PIHMgis: A “*Tightly Coupled*” GIS Framework for PIHM

<u>Version: 2.3</u>

Tutorial: V-Catchment Watershed

Aug 02, 2010

Hydrology Group
Civil & Environmental Engineering
Pennsylvania State University, University Park, USA.

Table of Content

1. Introduction

Physically-based fully-distributed hydrologic models seek to simulate hydrologic state variables in space and time while using heterogeneous input data for climate, land use, topography and hydrogeology. The process of incorporating several physical data layers in a hydrologic model requires intensive effort in data gathering, development as well as topology definitions and physical parameter assignments. Traditionally Geographic Information System (GIS) has been used for data management, data analysis and visualization. Joint use and development of sophisticated numerical models and commercial GIS systems poses challenges that result from proprietary data structures, platform dependence, inflexibility in their data models and non-dynamic data-interaction with pluggable software components. PIHMgis is an open-source, platform independent, extensible and "tightly-coupled" GIS interface to Penn State Integrated Hydrologic Model (PIHM). The tight coupling between the GIS and the model is achieved by developing the PIHMgis data-model to promote minimum data redundancy and optimal retrievability. Minimum data redundancy and optimal retrievability are facilitated through carefully designed data-model classes, relationships and integrity constraints.

This tutorial has been designed to facilitate the users with a step-by-step navigation through individual PIHMgis modules so that they can quickly learn to setup a model using the software and then easily adapt to any watershed application. Starting from a Digital Elevation Model (DEM) of any region of interest, the model simulation and analysis can be performed to understand the dynamics of different hydrologic processes and the influence of hydro-geologic parameters on the prediction variables. This document also briefly explains the algorithms behind each of the modules.

2. PIHMgis Framework

PIHMgis interface is procedural and interactive in nature. Figure 2.1 shows the procedural framework of the interface. PennState Integrated Hydrologic Model (PIHM) is the core hydrologic model. PIHM uses projection of triangular mesh (also called triangular irregular network, TIN) as model kernel (prismatic cells) on which system of ordinary differential equation representing hydrologic processes are coupled. For a simple application (1) watershed boundary and (2) stream networks are used as constraining layer for descritization of the modeling domain into TINs. Geo-hydrologic data are then assigned to each cell (elements). Which is then fed to the hydrologic model so perform simulation for the desired modeling period.

In the first step, RASTER-DEM-processing is used for watershed delineation and stream definition, which will be used as constraining layers. The Vector Processing module aids to assembling watershed properties and constraints using points (such as stream gauge, ground water observation-well locations), polygons (such as watershed and subshed boundary obtained in raster processing step, physiographic boundaries) and polylines (streams obtained from raster processing). Vector processing modules facilitates pre-processing of several constraining layers for intended domain decomposition. This constraining layer is then used as input for Delaunay triangulations along with certain quality constraints. Once, the domain has been descretized into triangular elements topological and spatial data are assigned in an automated way using Data Model Loader Modules. Further, prepared input database is fed into the model (PIHM) – facilitated by RunPIHM module within the framework. After simulation is complete model results can be visualized using InfoViz modules.

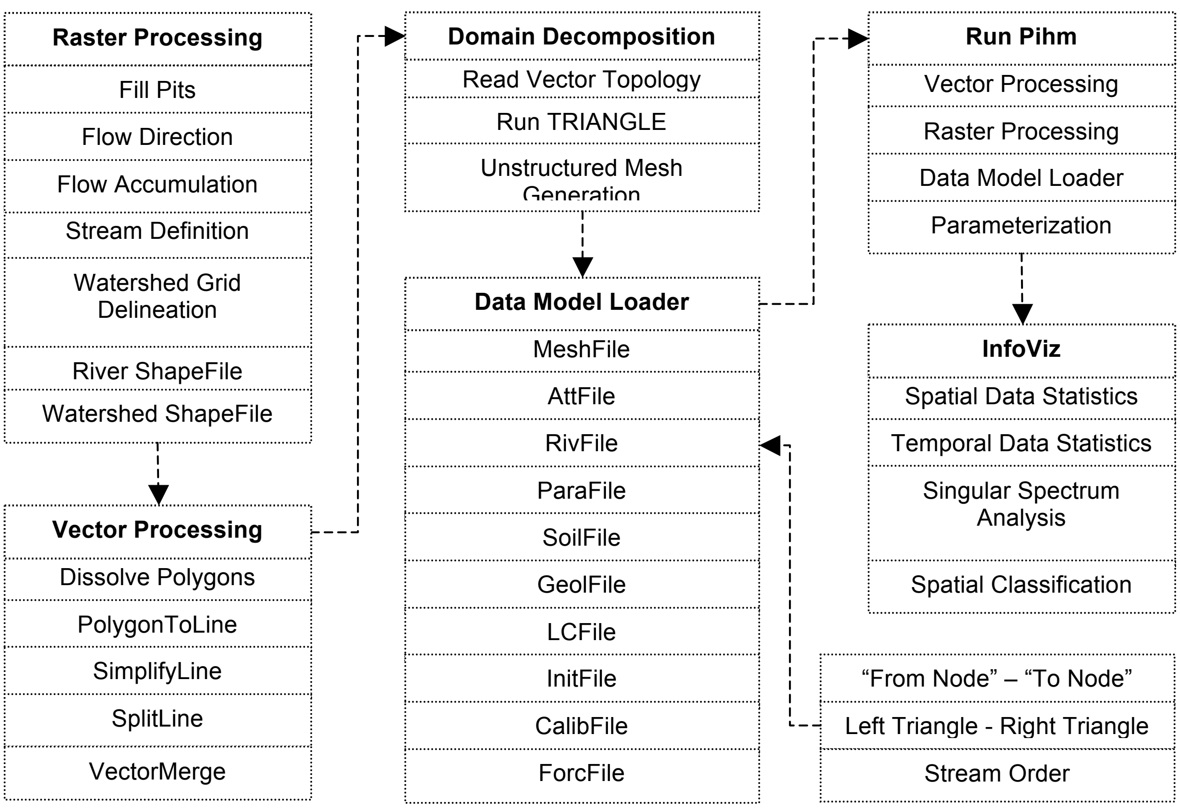


Figure 2.1: PIHMgis Procedural Framework

3. Download

PIHMgis is platform independent. It can be downloaded from the PIHMgis website [http://www.pihm.psu.edu/pihmgis_home.html] under **Downloads** tab. You need to select the correct version according to the operating system. Windows and Macintosh versions can be installed using the pre-compiled binaries. Detailed instructions are available on Installation Guide. For Linux systems, you will need to compile the source code natively and meet the library dependencies.

4. Initial Setup

The PIHMgis interface is shown in the Figure 4.1. PIHMgis uses QuantumGIS (QGIS) as base GIS framework. If the PIHMgis plugin doesn't show up automatically it should be loaded by selecting ***Plugins >> Plugin Manager*** and then "checking" the box next to PIHMgis. Click OK to load the PIHMgis plugin.

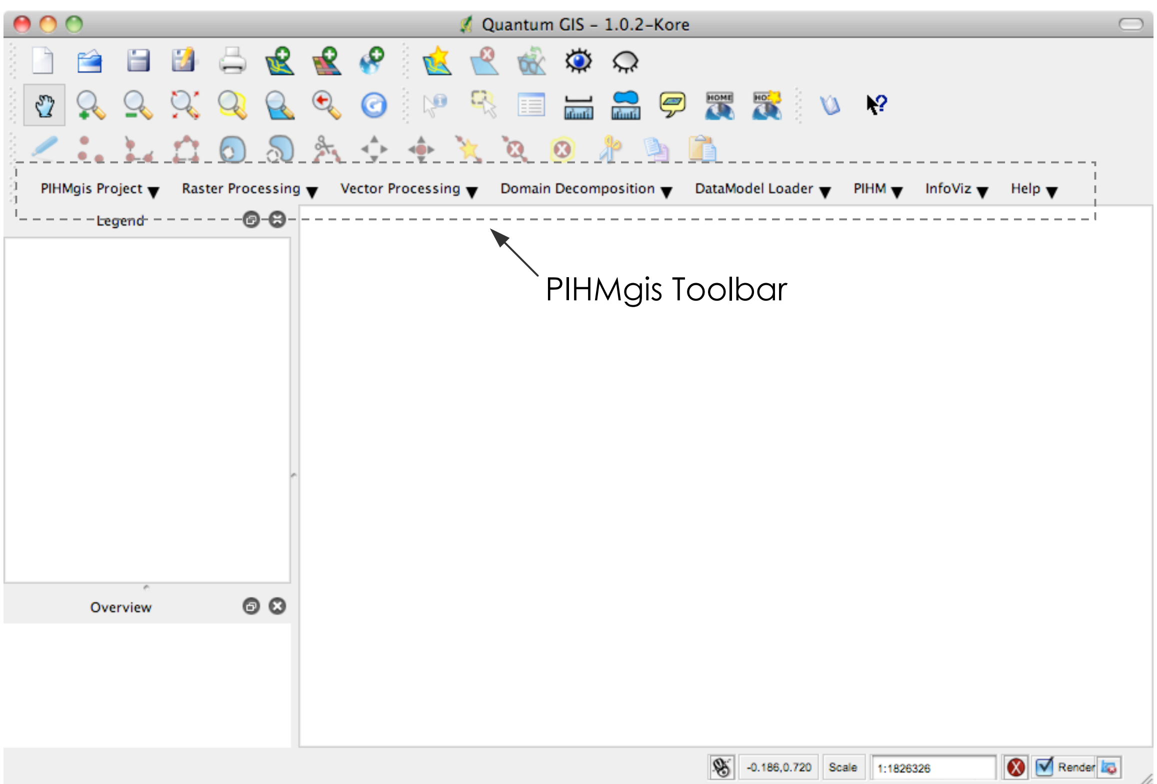


Figure 4.1: PIHMgis Interface

5. V-Catchment Watershed

Conceptual Description: In this tutorial we will use a simple example watershed (V-Catchment) and setup PIHM using the procedural framework of PIHMgis. The surface of the domain depicts a typical hill-slope watershed that has a compound slope. Also, notice that the domain is symmetric to the y-axis and a stream flows across the domain [Figure 5.1]. Boundary condition to the stream outlet is treated as a Zero-Depth gradient boundary condition (i.e. dh/dx=0, dH/dx=dZ/dx). Rest of the boundaries (up-stream river and watershed boundary) is considered as No-Flux boundary.

`

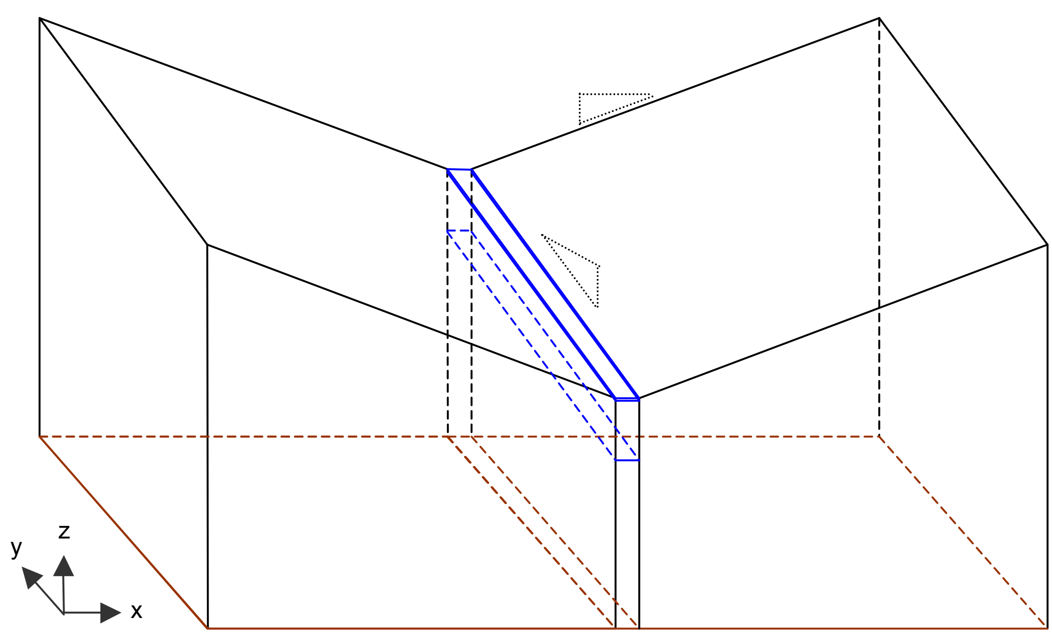


Figure 5.1: Schematic of the V-Catchment watershed

6. Data Requirement

It is very important to understand the data requirement for setting up a model simulation using PIHMgis. PIHM is a physically based distributed hydrologic model hence requires several physical parameters to accurately represent the watershed and streams properties. Basic data need for successfully setting up the model is:

(1) Digital Elevation Model (DEM) and bedrock elevation map
(2) Soil Texture or soil hydrologic properties,
(3) Geology texture or hydrologic properties of the aquifer,
(4) Land cover and vegetation parameters, and
(5) Climate variables for the modeling period.

National products are available for these datasets. However, locally available data products can be easily incorporated as well.

Above-mentioned data products can be obtained from various sources depending on location and availability of data. User needs to ensure that all the data (in the form of shape or grid files) are in GIS format and have same geographic projection system (also with length unit in **meters**) because various dataset may have different projection system depending on source. **PIHMservices** is under development, which is intended to serve as a hub to distribute all the required data sets for the modeling purposes under one roof. For the purpose of this tutorial, data for the example v-catchment watershed has already been pre-compiled which can be downloaded under the Downloads tab from http://www.pihm.psu.edu/pihmgis_home.html

<u>Data provided:</u>

Raster Files: DEM, Bed Elevation, Soil Class, Land Cover, Climate classes

Data Files: Soil Texture, Geology Texture, NLCD class, Climate data

After you have obtained the dataset you can start working with PIHMgis. It is always a good strategy to meet the data need as a first step towards PIHMgis project. It is intended that you will have good understanding of the data needs and processing steps involved by the time you finish this tutorial.

6.1 Pre-Processing of Geo-Hydrologic Data

There may be some pre-processing of data required apart from re-projecting the dataset into one the geographic project system as mentioned earlier. Several dataset such as Soil, Land Cover have their own way of assigning classification information. Before you could use them into PIHMgis you will need to re-assign those class numbers starting with 1 (one) and so on for individual classes.

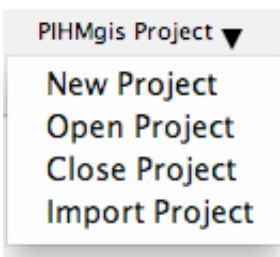
Also, you are required to develop a database from pre-existing Soil and Geology database. More information is provided later in this document in respective sections.

However, PIHMservices takes care of those pre-processing requirements by default and no manipulation should be required.

7. Projects

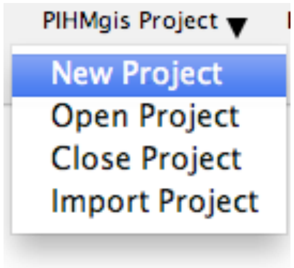
Project menu is an essential component of PIHMgis as it has been designed to keep tracks of your activity when you are working on a project and accordingly pre-populates many inputs fields, which is meant to assist users. It essentially helps user save time and also ensures proper input of data to individual PIHMgis modules.

Four options are available under PIHMgis Project menu as shown in figure below that are helpful in various circumstances and their utility has been explained in following sub-sections.



7.1 New (PIHMgis) Project

A modeling exercise of a watershed can be treated as a Project. There may be multiple iterations required in order to successfully meet the objective – "PIHMgis project". The folder where you downloaded the dataset can be treated as your Home-Directory (or Home-Folder – note that this is (can be) different from User's-Home-Directory) and you may create several projects under this folder corresponding to every iteration. Each project will have its own folder basically to keep things organized. New (PIHMgis) Project module creates a new PIHMgis-project-folder and some other sub-folders within that to organize all the intermediate outputs from individual PIHMgis modules. *.pihmgis file is also created within the project-folder which is the PIHMgis-project-file where all the changes (input/output) are recorded.



Select ***New Project*** from the ***PIHMgis Project*** drop down menu. This should bring New Project dialog [Figure 7.1] on the screen.

Processing steps: Here you will browse your Project-Home-Directory (i.e. where you have downloaded the dataset). Based on Date and Time you create a project it will automatically assign a PIHMgis-Project-Folder and the project file name. After you click on the Create Project button it will create a new project folder along with other empty folders, which will be used later. If the process executes successfully you will be greeted with the success message.

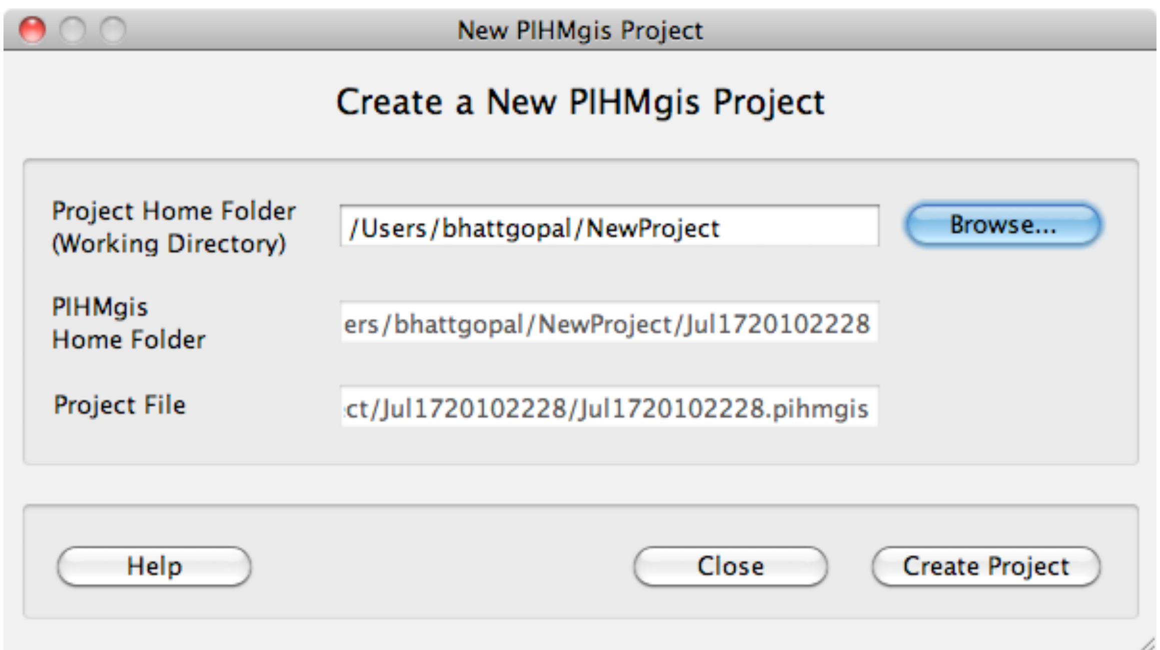
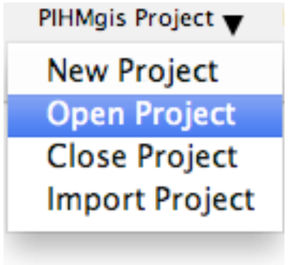


Figure 7.1: Create a new PIHMgis project dialog

7.2 Open Project

As the name suggests you can open any existing project, which was closed earlier (or is not currently active). Note that if you closed a project but changed the location of the project folder you will need to import it first.



Select ***Open Project*** from the ***PIHMgis Project*** drop down menu. This should bring Open Project dialog [Figure 7.2] on the screen.

Processing steps: On this dialog you need to browse for the PIHMgis project file. After you make your selection it will extract Home and Project folders. After you click the Open Project button it will load the project. If the process executes successfully you will be greeted with the message.

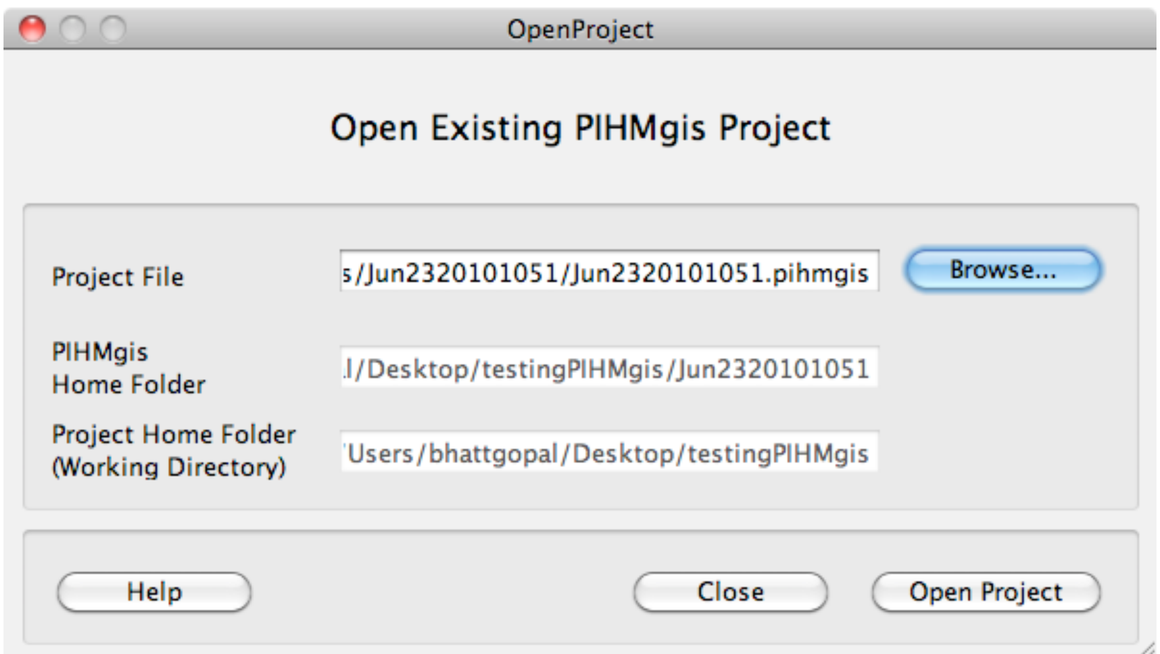
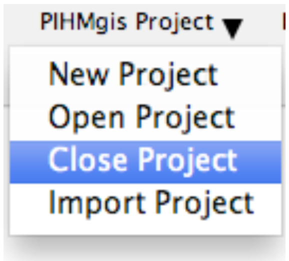


Figure 7.2: Open an existing PIHMgis Dialog

7.3 Close Project

Supposing you are working on multiple projects and want to switch between them without corrupting the PIHMgis project file (recall it stores /

updates the information when you execute a module). You should close the current project so that project file is not over written.



Select ***Close Project*** from the ***PIHMgis Project*** drop down menu. This should bring Close Project dialog [Figure 7.3] on the screen.

Processing steps: On this dialog you will see the current PIHMgis project that is open. If you are sure that this is the project you want to close, click Yes to close the project. If the process executes successfully you will be greeted with the message.
Tip: This module can also be used to make sure what is the name of the currently open project.

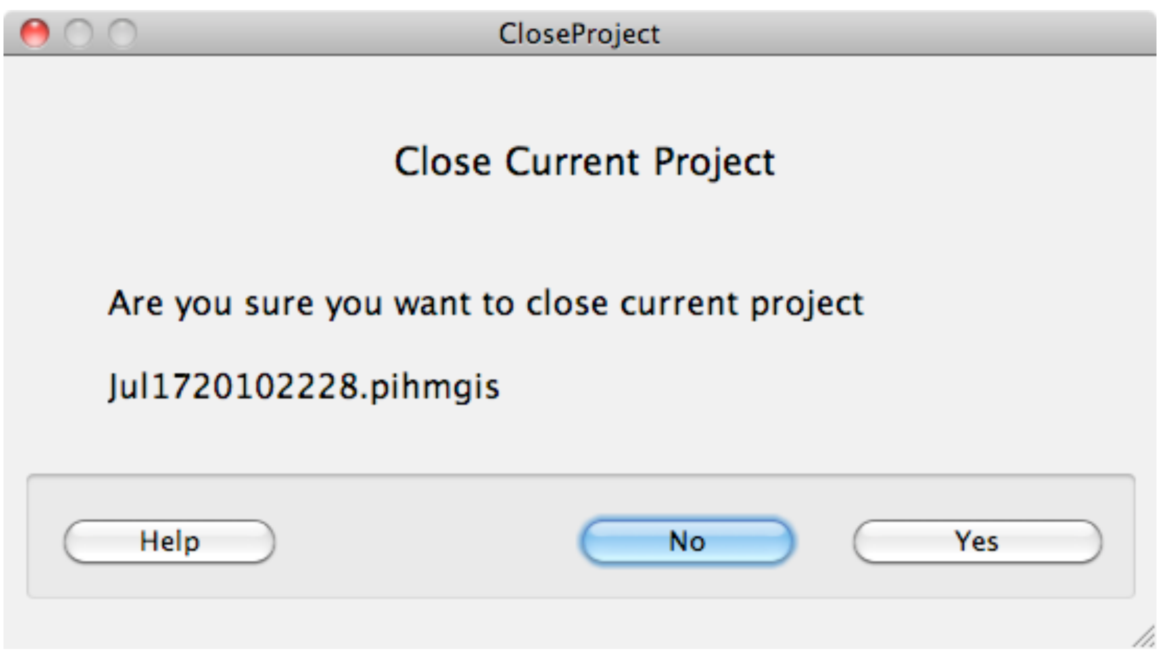
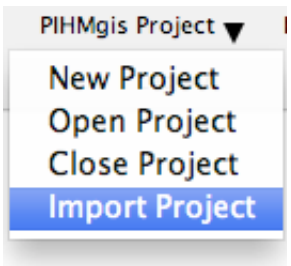


Figure 7.3: Close currently open PIHMgis project

7.4 Import Project

Import Project module allows you to import an existing project. Since the path of the file that is recorded in the PIHMgis project file is an absolute path – if you change the name of any directory on the root directory tree those files won't be accessible. Also, if you happen to move the folders around (including sending or receiving from different computers), Import project will update the existing project file so that you can continue to work on the project.



Select ***Import Project*** from the ***PIHMgis Project*** drop down menu. This should bring Import Project dialog [Figure 7.4] on the screen.

Processing steps: On this dialog you will need to browse an existing PIHMgis project that you want to import. After you load the project it will automatically extract the old directory information and will also show you the current directory where the project is being imported. If everything looks good, click on the Import Project to import project. If the process executes successfully you will be greeted with the message.

Tip: This also loads the imported project as current project by default.

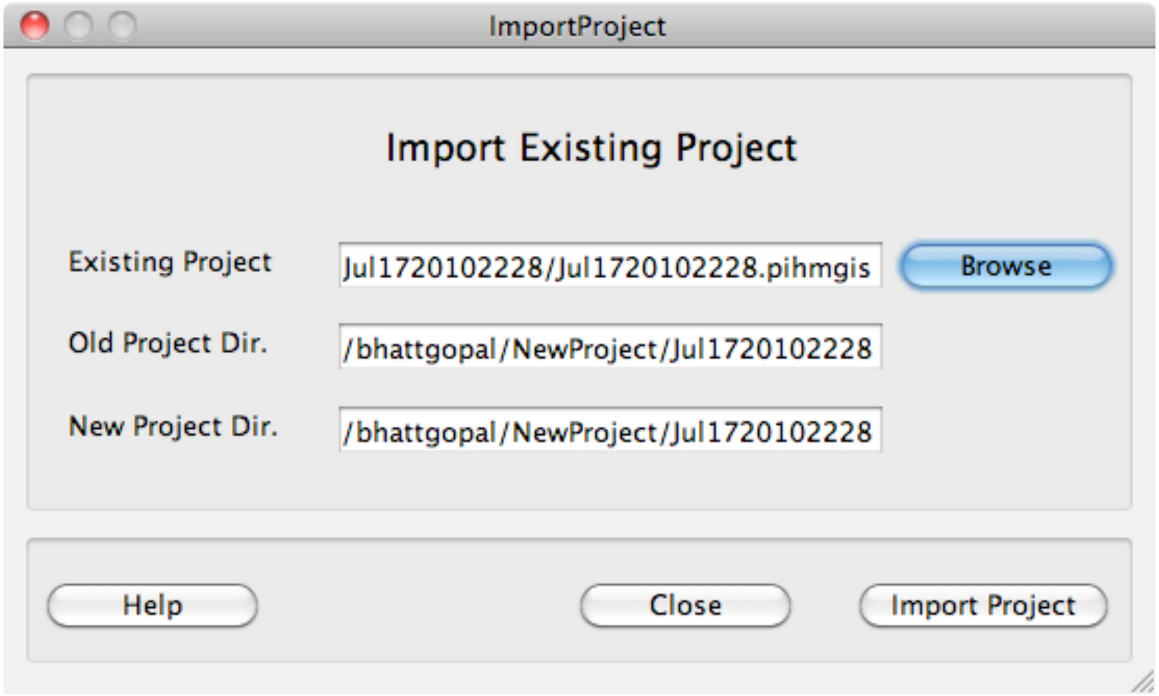
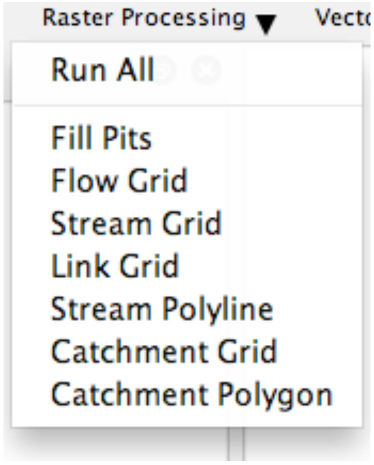


Figure 7.4: Import any existing PIHMgis project

8. Raster Processing

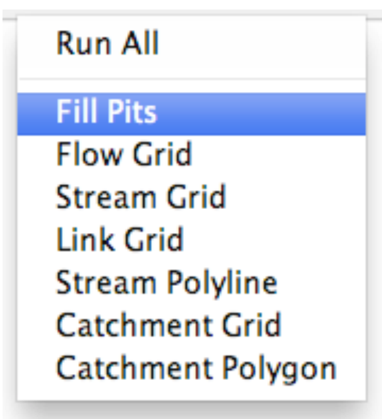
Earlier we discussed that for a simple model setup we can use watershed boundary and stream network as constraining layer (constraining layer is a feature file which is used for descretizing the modeling domain). Raster processing facilitates subshed/watershed delineation and stream definition from the Digital Elevation Model (DEM) of any region.

A raster dataset can be loaded into the QGIS framework by selecting ***Layers >> Add Raster Layer*** from the top menu or simply clicking icon in the File Toolbar. In order to successfully complete raster processing one needs to step through seven (7) modules (Start RunAll module after you are comfortable using these 7 modules individually). The steps performed are shown in the following Raster Processing m enu:



8.1 Fill Pits

Fill Pits fills pits in a grid. If a cell is surrounded by higher elevation cells, the water is trapped in that cell and cannot flow out. They are generally considered to be artifacts that interfere with the routing of flow across grid, and are removed by raising their elevation to the point where they essentially drain off the edge of the grid. Original pit locations can be identified and "protected" from getting modified by this function. (TAUDEM, accessed 2006).



Select ***Fill Pits*** from the ***Raster Processing*** drop down menu. This should bring Fill Pits dialog [Figure 8.1] on the screen.

<u>Processing steps:</u> In the Input group box of the dialog browse to the raw DEM file. It could be either an ESRI binary (*.adf) or Arc/Info ascii file (*.asc). In the Output section browse a file name to which the pit-filled grid could be saved. At this point the module is ready for processing. Select ***Run*** to start processing. Depending on the size and resolution of

DEM the processing could take several minutes. The text browser at the bottom of the dialog provides information related to status of processing. If the Load in Data Frame is checked the Pit-Filled Grid will be automatically loaded in the Qgis display window. After the processing is complete you need to press ***Close*** to proceed to next step.

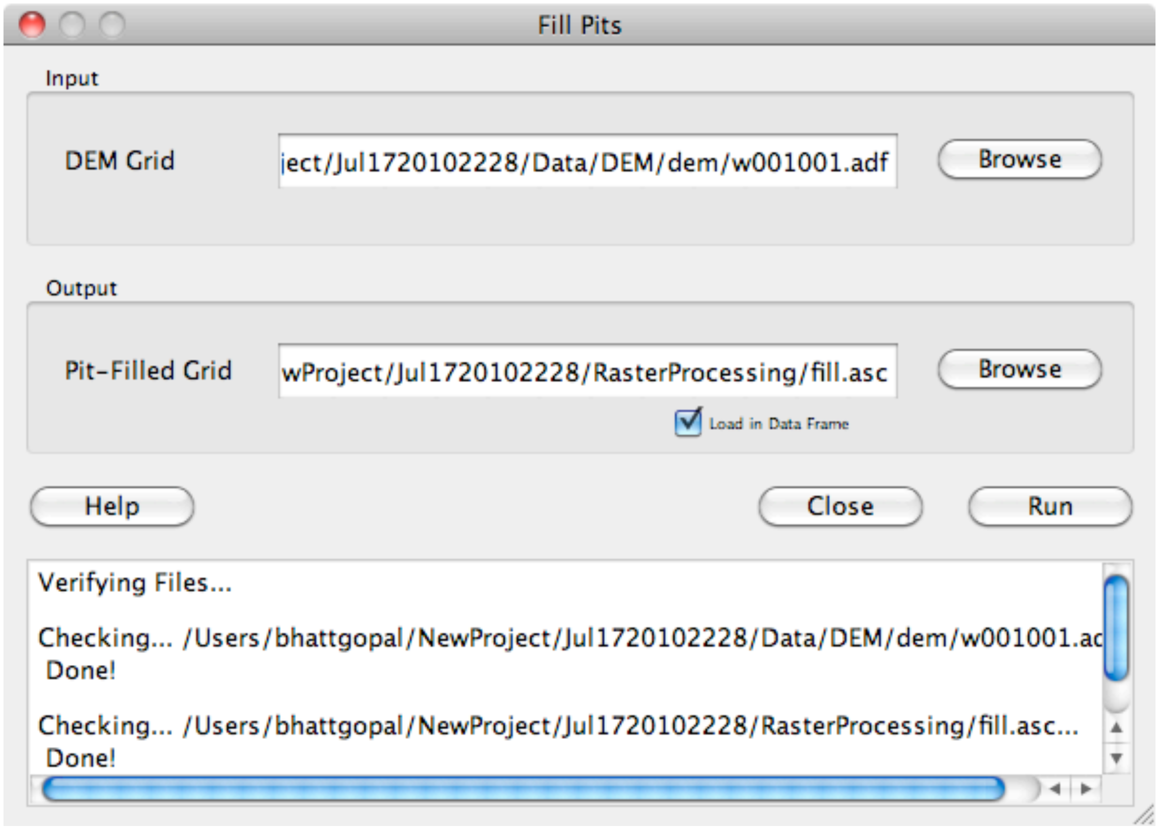


Figure 8.1: Fill Pits Dialog

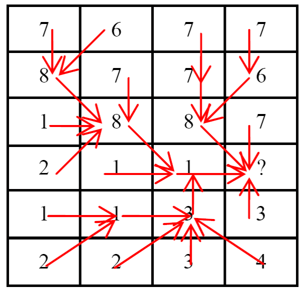
8.2 Flow Grid

Flow Direction produces an encoded grid with the neighboring cell direction to which the steepest descent is found using D8 algorithm [O'Callaghan and Mark (1984)]. The encoding used are 1 - east, 2 - North east, 3 - North, 4 - North west, 5 - West, 6 - South west, 7 - South, 8 - South east.

Flow Accumulation outputs an accumulation grid that contains the accumulated number of cells upstream (drains) to it, for each cell in the input (fill pit) grid using a recursive procedure explained in (Mark, 1988).

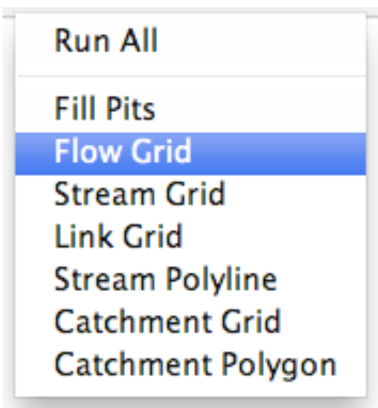
Figure 8.2 shows the Flow direction and Flow accumulation calculations performed on a synthetic DEM grid.

52	51	50	39
35	40	37	38
39	34	32	34
41	54	29	10
51	35	30	35
49	43	49	40



0	0	0	0
2	0	1	1
0	6	4	0
0	0	15	23
0	2	6	0
0	0	0	0

Figure 8.2: Flow direction and Flow accumulation for synthetic grid.



Select ***Flow Grid*** from the ***Raster Processing*** drop down menu. This should bring Flow Grid dialog [Figure 8.3] on the screen.

Processing steps: In the Input section of the dialog browse the Pit Filled Grid generated by the step 8.1. In the Output section browse the file names to which the flow direction and flow accumulation grid could be saved. At this point the module is ready to run; select ***Run*** to start processing. Please be patient until it completes processing. The text browser at the bottom of the dialog should provide information related to any error or processing. If the Load in Data Frame is checked the Flow Direction Grid and Flow Accumulation Grid will be automatically loaded in the Qgis window. After the processing is complete you need to press ***Close*** to proceed to next step.

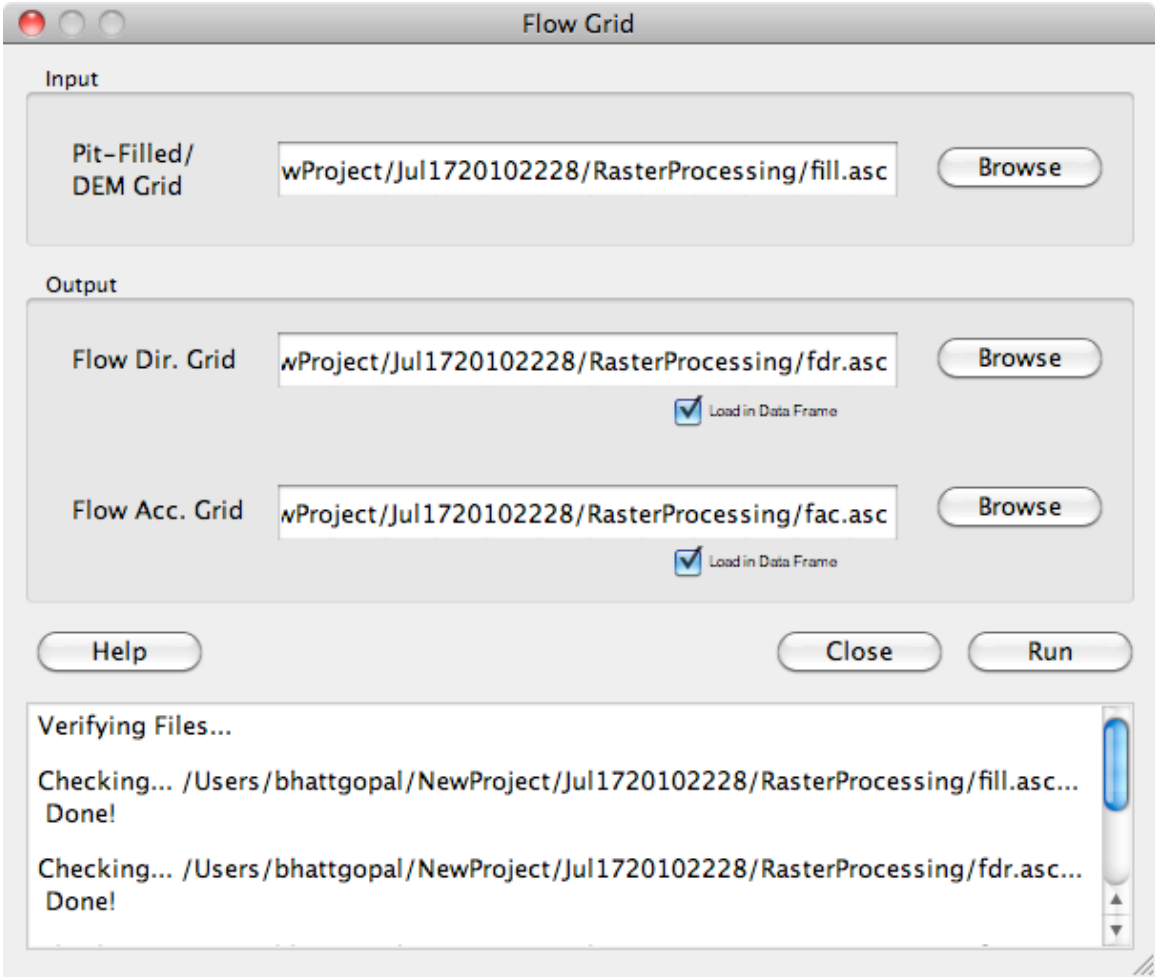


Figure 8.3: Flow Grid Dialog

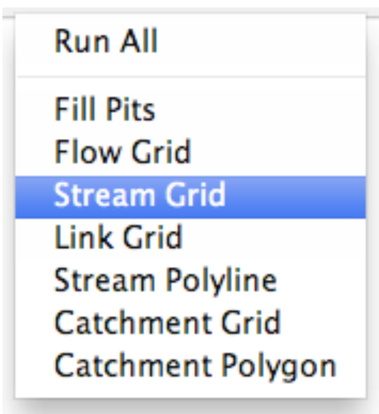
8.3 Stream Grid

Stream Grid is a raster equivalent of the stream network. Those Flow Accumulation Grid having value equal to or greater than the user provided threshold value, are marked 1. Physically, chosen threshold value represents the condition that a grid should be identified as a Stream grid if number of cells draining to that particular cell is equal to or greater than specified threshold. Rest of the grid assumes No Data Value (Non-stream grids).

Figure 8.4 shows stream grid generated when a threshold value of 2 is applied to the flow accumulation grid produced in the previous section.

0	0	0	0
1	0	0	0
0	1	1	0
0	0	1	1
0	1	1	0
0	0	0	0

Figure 8.4: Stream Grid for the synthetic grid.



Select ***Stream Grid*** from the ***Raster Processing*** drop down menu. This should bring Stream Grid dialog [Figure 8.5] on the screen.

Processing steps: In the Input section of the dialog browse the Flow Accumulation Grid generated by the step 8.2. In the Output section of the dialog browse the file name to which the stream grid could be saved (Notice the optional file naming convention used). Also it is required to provide an "*integer*" value for the threshold (A Value of **314** is used on purpose. However you can try and see the effect of different threshold). Any Flow Accumulation Grid with value greater than threshold will be classified as a stream grid. At this point module is ready to run; select ***Run*** to begin processing. Please be patient until it completes processing. The text browser at the bottom of the dialog should provide information related to any error or processing. If the Load in Data Frame is checked the Stream Grid will be automatically loaded in the Qgis window. After the processing is complete you need to press ***Close*** to proceed to next step.

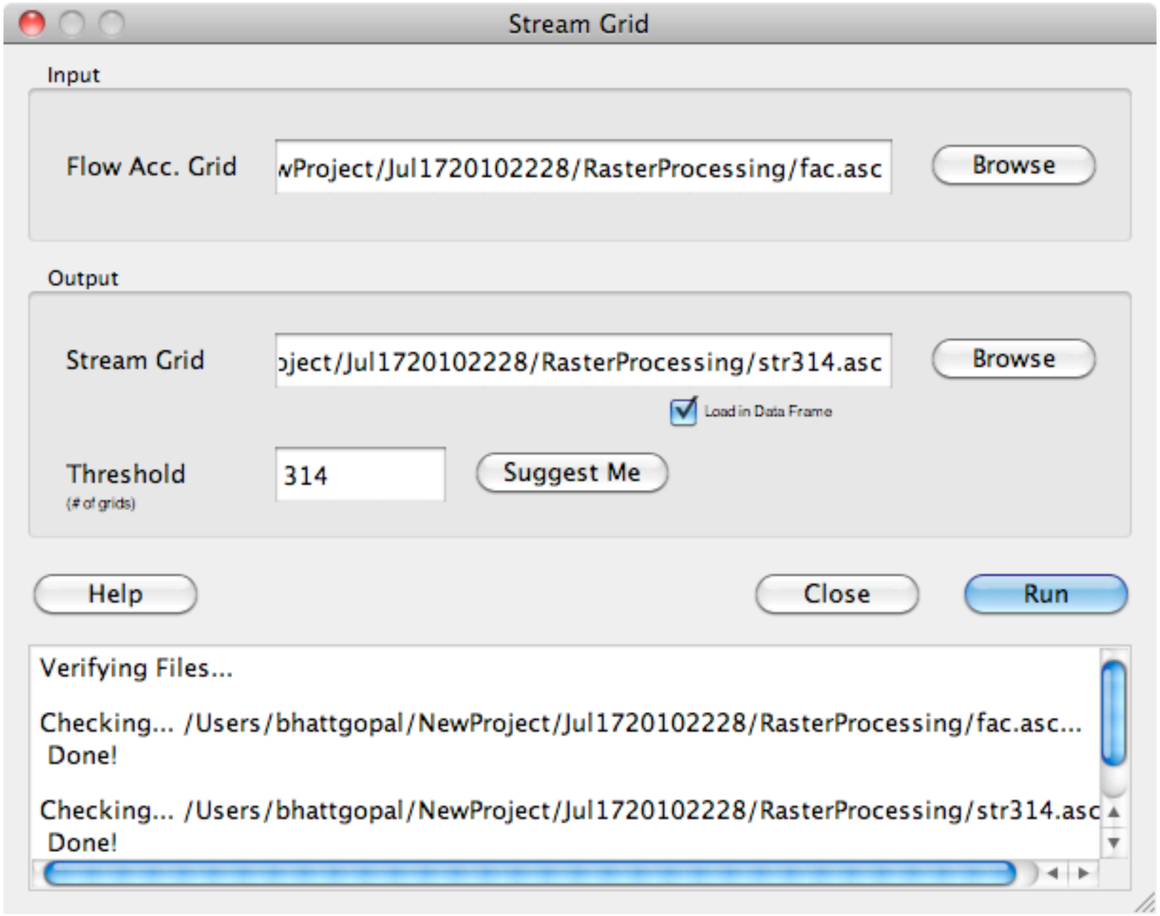


Figure 8.5: Stream Grid Dialog

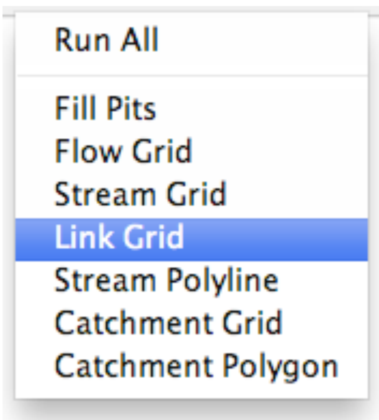
8.4 Link Grid

Link Grid separates the stream grid segments at the junctions (junction – where two streams intersects/meets). Each Link Grid segment is assigned a unique integer value starting with 1. The rest of the grid assumes NoData value similar to that of Stream Grid.

Figure 8.6 shows the Link Grid generated corresponding to the Stream Grid obtained in the previous section.

0	0	0	0
3	0	0	0
0	3	3	0
0	0	1	1
0	2	2	0
0	0	0	0

Figure 8.6: Link Grid for the synthetic grid.



Select ***Link Grid*** from the ***Raster Processing*** drop down menu. This should bring Link Grid dialog [Figure 8.7] on the screen.

Processing steps: In the Input section of the dialog browse the Stream Grid and Flow Direction Grid generated by the step 8.3 and 8.2 respectively. In the Output section of the dialog browse the file name to which the Link Grid could be saved (Notice the optional file naming convention used). At this point module is ready to run; select ***Run*** to begin processing. Please be patient until it completes processing. The text browser at the bottom of the dialog should provide information related to any error or processing. If the Load in Data Frame is checked the Link Grid will be automatically loaded in the Qgis window. After the processing is complete you need to press ***Close*** to proceed to next step.

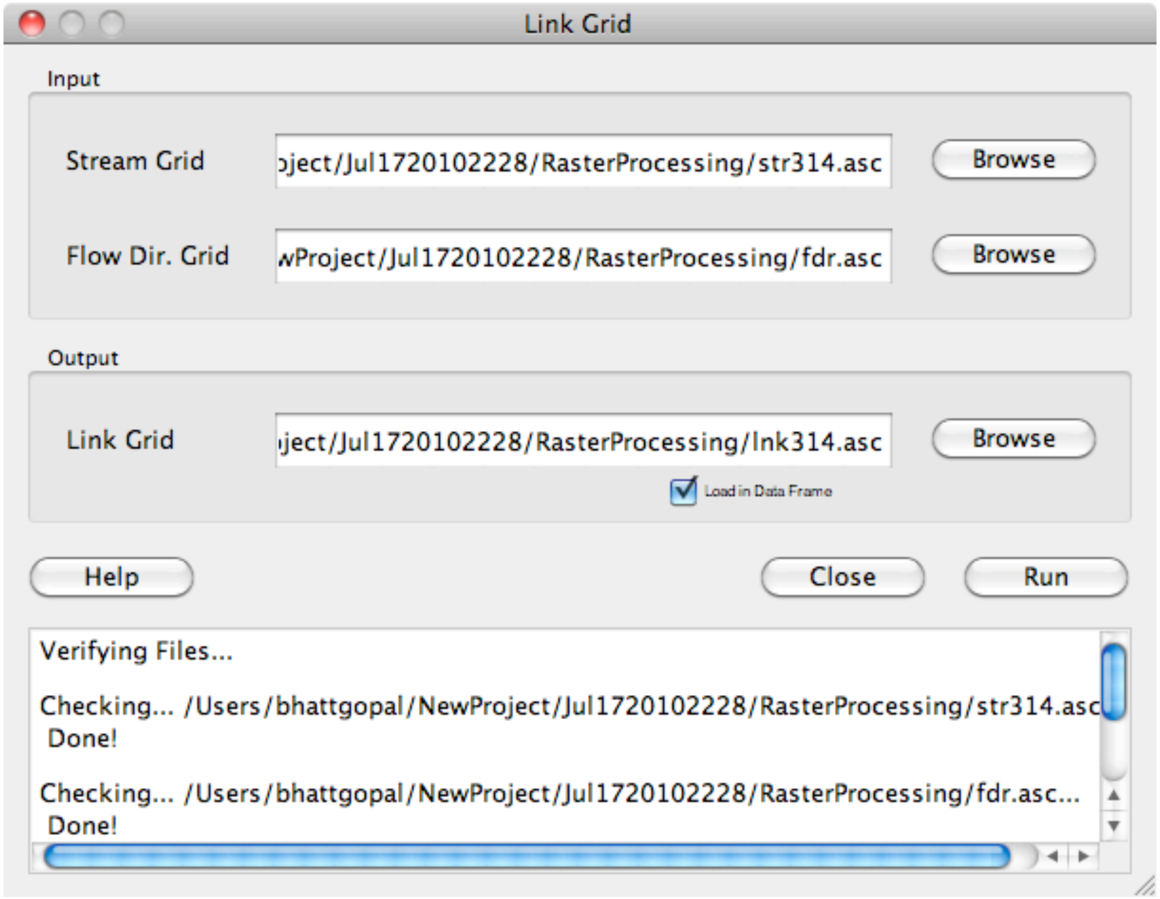


Figure 8.7: Link Grid Dialog

8.5 Stream Polyline

Stream polyline (polyline is a collection of connected lines) is the drainage network for the region of interest obtained by the conversion of the Link Grid to the vector format from the raster (link grid). Each link segment forms an individual stream segment and connected at the junction points. Flow direction is used to ensure that the segments are topographically correct (i.e. From-Node and To-Node are consistent with the flow direction; Points on a stream polylines are arranged FROM upstream TO downstream manner).

Figure 8.8 shows the stream polyline obtained corresponding to the link grid and flow direction grid obtained in the previous section for the synthetic grid.

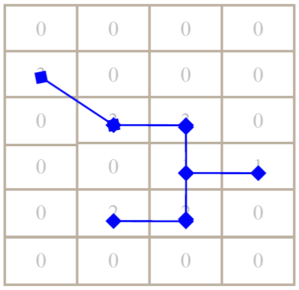
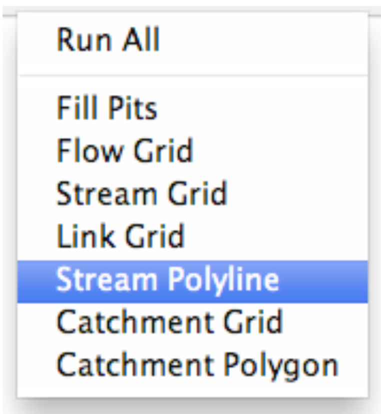


Figure 8.8: Stream Polyline for the synthetic grid.



Select ***Stream Polyline*** from the ***Raster Processing*** drop down menu. This should bring Stream PolyLine dialog [Figure 8.9] on the screen.

Processing steps: In the Input section of the dialog browse the Link Grid and Flow Direction Grid generated by the step 8.4 and 8.2 respectively. In the Output section of the dialog browse the shape file name to which the Stream Polyline could be saved (Notice the optional file naming convention used). At this point module is ready to run; select ***Run*** to begin processing. Please be patient until it completes processing. The text browser at the bottom of the dialog should provide information related to any error or processing. If the Load in Data Frame is checked the generated Stream Lines will be automatically loaded in the Qgis window. After the processing is complete you need to press ***Close*** to proceed to next step.

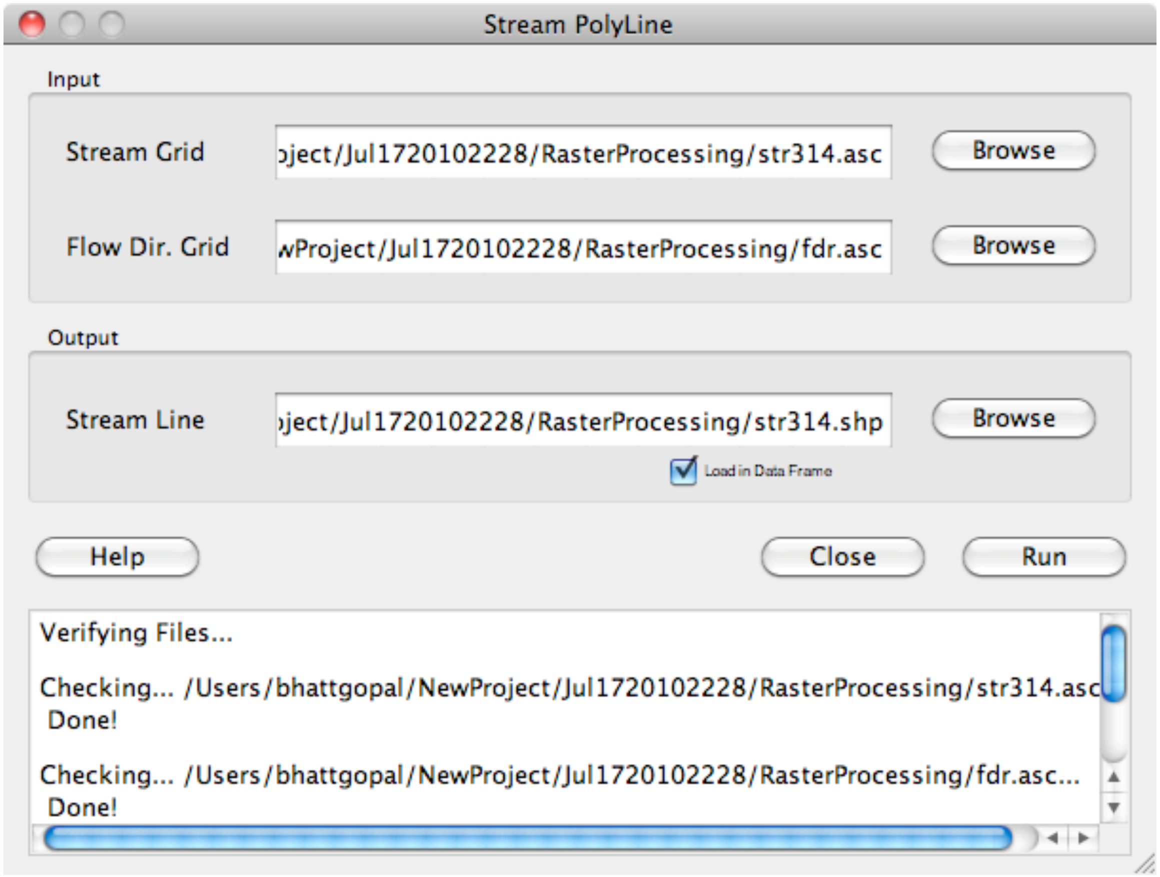


Figure 8.9: Stream Polyline Dialog

8.6 Catchment Grid

All the grids draining to a particular stream polyline element (stream segment) are grouped into one catchment grid. Catchment grids are marked according to the stream polyline nomenclature with integer numbers starting with 1 (Catchment grids marked as N drains to polyline ID N in stream shape file).

Figure 8.10 shows the catchment grid obtained for the synthetic grids using Stream and Flow direction grid discussed in section 8.3 and 8.2 respectively. Different colors are used for clear representation of the catchment grid.

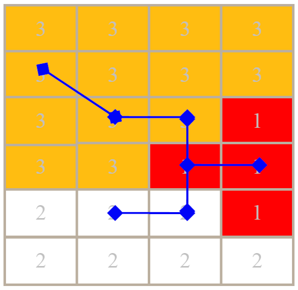
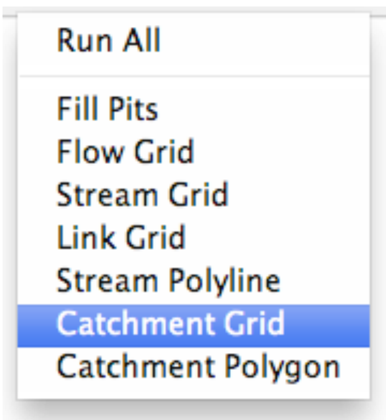


Figure 8.10: Catchment grid for the synthetic grid.



Select ***Catchment Grid*** from the ***Raster Processing*** drop down menu. This should bring Catchment Grid dialog [Figure 8.11] on the screen.

Processing steps: In the Input section of the dialog browse the Link Grid and Flow Direction Grid generated by the step 8.4 and 8.2 respectively. In the Output section of the dialog browse the file name to which the Catchment Grid could be saved (Notice the optional file naming convention used). At this point the module is ready to run; select ***Run*** to begin processing. Please be patient until it completes processing. The text browser at the bottom of the dialog should provide information related to any error or processing. If the Load in Data Frame is checked the generated Catchment Grid will be automatically loaded in the Qgis window. After the processing is complete you need to press ***Close*** to proceed to next step.

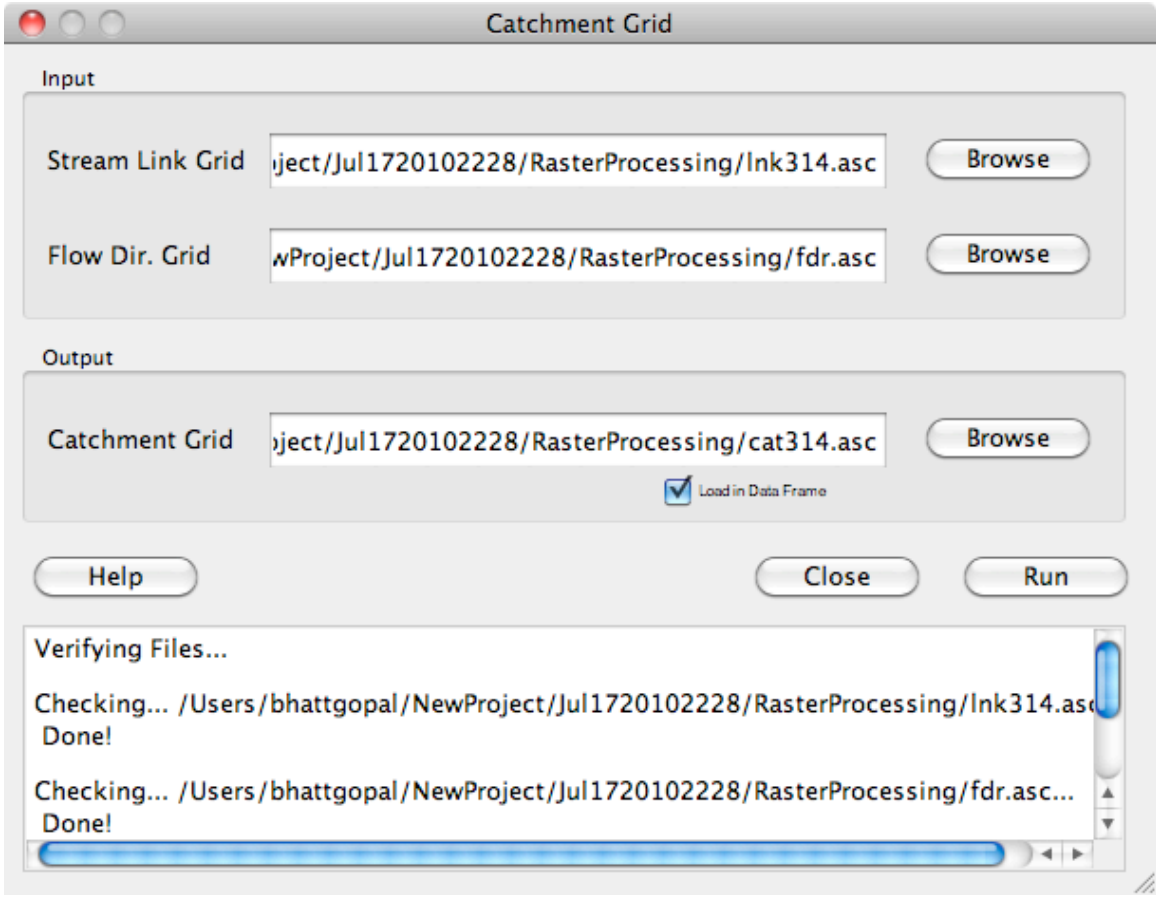


Figure 8.11: Catchment Grid Dialog

8.7 Catchment Polygon

Catchment polygons are the vector representation of the catchment grid. Similar to the catchment grid a catchment polygon bounds the region which has a single drainage outlet. Boundary of all catchment grids draining to the same stream segment forms a unique catchment polygon.

Figure 8.12 shows the catchment polygon obtained corresponding to the catchment grid obtained in the section 8.6 for the synthetic grid.

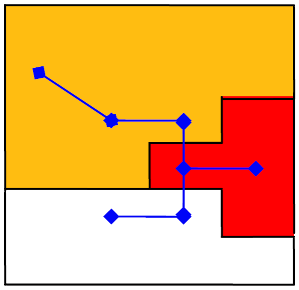
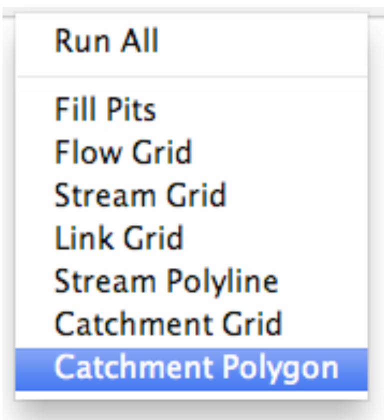


Figure 8.12: Catchment Polygon for the synthetic grid.



Select ***Catchment Polygon*** from the ***Raster Processing*** drop down menu. This should bring Catchment Polygon dialog [Figure 8.13] on the screen.

Processing steps: In the Input section of the dialog browse the Catchment Grid generated by the step 8.6. In the Output section of the dialog browse the file name to which the Catchment Polygon could be saved (Notice the optional file naming convention used). At this point module is ready to run; select ***Run*** to begin processing. Please be patient until it completes processing. The text browser at the bottom of the dialog should provide information related to any error or processing. If the Load in Data Frame is checked the generated Catchment Polygon will be automatically loaded in the Qgis window. After the processing is complete you need to press ***Close*** to proceed to next step.

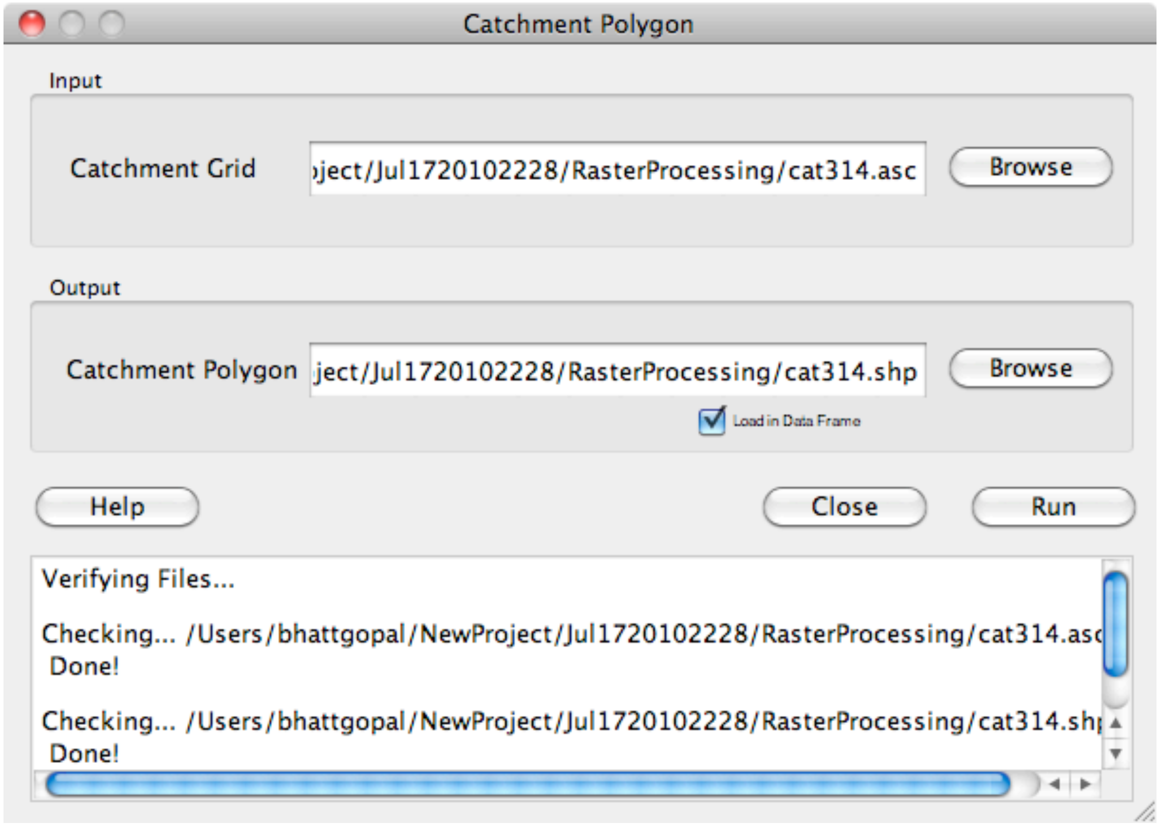


Figure 8.13: Catchment Polygon Dialog

After successful completion of Raster Processing in this tutorial we have generated 6 grid files and 2 shape files (Note that "a shape file" by itself consists of a number of related file. If you want to transfer or copy a shape file you will need to carry all the associated files. All associated files have the same name but different extensions). Now it is time to move on to Vector Processing. Before doing so, note that copies of Watershed boundary and stream network shapefiles (Figure 8.14) are made into Vector Processing directory. Also, behavior of copy by default is such that if the file with same name exists in destination folder already it will NOT be overwritten and preserve the files just in case you made any changes by editing.

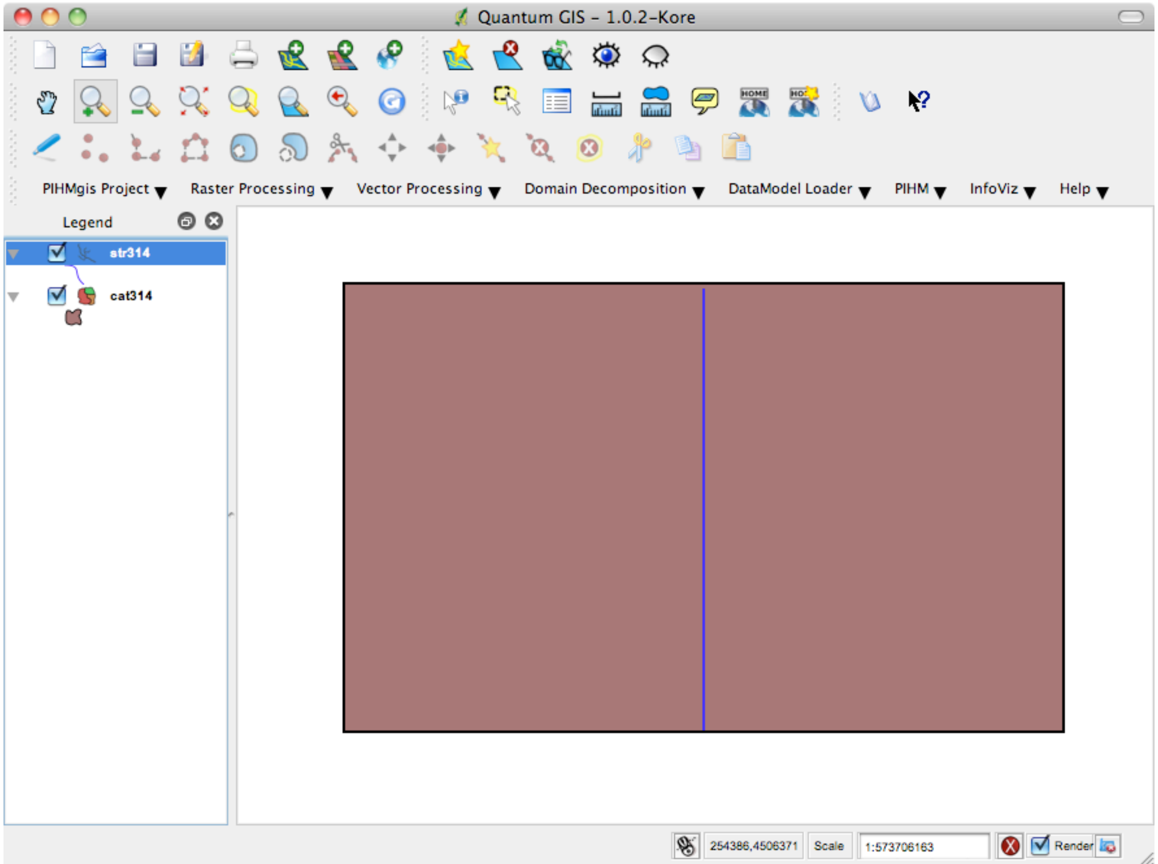


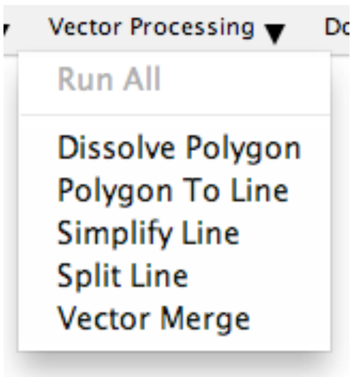
Figure 8.14: Watershed boundary and stream network

9. Vector Processing

Again recall from earlier that objective was to obtain watershed boundary and stream network, which can be used as constraining layer to decompose the modeling domain. Before we could use these vector files we need to (1) condition them and (2) merge into one vector file. Condition is done in terms of getting rid of irregularities/flucutation present in the generated shapefile because of inheritance from the raster nature of the DEM. Also, in order to merge the vector files we need to have all the features of same type (i.e. lines) – watershed boundary is a polygon and stream network is a polyline. Vector processing consists of a set of operations, which addresses issues mentioned above and enables efficient discretization of the modeling domain. Stream polyline and Catchment polygon generated in section 8.5 and 8.7 can be used primarily for this purpose. However, other hydrologic constraint such as soil coverage, land cover type coverage can also be incorporated.

Eventually, it prepares a GIS layer which is used as input constraint for the purpose of domain decomposition.

These steps are accomplished by performing steps described in the following Vector Processing menu:

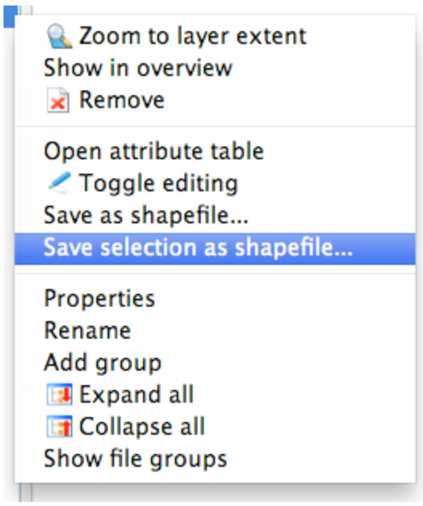


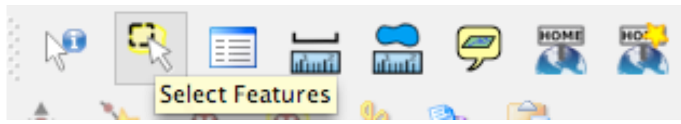
Tip: Note the project organization structure is defined in such a way that it is dependent on the way it populates the fields in following vector-processing modules. It is strongly recommended that you don't change any sequence in the way data is populated on the module. However, if you need to use any layer other than what is populated you could replace them with a similar feature or if you want to use other layers than watershed boundary and stream network, those layers can simply be add to the project when you are working on a particular module.

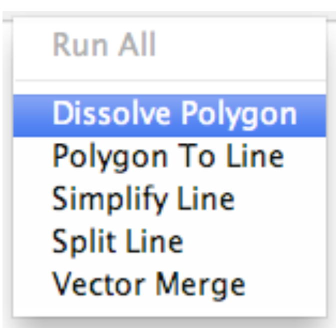
9.1 Dissolve Polygons

After the completion of Raster Processing we are going to carry forward with Watershed boundary and Stream network. However, depending on (1) Size of the DEM used you may get sub-watersheds those fall outside the modeling domain however they will still be part of the catchment polygon generated. (2) Depending on the threshold used for stream generation you can get multiple sub-watersheds within the modeling domain. You can keep those sub-watersheds but those internal boundaries lead to more number of triangle elements and in most modeling purposes it may not be important to keep those.

In order to rectify the issues outline you will need to select all the subshed boundaries that are part of the modeling domain. Select tool has been shown in the diagram blow. After you are done selecting right click on feature legend shown to the left, which provide you with the option to save the selection as a shapefile. **Important**: Also make sure to follow the same process for the stream network as well. **Note:** It is also important that you save these selections to the Vector Processing folders with the same file name. You will find that those files already exist in the folder – you can comfortably overwrite on them.







Select ***Dissolve Polygon*** from the ***Vector Processing*** drop down menu. This should bring Dissolve Polygon dialog [Figure 9.1] on the screen.

<u>Processing steps:</u> Make sure the catchment in the input is the one, which you exported after selecting out of the catchment polygons generated from the Raster Processing module. Click Run to execute the module.

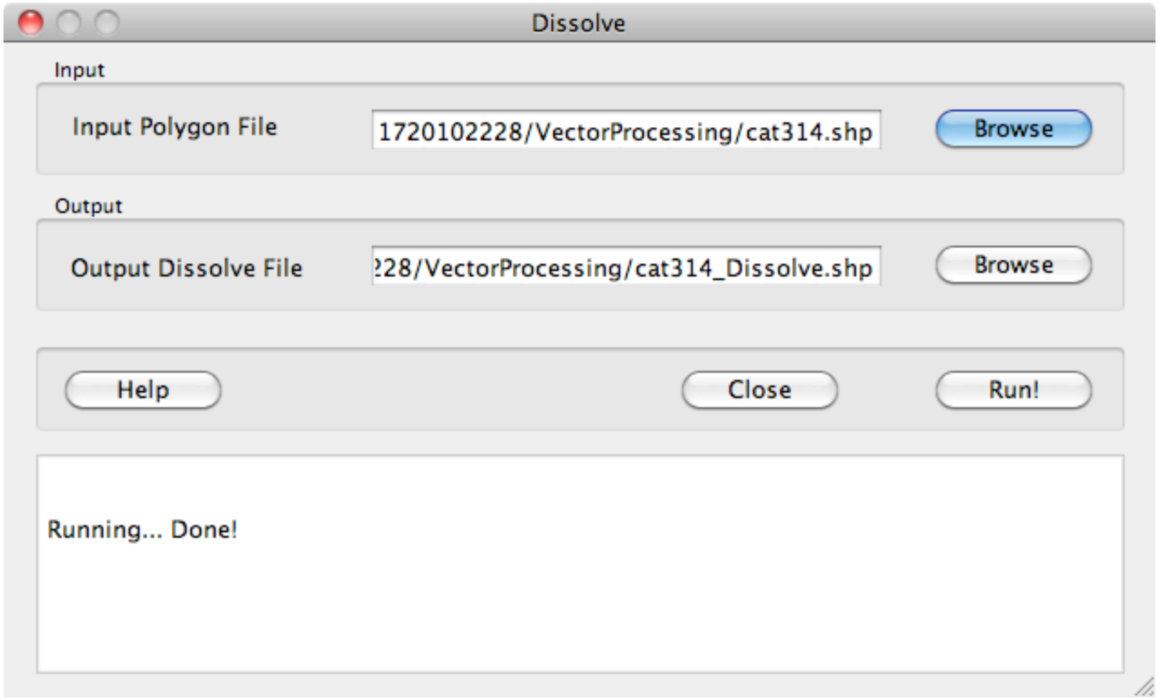


Figure 9.1: Dissolve polygons dialog

9.2 Polygon To Line

The geo-data feature can exist as point, line or polygon objects. In order to merge all the data features together before domain decomposition modules can use it, the object properties for all the features should be same. So polygons like lake feature and watershed boundaries are converted to polylines and then simple lines before they can be merged with already existing line features like river (also converted from polylines to lines). Figure 9.2 shows a schematic of the steps involved in the algorithm.

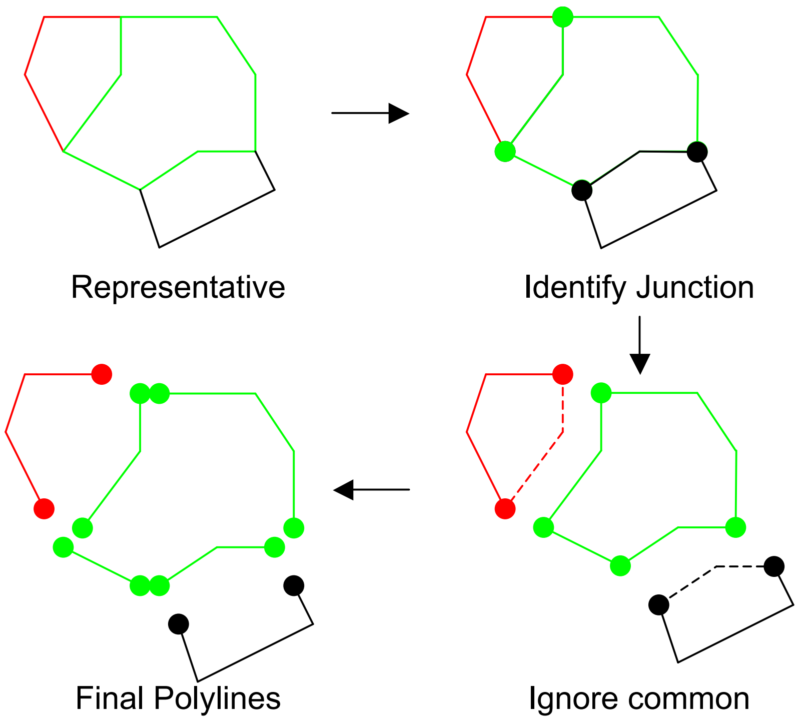
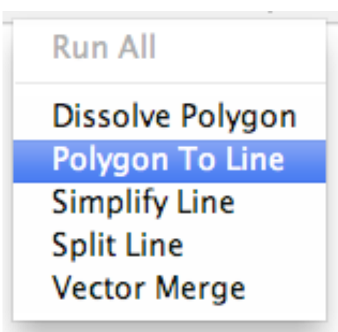


Figure 9.2: Algorithm for Polygon to Polyline



Select ***Polygon To Line*** from the ***Vector Processing*** drop down menu. This should bring Polygon To Line dialog [Figure 9.3] on the screen.

Note: This step should be performed on POLYGON features ONLY.

Processing steps: First of all add shape file(s) of "polygon" type only using the button on the top right of the dialog. A default output file name for the corresponding polyline file is generated (editable if desired). button can be used to remove any file from the simplification routine by selecting the file from the left bar of the spread sheet. button clears everything from the dialog setting it afresh. After adding the polygon files module is ready to execute using ***Run*** button. The text browser provides information regarding any error or processing. After the operation is complete for all the files you can ***Close*** the dialog and processed to next step.

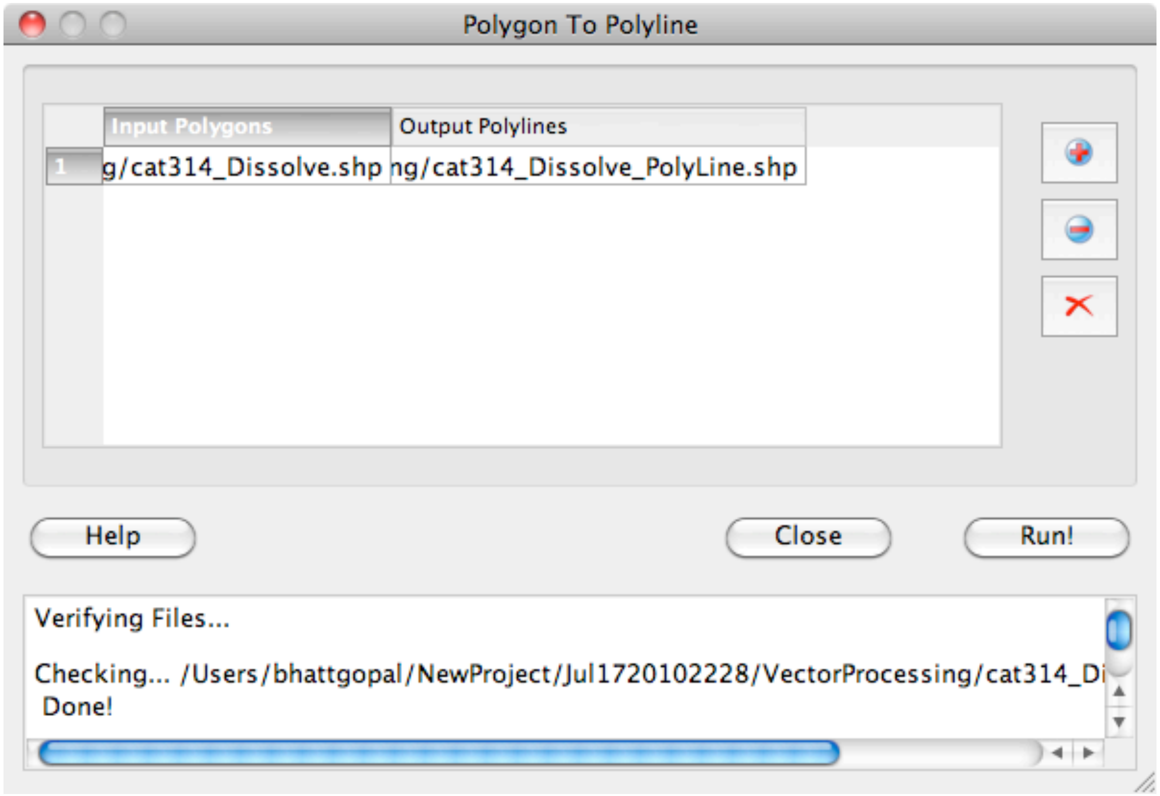


Figure 9.3: Polygon To Polyline Dialog

9.3 Simplify Line

Simply Line is used to simplify a polyline by removing small fluctuations or extraneous bends from it while preserving its essential shape and nodes in case of streams. This step becomes particularly crucial for the quality and efficient domain decomposition as an unsimplified feature can have (1) unnecessarily large number of nodes present and (2) sharp bends, which in turn determines the number of triangles generated. Needless to say, larger is the number of decomposed triangle elements, larger will be the computational requirement of the numerical model simulation. Figure 9.4 shows intermediate steps in polyline simplification that is an implementation of the Douglas-Peucker algorithm.

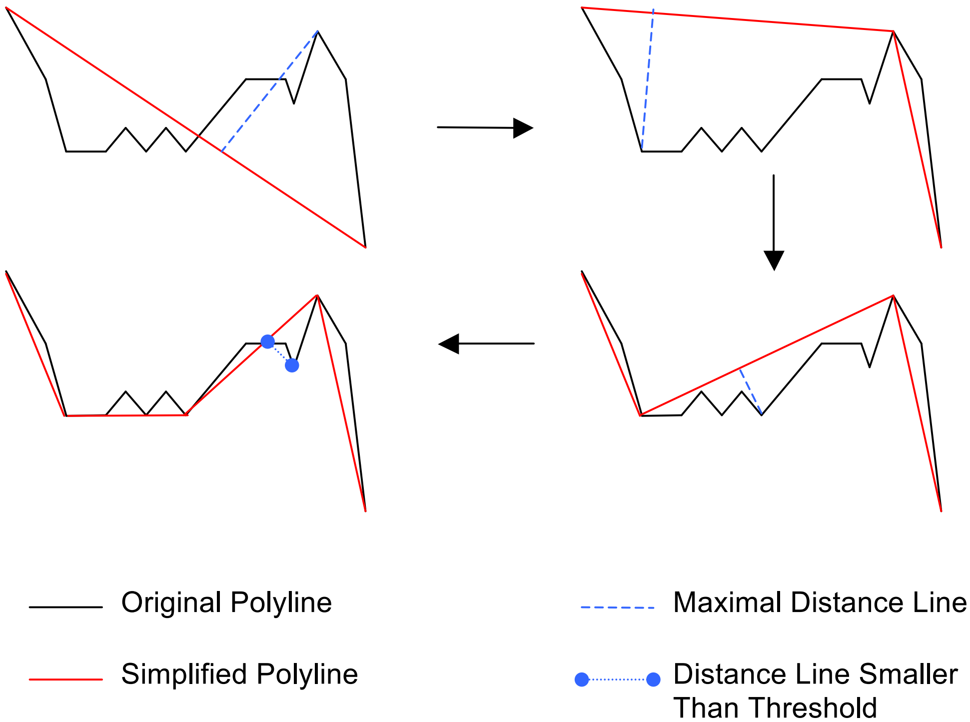
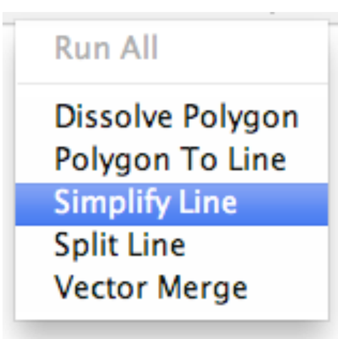


Figure 9.4: Algorithm for Simplification



Select ***Simplify Line*** from the ***Vector Processing*** drop down menu. This should bring Simplify Line dialog [Figure 9.5] on the screen.

NOTE: You will not see any change in stream network in the scope of v-catchment exercise. However, this step is generally important for more complicate modeling exercises, where stream network is more complicated than a simple straight line.

<u>Processing steps:</u> Browse all shape file(s) of "polyline" type only which you want to simplify. A tolerance (double variable type) should be provided which acts as the maximum distance tolerance value for the simplification algorithm. After that button allow you to add the file and the specified tolerance to the project. A default output file name for the corresponding simplified polyline file is generated that is editable if desired. The tolerance value field in the table is also editable. button can be used to remove

any file(s) from the simplification routine by selecting the file(s) from the left bar of the table.  button clears everything from the dialog setting it afresh.  button is provided for simplicity in editing the threshold value for a bunch of flies at a time. After adding the polyline files module is ready to execute using ***Run*** button. The text browser provides information regarding any error or processing. After the operation is complete for all the files it is required to ***Close*** the dialog to processed to next step.

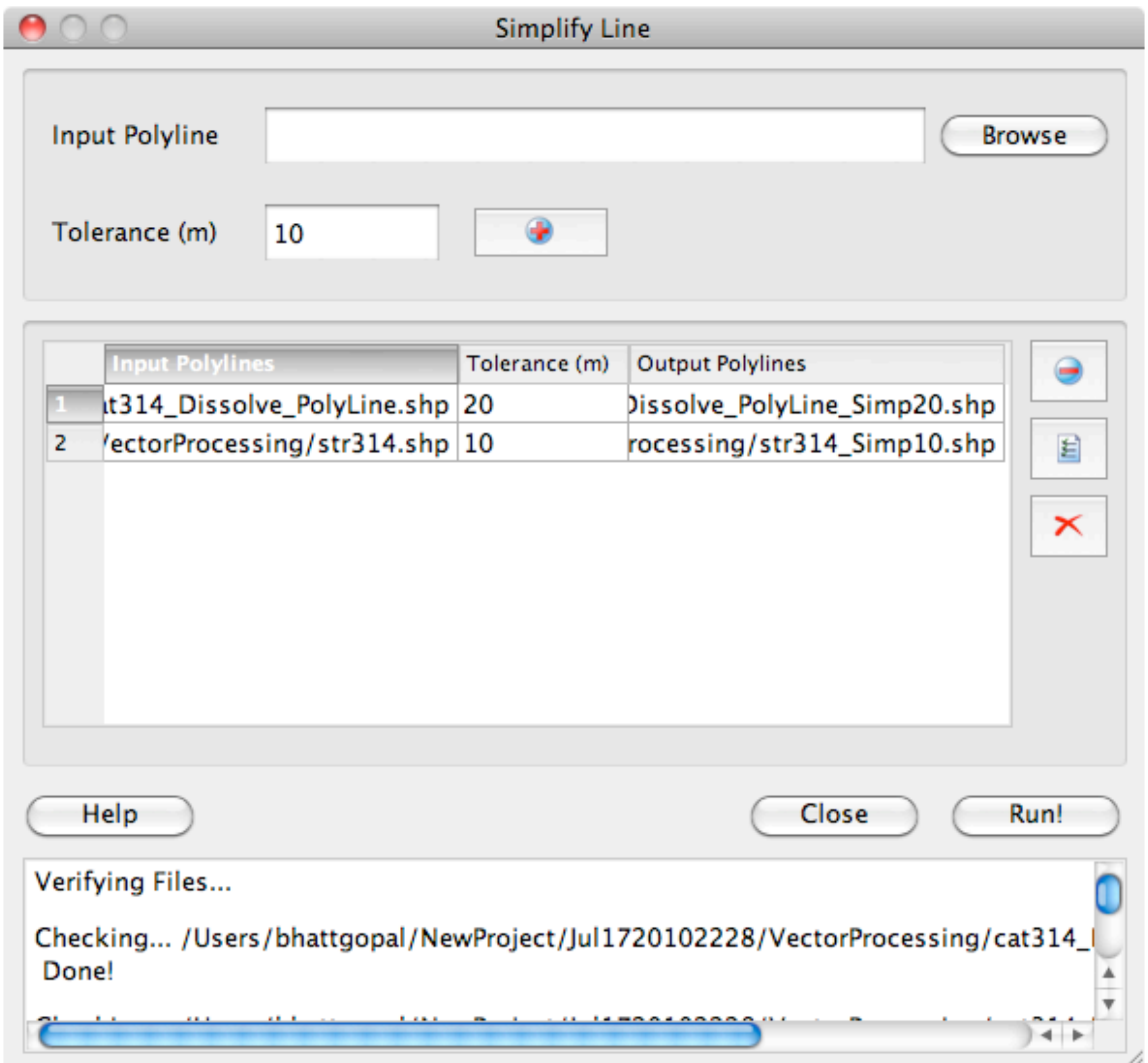


Figure 9.5: Simplify Line Dialog

Tip: Maintain the sequence of Watershed polyline first and stream polyline second.

9.4 Split Line

After execution of earlier steps we have features in polyline form (recall, a polyline is a collection of several lines). Before merging all the features together it is necessary to have them all in line feature type. Split line splits polylines at each vertex. Therefore, this turns a single polyline feature into a multiple line feature depending upon the number of vertices present in the original polyline.

Figure 9.6 schematically describes split line operations.

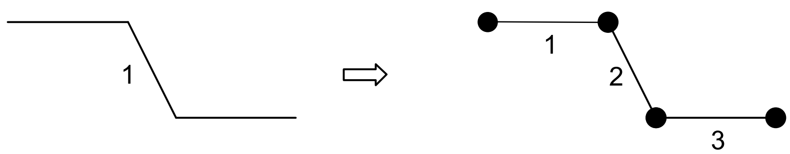
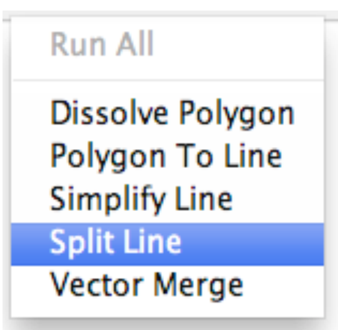


Figure 9.6: Split Line: Input has only one feature which is of type polyline. Output has three features which are of the type lines.



Select ***Split Line*** from the ***Vector Processing*** drop down menu. This should bring Split Line dialog [Figure 9.7] on the screen.

<u>Processing steps:</u> Add all the shape file(s) of "polyline" type (simplified polylines obtained in the section 9.3 using the  button. A default output file name for the corresponding splitted line feature file is generated for each input file. The output file name is editable if desired.  button can be used to remove any file(s) from the split line routine by selecting the file(s) from the left bar of the spread sheet.  button clears everything from the dialog setting it afresh. After adding the polyline files module is ready execute using ***Run*** button. The text browser provides information regarding any error or processing. After the operation is complete for all the files it is required to ***Close*** the dialog to processed to next step.

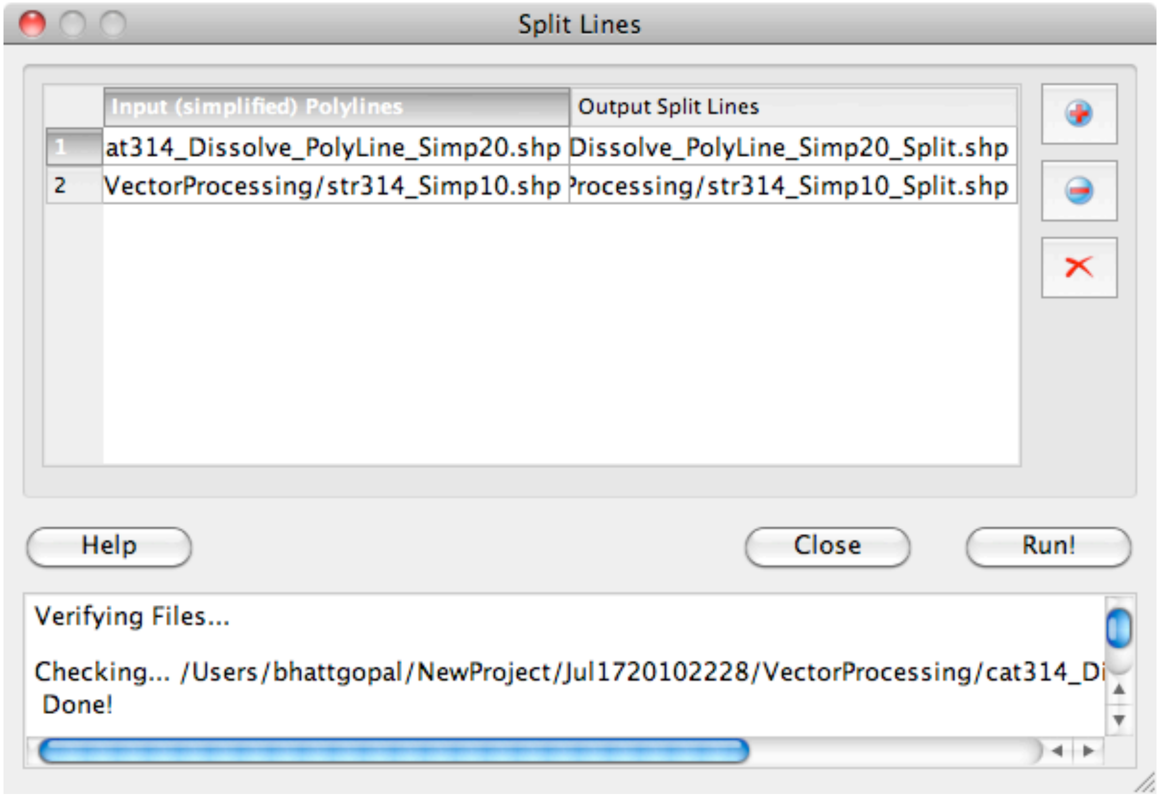


Figure 9.7: Split Line Dialog

Tip: Maintain the sequence of Watershed split line and stream split line.

Note, before you move further…

After previous processing steps we have obtained our constraining layer, which are optimized for efficient domain decomposition and most importantly have been converted into the same feature type (i.e. simple lines) and is ready to be merged into one shapefile. However, before we combine them all there are few issues that we need to address. First is generic to all the modeling exercise: (1) Editing the river outlet that we will discuss in next paragraph. Another step might be required in order to make decomposition more effective ("effective" word is used in the last sentence to reflect how a configuration results in terms of number of triangles – number of triangle is an important factor in computation time required by PIHM). If situation allows some changes (i.e. editing) on shapefiles can result in reduced number of triangles. If you happen to move forward without making those changes and realize after you have generated mesh that there is possibility of improvement you can always come back to this step and repeat following steps.

A careful observation of the stream outlet [see figure] will show that the reach terminates just before the catchment boundary. This happens because center of each elevation pixel is used in stream network generation. Two simple steps are required to fix this issue. To describe the problem precisely – essentially what we are trying to do is extend the stream network all the way such that it meets the watershed boundary. We achieve this by snapping tool that ships with the QGIS. You need to first set the **Snapping options** on **General** tab by selecting **File >> Project Properties** [Figure 9.8].

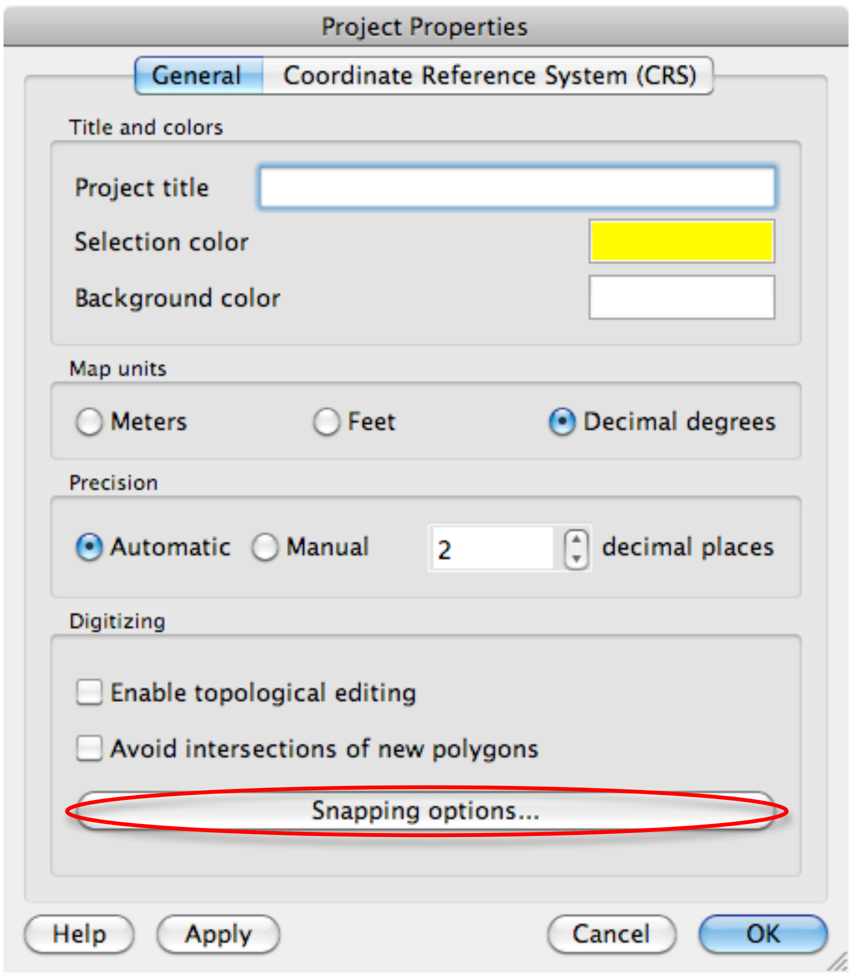


Figure 9.8: Dialog from where you choose snapping options

After you click on the Snapping options it will open a dialog. There are several ways you can perform this task. But here we will edit the stream file to match with the watershed boundary. So on the Snapping options dialog we are going to select watershed boundary as a layer to which we will snap our stream edits. For mode choose to vertex and segments and a tolerance of 50 [Figure 9.9].

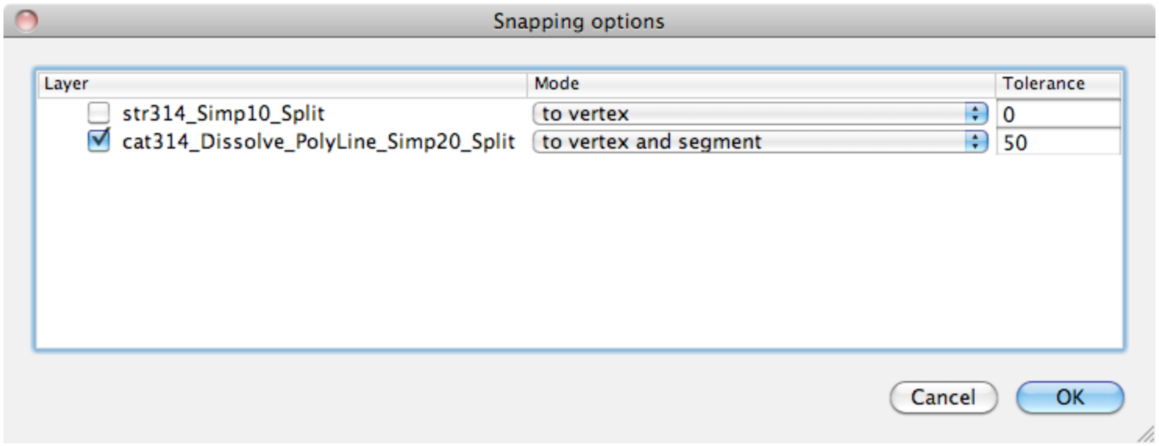


Figure 9.9: Dialog from where you can set snapping options

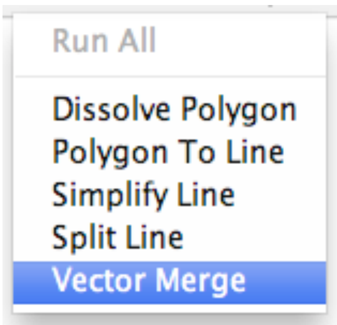
1. Select the stream shapefile (that we are going to edit) and click on Toggle Editing from the Digitizing toolber. This step makes the stream shapefile editable.

2. Carefully move the node of the river towards the watershed boundary [Figure 9.10] . Node should easily snap to the boundary. Toggle the edit mode to save changes.

Figure 9.10: Editing with the snapping options enabled

9.5 Vector Merge

Vector Merge is the final step of Vector processing. It merges all the layers into one shape file. The merged shape file is meant to acts as constraints in domain decomposition process (next 3 modules).



Select ***Vector Merge*** from the ***Vector Processing*** drop down menu. This should bring Vector Merge dialog [Figure 9.11] on the screen.

First of all add shape file(s) those need to be merged, in the input section using the button. User should be careful to provide only line features as input. In the output section browse an output shape file name to which the merged features will be saved. Cluster tolerance is a double type number. Any two nodes separated by the distance less than the cluster tolerance will be merged into one. button can be used to remove any file(s) from the routine by selecting the file(s) from the left bar of the spread sheet. button clears everything from the dialog setting it afresh. After adding the layers module is ready to execute using ***Run*** button. The text browser provides information regarding any error or processing. After the operation is complete for all the files it is required to ***Close*** the dialog to processed to next step.

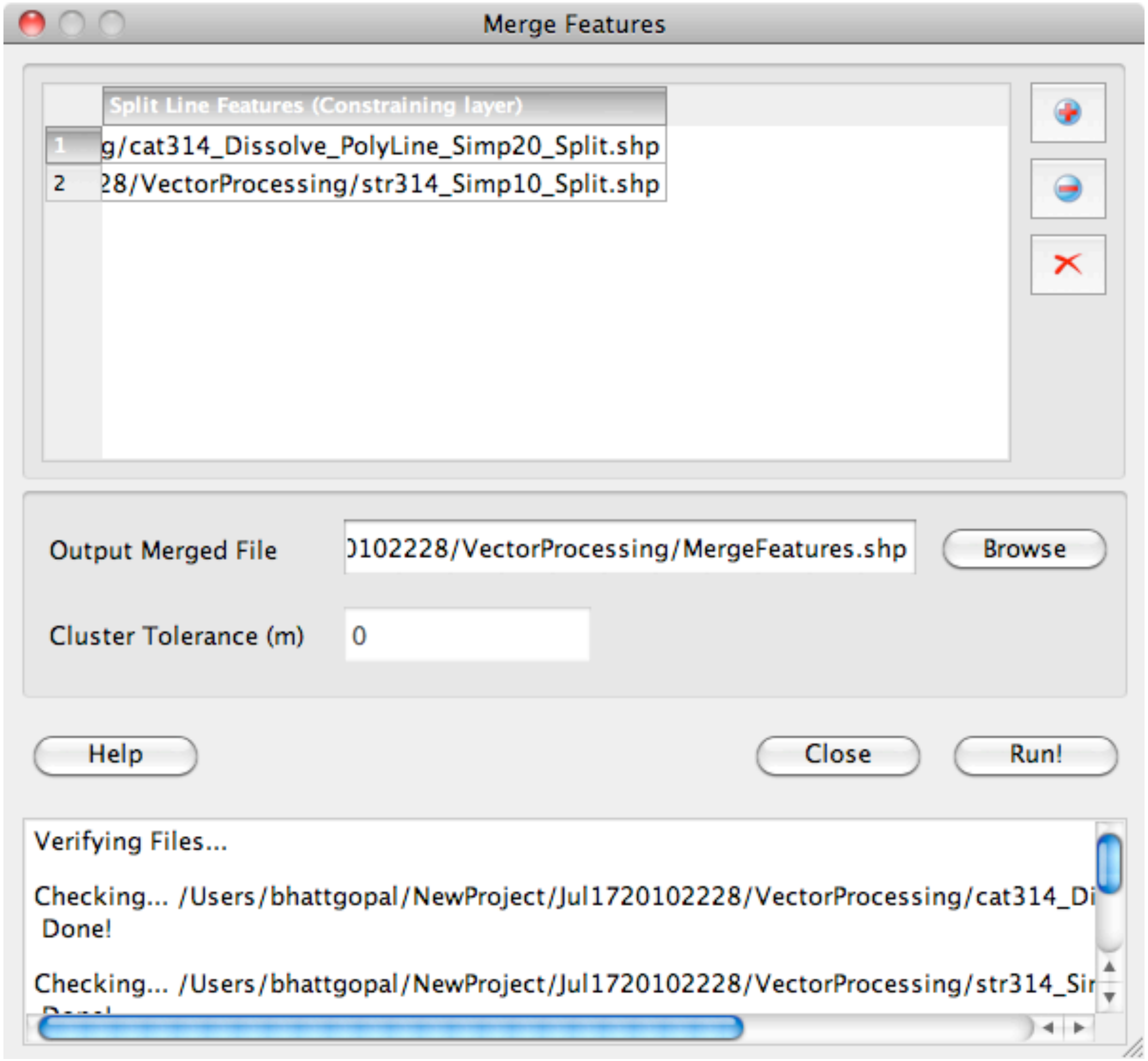
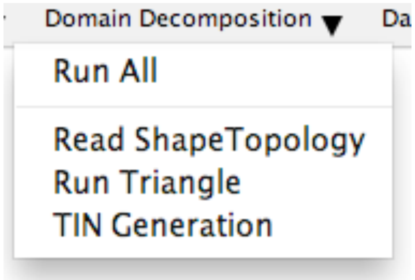


Figure 9.11: Merge Features Dialog

10. Domain Decomposition

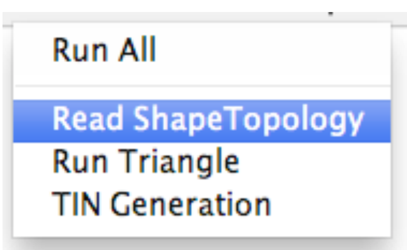
Domain decomposition applies Delaunay Triangulation [Delaunay, 1934] to decompose the modeling domain into triangular irregular mesh or triangular irregular network (TIN). An irregular mesh better represents a terrain if all critical terrain and hydrographic points are taken into account while performing domain decomposition. This may include watershed boundary, different types of contours (e.g. hypsometry, soil), stream network, hydraulic structures (e.g. dams, gages) for generating those points.

Domain Decomposition is performed using the following three steps shown in the following Domain Decomposition menu:



10.1 Read ShapeTopology

All the vector line information from the contrainig layer shapefile is extracted into planer straight line graph (PSLG) in this step that is saved to a *.poly file which in input to running the TRIANGLE in the next module. All the node and line information from the input shape file (vector merged shape file obtained in the section 9.5) is organized into specific poly file format.



Select ***Read ShapeTopology*** from the ***Domain Decomposition*** drop down menu. This should bring Read ShapeTopology dialog [Figure 10.1] on the screen.

<u>Processing steps:</u> In the input section of the dialog browse the shape file (merged shape file obtained in the section 9.5), that is constraining layer for domain decomposition. In the output section browse an output file name to which the poly file will be saved. After this, the module is ready to execute using the ***Run*** button. The text browser provides information regarding any error or processing. After the operation is complete ***Close*** the dialog to processed to next step.

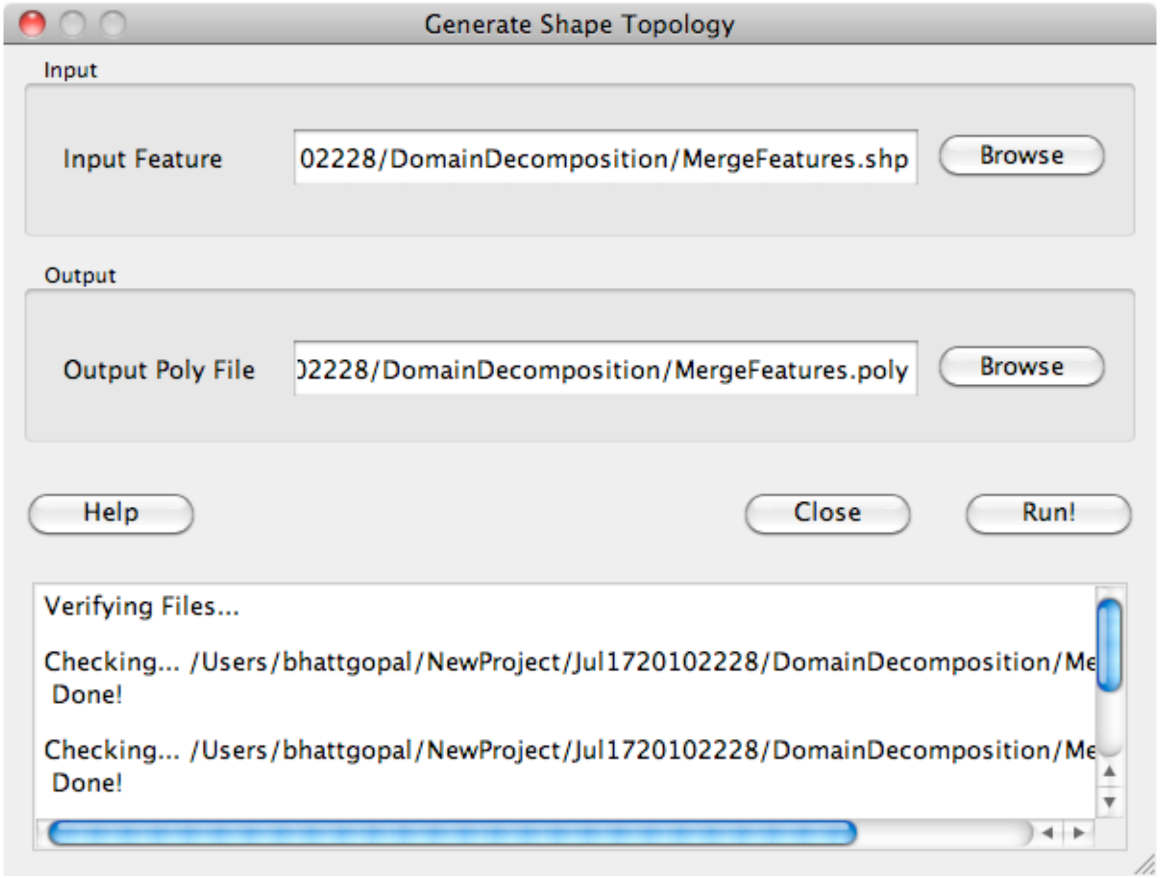
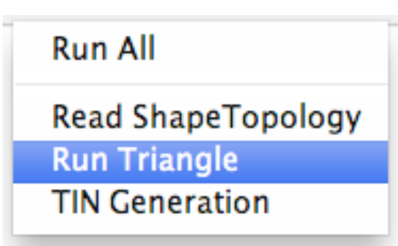


Figure 10.1: Read Shape Topology Dialog

10.2 Run TRIANGLE

'TRIANGLE' developed by Shewchuk [2001] is one of the efficient implementation of Delaunay triangulation algorithm which provides several flexibility and options to user. It is also computationally efficient. It takes planner straight line graph (PSLG) as input. The algorithm works in such a way that it refines the Delaunay triangulation by inserting carefully placed vertices until the generated mesh meets a provided quality and size criterion.



Select ***Run Triangle*** from the ***Domain Decomposition*** drop down menu. This should bring Run TRIANGLE dialog [Figure 10.2] on the screen.

<u>Processing steps:</u> In the Options section of the dialog there are several options that can be provided to TRIANGLE program. At the same time "Other" options can be used for the options that are not listed directly on the dialog. The available options and their purpose are listed on the table below. In the input poly file section browse the poly file generated in the section 10.1. A click on ***Run*** executes TRIANGLE. TRIANGLE generates four (4) output files: (1) .poly file; (2) .node file; (3) .ele file; and (4) .neigh file. They are stored in the same directory as the input poly file. It is advised to note that the value provided as minimum angle and maximum area has tremendous influence on the number of triangles generated. Number of triangles generated on the console or can be seen on the attribute table of the shapefile generated by the next (TIN Generation) module.

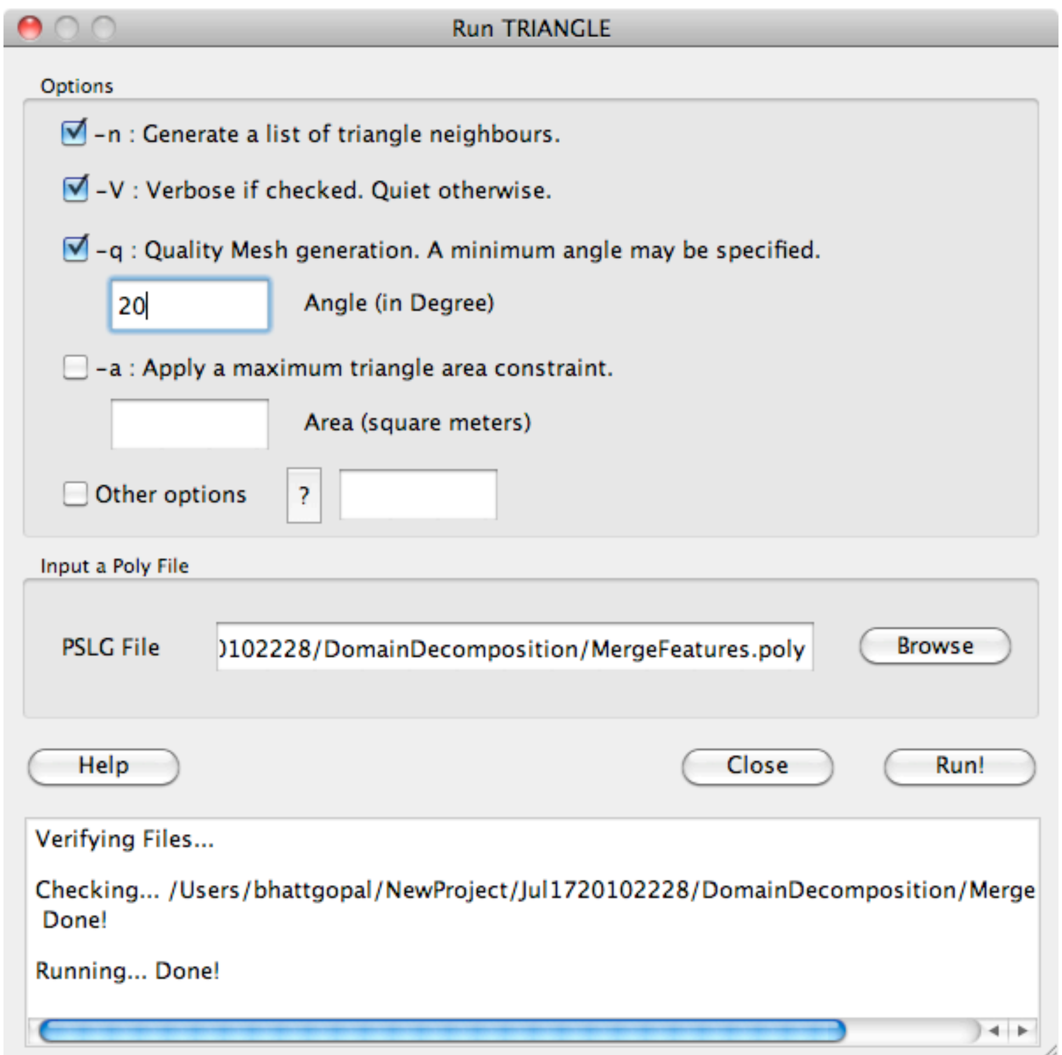


Figure 10.2a: Run TRIANGLE Dialog

```
  Angle histogram:
      0 -  10 degrees:         0    |     90 - 100 degrees:         5
     10 -  20 degrees:         0    |    100 - 110 degrees:         4
     20 -  30 degrees:        17    |    110 - 120 degrees:         2
     30 -  40 degrees:        15    |    120 - 130 degrees:         0
     40 -  50 degrees:        11    |    130 - 140 degrees:         1
     50 -  60 degrees:         5    |    140 - 150 degrees:         0
     60 -  70 degrees:        11    |    150 - 160 degrees:         0
     70 -  80 degrees:        18    |    160 - 170 degrees:         0
     80 -  90 degrees:        10    |    170 - 180 degrees:         0

Memory allocation statistics:

  Maximum number of vertices: 27
  Maximum number of triangles: 34
  Maximum number of subsegments: 21
  Maximum number of encroached subsegments: 1
  Maximum number of bad triangles: 52
  Maximum number of stacked triangle flips: 3
  Approximate heap memory use (bytes): 4336
```

Figure 10.2b: RunTriangle statistics displayed to the console

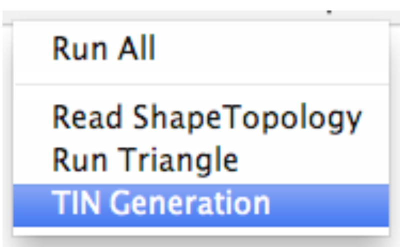
Table 10.1: TRIANGLE options

Options	Description
-p	Triangulates a Planar Straight Line Graph (.poly file).
-r	Refines a previously generated mesh.
-q	Quality mesh generation. A minimum angle may be specified.
-a	Applies a maximum triangle area constraint.
-u	Applies a user-defined triangle constraint.
-A	Applies attributes to identify triangles in certain regions.
-c	Encloses the convex hull with segments.
-w	Weighted Delaunay triangulation.
-W	Regular triangulation (lower hull of a height field).
-j	Jettison unused vertices from output .node file.
-e	Generates an edge list.
-v	Generates a Voronoi diagram.
-n	Generates a list of triangle neighbors.
-g	Generates an .off file for Geomview.
-B	Suppresses output of boundary information.
-P	Suppresses output of .poly file.
-N	Suppresses output of .node file.
-E	Suppresses output of .ele file.
-I	Suppresses mesh iteration numbers.

-O	Ignores holes in .poly file.
-X	Suppresses use of exact arithmetic.
-z	Numbers all items starting from zero (rather than one).
-o2	Generates second-order subparametric elements.
-Y	Suppresses boundary segment splitting.
-S	Specifies maximum number of added Steiner points.
-L	Uses equatorial circles, not equatorial lenses.
-i	Uses incremental method, rather than divide-and-conquer.
-F	Uses Fortune's sweepline algorithm, rather than d-and-c.
-l	Uses vertical cuts only, rather than alternating cuts.
-s	Force segments into mesh by splitting (instead of using CDT).
-L	Uses Ruppert's diametral spheres, not diametral lenses.
-C	Check consistency of final mesh.
-Q	Quiet: No terminal output except errors.
-V	Verbose: Detailed information on what I'm doing.
-h	Help: Detailed instructions for Triangle.

10.3 TIN Generation

This step reads the *.ele and *.node file produced by the TRIANGLE to generate a shapefile with all the triangle elements.



Select ***TIN Generation*** from the ***Domain Decomposition*** drop down menu. This should bring Unstructured Mesh Generation dialog [Figure 10.3] on the screen.

Processing steps: In the input section of the dialog, browse input files (*.ele file and *.node file) generated by the TRIANGLE in the section 10.2. In the output section browse an output file name where the unstructured mesh shape file could be saved. Click on the ***Run*** executes the routine. Any error or progress information is displayed in the text browser at the bottom of the dialog.

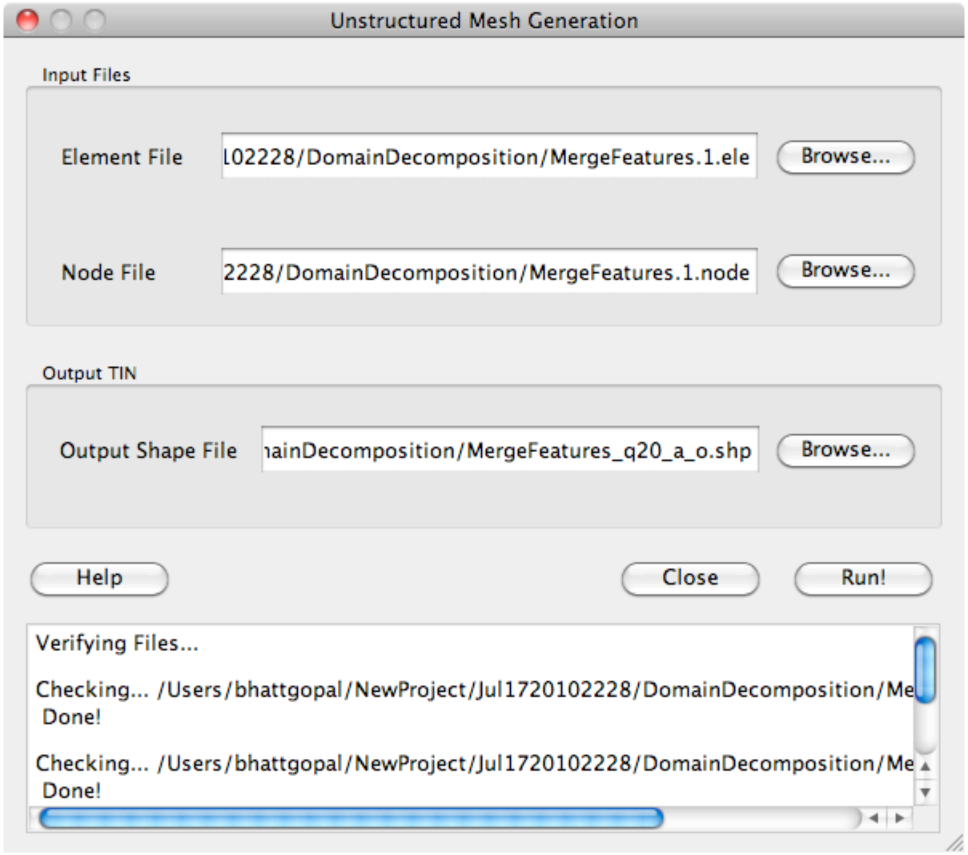


Figure 10.3a: Unstructured Mesh Generation Dialog

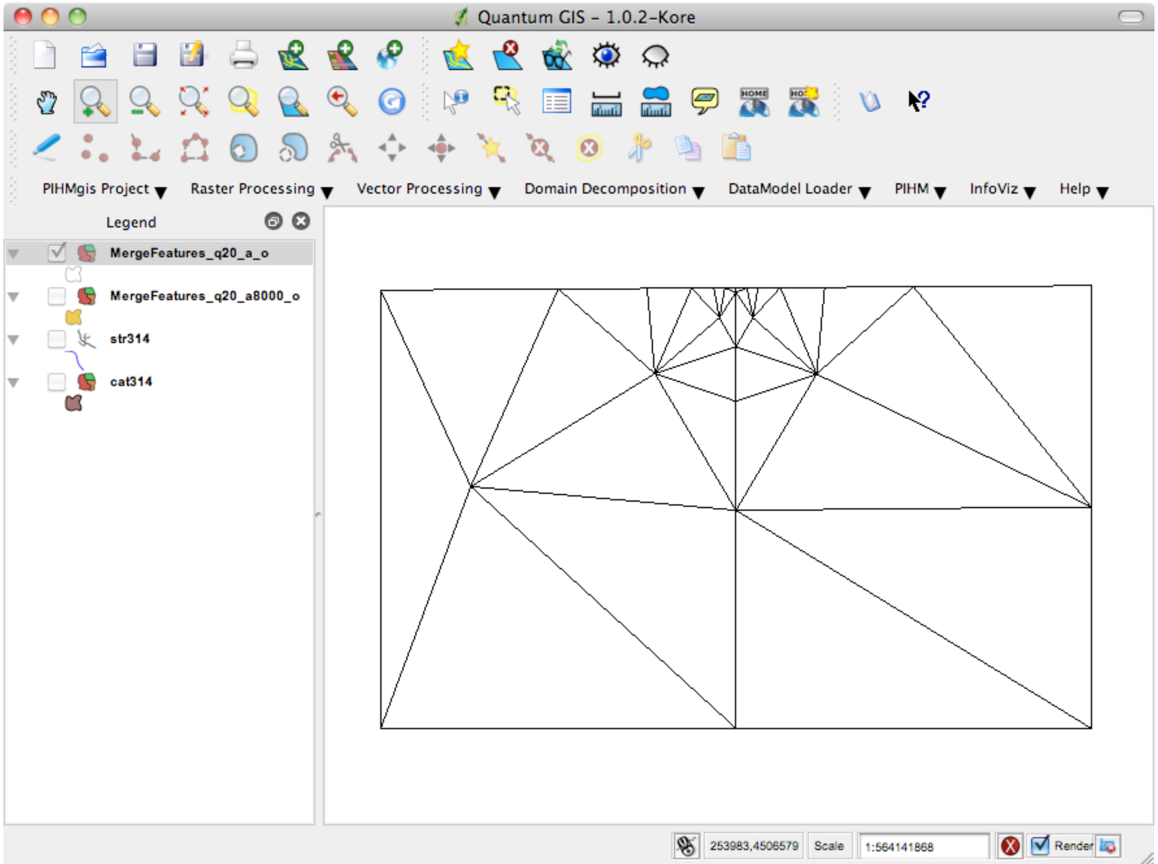


Figure 10.3b: Mesh generated generated.

<u>Unevenly sized triangular elements:</u> This situation is arising because of the nature of the constraining layer. Also, since most of the area in the domain does not have any constraint to follow and the stream segment is a straight line with only nodes at the beginning and end, the decomposition results as shown in Figure 10.3b. How you want your decomposition to look like might depend on several factors. Mainly the decision lies in the modeling purpose. Here in case of V-Catchment exercise we will try to keep the decomposition evenly distributed. There are several options available with RunTRIANGLE module. Area constraint will come in handy for this purpose. You may use the same angle constraint of 20 degrees along with an **area** constraint of 8000 m^2 for the purpose discussed in the last paragraph [Figures 10.4 a through c].

After we have done desired editing (if required – generally when excess number of triangular mesh are generated because of any artifact) we will have to repeat the steps described in Section 9.5 and onwards.

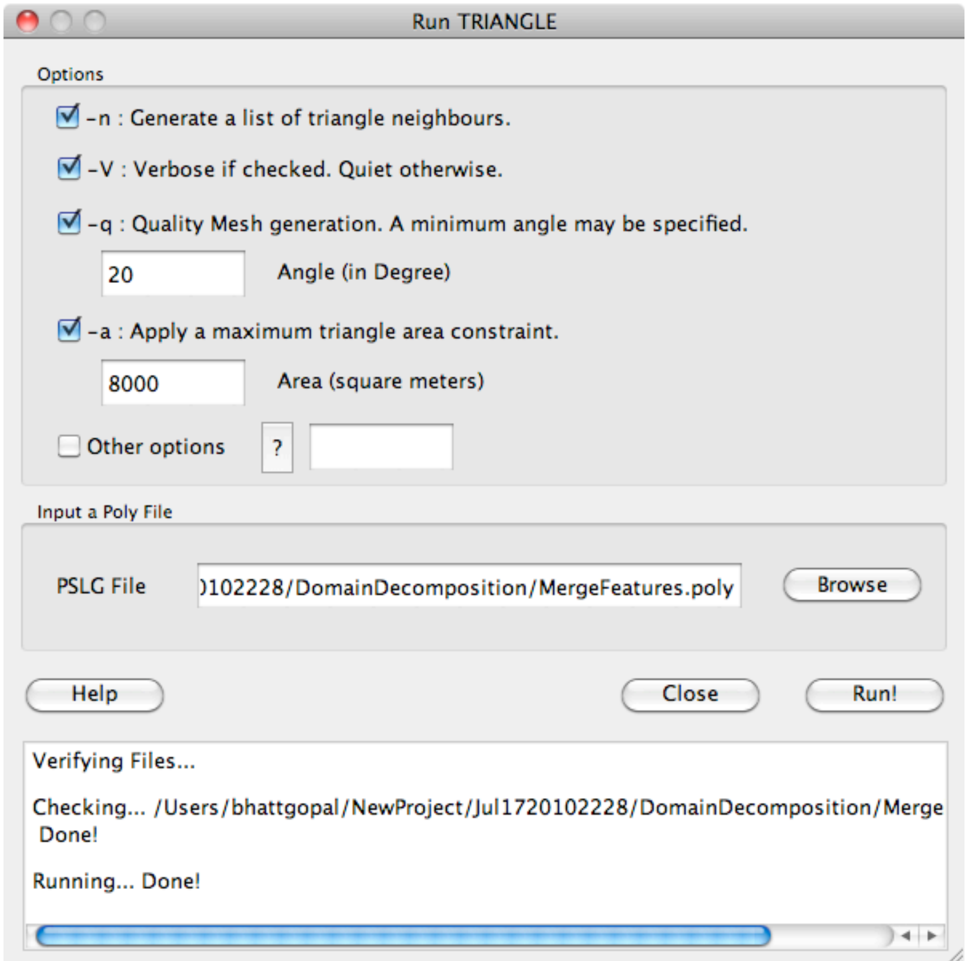


Figure 10.4a: Re-run of TRIANGLE with maximum area option

```
  Angle histogram:
      0 -  10 degrees:        0    |     90 - 100 degrees:       27
     10 -  20 degrees:        0    |    100 - 110 degrees:       12
     20 -  30 degrees:       15    |    110 - 120 degrees:        6
     30 -  40 degrees:       81    |    120 - 130 degrees:        0
     40 -  50 degrees:      173    |    130 - 140 degrees:        1
     50 -  60 degrees:      219    |    140 - 150 degrees:        0
     60 -  70 degrees:      217    |    150 - 160 degrees:        0
     70 -  80 degrees:      113    |    160 - 170 degrees:        0
     80 -  90 degrees:       60    |    170 - 180 degrees:        0

Memory allocation statistics:

  Maximum number of vertices: 179
  Maximum number of triangles: 308
  Maximum number of subsegments: 56
  Maximum number of encroached subsegments: 1
  Maximum number of bad triangles: 488
  Maximum number of stacked triangle flips: 4
  Approximate heap memory use (bytes): 32540
```

Figure 10.4b: Statistics from Delaunay triangulation

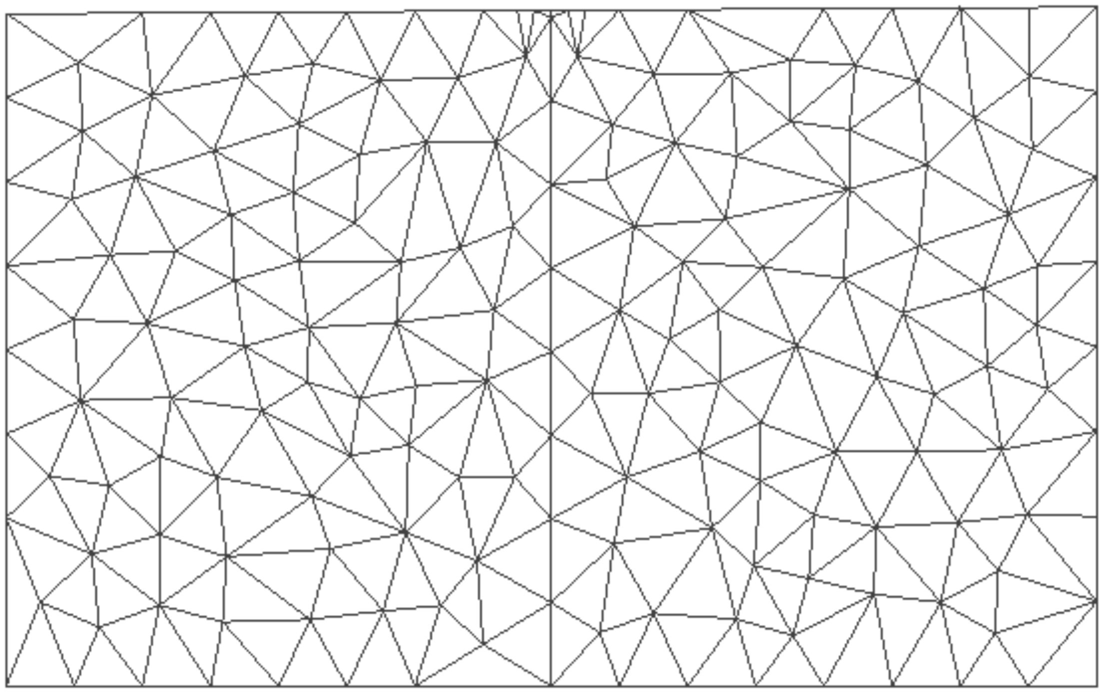


Figure 10.4c: Mesh generated for the v-catchment watershed

<u>Tip</u>: If the mesh generated is not as desired -- generally more number of triangles (related to computational time issue) or cluster of triangle at a location where it is not necessary and does not add any value to the modeling purpose, you will need to make edits to the shapefile in order to improve your decomposition. To successfully do this you may be have to go back to Vector Merge and repeat following steps as noted earlier.

11. DataModel Loader

Data model loader reshapes the structure of GIS maps into a geo-database, which is specifically designed according to the PIHM data model template. The files prepared constitute spatial and relational attributes to the modeling domain. PIHM takes required input in the form of 11 separate data files: (1) mesh file; (2) att file; (3) soil file; (4) lc file; (5) riv file; (6) forc file; (7) ibc file; (8) para file; (9) calib file; (10) geol file; and (11) init file. More details about these files can be found in the PIHM Input File Formats

Tiny: **http://tiny.cc/z65ry**

Original:
http://sourceforge.net/projects/pihmmodel/files/PIHM/FileFormats_PIHM2.0.pdf/

Data model loader will facilitate/guide you prepare these files. For proper management of data it is advised that you create a new folder where you will store output files produced by modules here. In this tutorial a folder named DataModel has been added, and all the new output files will be stored there. Remember all the files produced in DataModel Loader modules will be direct input to the RunPIHM.

The steps are described in the following DataModel Loader dropdown menu:

11.1 Mesh File

Mesh file has all the irregular mesh (TIN) geometry information in it. It contains all the nodes and elements. For nodes, it records its location in space and for elements, it saves index of nodes of which elements comprise of and some topological relations in the form of its neighbor elements.

Select ***Generate MeshFile*** from the ***DataModel Loader*** drop down menu. This should bring Mesh File Generation dialog [Figure 11.1] on the screen.

Processing steps: In the input section of the dialog browse the 5 input files. First three: (1) ele file - *.1.ele, (2) node file - *.1.node, and (3) neighbor file - *.1.neigh are those obtained by the RunTRIANGLE in the section 10.2. Rest two files are the ESRI binary or ascii raster files corresponding to the surface (DEM) and bedrock elevation of the modeling domain respectively. Bed elevation map can be obtained based on geologic formation of the domain. However, in this case we are considering a uniform bed depth of 5 meters and there is a provision do that simply by entering bed depth. In the output section browse a mesh file to which the output will be saved. Execute the module by clicked ***Run***. Text browser should provide with any information regarding error or execution of the module. Click ***Close*** to close the dialog and proceed to next step.

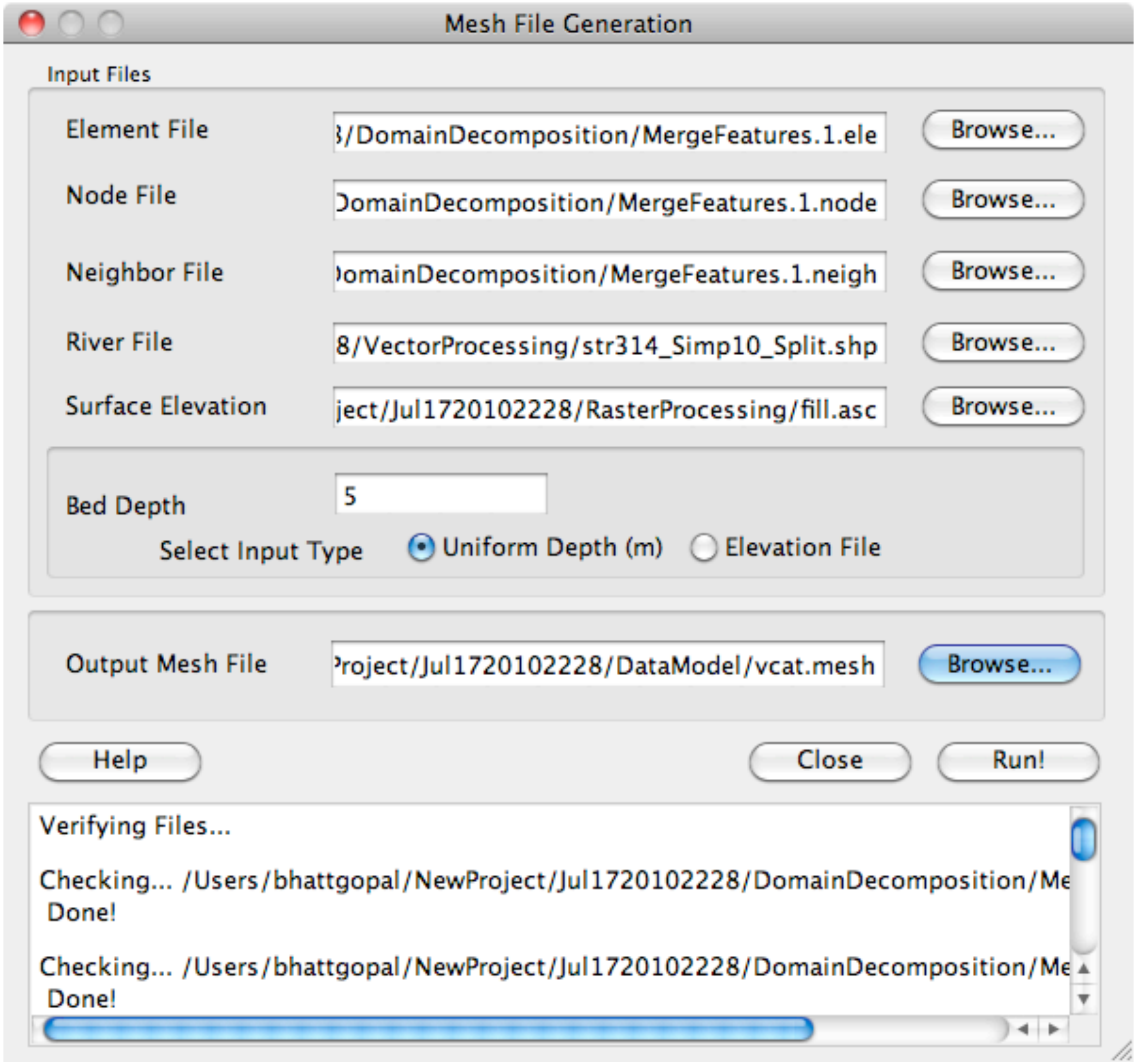
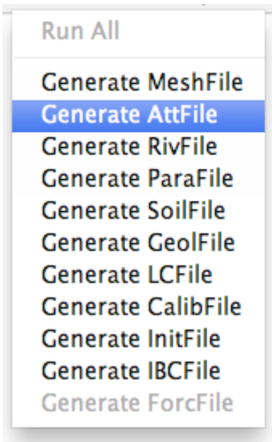


Figure 11.1: Mesh File Generation Dialog

11.2 Att File

An att (attribute) file is a record which stores all the physical parameters class for each triangular element such as soil type, land cover type, several climate types – precipitation, temperature and others. Att file allows efficient data storage by use of relational database.



Select ***Generate AttFile*** from the ***DataModel Loader*** drop down menu. This should bring Att File Generation dialog [Figure 11.2] on the screen.

Processing steps: In the input section of the dialog browse the TIN shape file generated in the section 10.3. All the data classes will be assigned to each element. Data classes have been categorized under three tabs for easy assignment. On Climate tab browse the individual climate/forcing raster classification maps for the domain (see Figure 11.2). Just as a soil classification map climate classification represents the area with different forcing data. Note that, the class number/ID should be integers starting with 1 for each climate classification map. For example, if the modeling domain has two precipitation observations a class map dividing the domain into two precipitation classes should be prepared in advance. In this tutorial we will consider uniform distribution of climate variable. Soil and Land cover types are prepared as optional two different setting – one with uniform properties on both side of hills slope and another with different property types on either side. The data is extracted on the basis of centroid of each triangle. In GridFiles folder under the project directory three (3) grid files are provided:

uniform1 – it is a gridded data file with all the data being 1

uniform0 – it is a gridded data file with all the data being 0

divided – it is a gridded data file with 1 on the left and 2 on the right side

It is advised to make sure that all the grid coverages span well enough to cover the decomposed domain or TINs generated. Also, the spatial reference used for each of the classification map must be the same. Click ***Run*** to begin processing. Any error or processing information is displayed on the text browser at the bottom. After successful completion of the routine press ***Close*** to close the dialog and proceed to the next step.

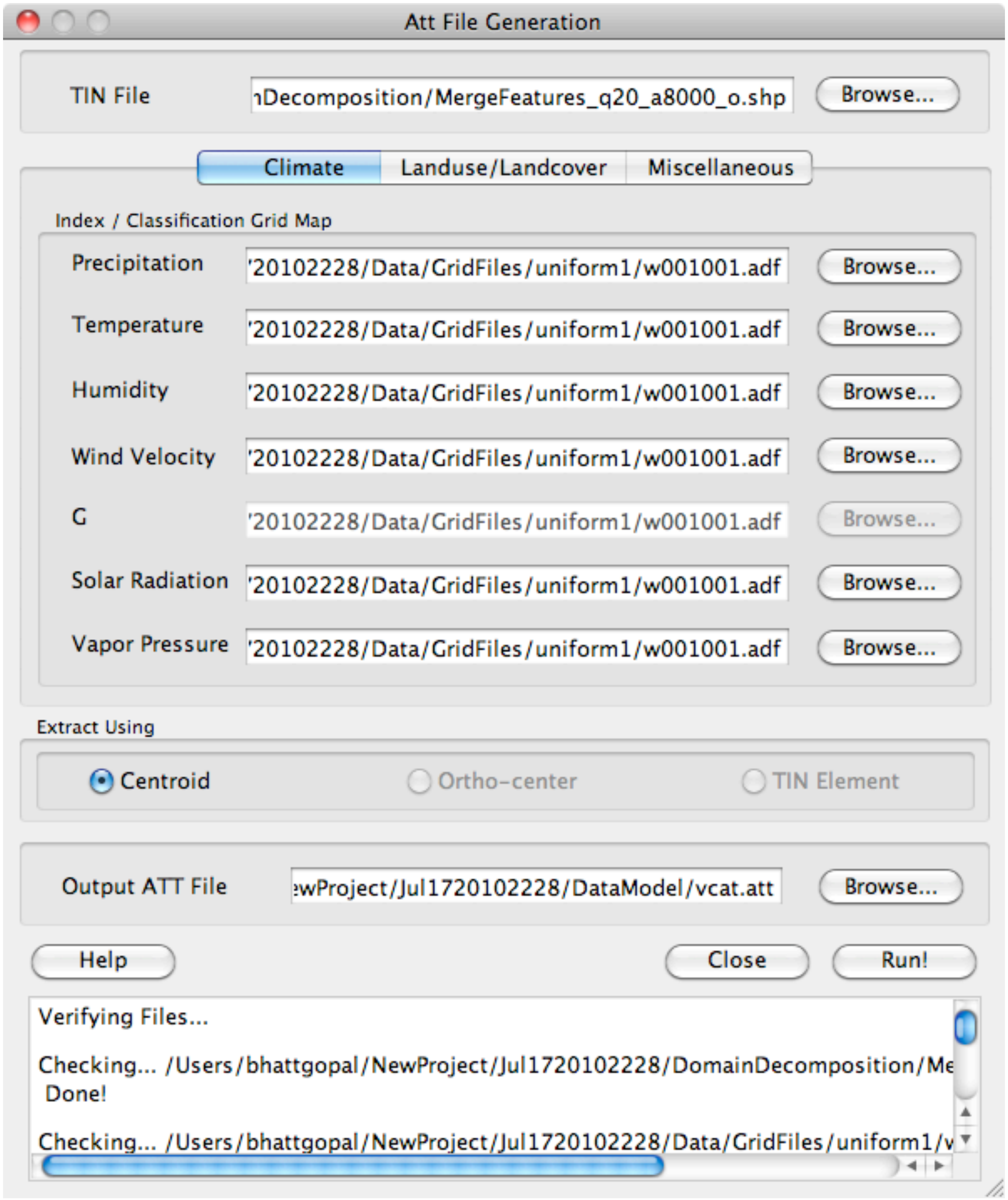


Figure 11.2a: Att File Generation Dialog

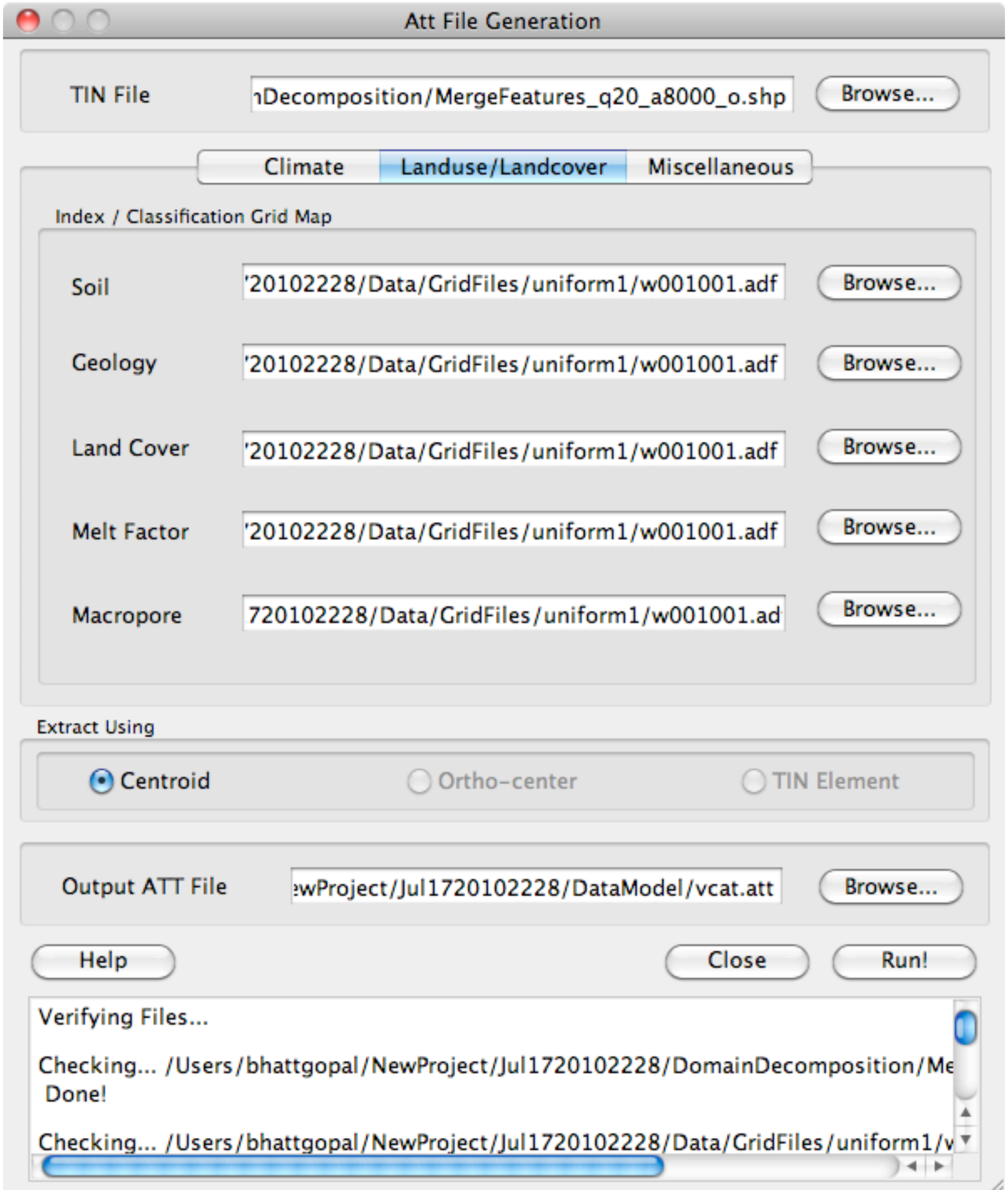


Figure 11.2b: Att File Generation Dialog

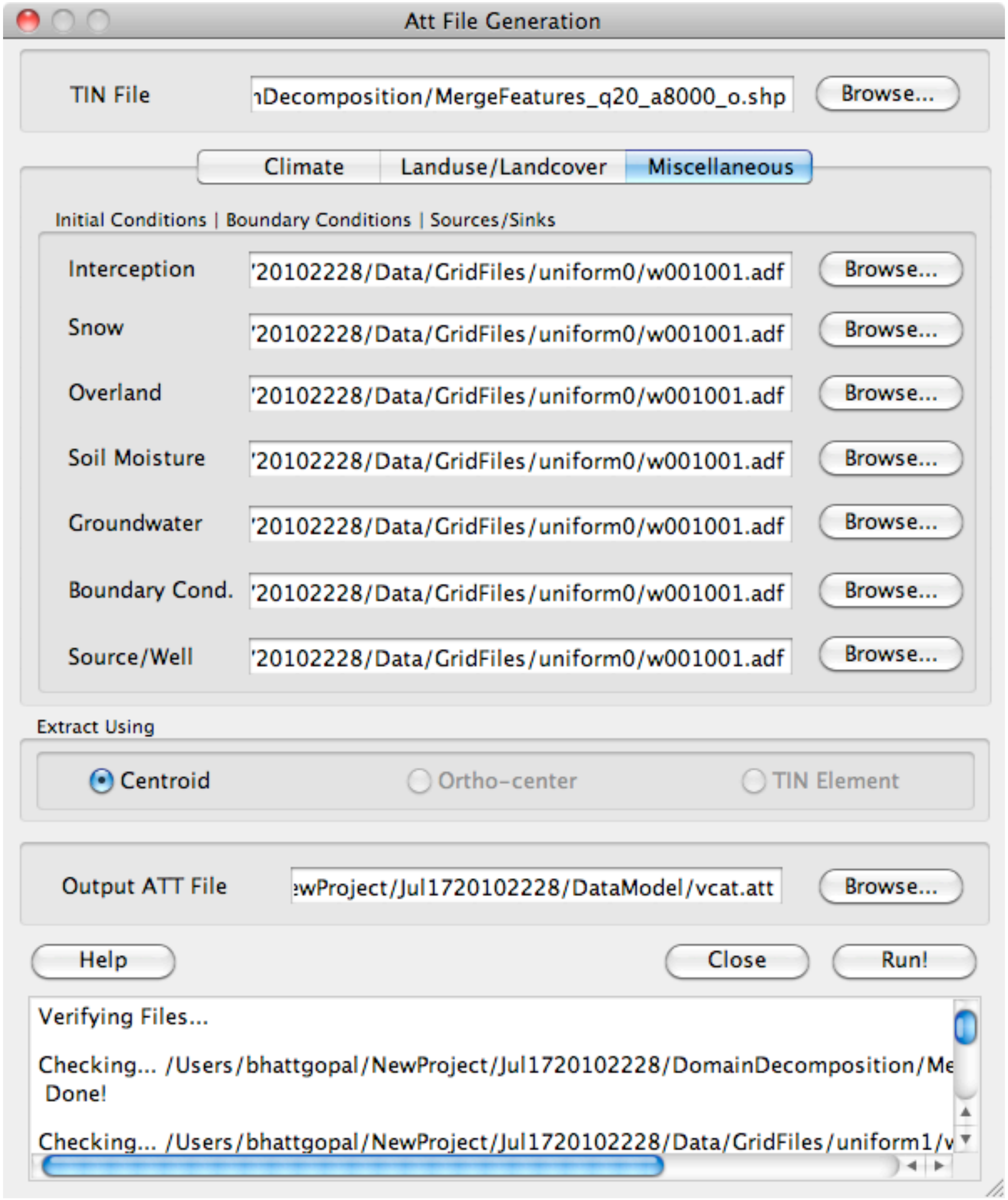
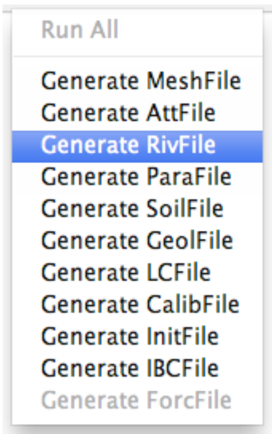


Figure 11.2c: Att File Generation Dialog

11.3 Riv File

Topological information related to river segments (such as Node information; Left and Right Element) is stored in this file. Also different shape and material properties of river segments are provided. Other variables such as Initial and Boundary condition pertaining to river segments are placed at the end of this file as well.



Select ***Generate RivFile*** from the ***DataModel Loader*** drop down menu. This should bring Riv File Generation dialog [Figure 11.3] on the screen.

Processing steps: In the input section of the dialog browse the River shape file (simplified and split) that was used in the Vector Merge [section 9.5]. Note, that if you used the **edited** stream file in domain decomposition module make sure to use the one with edits instead. Also, browse the ele (*.1.ele), node (*.1.node) and neighbor (*.1.neigh) files generated by the RunTRIANGLE in the section 10.2. In the output section of the dialog browse a (*.riv) file, where the output will be saved. Click ***Run*** to begin processing. Any error or processing information will be displayed in the text browser. Please note that the *.riv file generated by this module is has default values generated for stream properties such as shape and material. User needs to edit shape and material information at the end of the file with available information. It is advised to refer the PIHM input file format for more details on the file format. Click ***Close*** to close the dialog and proceed to next step.

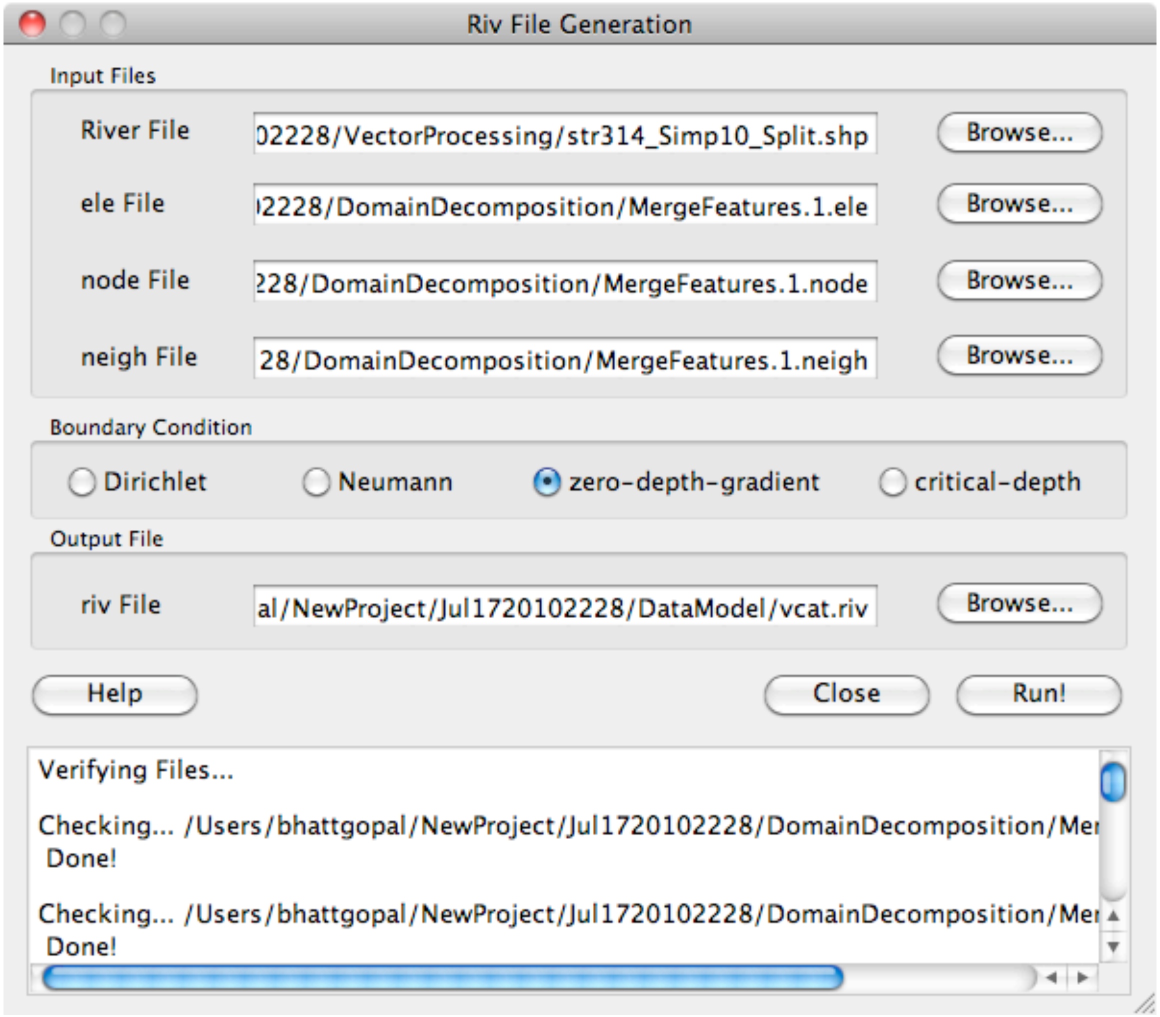
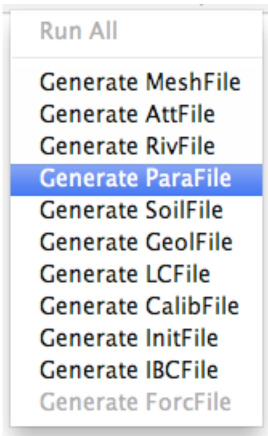


Figure 11.3: Riv File Generation Dialog

11.4 Para File

Para file provides all the control data to the model. It contains solver options; model modes; also parameters that govern model error.



Select ***Generate ParaFile*** from the ***DataModel Loader*** drop down menu. This should bring Parameter File Generation dialog [Figure 11.4] on the screen.

Processing steps: For detailed information on different variables used on this module refer to the PIHM Input File Format document. Choose the several model and control parameter from the list provided on the dialog. In most of the situation user may leave the default values provided on the dialog except length of simulation and output variables to print and interval at which user wants predictions. In the output section of the dialog browse output file name where the para file will be saved. Click ***Run*** to generate the output file with the specified options. Any error or processing information is displayed at the text browser. Click ***Close*** to close the dialog.

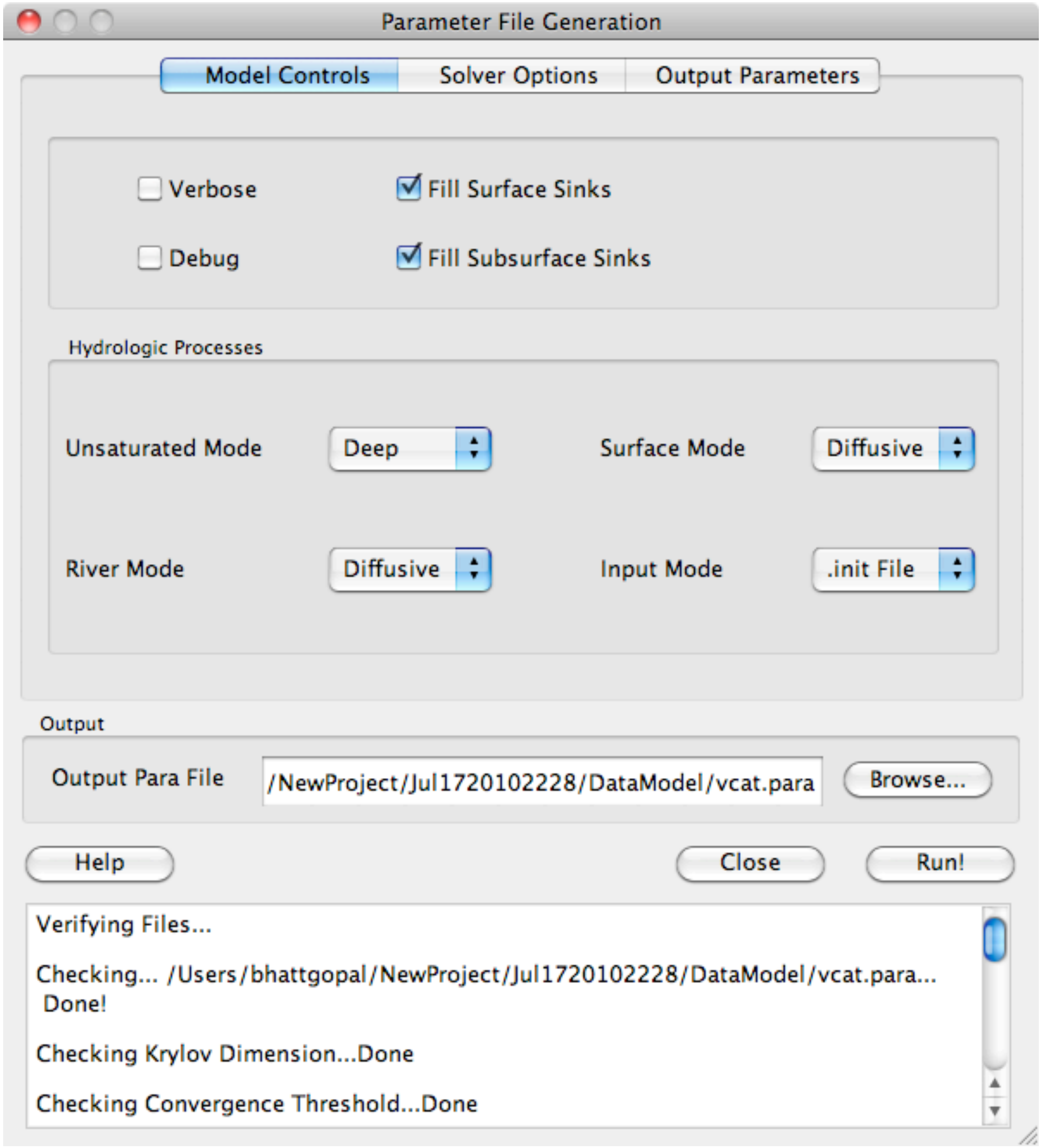


Figure 11.4a: Para File Generation Dialog

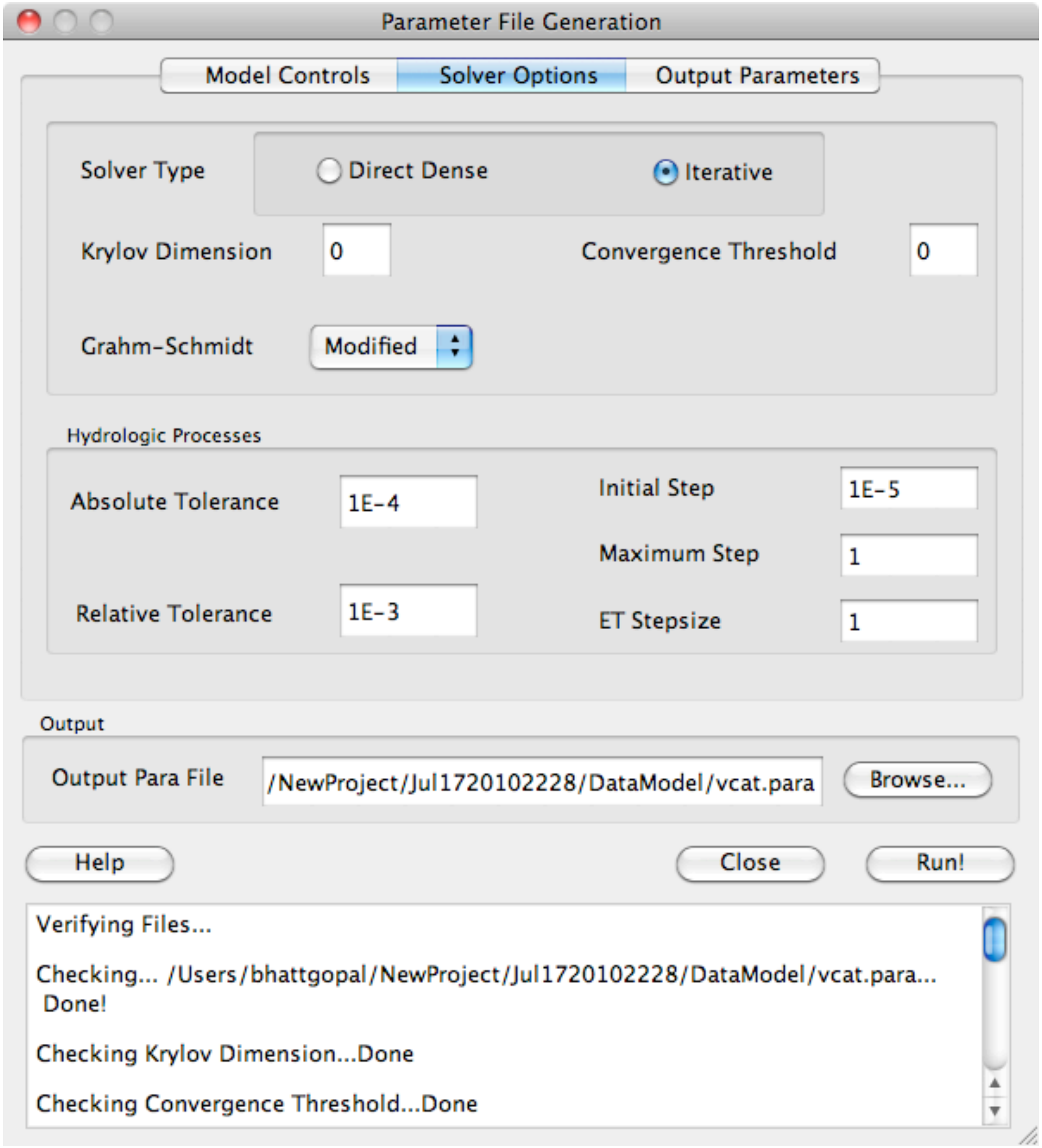


Figure 11.4b: Para File Generation Dialog

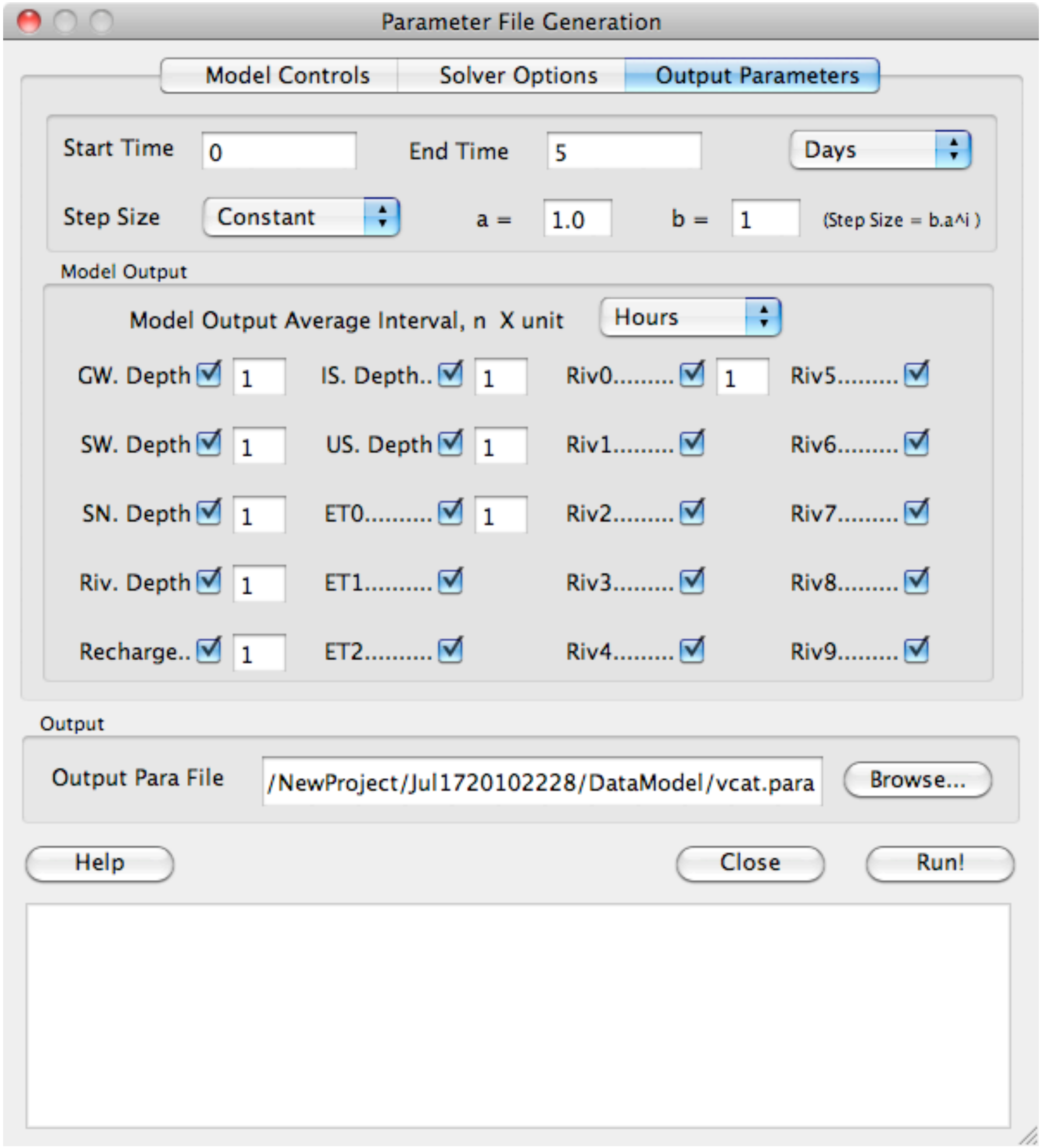
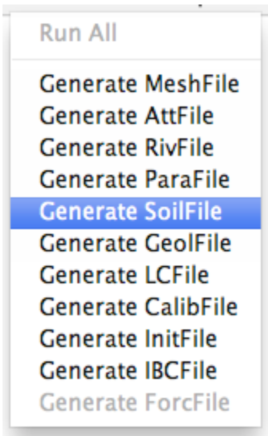


Figure 11.4c: Para File Generation Dialog

11.5 Soil File

Soil module uses pedotransfer functions to estimate soil hydraulic parameters from soil texture data. Soil hydraulic parameters required by the PIHM are: vertical soil hydraulic conductivity, porosity, residual porosity, macropore depth, macropore vertical hydraulic conductivity, Ven-

Genechten alpha and beta parameters. Data needed for soil textural properties are: (1) Silt percentage, (2) Clay percentage, (3) Organic matter (optional) and (4) Bulk density (optional). If working with SSURGO dataset – it is advised that you average all layer values to get above properties.



Select ***Generate SoilFile*** from the ***DataModel Loader*** drop down menu. This should bring Soil File Generation dialog [Figure 11.5] on the screen.

Soil module takes soil texture file as an input. User need to prepare mean representative soil texture property file (*.txt file). The format used to organize input file is as follows:

One line description about the soil texture file				
MUKEY 1	SILT %	CLAY %	ORGANIC MATTER	BULK DENSITY
MUKEY 2	SILT %	CLAY %	ORGANIC MATTER	BULK DENSITY
…	…	…	…	…
…	…	…	…	…
MUKEY n	SILT %	CLAY %	ORGANIC MATTER	BULK DENSITY

<u>Processing steps:</u> In the input section of the dialog browse the prepared Soil Texture file (.txt extension). For this modeling exercise soil texture file has already been prepared and you should be able to find them in folder called Misc. However, for the purpose of this tutorial you will need to choose the right input file based on what you choose for soil coverage in ATT file module (I am using uniform.txt as input since I considered uniform soil coverage for the whole watershed). In the output section of the dialog browse a (*.soil) file, where the output will be saved. Click ***Run*** to begin processing. Any error or processing information will be displayed in the text browser. It is advised to refer the PIHM input file format for more details on the file format. Click ***Close*** to close the dialog and proceed to next step.

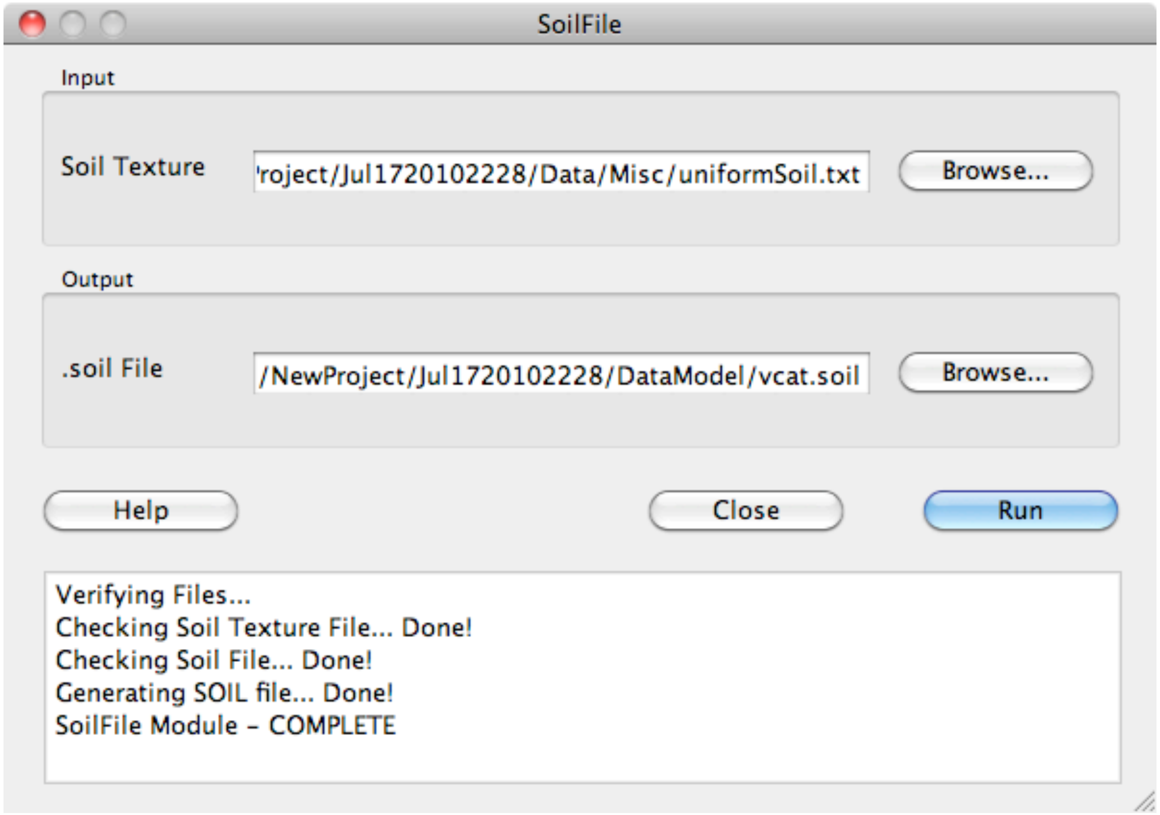
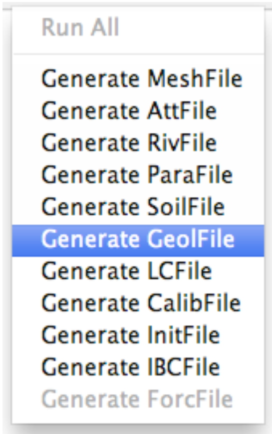


Figure 11.5: Soil File Generation dialog

11.6 Geol File

If you don't have any form of measured data regarding hydraulic properties of the geology you can use Geol module to generate one. Please refer to the PIHM File Format document for details regarding the parameters needed. Similar to Soil module, Geol module also uses pedotransfer functions to estimate aquifer hydraulic properties from texture data. Data needed for soil textural properties are: (1) Silt percentage, (2) Clay percentage, (3) Organic matter (optional) and (4) Bulk density (optional). If you are using SSURGO dataset to obtain above parameters it is desirable that you use only available bottom layer's texture information.



Select ***Generate GeolFile*** from the ***DataModel Loader*** drop down menu. This should bring Soil File Generation dialog [Figure 11.6] on the screen.

Input File format is same as the one used for Generate SoilFile module. But, as mentioned earlier use only bottom layer's texture data instead of average. File format is as follows:

One line description about the geol texture file				
MUKEY 1	SILT %	CLAY %	ORGANIC MATTER	BULK DENSITY
MUKEY 2	SILT %	CLAY %	ORGANIC MATTER	BULK DENSITY
…	…	…	…	…
…	…	…	…	…
MUKEY n	SILT %	CLAY %	ORGANIC MATTER	BULK DENSITY

Processing steps: In the input section of the dialog browse the prepared Geol Texture file (.txt extension). For this modeling exercise geol texture file has already been prepared and you should be able to find them in folder called Misc. You will still need to select the correct file as you did in the Generate SoilFile module. In the output section of the dialog browse a (*.geol) file, where the output will be saved. Click ***Run*** to begin processing. Any error or processing information will be displayed in the text browser. It is advised to refer the PIHM input file format for more details on the file format. Click ***Close*** to close the dialog and proceed to next step.

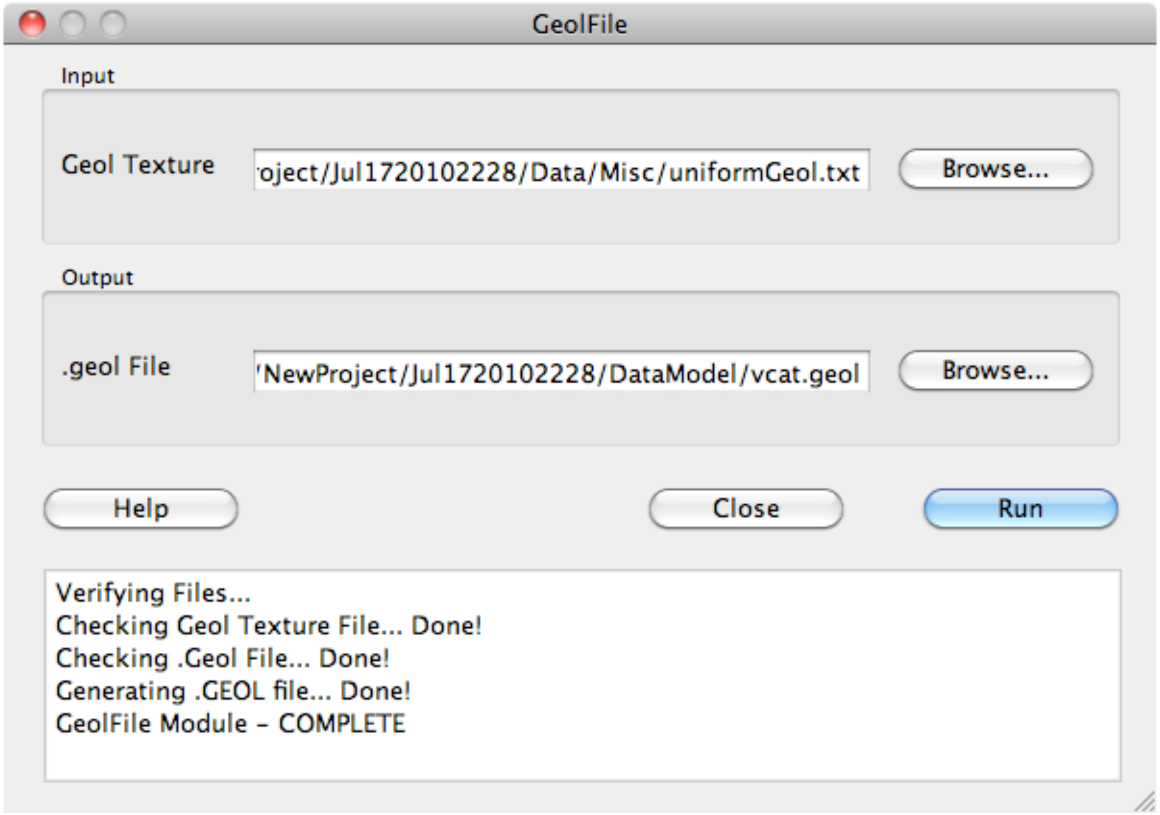
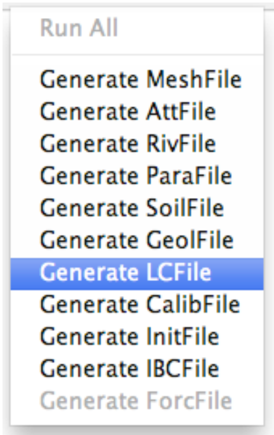


Figure 11.6: GeolFile Generate Module

11.7 LC File

This module has been designed keeping NLCD 2001 classification at its core. Generate LCFile module prepares vegetation parameters for any class present in NLCD 2001 classification. Please refer to following link for more description about the classification:

http://www.epa.gov/mrlc/definitions.html#2001



Select ***Generate LCFile*** from the ***DataModel Loader*** drop down menu. This should bring LC File Generation dialog [Figure 11.7] on the screen.

Input File format for this modules simply comprise of different NLCD land cover type present in the modeling domain. You should arrange all the class numbers in ascending order as a column in .txt file format and follow the same for re-arranging the raster file used in att module. File format is as follows:

Class_1
…
…
Class_n

<u>Processing steps:</u> In the input section of the dialog browse the prepared land cover class files (.txt extension). For this modeling exercise this file has already been prepared and you should be able to find them in folder called Misc. You will still need to select the correct file as you did in the Generate SoilFile/GeolFile module depending on if you choose uniform or divided land cover classification in *.att file. In the output section of the dialog browse a (*.lc) file, where the output will be saved. Click ***Run*** to begin processing. Any error or processing information will be displayed in the text browser. It is advised to refer the PIHM input file format for more details on the file format. Click ***Close*** to close the dialog and proceed to next step.

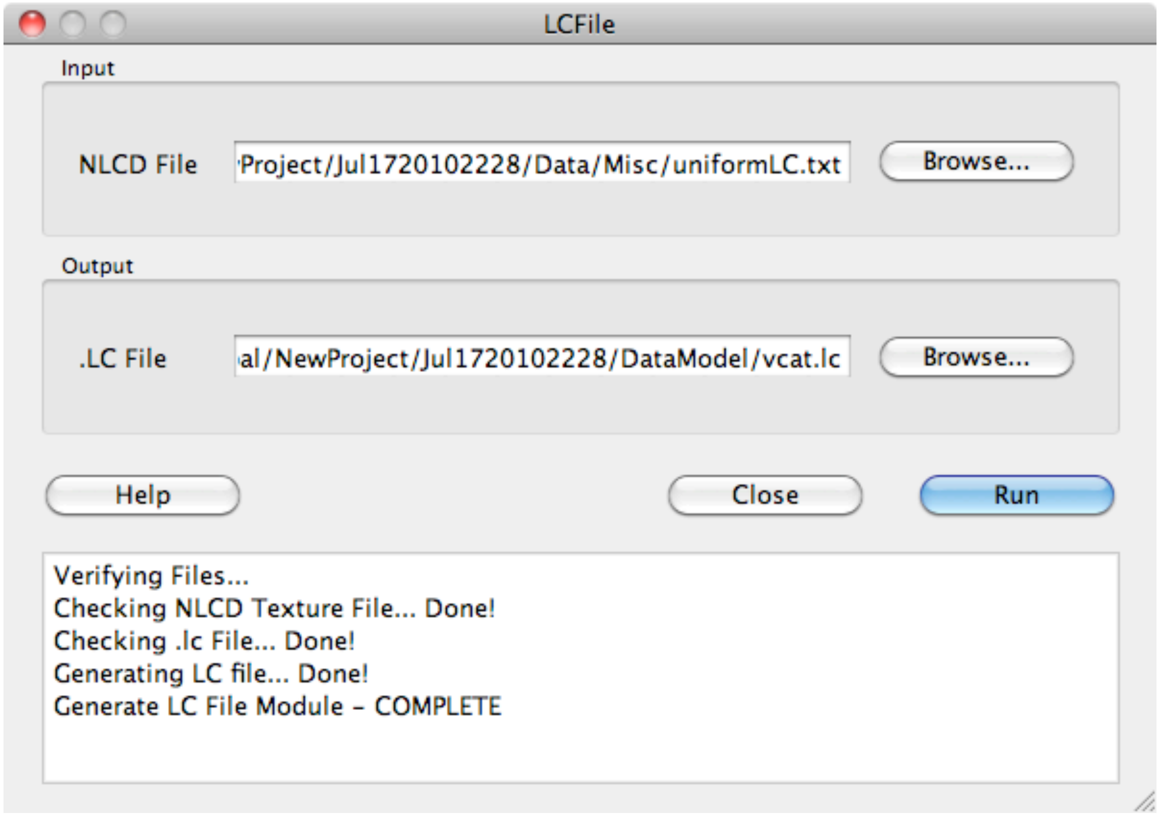
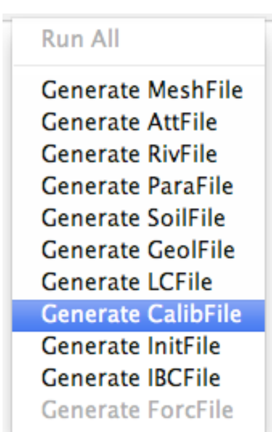


Figure 11.7: Generate LCFile modules

11.8 Calib File

As the name suggests calib file is used as a part of calibration. Several physical parameters have been identified for the purpose of calibration. The values you choose in the module for specific parameter is a multiplier hence it globally and uniformly nudges the a-priori parameter value across the whole modeling domain.



Select ***Generate CalibFile*** from the ***DataModel Loader*** drop down menu. This should bring Calib File Generation dialog [Figure 11.8] on the screen.

<u>Processing steps:</u> On the dialog you will notice a default value has been provided for most of the parameters. A value of 1.0 implies that it would not change the parameter (since it is a multiplier to any parameter). You can accept these default values for the purpose of first run and come back later if you decide to change any of them.

In the output section of the dialog browse a (*.calib) file, where the output will be saved. Click ***Run*** to begin processing. Any error or processing information will be displayed as a pop up window. It is advised to refer the PIHM input file format for more details on the file format and understand several variable names used. Click ***Close*** to close the dialog and proceed to next step.

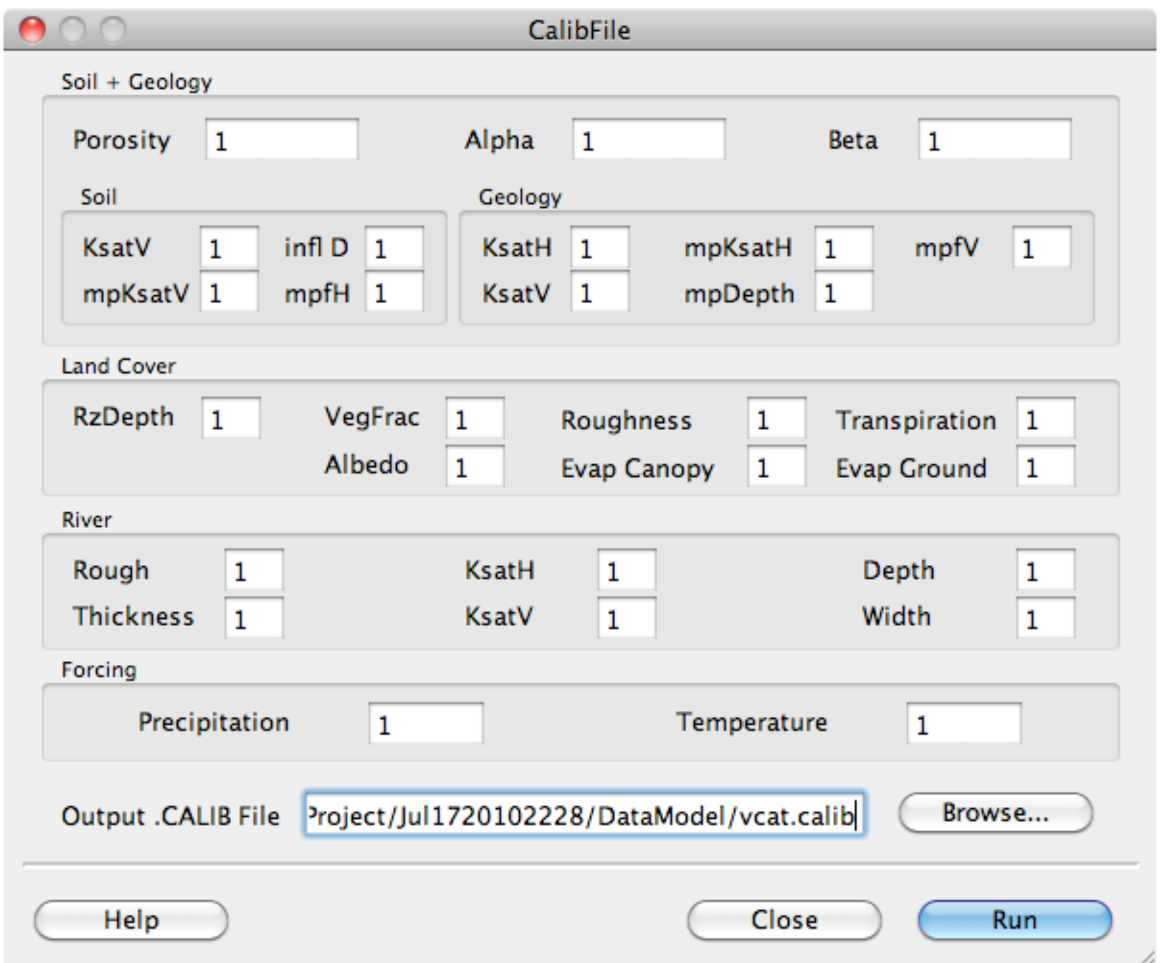


Figure 11.8: Generate CalibFile module

11.9 Init File

Init (initial condition) file has initial state for all the hydrologic variables. This module facilitates a uniform initialization of individual state variable for the domain. However, this file is an optional file. If you have some observation data with which you would like to initialize the model you should use the interpolated raster data in the *.att file in stead. Remember you will have to adjust initialization mode in the .para file to switch between these two options. If you don't have any information you could proceed with this module with a uniform initial guess.

Select ***Generate InitFile*** from the ***DataModel Loader*** drop down menu. This should bring Init File Generation dialog [Figure 11.9] on the screen.

<u>Processing steps:</u> In the input section of the dialog browse the previously generated .mesh and .riv files. It is adviced that you adjust the initialization values for elements according the bed depth you are considering and for river elements according to the specified channel depth. For this modeling exercise since we are considering 5 m thinkness of the domain we can initialize the ground water with 4.5 meters. Rest can be left as default. In the output section of the dialog browse a (*.init) file, where the output will be saved. Click ***Run*** to begin processing. Any error or processing information will be displayed in the text browser. Click ***Close*** to close the dialog and proceed to next step.

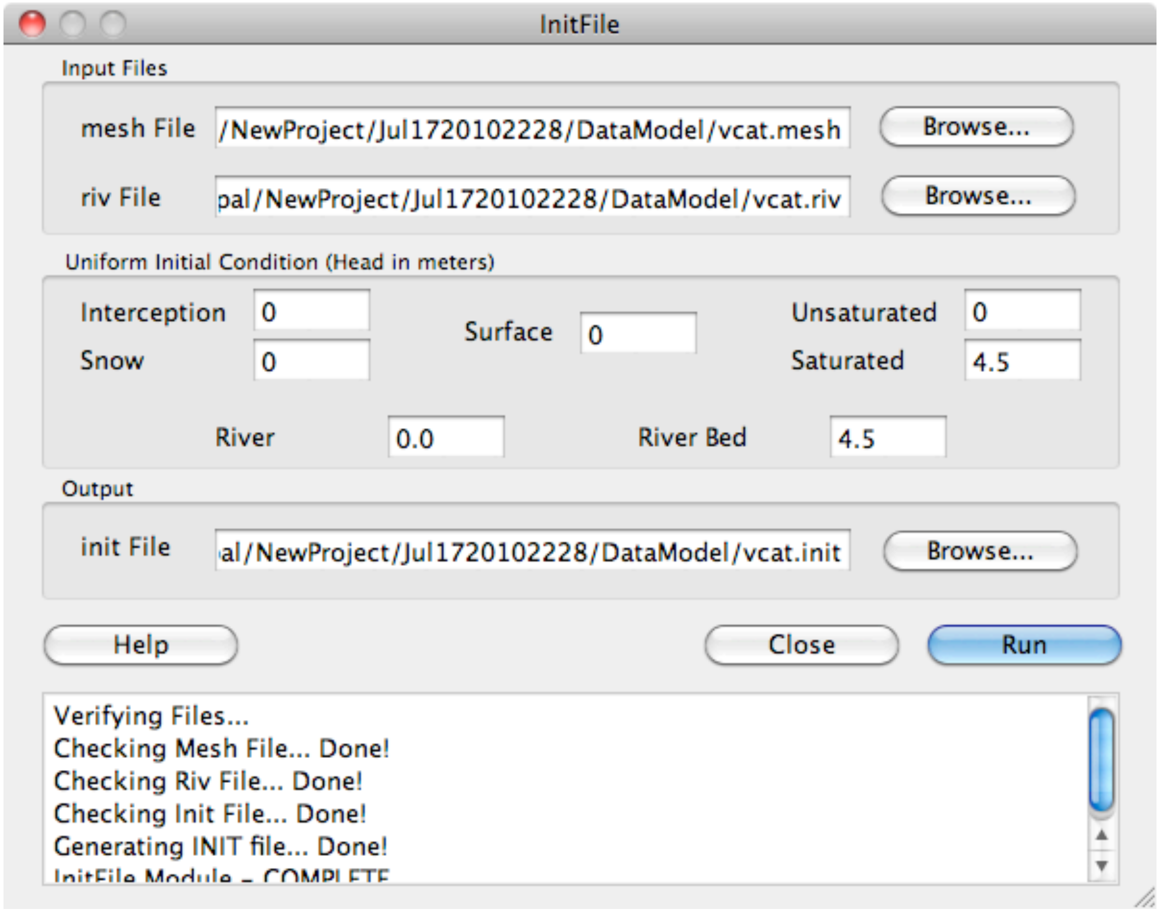


Figure 11.9: Generate InitFile module

11.10 IBC File

This file contains any boundary condition information if present for the modeling domain. PIHM has provision to specify both Dirichlet and Newman boundary conditions.

Select ***Generate IbcFile*** from the ***DataModel Loader*** drop down menu. This should bring IBC File Generation dialog [Figure 11.9] on the screen.

Processing steps: The scope of this module is limited to just creating a dummy file, which is generally the case for a modeling domain. It generates a dummy file, which allows setting up a simulation.

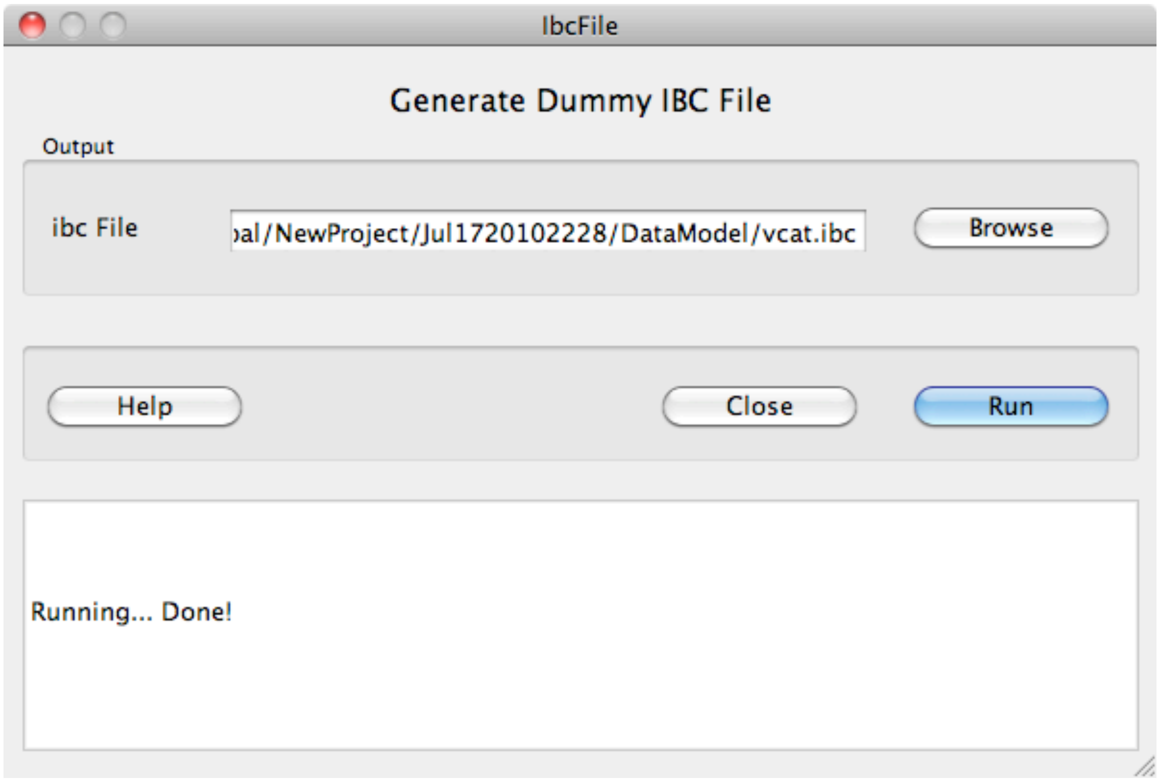


Figure 11.9: Dummy IBC file dialog to generate an ibc file

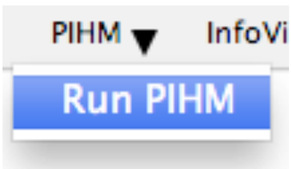
11.11 Forc File

... NOT IMPLEMENTED ...

... Use PIHMServices ...

12. Run PIHM

RunPIHM is the module where PIHM has been integrated to Qgis. For a successful run of the model it is required to have all the input files prepared in accordance with the file format of the PIHM. It should be noted that PIHMgis helps prepare all files needed by RunPIHM module but is contingent to files you input in individual module.



Select ***Run PIHM*** from the ***PIHM*** drop down menu. This should bring PIHM dialog [Figure 12.1] on the screen.

In the input section of the dialog browse the folder where the input files for the model has been stored (It is necessary that you store all the DataModel Loader generated files in one folder). The model assumes that each input file has the same first syllable for the file name followed by "." and file type (for e.g. domain.mesh; domain.att; domain.soil). Input File ID corresponds to that first syllable of the input files (for the "domain" example - input **domain** in the Input File ID textbox). In this tutorial all the files have been stored in "DataModel" folder and the prefix to all the file name is "vcat". Click ***RUN*** to begin running PIHM. A copy of input files along with the simulation results are stored in a new sub-folder created with each run. The interacting hydrologic processes of different timescales introduce stiffness into the system of ordinary differential equation. Therefore, the time a model run will take to complete depends on several factors, which include: number of triangular elements, topography of the domain, climate data, and system state. It will also depend on the period of simulation specified (start and end time indicated in ParaFile Generate module - *.para file). A progress-bar has been provided to keep track of progress of the simulation. After the completion of the model run ***Close*** the dialog.

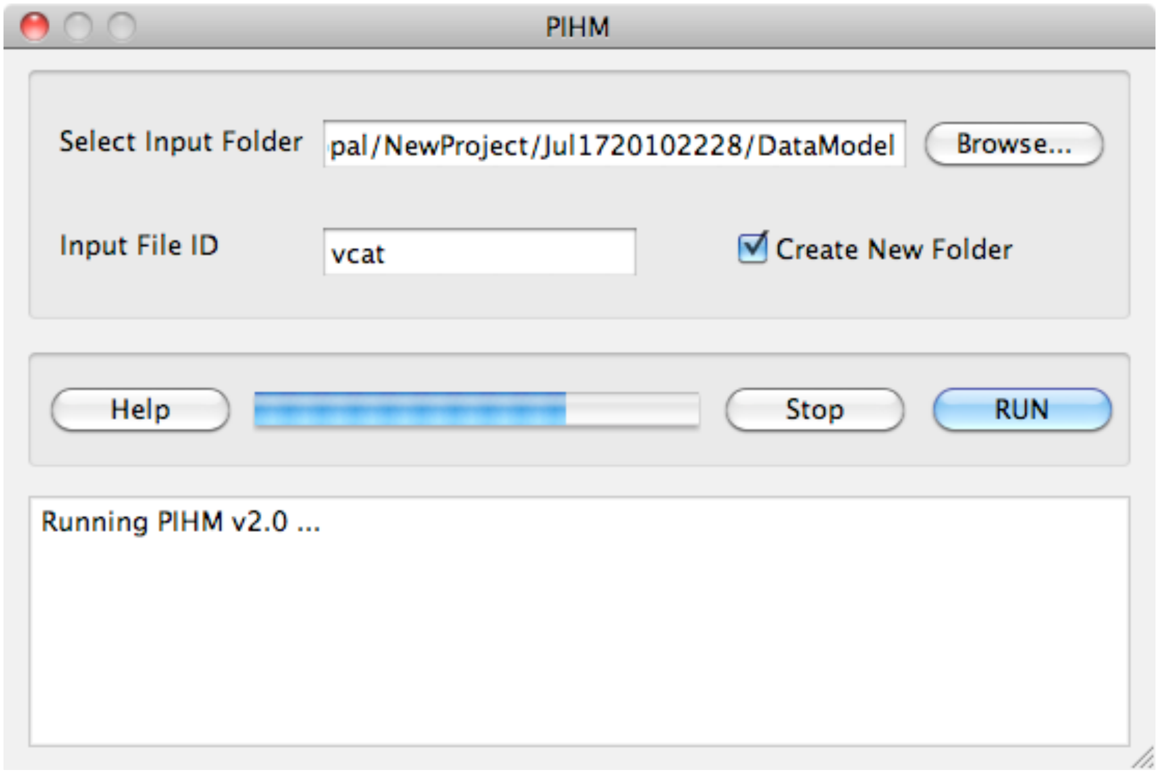
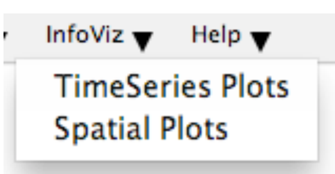


Figure 12.1: Run PIHM Dialog

13. InfoViz

However there could be several ways of visualizing the simulation results and analyze them depending on the modeling purpose. Two basic types of analysis module have been integrated in the current version. They enable analysis in the form of time series and spatial distribution of several state variables corresponding to the modeling domain.

These modules can be accessed from the following Analysis drop down menu:



Before we get into the analysis part it is important to understand output file format. Depending on what you specified as print variables (in .para file) the files would contain either state variable for flux variable for either triangular elements or stream segments.

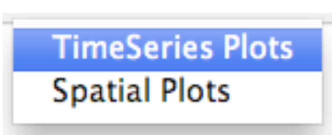
Depending on the print interval you specify for each variable in .para file a mean value for the specified period are printed for each element or stream element in a row. The same carry on in the following rows. Suppose for any variable of interest you specified a print interval value as 'n' time units. The modeling domain has 'E' number of elements and 'S' number of stream segments. Also the simulation period has been specified as from 0 to 'N' x 'n' time units. The output file for stream and element will look like the following:

str(n,1)	str(n,2)	…	str(n,s-1)	str(n,s)
str(2*n,1)	str(2*n,2)	…	str(2*n,s-1)	str(2*n,s)
…	…	…	…	…
str(N*n,1)	str(N*n,2)	…	str(N*n,s-1)	str(N*n,s)

Ele(n,1)	ele(n,2)	…	data(n,e-1)	data(n,e)
data(2*n,1)	data(2*n,2)	…	data(2*n,e-1)	data(2*n,e)
…	…	…	…	…
data(N*n,1)	data(N*n,2)	…	data(N*n,e-1)	data(N*n,e)

13.1 Time Series Plots

Time Series Plots can be used to plot temporal behavior of a state variable or the flux corresponding to either elements or river segments. It provides flexibility to visualize an element or river segment individually or as an average of all the feature elements.



Select ***TimeSeries Plots*** from the ***InfoViz*** drop down menu. This should bring Time Series Plots dialog [Figure 13.1] on the screen.

<u>Processing steps:</u> Select from Element feature or River feature type tab depending on the state variable type you want to plot. ***Select Plot Variable:*** allows visualize time series of individual feature element (***ID***: individual element or stream segment) or a group on feature elements or an ***Average*** of all the feature elements. It is only required to input the ID

value if "Select Feature by" ID has been selected. ***Select Plot Variables:*** provides option to choose from for multiple plot variables of interest. ***Time-Step*** is used to average the model output by the value provided (number that shows up by default is the interval at which simulation outputs were printed. User also needs to browse the simulation output folder and right file will be automatically selected. Model saves the output files in the sub-directory from where input files were fetched in the section 10. Click ***Plot*** to begin plotting. The process might take a while to process the model output file specially if simulation duration is long. After successful processing of the model output file, time series plot is displayed using Qwt widget. The data for the selected option is also saved as a text file which you could plot using your preferred plotting tool.

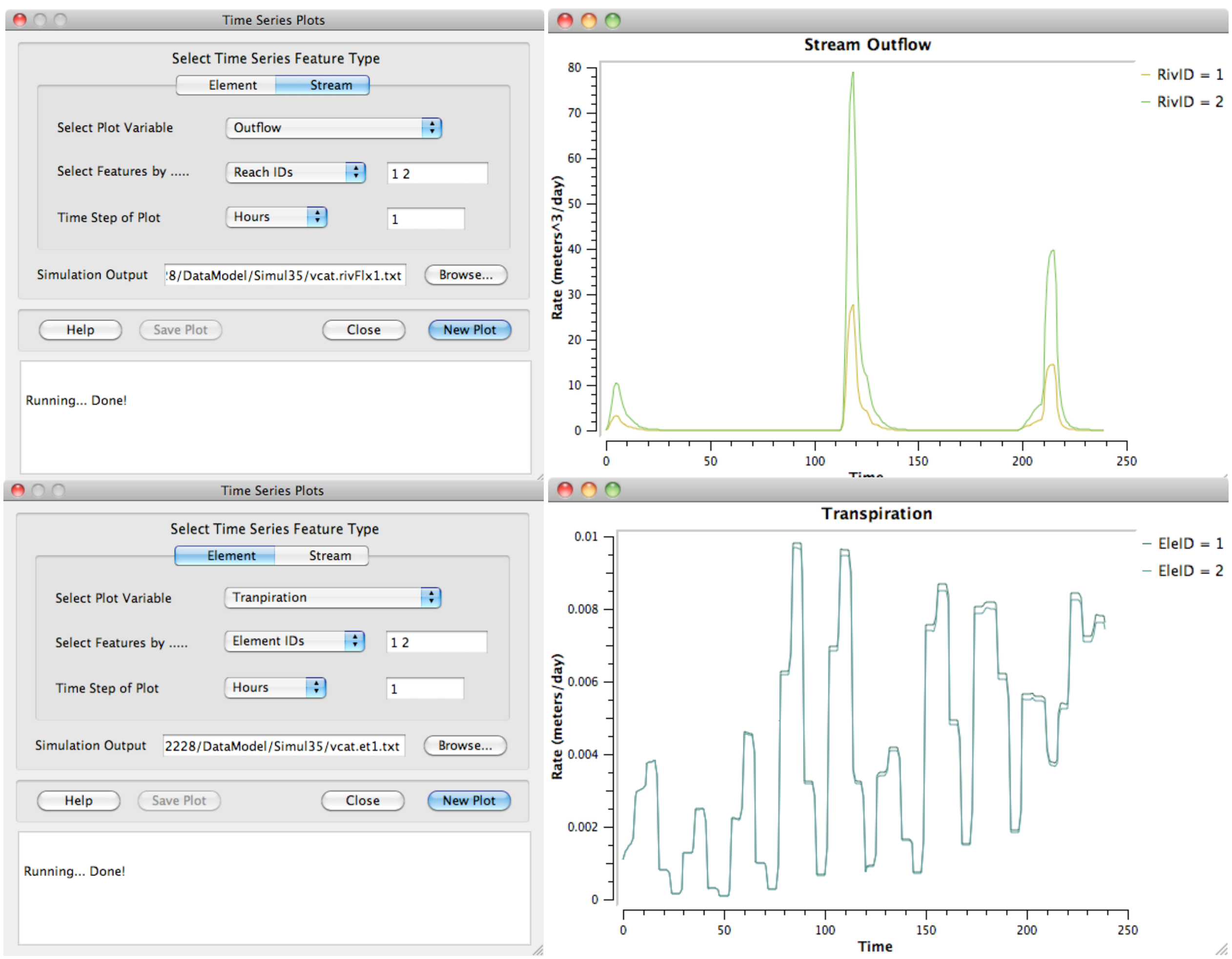
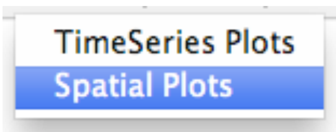


Figure 13.1: Time Series Plot Dialog (a) Discharge through stream (b) transpiration from selected mesh

13.2 Spatial Plots

Spatial Plots can be used to plot spatial behavior of several state or flux variables corresponding to either elements or river segments. It provides flexibility to visualize spatial distribution of any of those variables as a snapshot of time or as an average over a specified period of time.



Select ***Spatial Plots*** from the ***Analysis*** drop down menu. This should bring Spatial Plots dialog [Figure 13.2] on the screen.

Select Element feature or River feature type depending on the plot variable you want to plot. Input ***Start Time*** and ***End Time*** over which it is desired to obtain average spatial plot of the plot variable. ***Select Plot Variable:*** provides option to choose from for the plot variable of interest. TIN shape file or River shape file (the one obtained from RivFile module) also need to be browsed as the spatial map is generated based on them. User also needs to browse the model output folder containing the simulation outputs. Note that, model saves the output files in a new directory (SimulXX) by default for each simulation run. Click ***Generate*** to begin processing. The routine might take a while for processing the model output file. After successful processing of the model output file, a shapefile named "SpatialData.shp" is generated in the InfoViz directory.

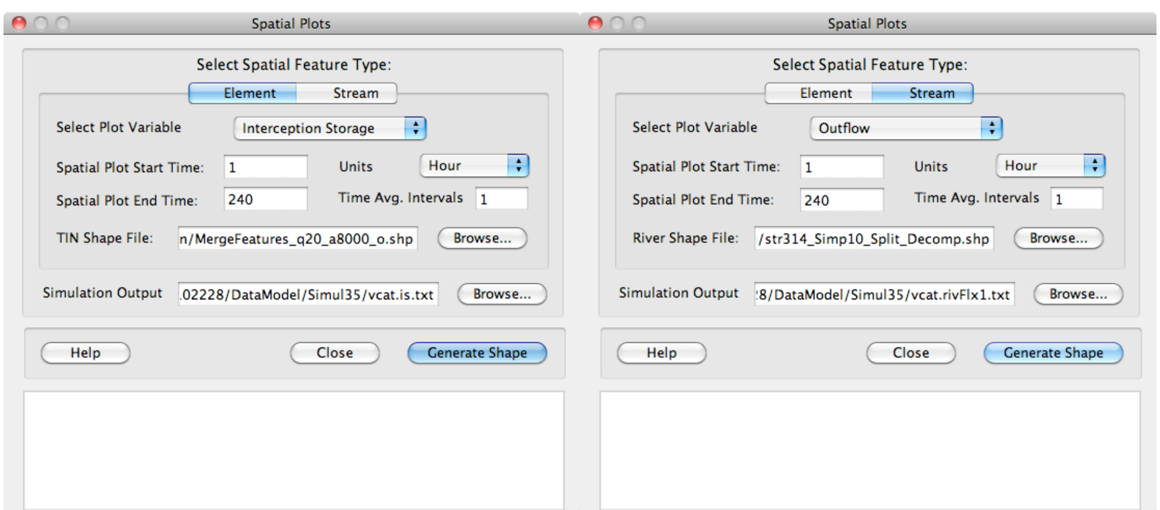


Figure 13.2: Spatial Plot Dialog

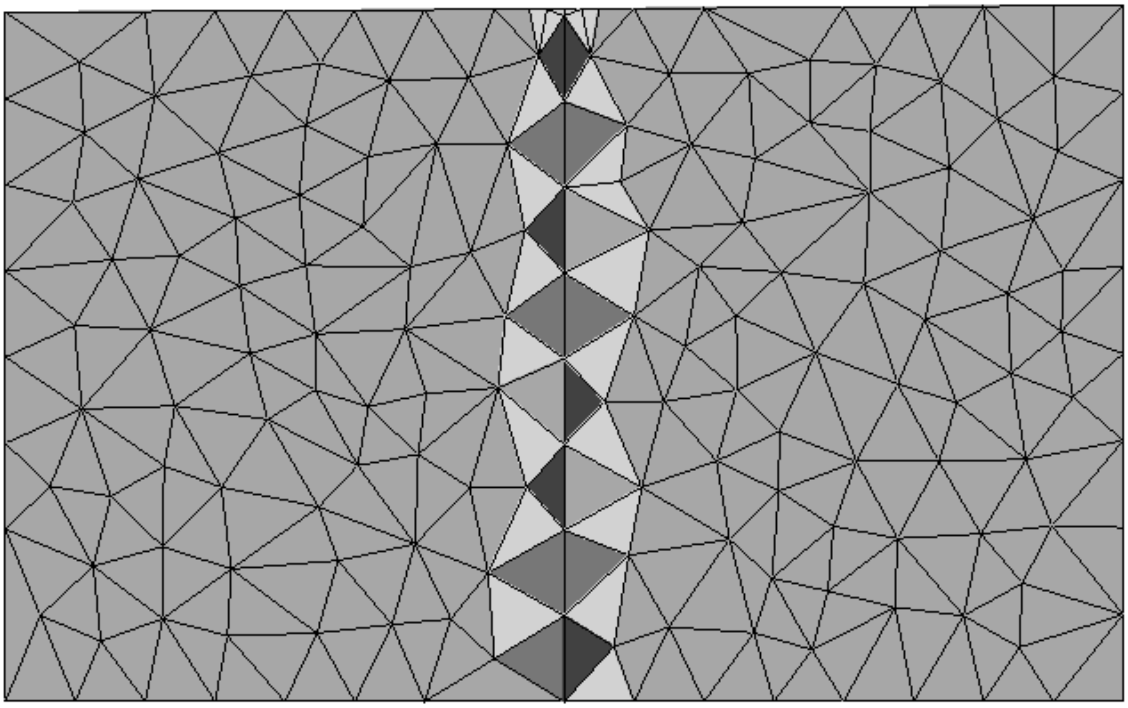


Figure 13.3: A simple demonstration of a spatial plot of recharge

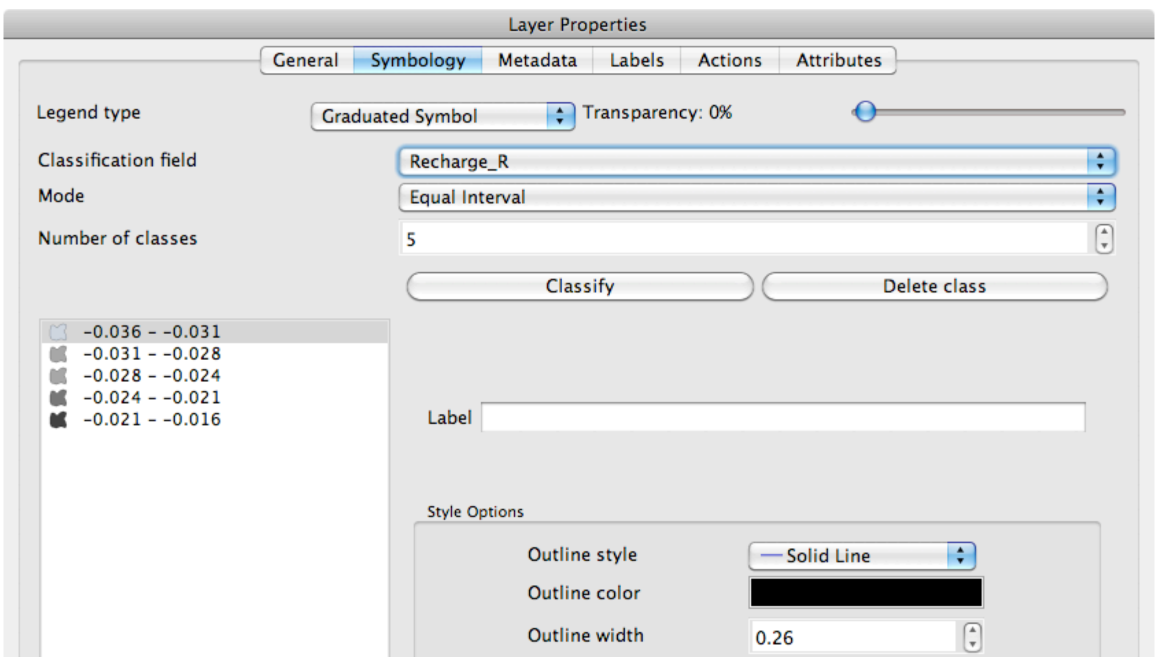


Figure 13.4: Classification and symbology preference for any shape file [Properties >> Symbology]