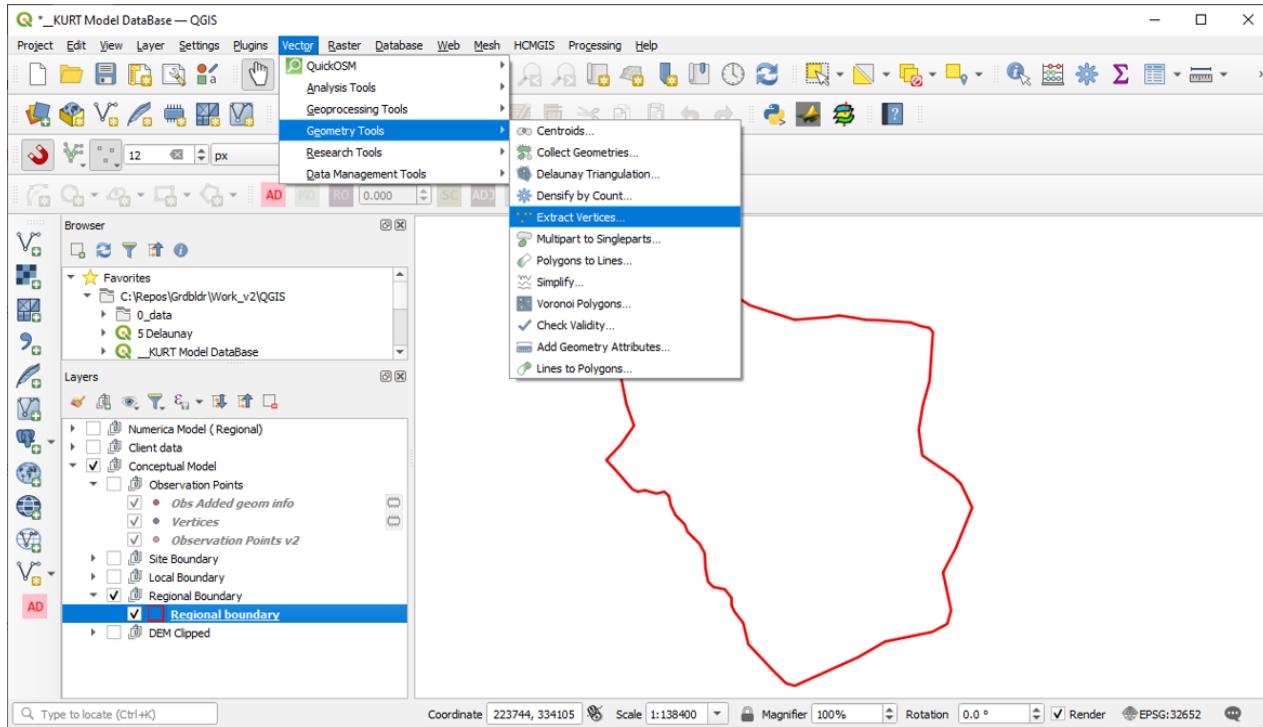
In QGIS features can be selected using the Selection Toolbar, shown below in the orange box:



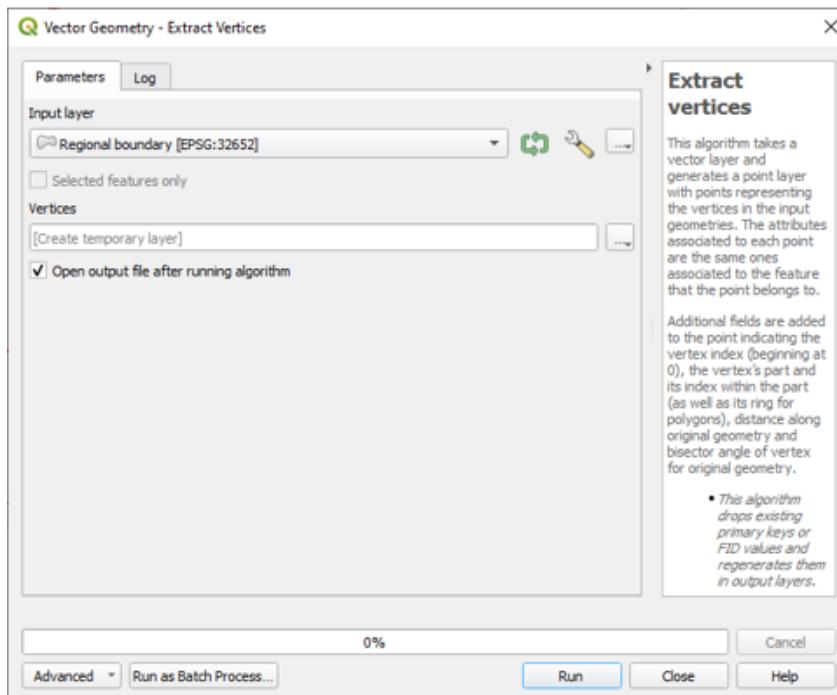
In this example, we are using the option **Select features by area or single click** to choose points (white circles) in a vector layer called **Regionalo.elements**. The currently selected points are shown here as yellow-highlighted circles. To choose multiple points, hold down the **shift** key while left-clicking on the feature.

### A.3.4.3 Extract Vertices

In QGIS vertices can be extracted from an active vector layer by choosing Vector\Geometry Tools\Extract Vertices as shown below:

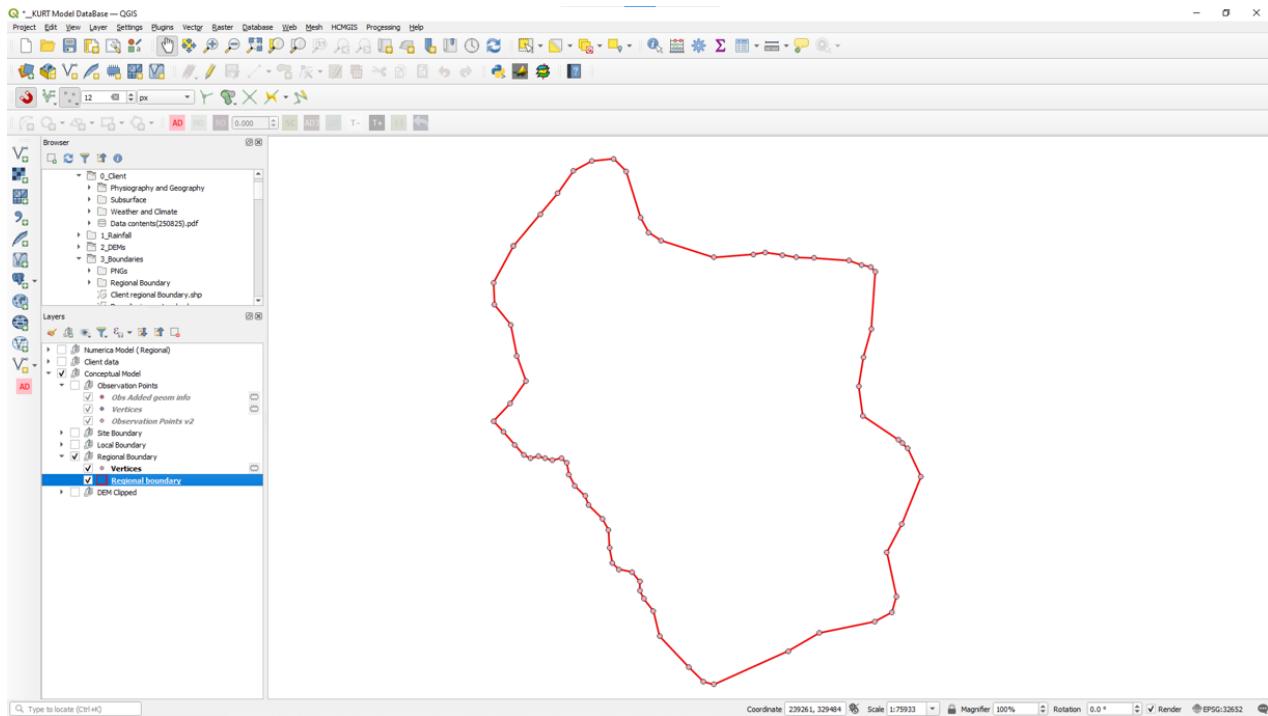


This opens the Extract Vertices dialogue:

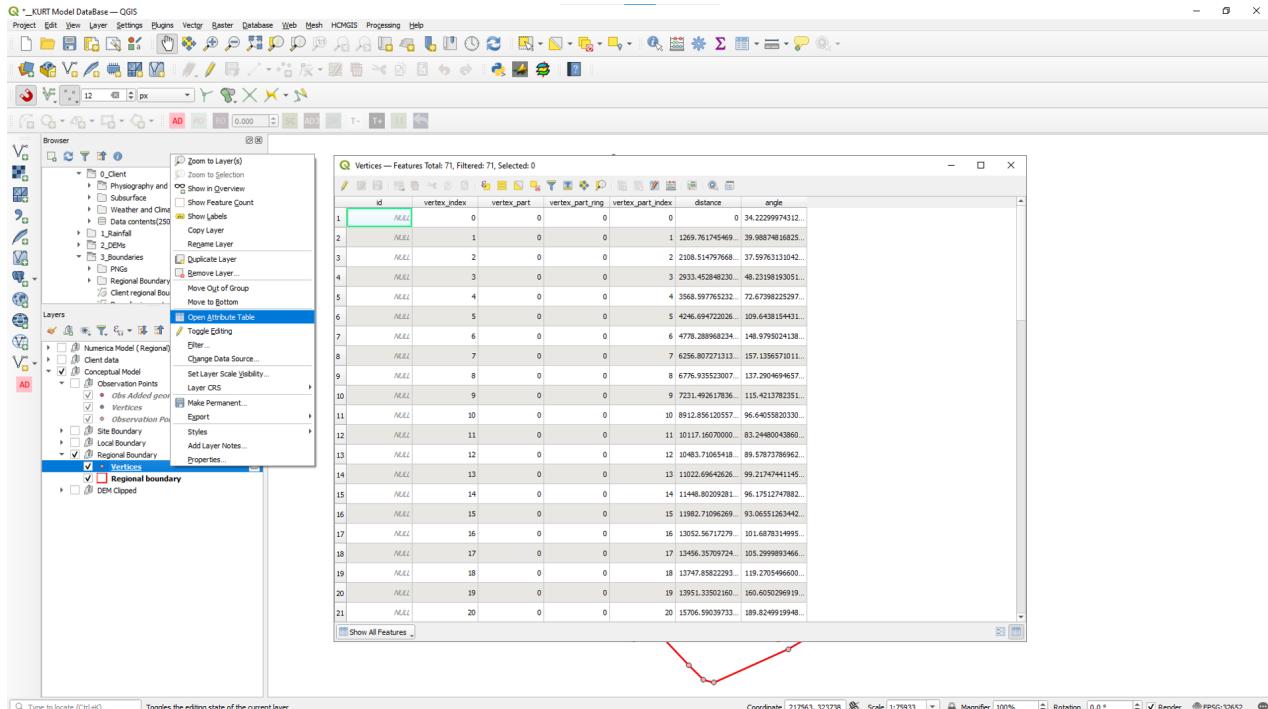


The active layer, in this case Regional Boundary, is shown in the Input layer field and can be changed if desired. Choose Run.

This creates a new layer called **Vertices**, and adds vertex locations along the local boundary as shown here by the pink circles:



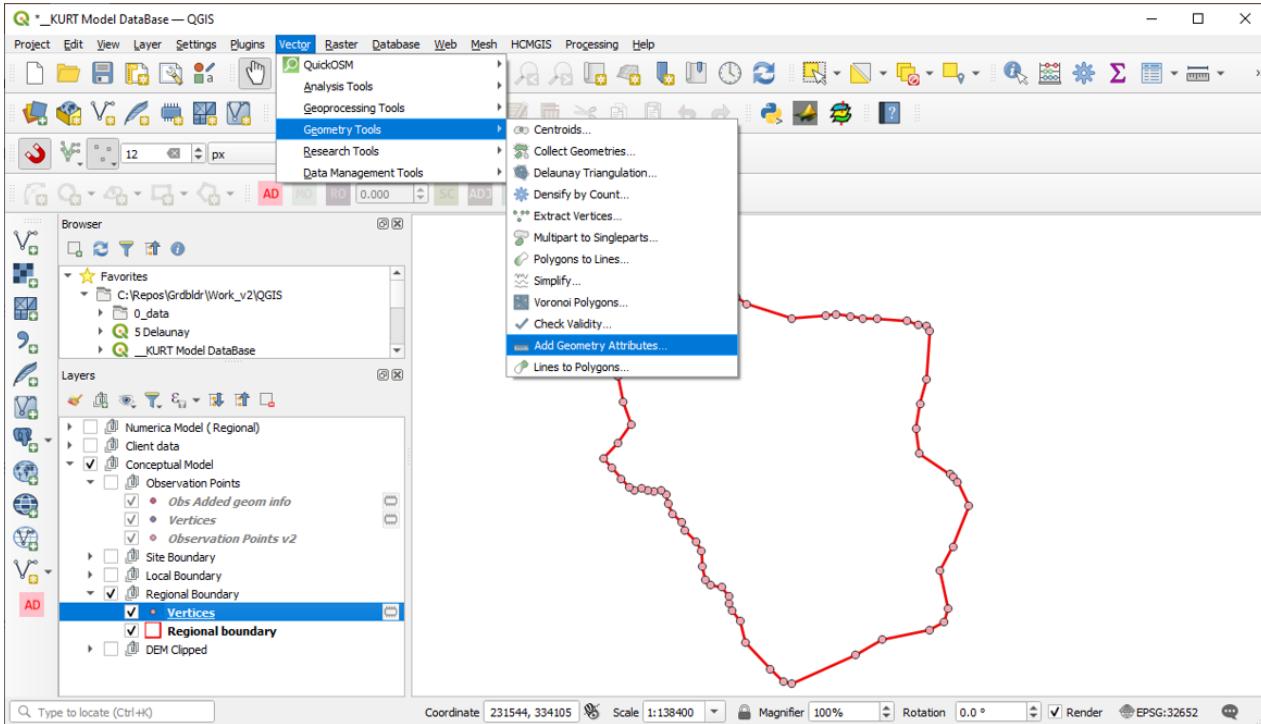
Right-click on the vertices layer and choose Open Attribute Table as shown below:



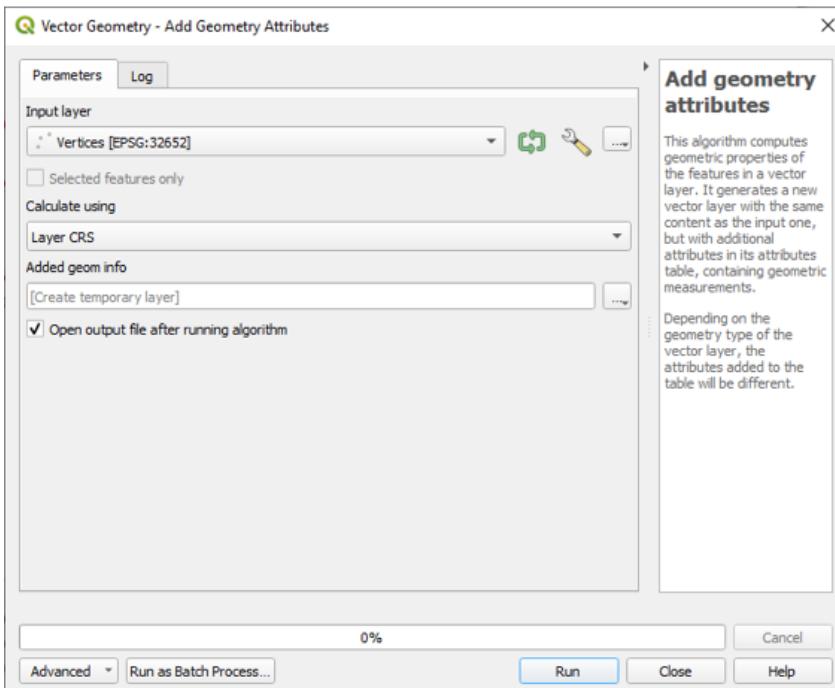
In this example, the table contains the **distance** and **angle** from **vertex\_index 0** to the other vertices in the polygon.

#### A.3.4.4 Add Geometry Attributes

Geometry attributes (e.g. *xy* coordinates, Delaunay triangle vertex id's) can be extracted from an active vector layer by choosing Vector\Geometry Tools\Add geometry attributes... as shown below:



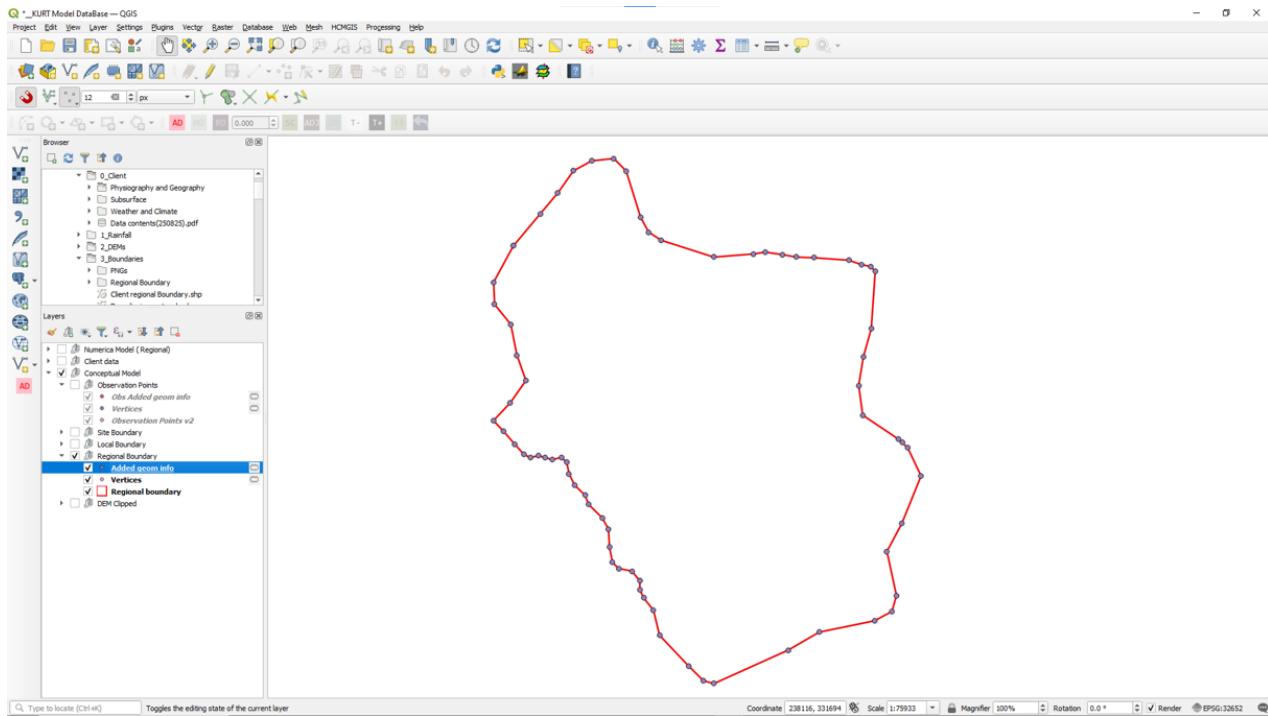
This opens the Add Geometry Attributes dialogue:



The active layer, in this case **vertices**, is shown in the Input layer field. Choose Run.

This creates a new layer called **Added geom info**, and adds vertex locations along the local boundary as

shown here by the purple circles:



Right-click on the Added geom info layer and choose Open Attribute Table to see the table shown below:

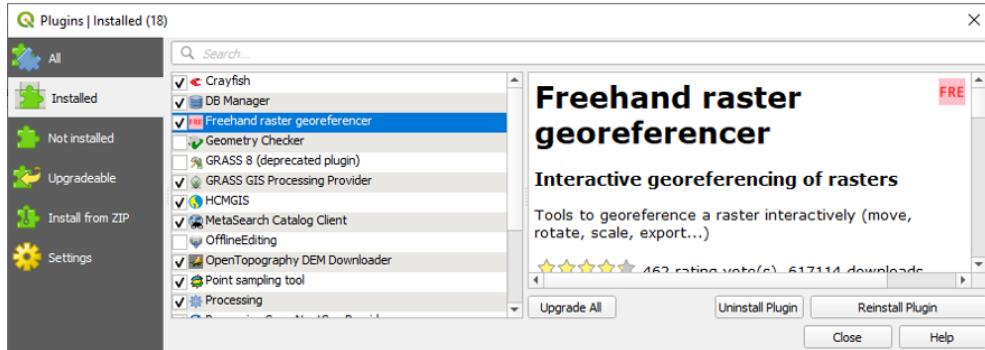
	id	vertex_index	vertex_part	vertex_part_ring	vertex_part_index	distance	angle	xcoord	ycoord
1	NULL	0	0	0	0	0	34.22299974312...	225036.4767607...	330762.0157023...
2	NULL	1	0	0	1	1269.761745469...	39.98874816825...	225851.6635518...	331735.5481290...
3	NULL	2	0	0	2	2108.514797668...	37.59763131042...	226391.2116437...	332377.7276515...
4	NULL	3	0	0	3	2933.452848230...	48.23198193051...	226866.2485507...	33052.1627664...
5	NULL	4	0	0	4	3568.597765232...	72.67398225297...	227423.3906022...	333357.1247314...
6	NULL	5	0	0	5	4246.694722026...	109.6438154431...	228097.8257171...	333427.5005695...
7	NULL	6	0	0	6	4778.288968234...	148.9795024138...	228472.1104063...	33050.0043257...
8	NULL	7	0	0	7	6256.807271313...	157.1356571011...	228911.4336776...	331638.2637562...
9	NULL	8	0	0	8	6776.935523007...	137.2904694657...	229159.1619990...	331180.9191629...
10	NULL	9	0	0	9	7231.492617836...	115.4213782351...	229540.2824934...	330933.1908415...
11	NULL	10	0	0	10	8912.856120557...	96.64055820330...	231140.9885699...	330418.6781741...
12	NULL	11	0	0	11	10117.16070000...	83.24480043860...	232341.5181272...	330513.9582977...
13	NULL	12	0	0	12	10483.71065418...	89.57873786962...	232703.5825969...	330571.1263718...
14	NULL	13	0	0	13	11022.69642626...	99.21747441145...	233237.1512891...	330494.9022729...
15	NULL	14	0	0	14	11448.80209281...	96.17512747882...	233656.3838329...	330418.6781741...
16	NULL	15	0	0	15	11982.71096269...	93.06551263442...	234189.9525251...	330399.6221493...
17	NULL	16	0	0	16	13052.56717279...	101.6878314995...	235257.0899094...	330323.3980505...
18	NULL	17	0	0	17	13456.35709724...	105.2999893466...	235638.2104038...	330190.0058774...
19	NULL	18	0	0	18	13747.85822293...	119.270549660...	235924.0507746...	330132.8378033...
20	NULL	19	0	0	19	13951.33502160...	160.6050296919...	236062.2200834...	329983.4656181...
21	NULL	20	0	0	20	15706.59039733...	189.8249919948...	235940.4247178...	328232.4409687...

In this example, the table has two additional columns *xcoord* and *ycoord* containing the *x* and *y* coordi-

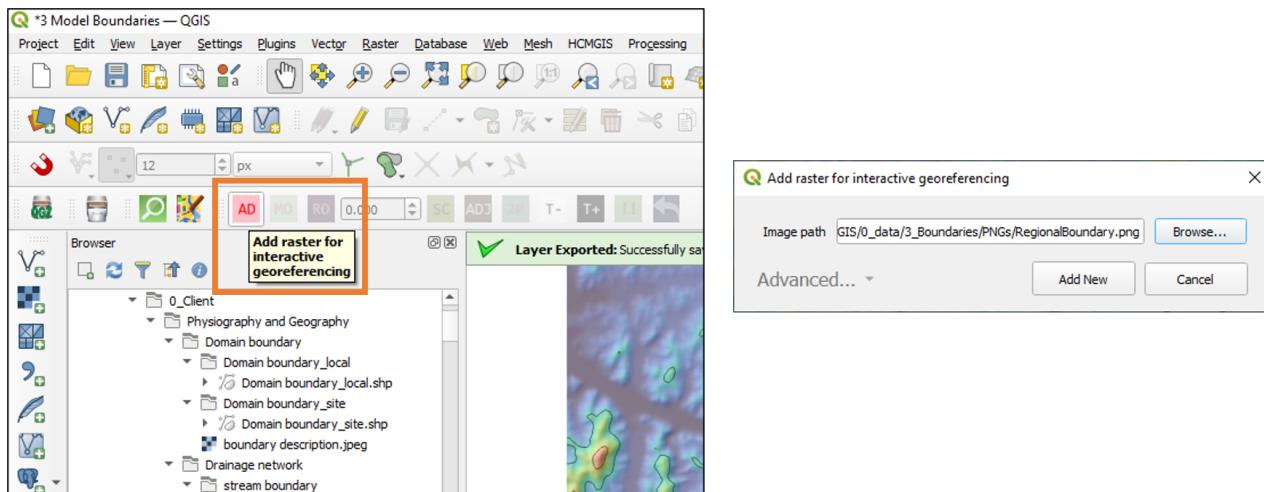
nates respectively.

#### A.3.4.5 Georeference an Image

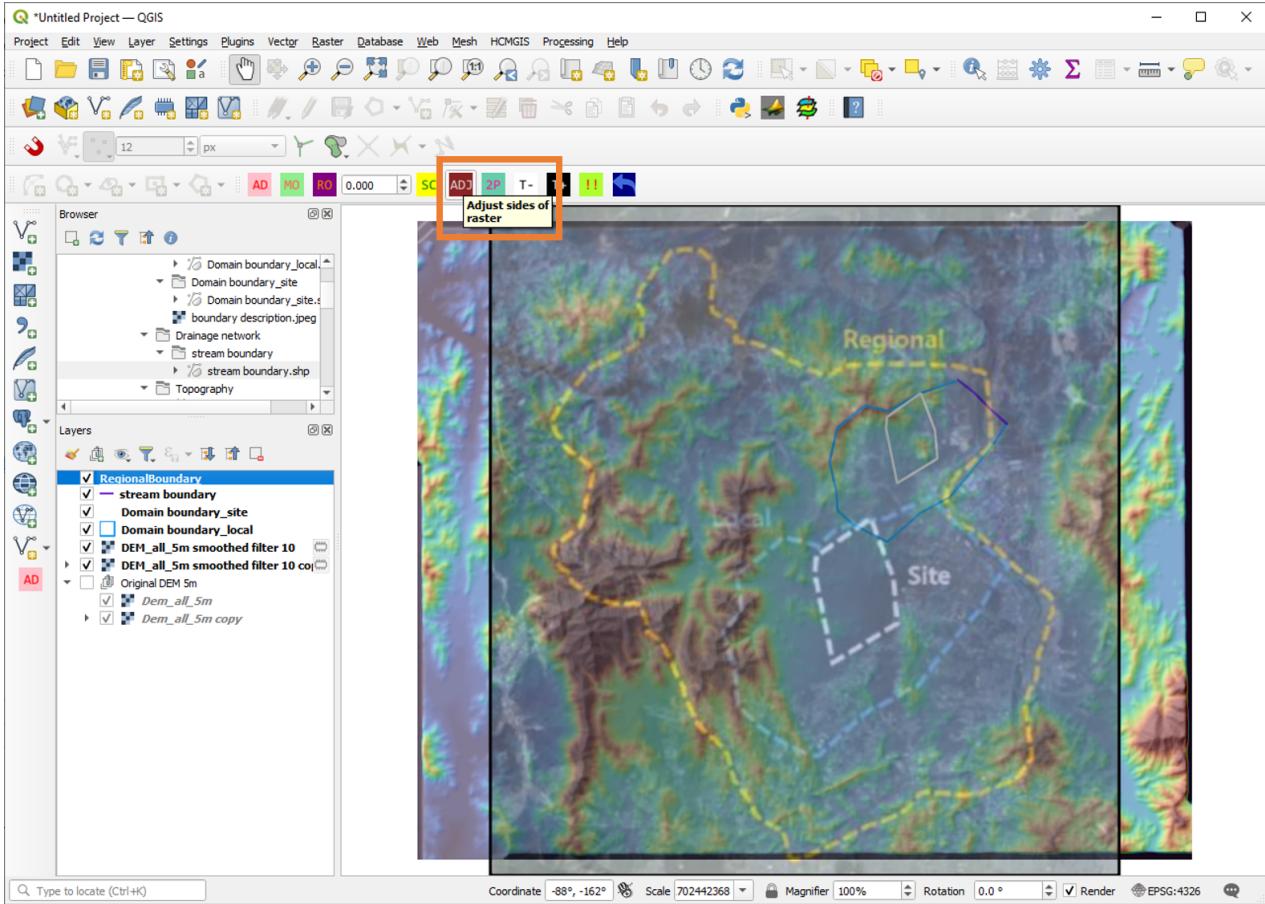
This image can be imported into QGIS using the Freehand Raster Georeferencer plugin, which is installed using Plugin\Manage and Install Plugins, as shown below:



Once installed, make sure the plugin toolbar is visible by choosing View\Panels\Freehand Raster Georeferencer. To import the regional boundary image file, click the AD button as shown below left and choose the file shown in the browser window below right:

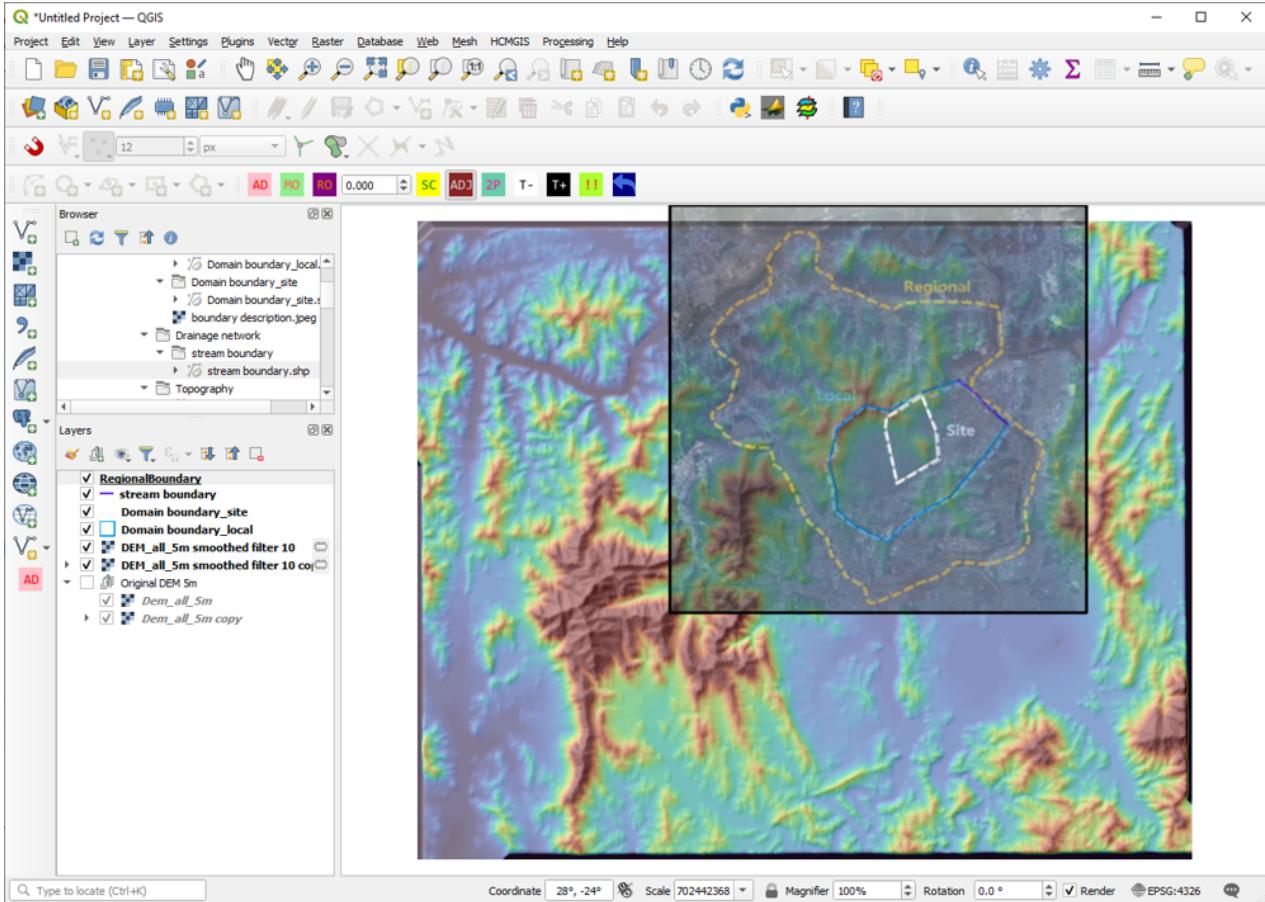


Below we see the imported image superimposed over the QGIS workspace.



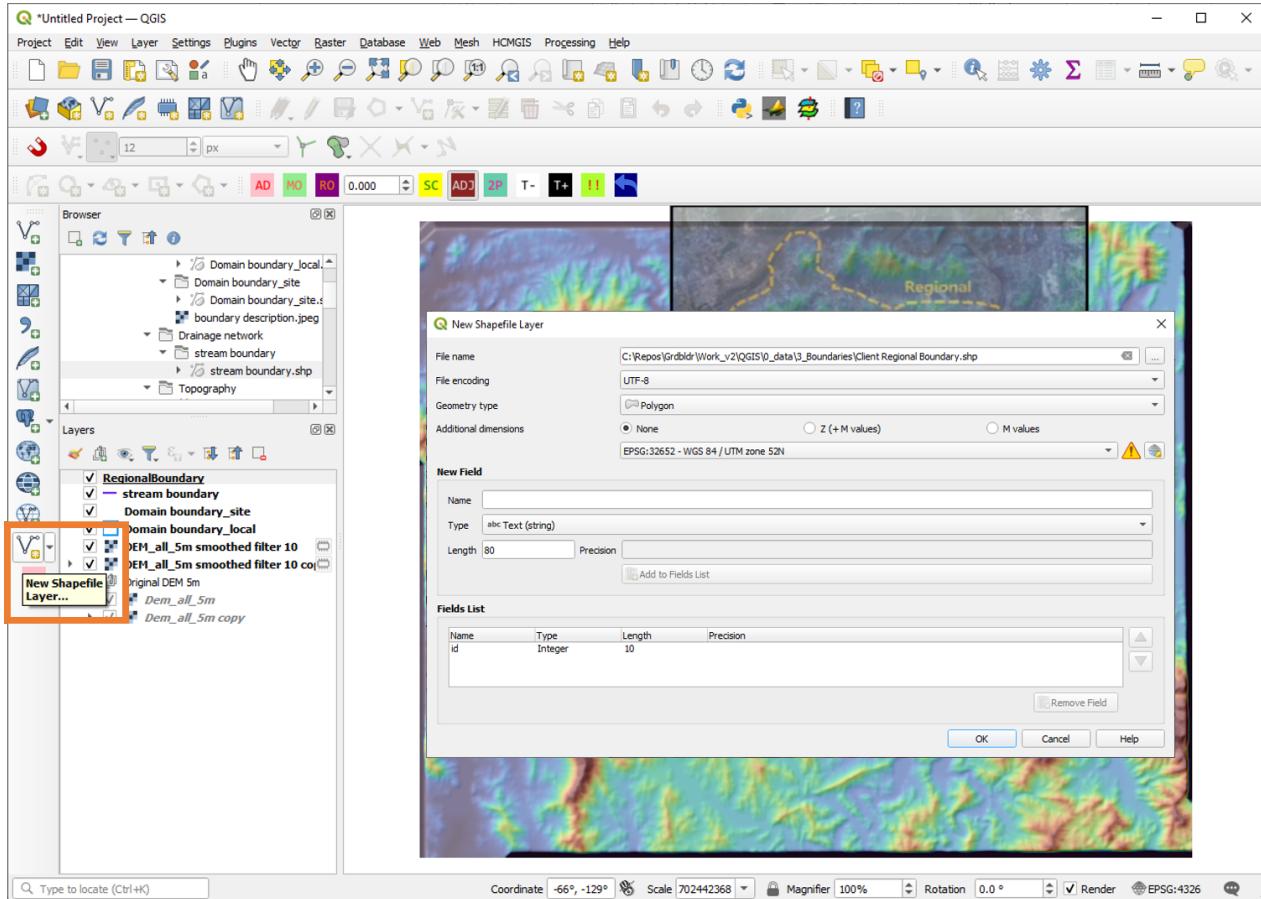
Note that the image boundaries are not lined up with the local and site shapefile boundaries.

Use the ADJ button to move the sides of the image around until the boundaries are aligned as shown below:



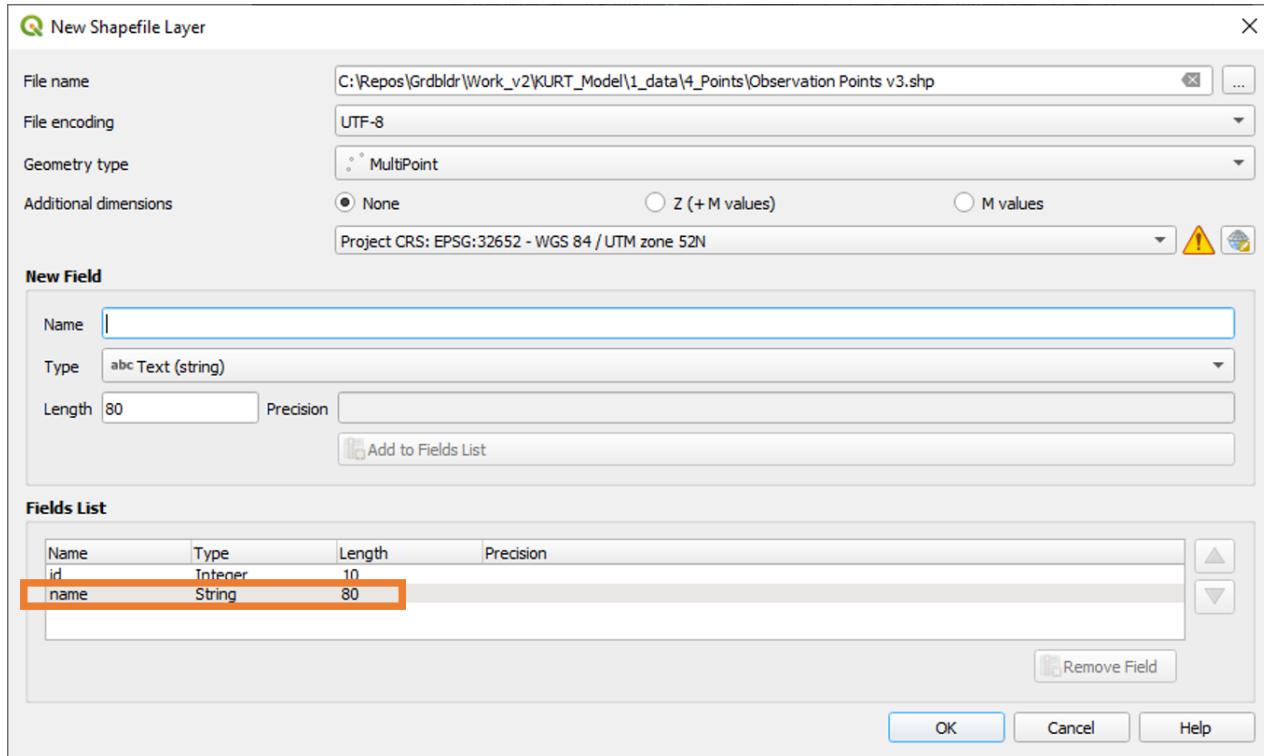
#### A.3.4.6 Add a new shapefile

If the Manage Layers Toolbar is not visible, choose View\Panels\Manage Layers Toolbar. To create a new shapefile, choose the New Shapefile Layer... button  to open the dialogue shown below:



In this example, the Geometry type is set to Polygon, the project CRS to EPSG:32652 - WGS 84 / UTM zone 52N and the shapefile name is Client Regional Boundary.

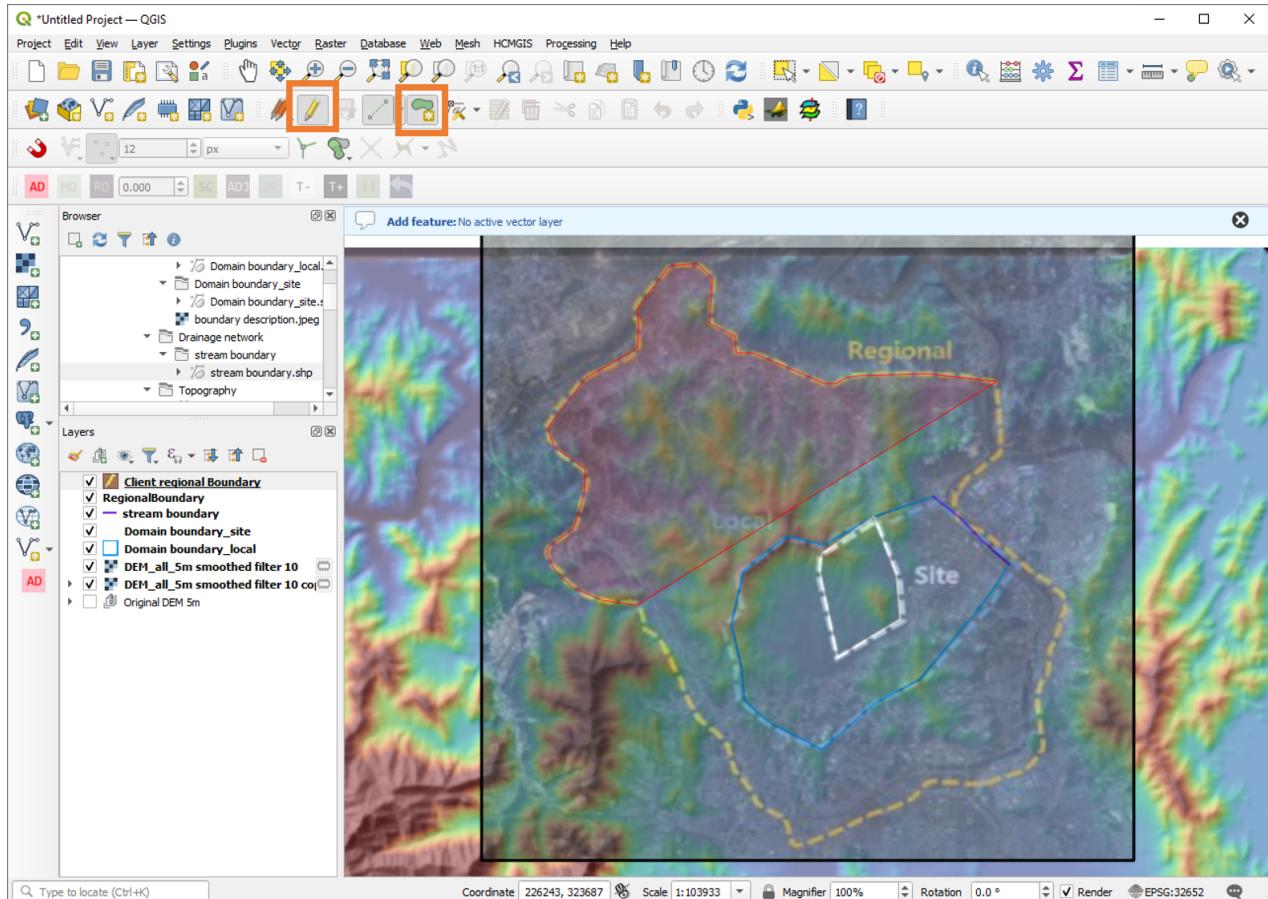
In the example below, a shapefile suitable for storing e.g. observation point data is defined:



Here the Geometry type is set to Multipoint and the shapefile name is Observation Points v3. A new field called `name` has been defined to store the observation point names of type string.

#### A.3.4.7 Digitize a new shapefile

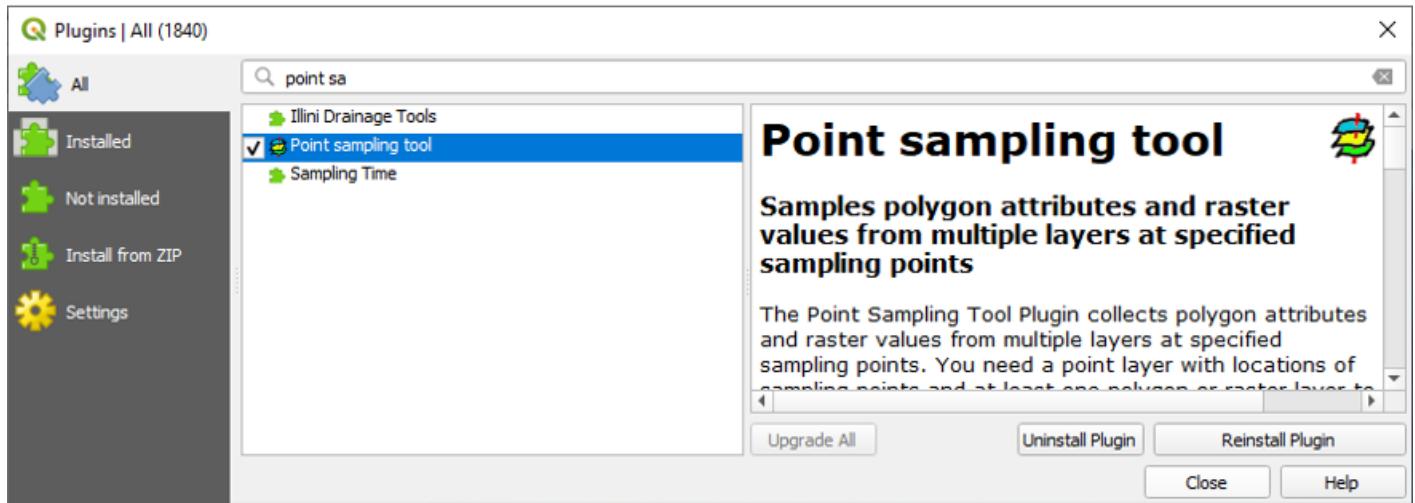
If the digitizing toolbar is not visible, choose View\Panels\Digitizing toolbar. To start digitizing choose the Toggle Editing button , then the Add Polygon Feature button . The mouse cursor should now be shown as a crosshair. Move the crosshair to a point on the regional boundary and click the left-mouse button to digitize a new boundary point. The example below shows a partially digitized polygon:



Proceed around the polygon boundary, left-clicking the mouse to define new polygon vertices, and finally right-clicking the mouse to close it.

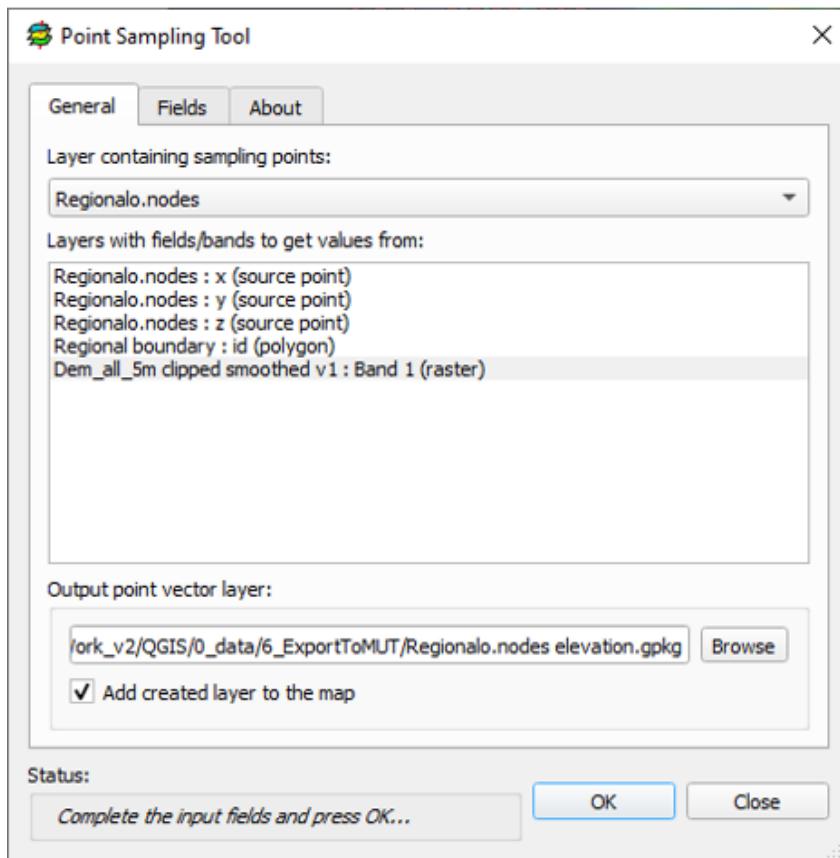
#### A.3.4.8 Point Sampling From a Raster Layer

The PointSamplingTool can be used to map elevations from a raster layer (i.e. a DEM) to the vertices of a vector layer. This plugin may need to be installed using Plugin\Manage and Install Plugins, as shown below:



Once installed, the Point sampling tool button should appear in the Plugins toolbar.

First, activate the layer, then click on Point sampling tool button to open the Point Sampling Tool dialogue shown below:



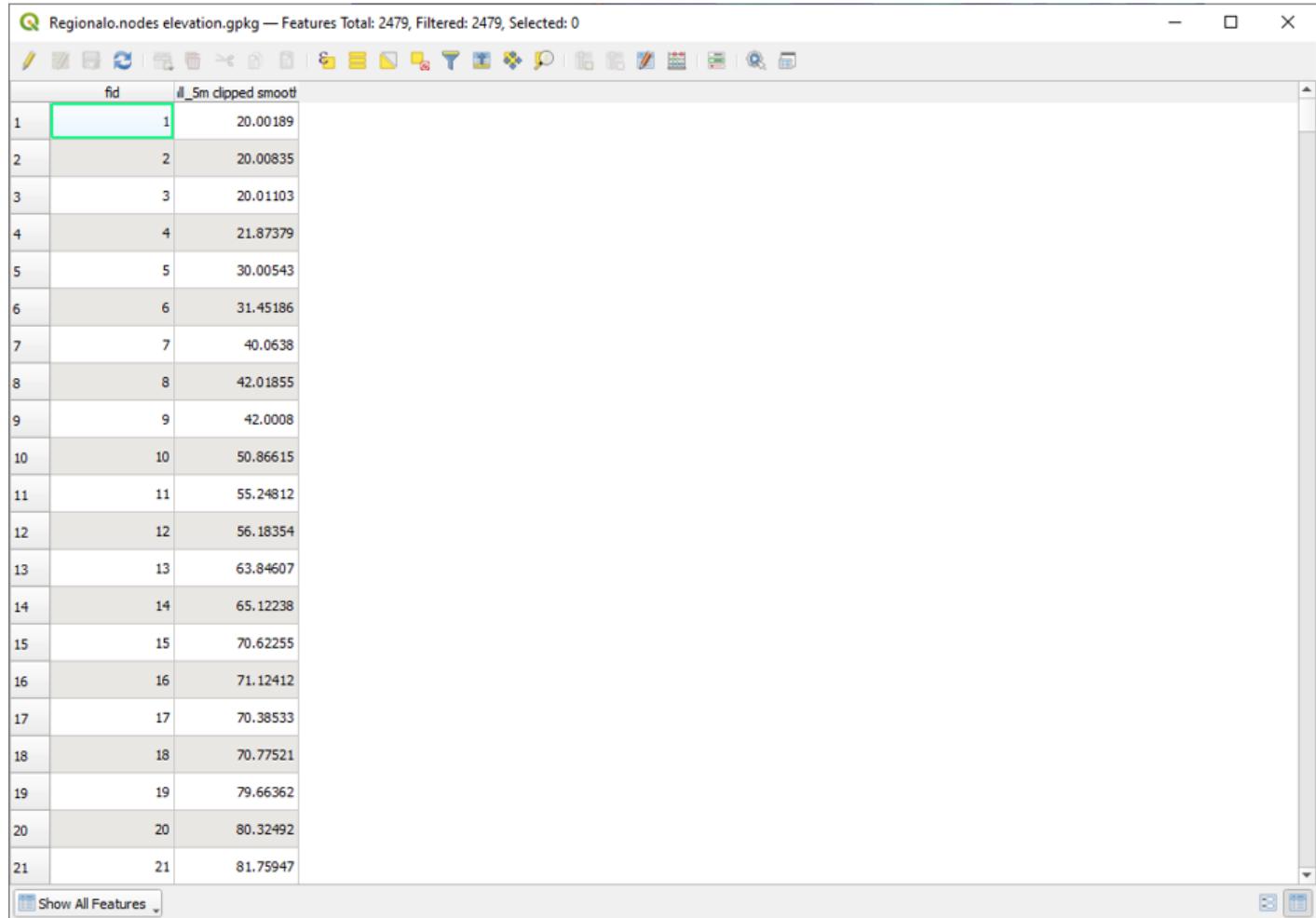
In this example the field Layer containing sampling points: is referring to a vector layer of points Regionalo.nodes.

alo.nodes.

The field **Layers with fields/bands to get values from:** contains a list of vector and raster layers available to sample from. Click on a layer to choose it and it will be indicated with a light gray shading. In this case the layer Dem\_all\_5m clipped smoothed v1: Band 1 (raster) is the only one chosen.

The field **Output point vector layer:** contains the full path and name of the file to store the sampled elevations in. Click the Ok button to do the sampling and then close the dialogue.

This creates a new layer called Regionalo.nodes elevation.gpkg. Right-click on it and choose Open Attribute Table, which is shown below:

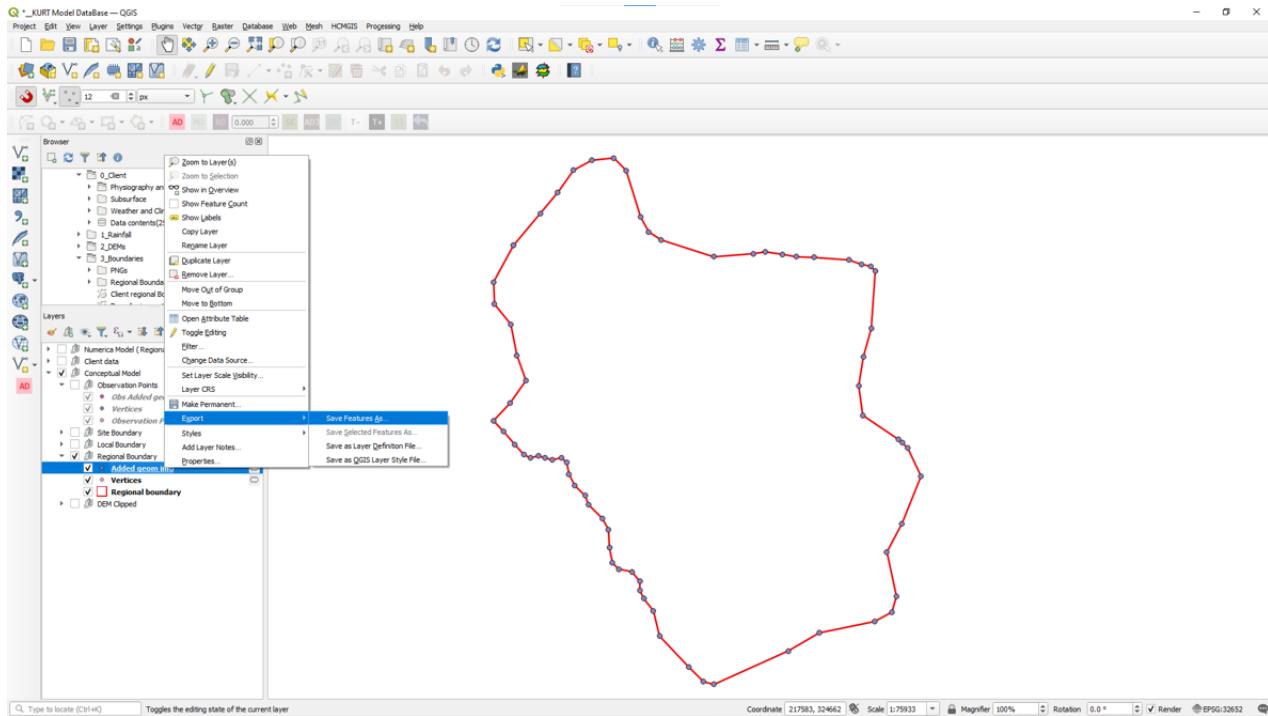


	fid	il_Sm clipped smooth
1	1	20.00189
2	2	20.00835
3	3	20.01103
4	4	21.87379
5	5	30.00543
6	6	31.45186
7	7	40.0638
8	8	42.01855
9	9	42.0008
10	10	50.86615
11	11	55.24812
12	12	56.18354
13	13	63.84607
14	14	65.12238
15	15	70.62255
16	16	71.12412
17	17	70.38533
18	18	70.77521
19	19	79.66362
20	20	80.32492
21	21	81.75947

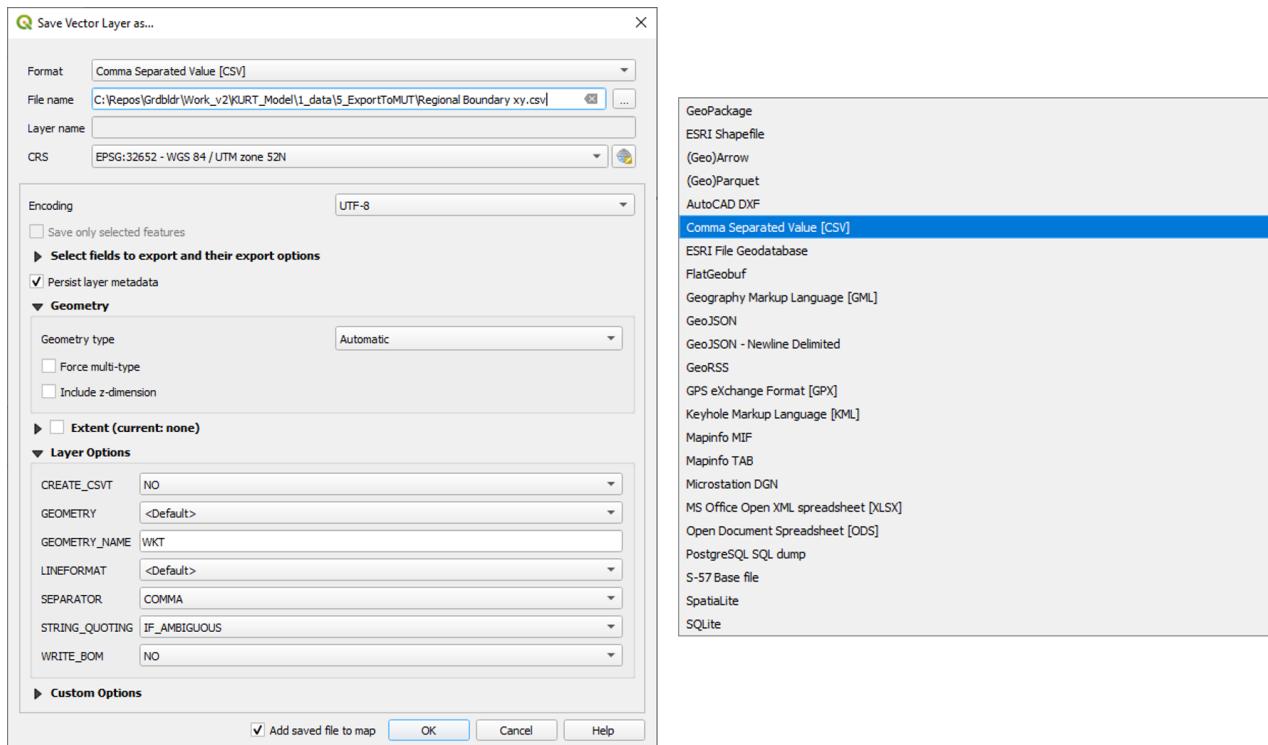
This is the list of sampled elevations, one for each point in the layer Regionalo.nodes.

### A.3.4.9 Export to a CSV File

The attribute table of an active vector layer can be exported to a CSV file by right-clicking on the vector name and choosing **Export\Save features as...** as shown below:



This opens the **Save Features as** dialogue shown below left:



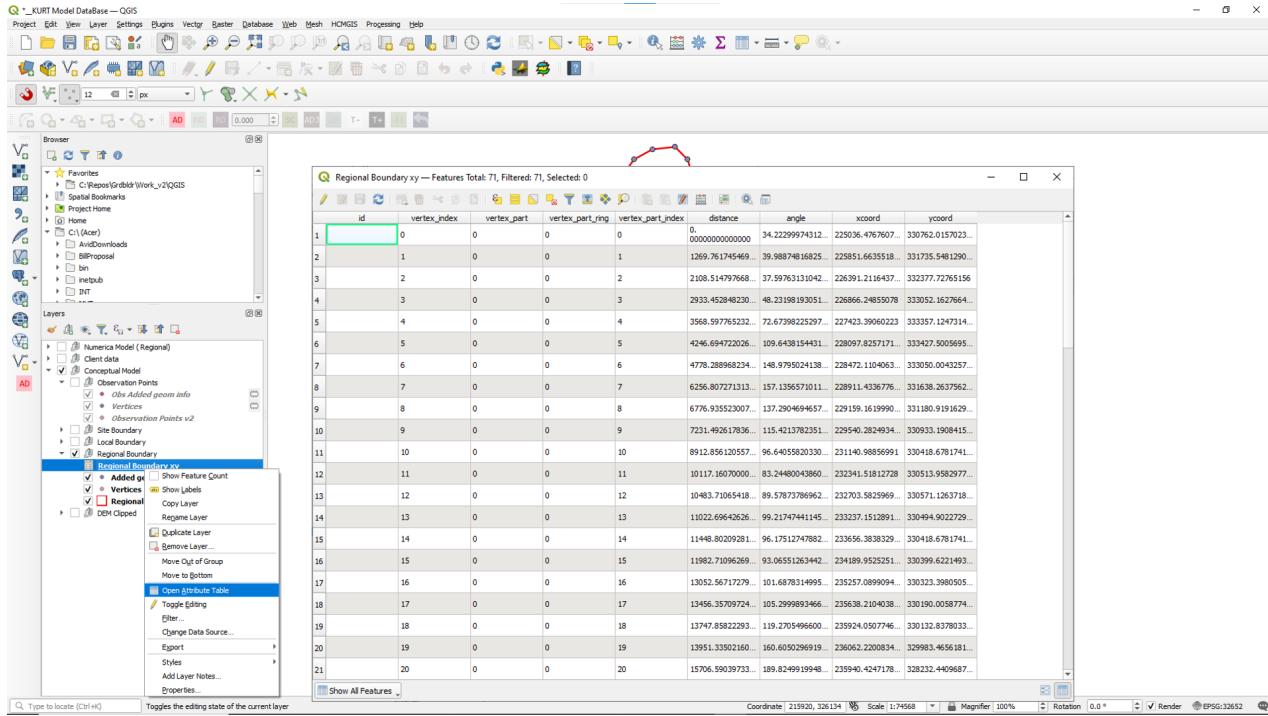
The **Format** field drop-down menu is shown to the right, with the **Comma Separated Value [CSV]** option

highlighted.

The **File Name** field requires the full path and name of the CSV file. The browse button ... at the end of the field is the easiest way to enter the information.

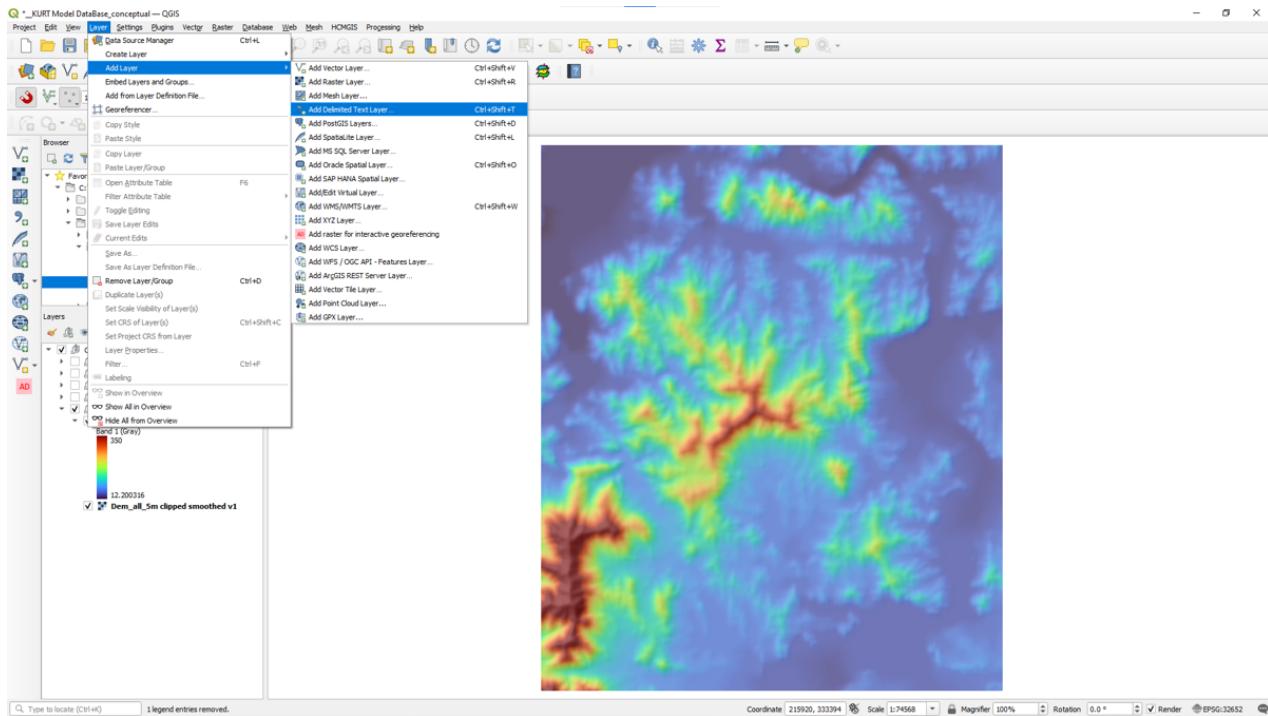
Check that the **CRS** field is set to the correct coordinate system. Choose **Ok**.

This creates a new layer whose name is taken from the CSV file name, in this example **Regional Boundary xy**. Right-click on the **Regional Boundary xy** layer and choose **Open Attribute Table** as shown below:

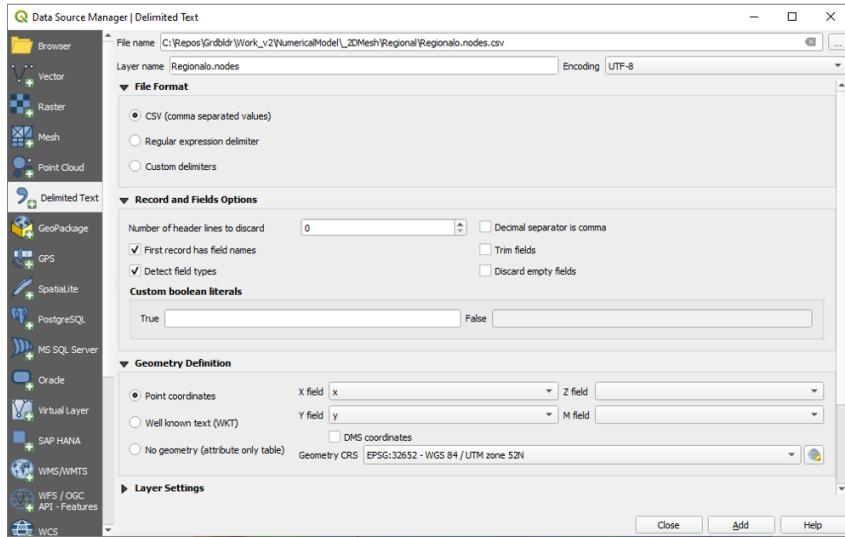


### A.3.4.10 Import from a CSV File

A new layer can be created by importing data from a CSV file. Choose Layer\Add layer\Add text delimited layer... as shown below:

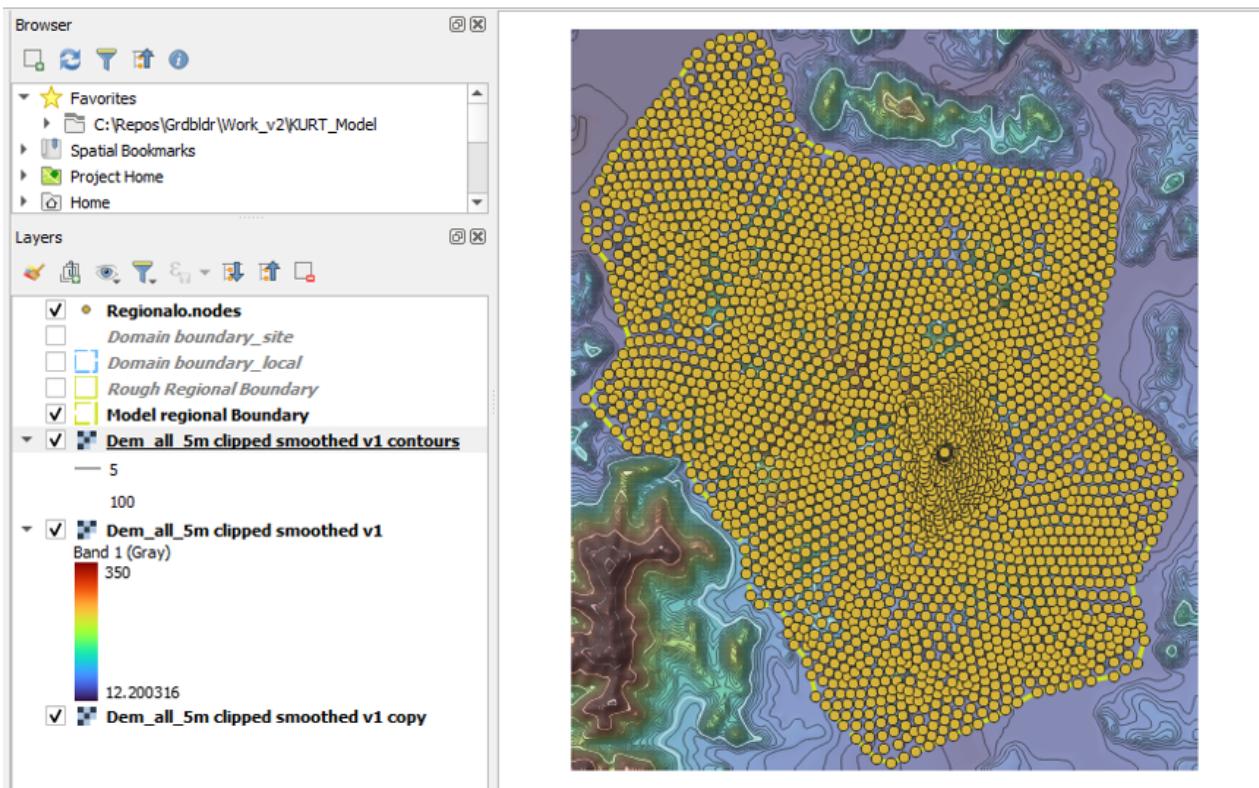


This opens the Delimited Text dialogue shown below:



The File Name field requires the full path and name of the CSV file. The browse button ... at the end of the field is the easiest way to enter the information. Choose Ok.

This creates a new layer whose name is taken from the CSV file name, in this example Regionalo.nodes, and plots the vertices (nodes) as yellow-orange circles shown below:



Right-click on the Regionalo.nodes layer and choose Open Attribute Table as shown below:

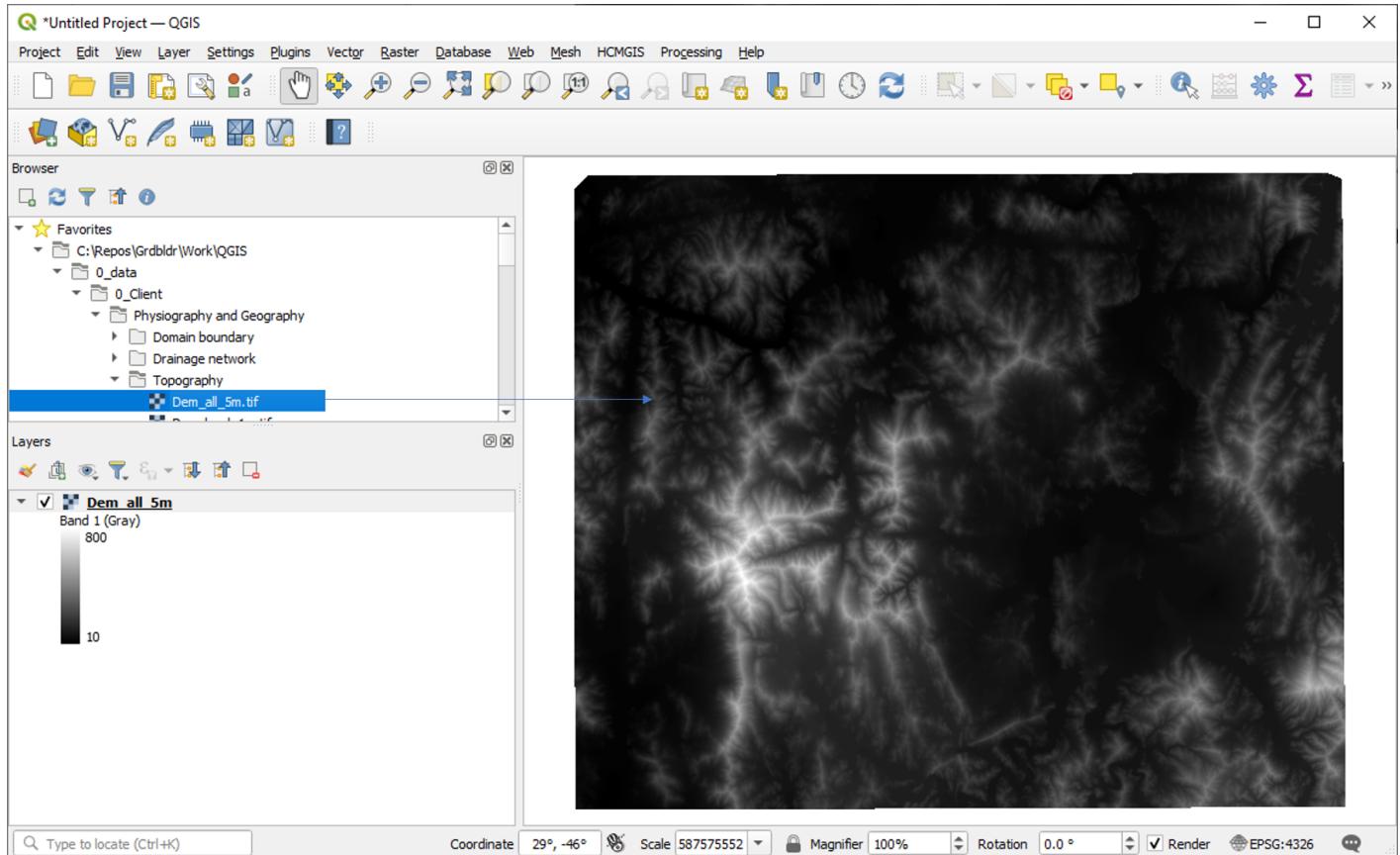
	x	y	z
1	224433.4057	329651.5234	0
2	224464.2135	328973.7527	0
3	224957.1376	328367.8667	0
4	225144.6924	327424.0277	0
5	225419.254	326652.9015	0
6	224935.4217	325964	0
7	224429.2788	325428.6565	0
8	224740.7514	325097.7169	0
9	225078.3293	324708.5673	0
10	225359.038	324391.3436	0
11	225550.7416	324302.3383	0
12	225806.3463	324357.1108	0
13	226002.355	324295.4661	0
14	226225.7113	324238.9852	0
15	226497.8483	324302.3383	0
16	226662.1656	324151.7142	0
17	226734.0393	323803.8256	0
18	226900.9147	323461.0893	0
19	227218.1277	323149.6959	0
20	227329.5562	322868.1925	0
21	227745.9465	322457.6668	0

This CSV file contains a set of  $xyz$  coordinates.

## A.3.5 Raster Layers

### A.3.5.1 Loading Raster Layers

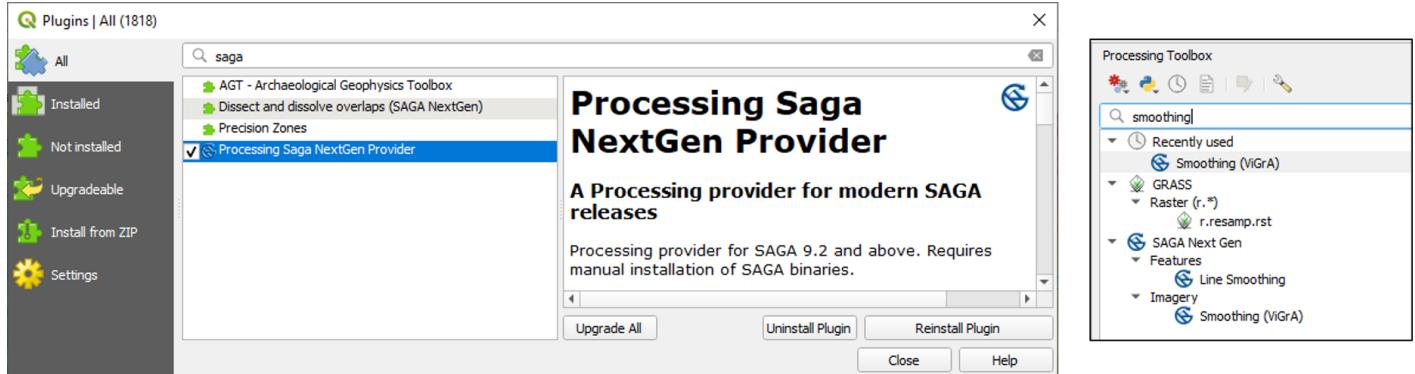
To load a raster layer, simply drag and drop the raster file into the QGIS workspace:



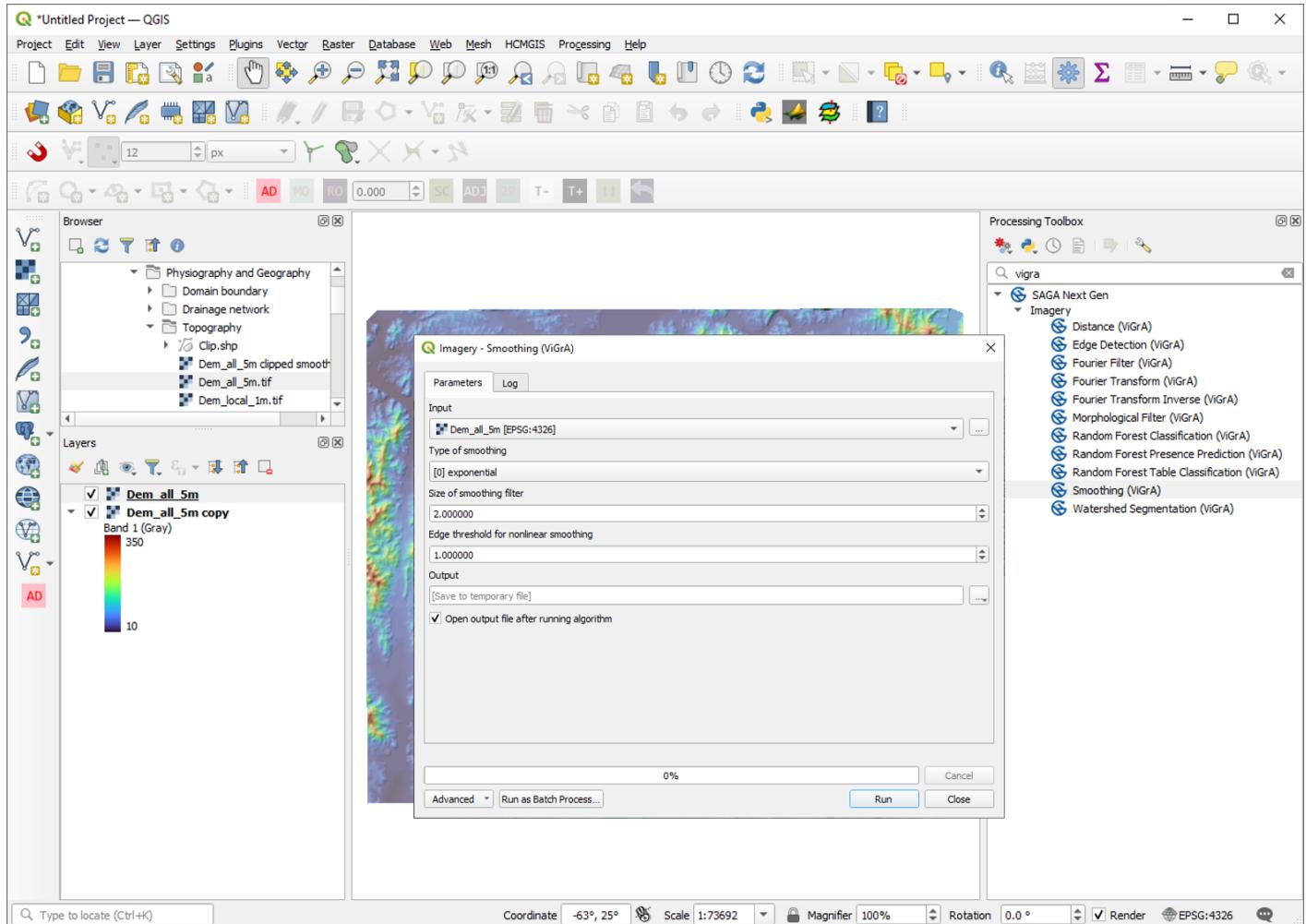
In this example, a raster file `Dem_all_5m.tif` was loaded. The raster appears in the display with a gray-shading indicating elevation and as an entry in the `Layers` window on the lower left part of the QGIS window. The elevation range (10 to 800 m) and shading scale for the DEM are indicated.

### A.3.5.2 Smoothing a Raster Layer

A smoothed version a raster layer can be created using the SAGA NEXT GEN plugin \Smoothing(ViGrA). This plugin must first be installed using Plugin\Manage and Install Plugins, as shown below on the left:

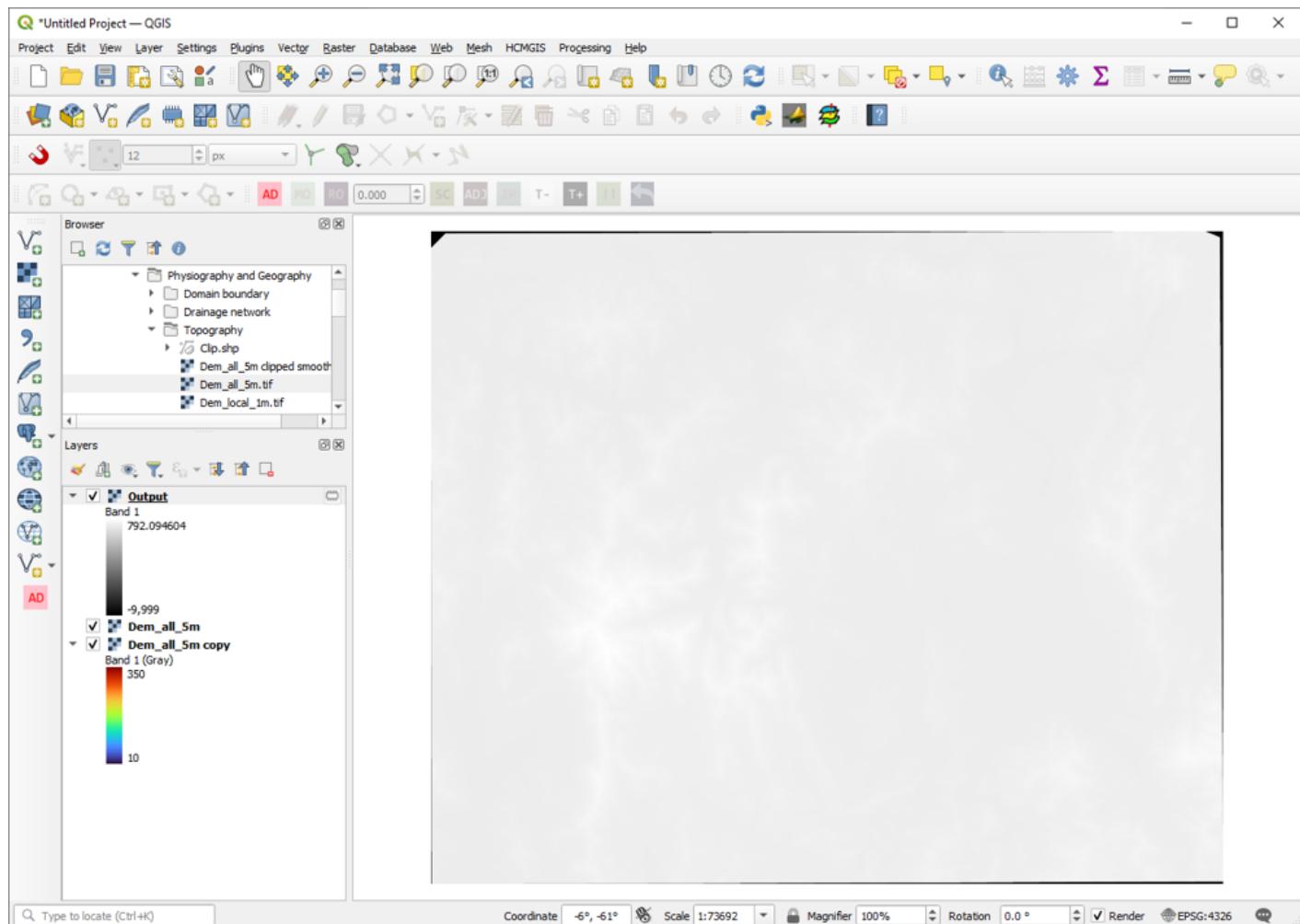


If the Processing Toolbox panel (above right) is not open choose View\Panels\Processing Toolbox, then search for the string e.g. 'smoothing' to locate the plugin.



In the example shown above we have chosen to smooth the raster layer Dem\_all\_5m\_clipped. Double-click the Smoothing(ViGrA) plugin in to open the dialogue shown below:

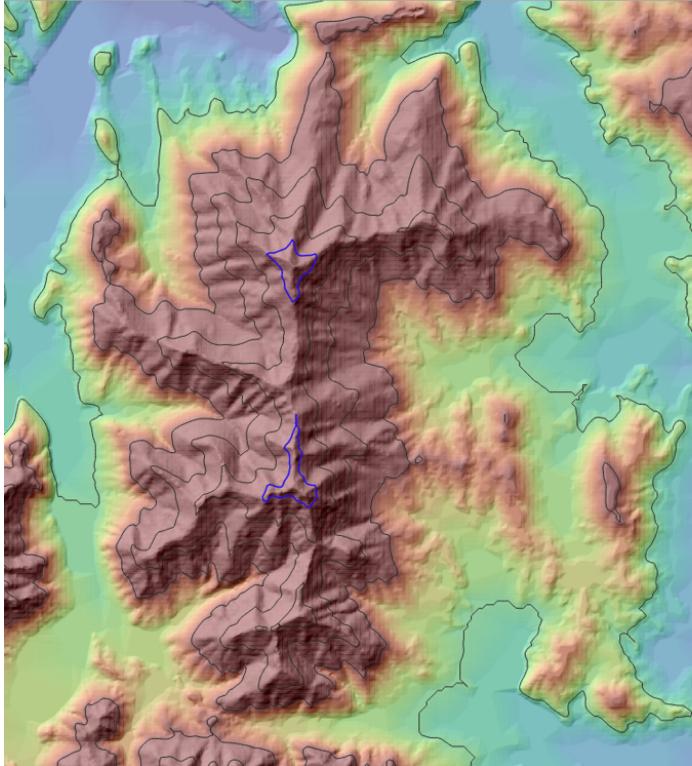
Click the Run button to generate a smoothed DEM with a default Size of smoothing filter of 2. The image below shows that a new DEM called Output has been created with an elevation range (-9,999 to 792 m).



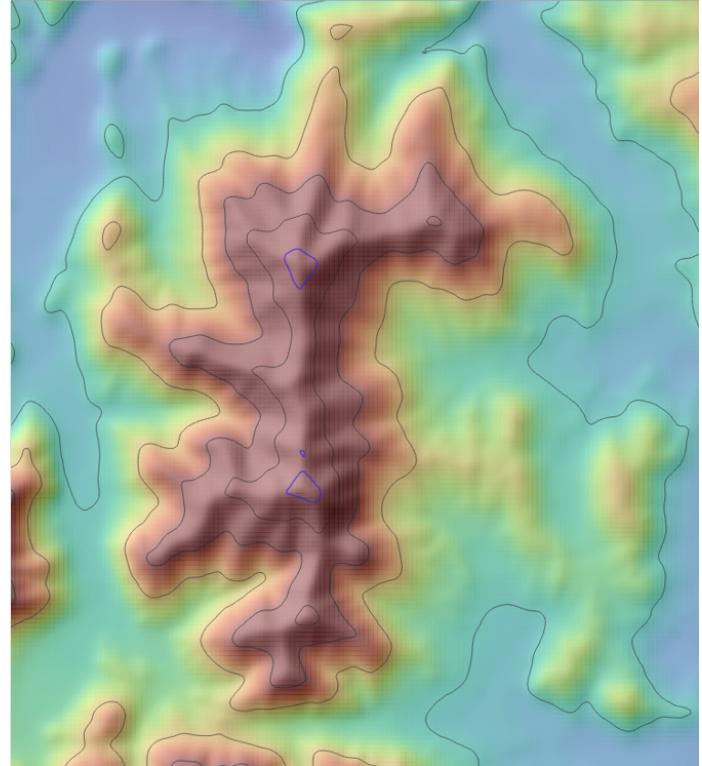
The regions of -9,999 m elevation show up as black areas around the outside edges of the DEM. To improve the image choose Output\Properties\Symbology and change the Color gradient\Min value to e.g. 10 m.

After testing with Size of smoothing filter of 2, 5, and 10 m, it was found that the 10 m was sufficient to remove the artefacts. A portion of the DEM is shown below both before and after smoothing:

Before smoothing



After smoothing



# Appendix B

## Microsoft Excel Database Files

MICROSOFT EXCEL files are used to store information in the following databases:

- **GWF.xlsx** for GWF domain material parameters.
- **SWF.xlsx** for SWF domain material parameters.
- **CLN.xlsx** for CLN domain material parameters.
- **SMS.xlsx** for solver parameters.
- **ET.xlsx** and **LAI.xlsx** for evapotranspiration (ET) and leaf area index (LAI) parameters respectively. *These are currently just placeholders for future development and will not be discussed further at this time.*

Modifications can be made to the database by editing the **.xlsx** file in MICROSOFT EXCEL and exporting the results to a **csv**-formatted version of the file which is then read and processed by MUT. We will use the GWF database to illustrate the modification workflow, which can then be applied to the other databases.

If you are a MUT end-user, you should edit the database files in the **USERBIN** directory, as outlined on page [10](#).

If you are a developer and have downloaded the MUT\_Examples repository, you should edit the database files in the directory *path\Grdbldr\\_MUT\_USERBIN*, where the string *path* represents the local path to the repository (e.g. *c:\repos*).

A third alternative is to copy the database files to a folder in your MUT project directory (e.g. *c:\MyProject\Datasets*) and direct MUT to look for material property files in the local folder using the following instruction:

---

### use local databases

- **Path<sub>str</sub>** Path to the directory containing the local databases.

In our example, we would insert the following lines in the MUT input file before any other database instructions:

```
use local databases
c:\MyProject\Datasets

gwf materials database
My_GWF.csv
```

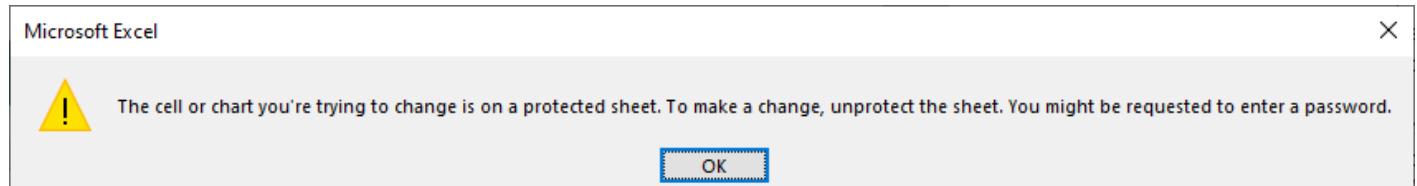
First, open the GWF.xlsx in MICROSOFT EXCEL:

1	ID	Material name	Porosity	Kh (Kx)	Kv (Kz)	Ky	Specific storage	Specific Yield	Unsaturated Function Type	Alpha	Beta	Sr	Brooks-Corey Exponent	Length Unit	Time Unit	Notes			
2	1 1D Column		0.3	10	10	0	1E-07	0.43	Van Genuchten	0.036	1.56	0.1814	6.5714	CENTIMETER DAYS		Rob copied these properties from			
3	2 1D Column Brooks		0.3	10	10	0	1E-07	0.43	Brooks-Corey	0.036	1.56	0.1814	6.5714	CENTIMETER DAYS		-			
4	3 2D Hillslope		0.1	31.536	31.536	31.536	1E-07	0.1	Van Genuchten	3.34	1.982	0.2771	6.5714	METERS YEARS		-			
5	4 2D Hillslope 100 m len.		0.3	0.000147	0.000147	0	0.0001	0.3	Van Genuchten	1	5	0.3	1	METERS	SECONDS	-			
6	5 3D Abduls Problem		0.34	0.00001	0.00001	0	1.2E-07	0.34	Van Genuchten	1.9	6	0.18	1	METERS	SECONDS	Properties from Joel Vanderkwaai			
7	6 Borden sand		0.34	0.00001	0.00001	0	1.2E-07	0.34	Van Genuchten	1.9	6	0.18	1	METERS	SECONDS	Properties from Joel Vanderkwaai			
8	7 Sand		0.3	0.00001	0.00001	0	0.0001	0.3	Van Genuchten	1	5	0.3	1	METERS	SECONDS	Made up for illustrative purposes			
9	8 Clay		0.5	1E-08	1E-08	0	0.0001	0.5	Van Genuchten	1	5	0.5	1	METERS	SECONDS	Made up for illustrative purposes			
10	9 Silt		0.4	0.000001	0.000001	0	0.0001	0.4	Van Genuchten	1	5	0.5	1	METERS	SECONDS	Made up for illustrative purposes			
11	10 2D Hillslope seconds		0.1	31.536	31.536	0.00001	0.00001	0.1	Van Genuchten	0.0334	1.982	0.2771	1	METERS	SECONDS	Rob copied these properties from			
12																			
13																			

Some key features to note are:

- The first row contains database field (i.e. column) names.
- The existing database contains data for 10 materials stored in rows 2 to 11.

As a precaution, the database is protected. If you try to modify any of the existing contents, in this case rows 1 to 11, you will receive the following warning:



To unprotect the sheet, choose Review, then Unprotect sheet:

Comments							Protect			
Kv (Kz)	Ky	Specific storage	Specific Yield	Unsaturated Function Type	Alpha	Unprotect Sheet				
.0	10	0	1E-07	0.43	Van Genuchten	0.0	Prevent unwanted changes from others by limiting their ability to edit.	Lengt	CENTI	
.0	10	0	1E-07	0.43	Brooks-Corey	0.0	For example, you can prevent people from editing locked cells or making formatting changes.	Lengt	CENTI	
.6	31.536	31.536	1E-07	0.1	Van Genuchten	3.		METE	METE	
.7	0.000147	0	0.0001	0.3	Van Genuchten	1	<a href="#">Tell me more</a>	METE	METE	
1	0.00001	0	1.2E-07	0.34	Van Genuchten	1.9	6	0.18	1	METE
1	0.00001	0	1.2E-07	0.34	Van Genuchten	1	5	0.3	1	MFTF
1	0.00001	0	0.0001	0.3	Van Genuchten					

You will be prompted to enter a password, which is **mut**.

To protect the sheet, choose **Review**, then **Protect sheet**, then enter and confirm a password. We suggest you use the **mut** password. If not, don't forget the new password!

The purpose for protecting the database is to prevent accidental changes from being made to critical material properties so future runs referring to protected material ID's yield repeatable results. This would be the case, for example, for material properties calibrated for an engineering project or verification example, which occupy the first 6 materials in this GWF database).

Although the database is protected, you can add your own materials starting at row 12. The easiest way to add a material is to copy an existing one. For example, we can copy row 10 (silt) to row 12:

1. Click on the number 10 at the left end of row 10 to select the entire row.

9	8 Clay	0.5	1E-08	1E-08	0	0.0001	0.5 Van Genuchten	1	5	0.5	1 METERS	SECONDS	Made up for illustrative purposes
10	9 Silt	0.4	0.000001	0.000001	0	0.0001	0.4 Van Genuchten	1	5	0.5	1 METERS	SECONDS	Made up for illustrative purposes
11	10 2D Hillslope seconds	0.1	31.536	31.536	0.00001	0.00001	0.1 Van Genuchten	0.0334	1.982	0.2771	1 METERS	SECONDS	Rob copied these properties from
12													
13													

2. Type **ctrl-C** to copy it.
3. Click on the number 12 at the left end of row 12 to select the entire row.
4. Type **ctrl-V** to paste it.

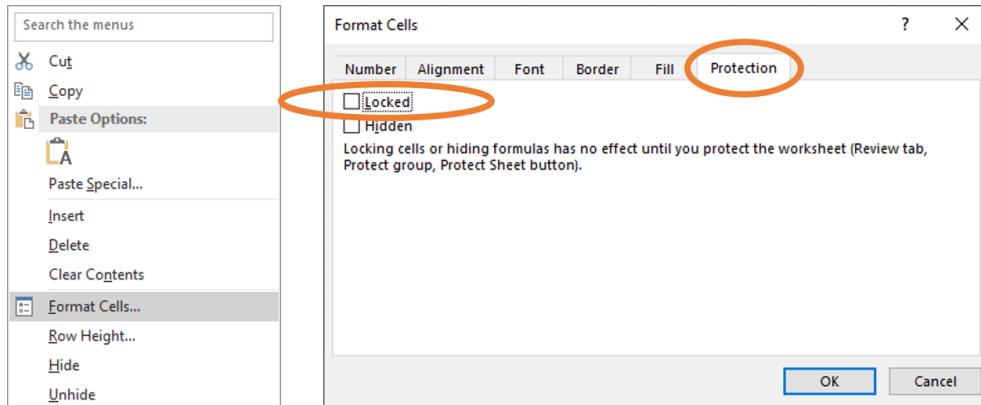
10	9 Silt	0.4	0.000001	0.000001	0	0.0001	0.4 Van Genuchten	1	5	0.5	1 METERS	SECONDS	Made up for illustrative purposes
11	10 2D Hillslope seconds	0.1	31.536	31.536	0.00001	0.00001	0.1 Van Genuchten	0.0334	1.982	0.2771	1 METERS	SECONDS	Rob copied these properties from
12	11 Silt	0.4	0.000001	0.000001	0	0.0001	0.4 Van Genuchten	1	5	0.5	1 METERS	SECONDS	Made up for illustrative purposes
13													

Note that the new material ID number is automatically set to 11, which is the previous number of materials plus 1. You should now change the material name and properties as desired.

If you want to extend the zone of protected cells to include the added row 12 you should do the following:

1. Unprotect the worksheet.
2. Select the entire worksheet by pressing **ctrl-A** or clicking on the button at the top left corner of the worksheet.

3. Right-click on any cell and choose **Format cells...** from the drop-down menu, then in the Protection tab uncheck the **Locked** checkbox then choose **OK**.



4. Select only the cells containing data that you want to protect, in this case from columns A to P and rows 1 to 12.
5. Right-click on any chosen cell, choose **Format cells...** from the drop-down menu, then in the Protection tab check the **Locked** checkbox then choose **OK**.
6. Protect the worksheet.

Columns I, N and O are special fields where the input is restricted by a list of choices. For example, if you click on cell I12, the **Unsaturated Function Type** for the new material, you will see a drop-down list button appear beside the cell. To choose, for example, the **van Genuchten** function type, select the button and highlight the option in the list and press enter:

0.5	Van Genuchten	1
0.4	Van Genuchten	1
0.1	Van Genuchten	0.0334
0.4	Van Genuchten	1
	Van Genuchten	
	Brooks-Corey	

The second worksheet, **Options**, contains the following data:

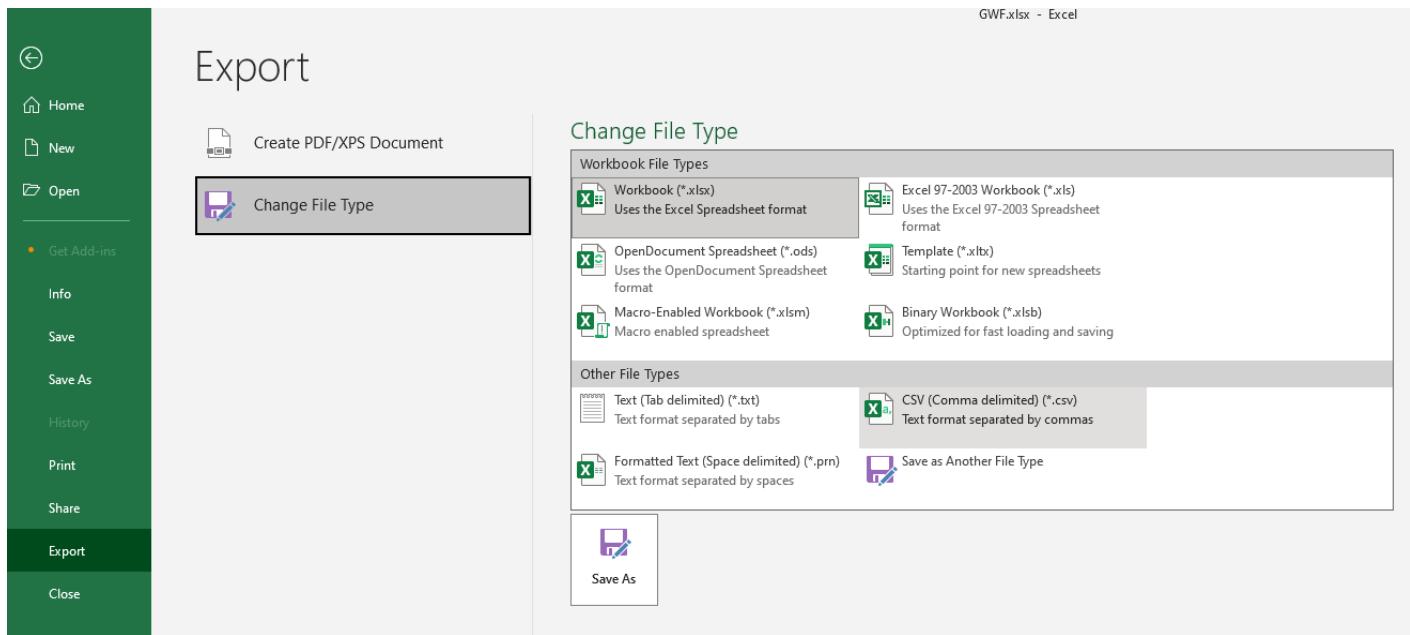
	A	B	C	D
1	Length	Time	Mass	Unsaturated Function Type
2	FEET	SECONDS	GRAM	Van Genuchten
3	METERS	MINUTES	KILOGRAM	Brooks-Corey
4	CENTIMETERS	HOURS		
5		DAYS		
6		YEARS		
7				

At the bottom of the sheet, there are tabs: GWF, Options (which is selected), and +.

These are the drop-down lists for the unsaturated function type, length unit, time unit in fields I, N and O respectively. This worksheet is also protected with the password **mut**.

Once you have finished modifying the material properties, you should save the **xlsx** file, then export them to a **csv** file by:

1. Selecting File/Export/Change File Type/CSV:



2. Selecting CSV (Comma delimited)(\*.csv)

3. Clicking the Save As button.

## Appendix C

# Microsoft Excel Modifications

Changes in version 2025.1:

- Batch file processing
- Triangular mesh generation
- QGIS useage

# Index

- CLN Domain
  - boundary conditions, 67
  - cell connection properties, 65
  - initial condition
    - initial (starting) depth, 66
  - material properties, 65
  - \_build.lay
    - CLN frame, 92
- GWF Domain
  - boundary conditions, 54
  - cell connection properties, 37
  - initial condition
    - initial (starting) head, 50
  - layering, 35
  - material properties, 39
  - material zones, 45
  - \_build.lay
    - GWF Cells frame, 83
    - GWF frame, 79
- QGIS Useage
  - Add a new shapefile, 140
  - Add geometry attributes, 134
  - Clipping a layer, 129
  - Coordinate reference systems (CRS's), 125
  - Digitize a new shapefile, 142
  - Export to a CSV file, 145
  - Extract vertices, 132
  - Georeference an image, 137
  - Import from a CSV file, 147
  - Installation and set-up, 123
  - Layer appearance (symbology), 128
  - Layer properties, 127
  - Loading a raster layer, 150
  - Loading a vector layer, 130
  - Point sampling tool, 143
  - Selecting features, 131
  - Smoothing a raster layer, 151
- SWF Domain
  - boundary conditions, 60
  - cell connection properties, 56
  - initial condition
    - initial (starting) depth, 59
  - material properties, 57
  - \_build.lay
    - SWF frame, 86
- TECPLOT
  - Custom labels, 86
  - data set information, 81
  - Equations and defining new variables, 104
  - frame order, 77
  - Infiltration Plot, 107
  - Isosurfaces, Water Table Plot, 106
  - probe tool, 81
  - Slices and Fence diagrams, 103, 106
  - value blanking, 102
  - Zoom Tool, 89
- Excel
  - cell protection, 155
- generate uniform rectangles, 6, 28
- generate variable rectangles, 28
- Grid generation
  - GWF Domain
    - New layer, 35
- Input instructions
  - 2d mesh from gb, 24
  - active domain, 39
  - build modflow usg, 20
  - build triangular mesh, 26
  - choose all cells, 39
  - choose all nodes, 62
  - choose all zones, 45
  - choose cell at xyz, 40
  - choose cells by chosen zones, 46
  - choose cells by layer, 40
  - choose cells by xyz layer range, 49
  - choose cells from file, 40

choose cells from gb elements, 40  
choose cells from gb nodes, 41  
choose cells from xyz list, 40  
choose gb nodes, 62  
choose node at xyz, 62  
choose zone number, 45  
chosen cells use gwf material number, 42  
chosen zones use cln material number, 65  
chosen zones use swf material number, 58  
clear chosen cells, 41  
clear chosen nodes, 62  
clear chosen zones, 45  
cln constant head, 67  
cln from xyz list, 64  
cln from xyz pair, 63  
cln initial depth, 66  
cln materials database, 65  
cln well, 67  
deltat, 70  
disable qgis output, 28  
disable tecplot output, 21  
duration, 70  
elevation constant, 32  
elevation from bilinear function in xy, 34  
elevation from gb file, 32  
elevation from list file, 32  
elevation from xz pairs, 33  
flag chosen cells inactive, 67  
generate cln domain, 63  
generate layered gwf domain, 31  
generate output control file, 72  
generate swf domain, 56  
generate uniform rectangles, 6, 28  
generate variable rectangles, 28  
gwf alpha, 45  
gwf beta, 45  
gwf brooks, 45  
gwf constant head, 55  
gwf drain, 55  
gwf initial head, 51  
gwf initial head equals surface elevation, 53  
gwf initial head from depth-saturation table, 54  
gwf initial head from tecplot file, 51  
gwf initial head function of z, 53  
gwf kh, 44  
gwf kv, 44  
gwf materials database, 41  
gwf recharge, 55  
gwf sr, 45  
gwf ss, 44  
gwf sy, 44  
gwf well, 55  
initial head from csv file, 51, 59, 66  
Layer name, 35  
Minimum layer thickness, 35  
modflow prefix, 21  
new layer, 35  
new zone, 48  
nodal control volumes, 30  
number of timesteps, 70  
observation point, 68  
observation points from csv file, 68  
Offset base, 36  
postprocess existing modflow model, 97  
Proportional sublayering, 36  
read mesh, 29  
refine inside polygon, 26  
saturated flow, 44  
sms database, 75  
sms parameter set number, 75  
stress period, 69  
swf constant head, 60  
swf critical depth, 61  
swf critical depth with sidelength1, 61  
swf depression storage height, 57  
swf depth for smoothing, 58  
swf initial depth, 59  
swf initial head from tecplot file, 59  
swf manning, 57  
swf materials database, 58  
swf obstruction storage height, 58  
swf recharge, 60  
swf to gwf connection length, 57  
swf well, 60  
tadjat, 71  
tcutat, 71  
tmaxat, 70  
tminat, 70  
top elevation, 31  
type, 70  
Uniform sublayering, 36  
units of length, 21  
units of time, 21  
use local databases, 155

wells from id\_x\_y file, [26](#)

Zone by template, [31](#)

Installation, [8](#)

Model Build

  visualization, [76](#)

Volumetric water budget, [97](#)