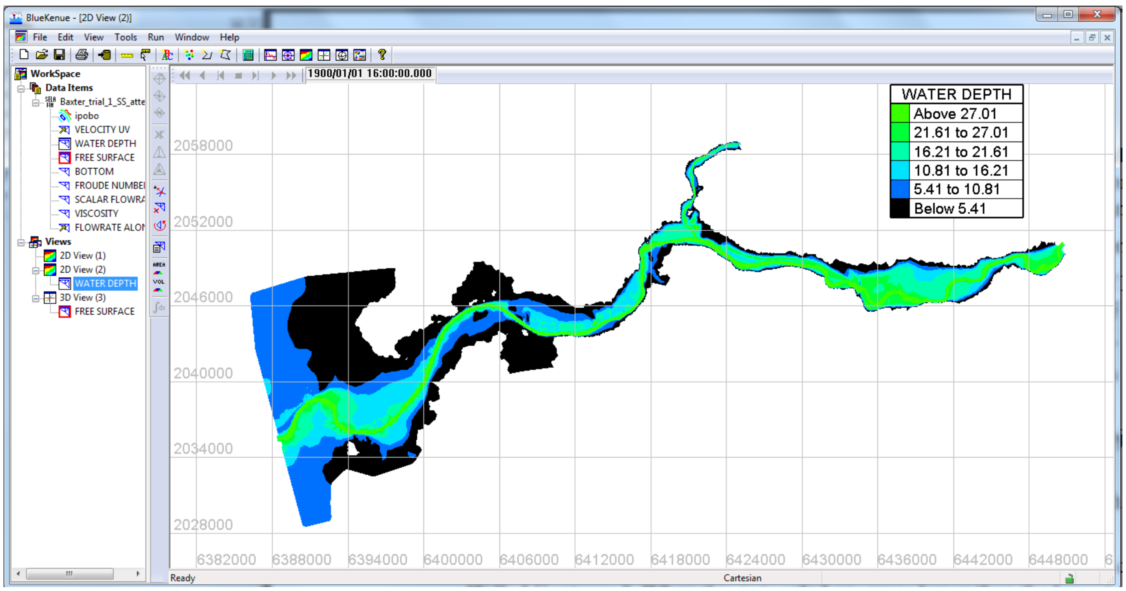
Tutorial on the use of TELEMAC-2D Hydrodynamics model and Pre-/Post-processing with BlueKenue for flood-inundation mapping in Unsteady Flow Conditions

Christopher Gifford-Miears[*], Arturo S. Leon[†]

Oregon State University
School of Civil and Construction Engineering
CE 540 **Unsteady flows in Rivers and Pipe Networks** , Fall 2013



Instructor: Arturo Leon, Ph.D., P.E.
E-mail: arturo.leon@oregonstate.edu
Office phone: (541) 737-2606
Mail address: 220 Owen Hall
School of Civil and Construction Engineering
Oregon State University
Corvallis, OR 97331-3212
Research Web page: `http://web.engr.oregonstate.edu/~leon`

Geometric data accessed and adapted from U.S. Army Corps. Hydrologic Engineering Center - HEC-GeoRAS Tutorial

Objectives of this Tutorial: The objective of this tutorial is to give a brief introduction on the use of TELEMAC-2D and BlueKenue for setting up and analyzing flood inundation in unsteady flow conditions.

KeyWords: Unsteady hydraulic routing; Two-dimensional modeling; Flood-inundation; BlueKenue; Finite-element mesh generation

[*]Graduate Research Assistant, School of Civil and Construction Engineering, Oregon State University, 233 Owen Hall, Corvallis, OR 97331, USA. E-mail: cgiffordmiears@gmail.com (Corresponding author)

[†]Assistant Professor, School of Civil and Construction Engineering, Oregon State University, 220 Owen Hall, Corvallis, OR 97331-3212, USA. E-mail: arturo.leon@oregonstate.edu

Contents

List of Figures

List of Tables

1 Pre-requisites

1.1 Computer Requirements

1. ArcGIS 10.1
2. OpenTELEMAC CFD suite (via auto- or self-install)
   Successful installation/compilation requires:
   (a) Python 2.7.3
   (b) gfortran compiler
   (c) TortoiseSVN client
3. BlueKenue pre-/post-processing software
4. FUDAA pre-processing software
5. 64-bit operating system (ideal, but not required if self-installing)

You can download the Open TELEMAC automatic installer and FUDAA pre-processor from the TELEMAC-MASCARET website at:
`http://www.opentelemac.org/`

The geometric dataset can be obtained from the course webpage or from the US Army Corps of Engineers Hydrologic Engineering Center website at:
`http://www.hec.usace.army.mil/software/hec-georas/downloads.aspx`

BlueKenue Pre-/Post-processing program is available from the "Canadian Hydraulics Centre of the National Research Council Canada" website at:
`http://www.nrc-cnrc.gc.ca/eng/solutions/advisory/blue_kenue_index.html`

If performing a self-installation of the TELEMAC-MASCARET suite, it is recommended to follow the **Python installation guidelines** along with a summary of installation/compilation tips for a gfortran/Python/Windows specific build (**Appendix B**).

1.2 Data Requirements

The data required for this tutorial is available at:
`http://web.engr.oregonstate.edu/~leon/Teaching_transients.html`
Download both zip files on your local drive (e.g. `C:\TELEMAC_tutorial`), and unzip their contents. The TELEMAC and GeoRAS Data folder contain three sub-folders; one TIN dataset, a RASModel and one aerial image (as raster grid) as shown in Figure 1.

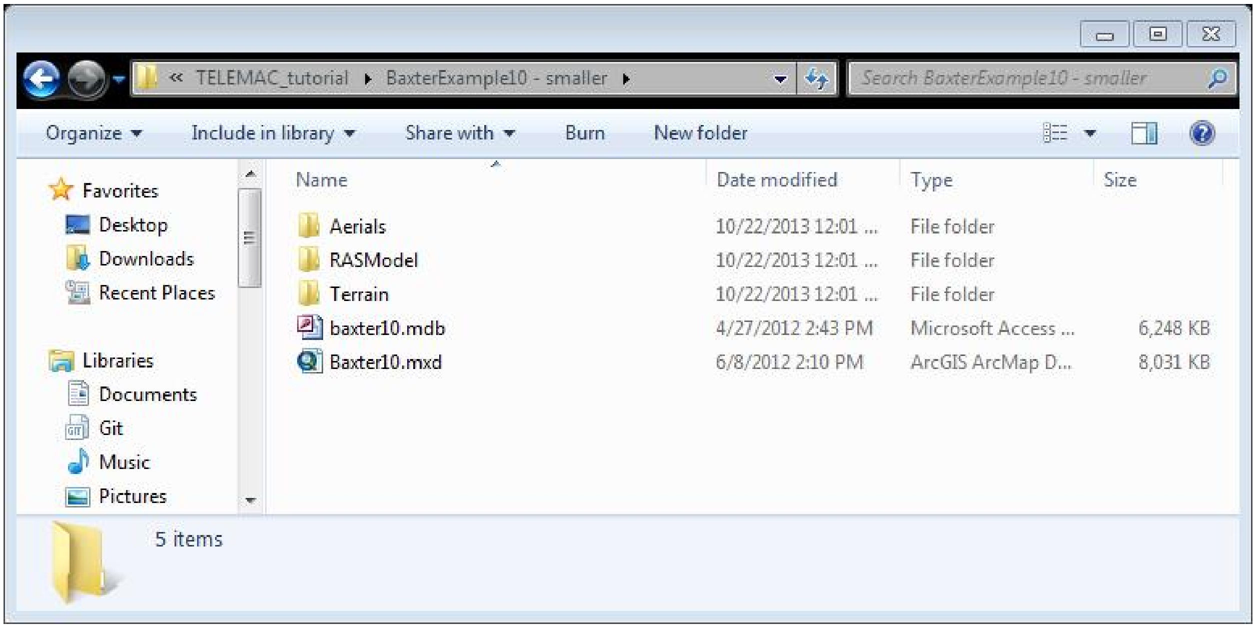


Figure 1: Contents of unzipped TELEMAC tutorial data. Note: extract this data to a directory with no spaces

2 Extract bathymetry from ArcGIS surface

Accurate representation of the waterway geometry is a crucial component for successful hydraulic modeling. In this section, two methods are illustrated for extracting a bathymetric data from an ArcGIS Triangulated Irregular Network (TIN) terrain model. The first method involves extracting *xyz* bathymetric points from the surface, into an *ASCII text file* and the second method exports the bathymetric data as an *ArcView Shape File (.shp)* .

2.1 ArcGIS TIN to ASCII text file

Start ArcMAP, and enable the *3D Analyst* extension from the $Customize > Extensions$ menu as shown in Figure 2.

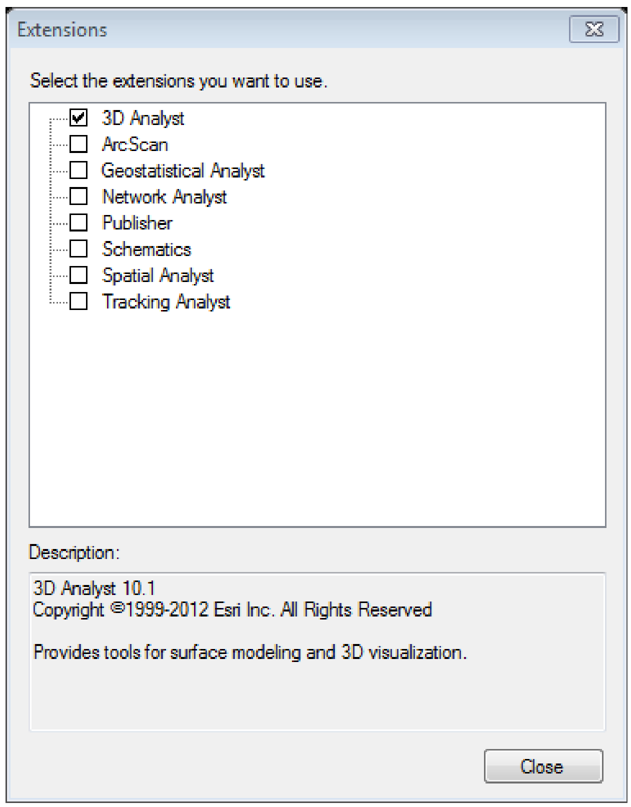


Figure 2: Activate *3D Analyst* extension from ArcMap $Customize > Extensions$ menu

Add the Baxter river TIN data set to the data frame using the 'Add Data' command, and open the 'ArcToolbox' window, as shown in Figure 3.

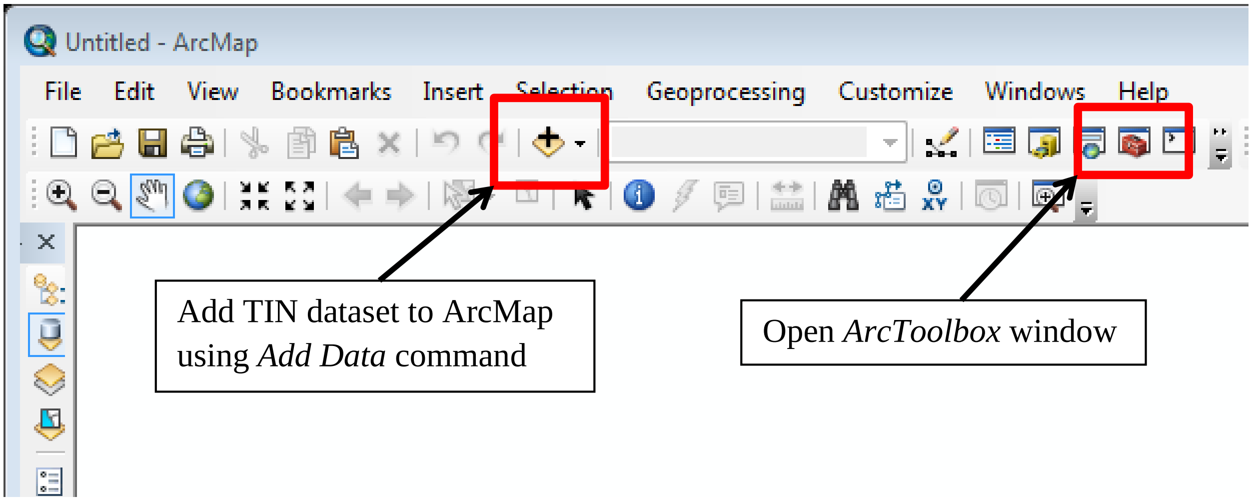


Figure 3: Add TIN data to ArcMap and initialize the ArcToolbox

In order to generate the finite-element mesh, the bathymetric and topographic information needs to be extracted or imported from a CAD program. A universal method to represent the elevation data of a domain is to have the xyz coordinates of each node representing the surface. This can be accomplished in three steps using ArcMap and Excel:

1. Extract TIN-Nodes using 3D-Analyst Toolbox command (Figure 4):
   $ArcToolbox > 3DAnalystTools > Conversion > FromTIN > TINNode$
2. Convert TIN-Node feature class to ASCII text via Toolbox command (Figure 4):
   $ArcToolbox > 3DAnalystTools > Conversion...$
   $> FromFeatureClass > FeatureClassZtoASCII$
3. Open text file using Excel, remove point index numbering column, and save-as a new text file (e.g. `Baxter_geometry.xyz`)

Steps 1 and 2 are illustrated in Figures 4 through 6, and Step 3 details are illustrated in Figures 7 through 11

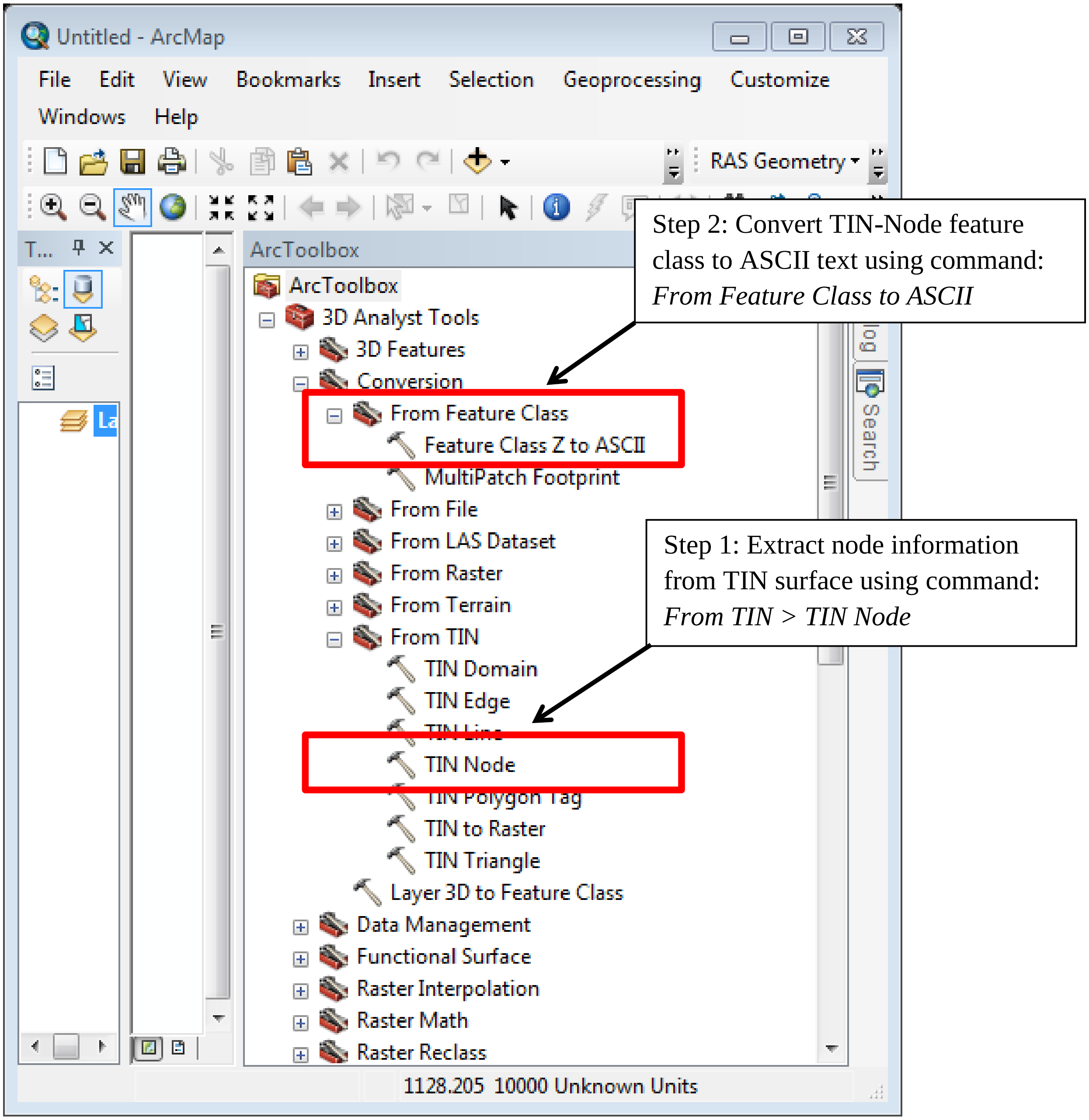


Figure 4: Extract and export ASCII xyz bathymetric data utilizing the **3D Analyst Tools** toolbox

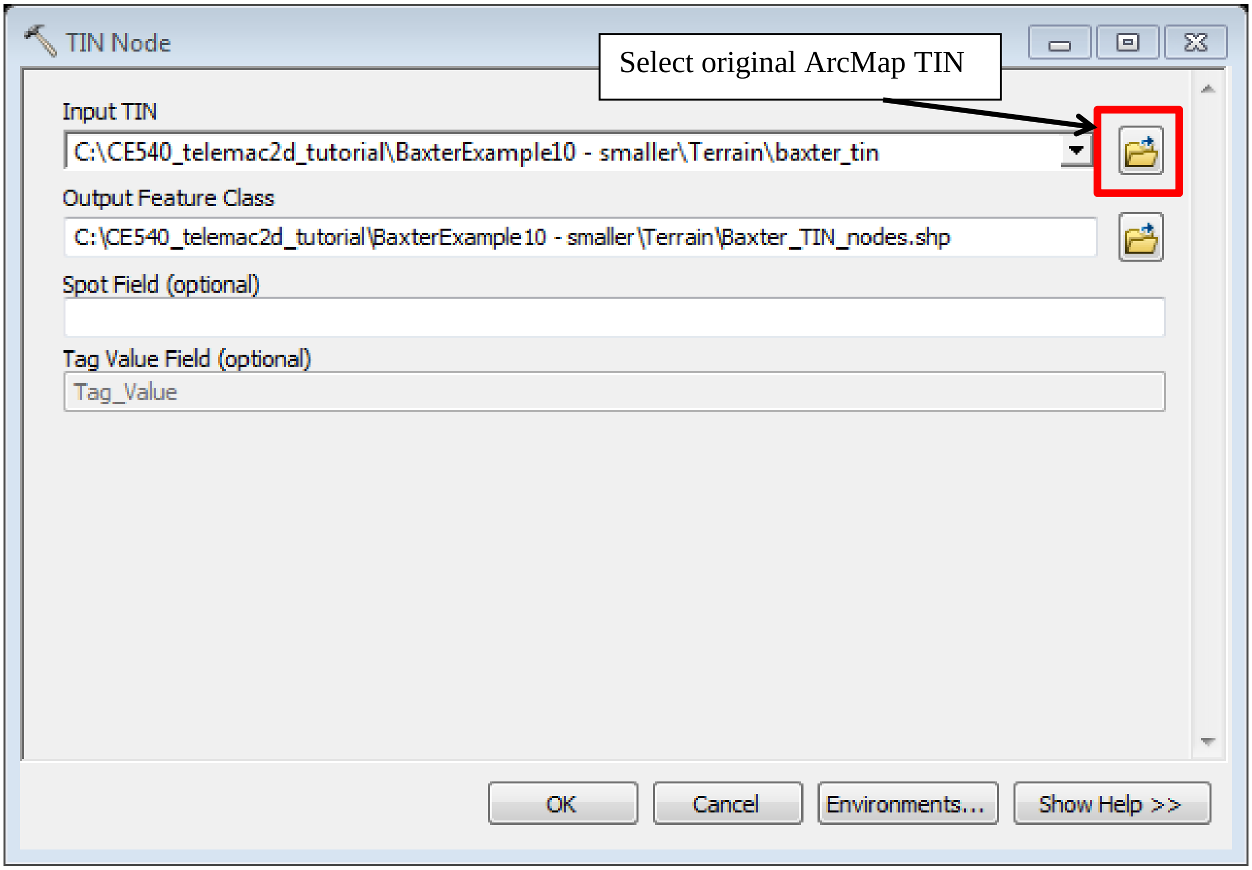


Figure 5: Prompt from **Step 1** , converting *TIN* to *TIN-Node Feature Class*

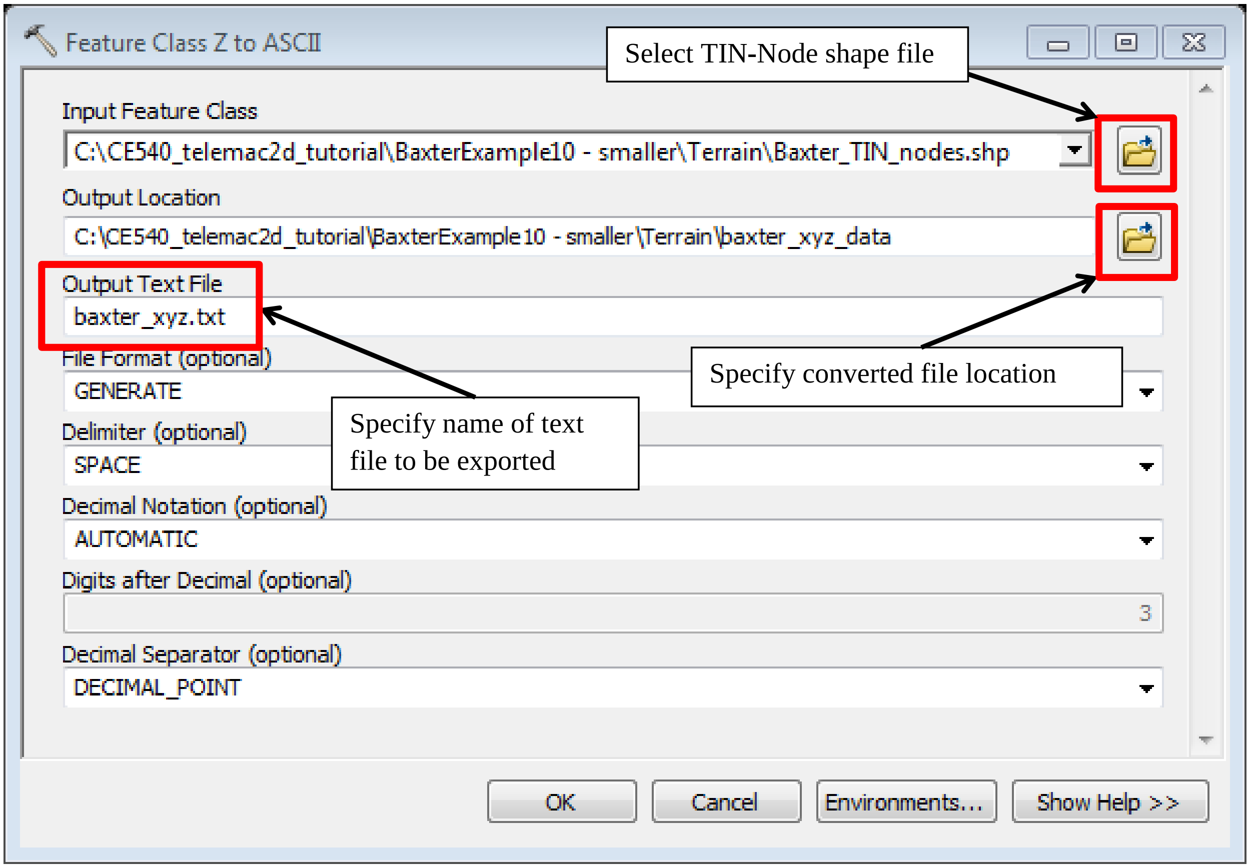


Figure 6: Prompt from **Step 2** , converting *TIN Node Feature Class* to *ASCII* text

2.2 Converting ArcGIS ASCII text file to XYZ text file

Now that the xyz point information has been converted from the ArcMap TIN to the corresponding ASCII format, the next step requires a minor change in formatting and unit conversion using Microsoft Excel. Import the ASCII bathymetric information from ArcMap to Excel (e.g. *File ¿ Open ¿* `baxter_xyz.txt`); each row of data represents an individual xyz point in the form illustrated in Table 1:

Table 1: Individual **xyz** point information from ArcMap ASCII bathymetry file

point-id (#)	x-coord (ft)	y-coord (ft)	z-coord (ft)
1	6417302.24	2048668.67	32.5123
.	.	.	.
.	.	.	.
.	.	.	.
238547	6417290.38	2048732.06	31.5000

For this tutorial, the original source data (e.g. **baxter tin**) had to be converted from US Survey Feet to the SI unit equivalent for use in TELEMAC-2D. This was done using the conversion described in Section 2.5.

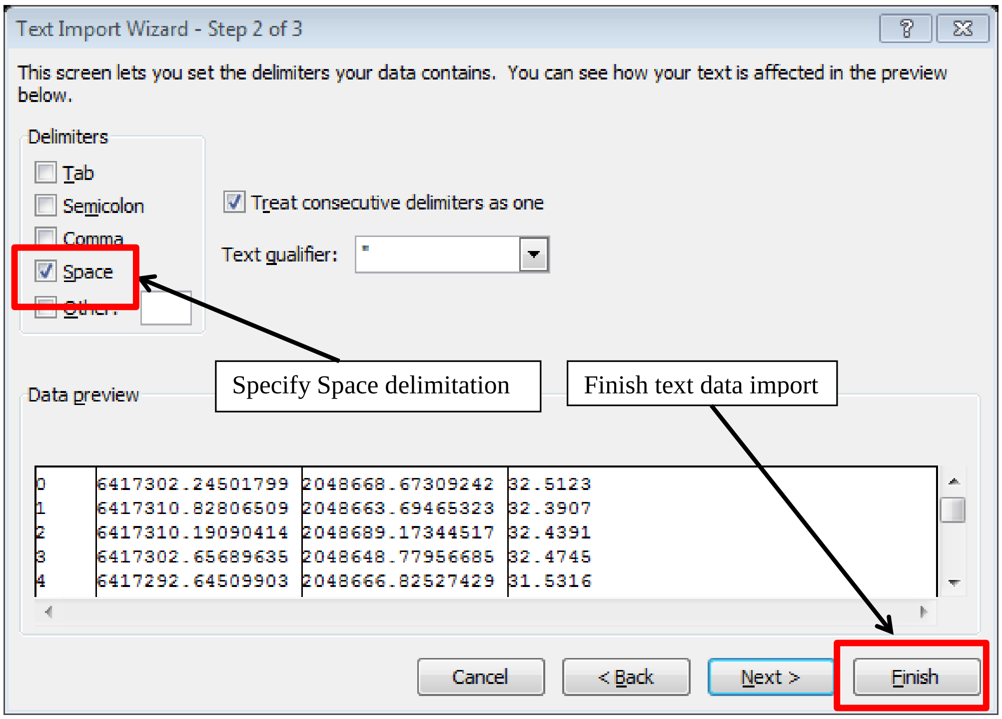


Figure 7: Open the Baxter River ASCII text file using Excel, importing as a space delimited file

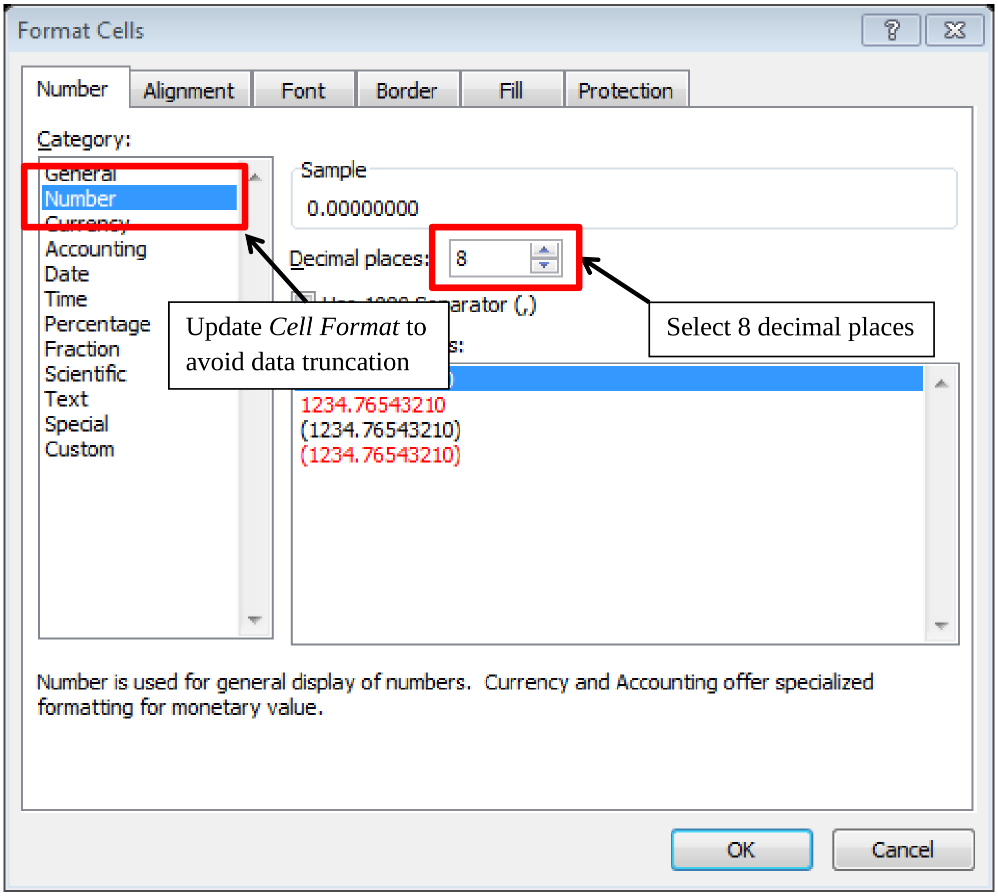


Figure 8: Format the cells in order to avoid truncating the data

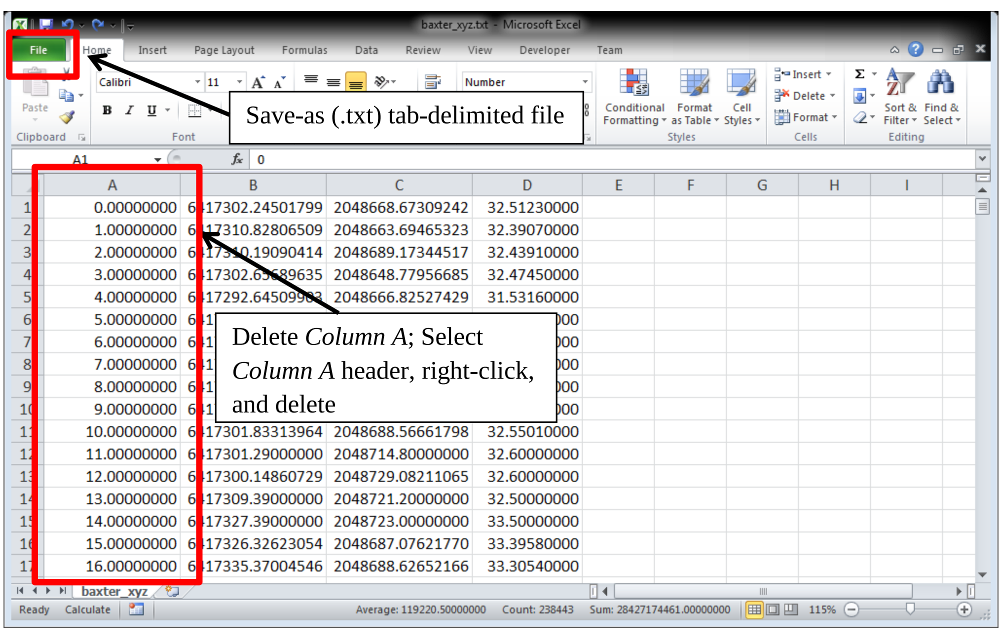


Figure 9: Node IDs are not necessary for *xyz* format; delete the entire column of IDs

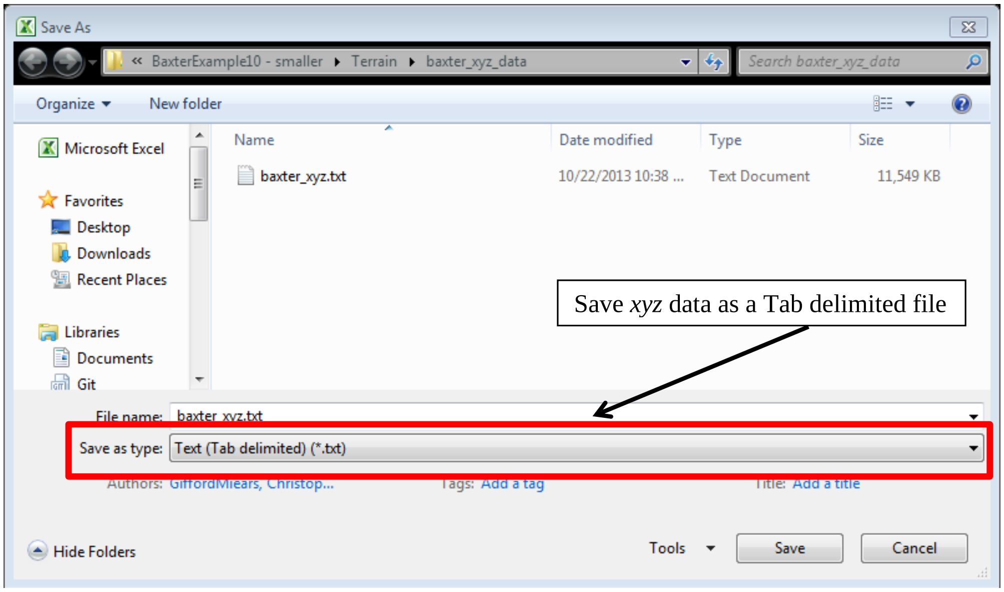


Figure 10: Save altered bathymetric ASCII text file as a Tab delimited text file

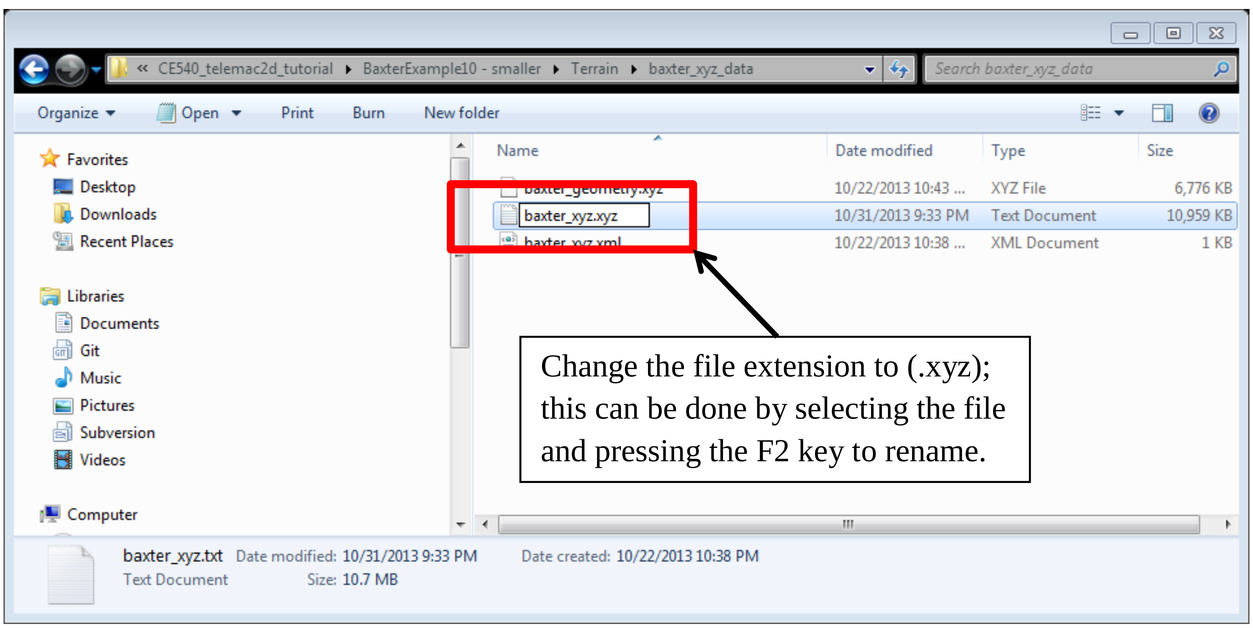


Figure 11: BlueKenue utilizes a (.xyz) file extension, therefore the extension is simply updated to (.xyz)

2.3 Export ArcMap data as a Shape File

The ability to import **ArcView Shape Files** into BlueKenue is a powerful feature. The previous topographic data for the *HEC-GeoRAS* tutorial (e.g. levees, bridges, reservoirs, etc.) can readily be included for analyzing TELEMAC-2D results and incorporating these features into the computational mesh.

For example, exporting the *HEC-GeoRAS* cross-sections, 3D levee and bridge information, can each be accomplished in a single step from ArcMap, as shown in Figures 12, 13 and 14.

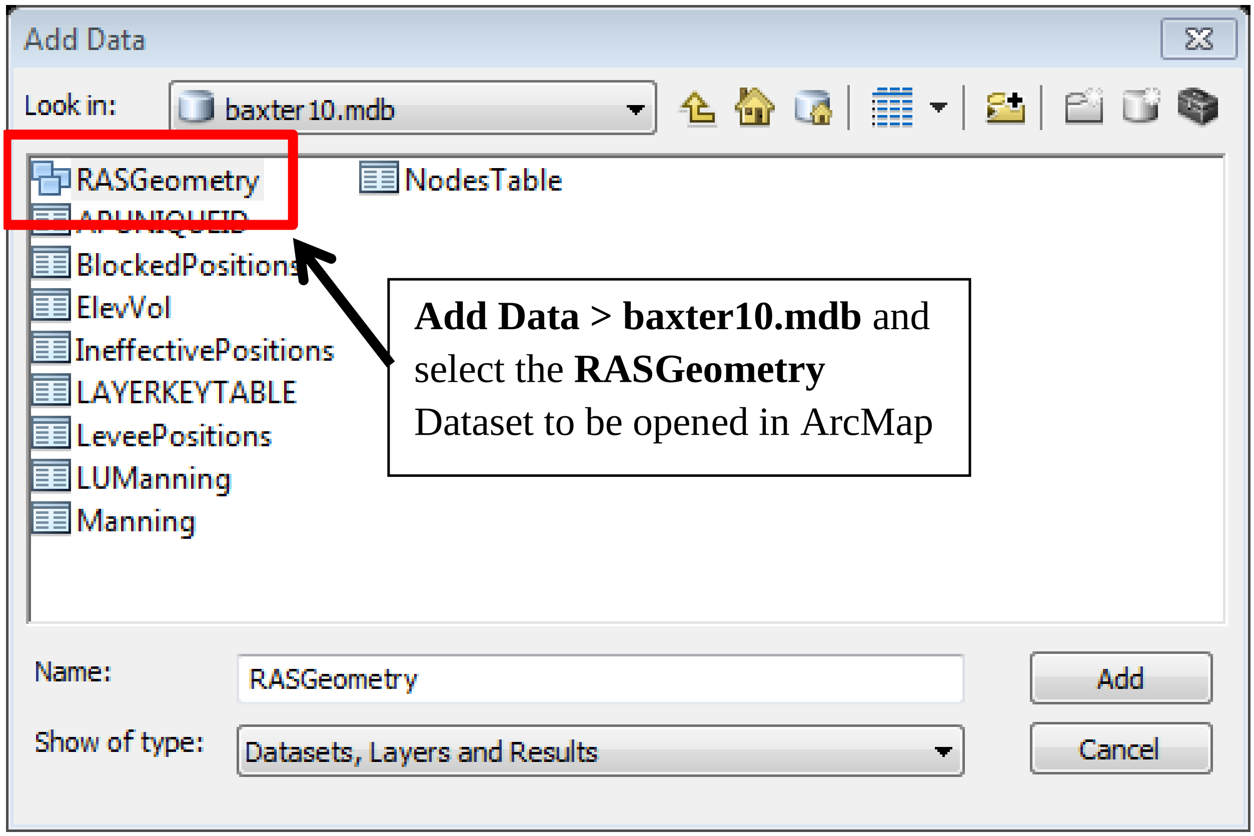


Figure 12: Open ArcMap and add the **RASGeometry** data layer from **baxter10.mbd** database

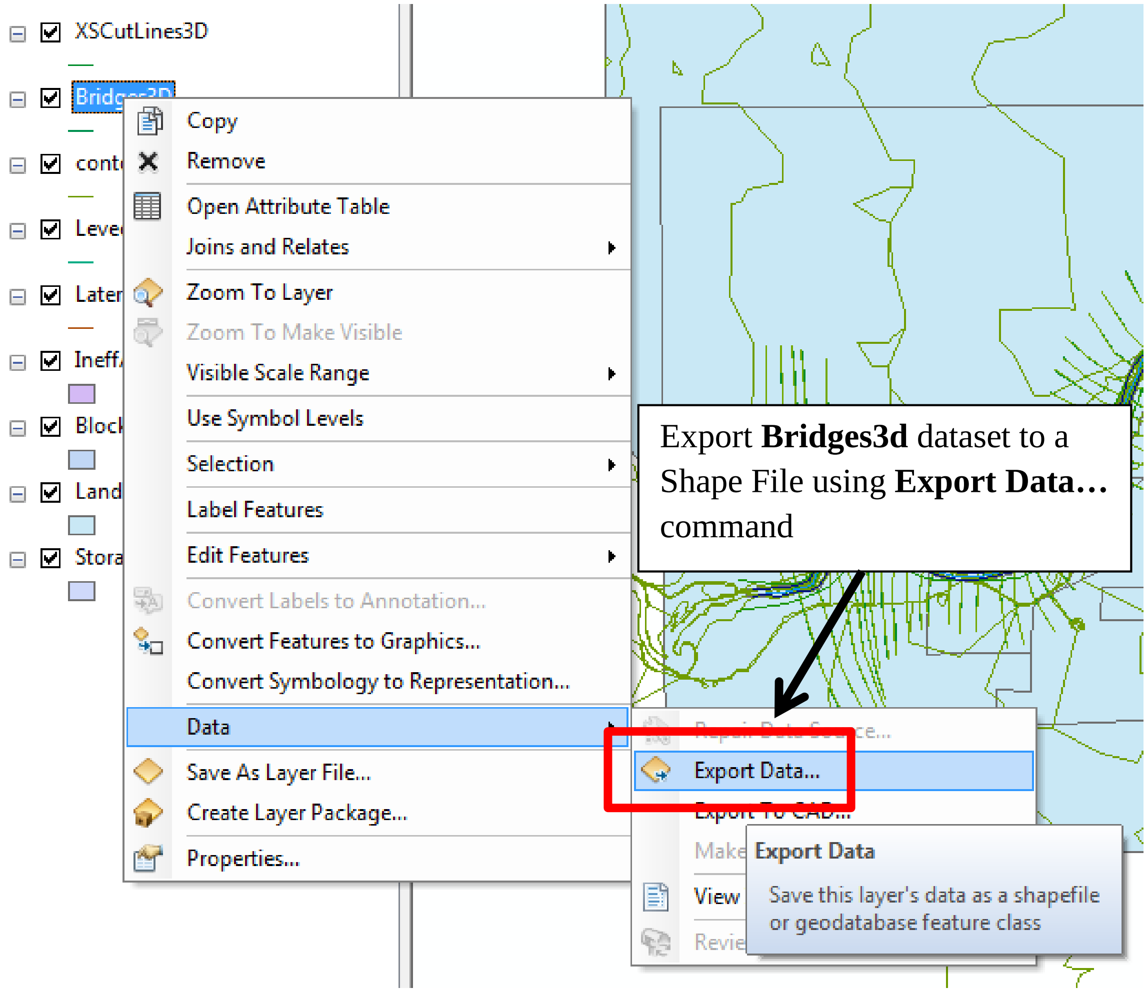


Figure 13: Select dataset of interest from RASGeometry and **Export Data**

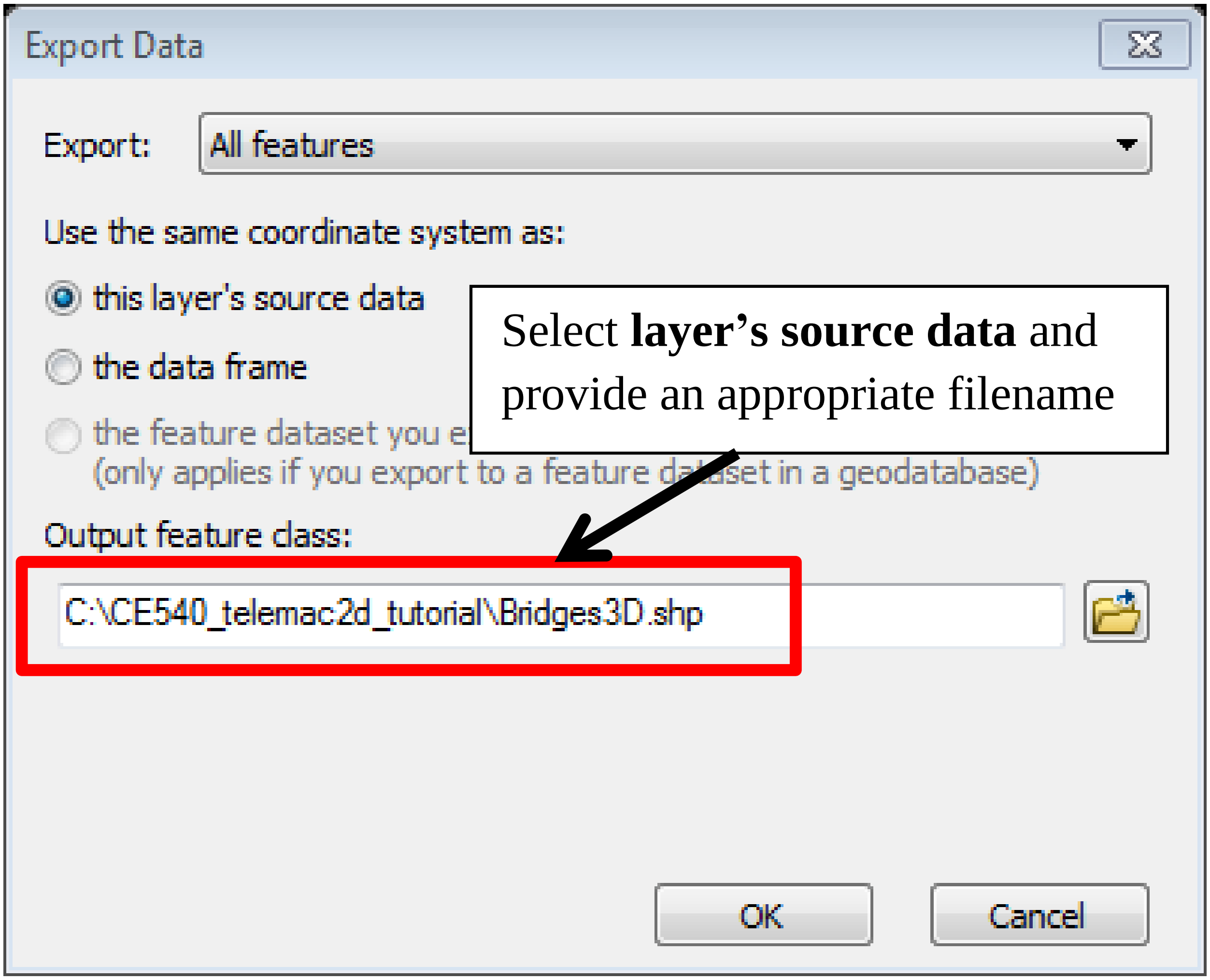


Figure 14: Exported data features, as **Shape Files (*.shp)** , are simple to import using BlueKenue

Once the **ArcView Shape Files** are exported, importing them to BlueKenue is achieved through the **File ¿ Import ¿ ArcView Shape File** command, detailed in the subsequent section.

2.4 Representing TIN elevation data using Shape file

If the raw *xyz* values are not required, then you can simply perform Step 1 of the process described in Section 2.1 (i.e. Figure 4), save the newly created Shape File (.shp), and import the Tin-Node file directly into BlueKenue. To import an ArcMap Shape File to BlueKenue, use the *File ¿ Import ¿ ArcView Shape File* command as depicted in Figure 15.

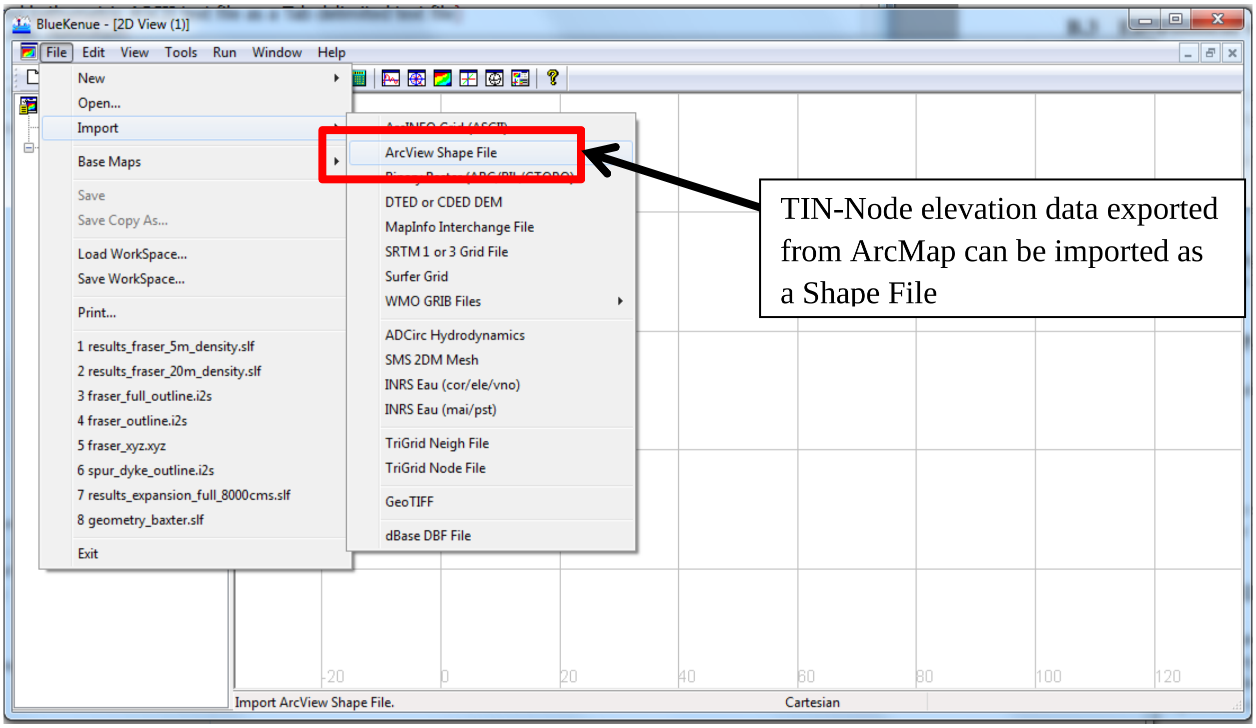


Figure 15: BlueKenue is capable of importing ArcMap/ArcView Shape Files directly

2.5 Convert source data from US Survey Feet to Meters

The geographic projection of the original dataset is Oregon State Plane Coordinates in United State (US) Survey Foot units (ft). The exact project information in ArcMap:

`NAD_1983_StatePlane_California_III_FIPS_0403_Feet`

TELEMAC-2D exclusively uses International System of Units (SI), therefore the projection of the elevation data **must** be converted to equivalent SI units (i.e. meters). The conversion of US Survey foot to meters is shown below in Table 2:

Table 2: Conversion equivalents from US Survey Feet to Meters

US Survey Foot (ft)	Meter (m)	Meter (m)
1.0	1200/3937	0.3048006096

Resulting information and datasets can be viewed in ArcMap after conversion so long as the geographic projection is changed to the meter equivalent:

`NAD_1983_StatePlane_California_III_FIPS_0403`

3 Pre-processing utilizing BlueKenue

BlueKenue is developed by the *Canadian Hydraulics Centre of the National Research Council* and is utilized in this tutorial to accomplish the following:

1. Generate the Baxter Finite-Element mesh
2. Create the boundary conditions influencing the system
3. Visualizing the TELEMAC-2D hydrodynamic results

3.1 Importing data to BlueKenue

Start BlueKenue and open the $(.xyz)$ geometry data extracted from the ArcMap TIN, or import the Tin Node Shape File (Figure 16). In order to see your XYZ dataset, change the $Filetype$ drop-down menu from $Selaphin$ to $AllFiles$ as shown in Figure 17.

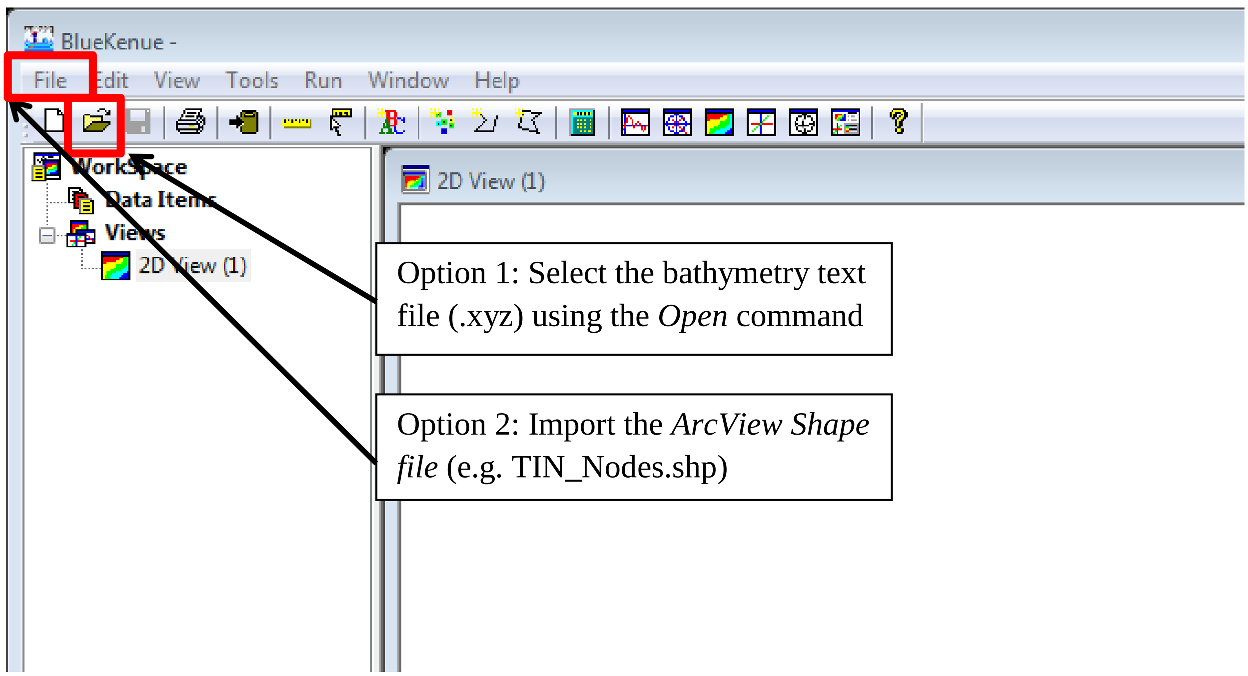


Figure 16: Import the Baxter bathymetry using either the *Open* command or the *Import ¿ ArcView Shape file* command

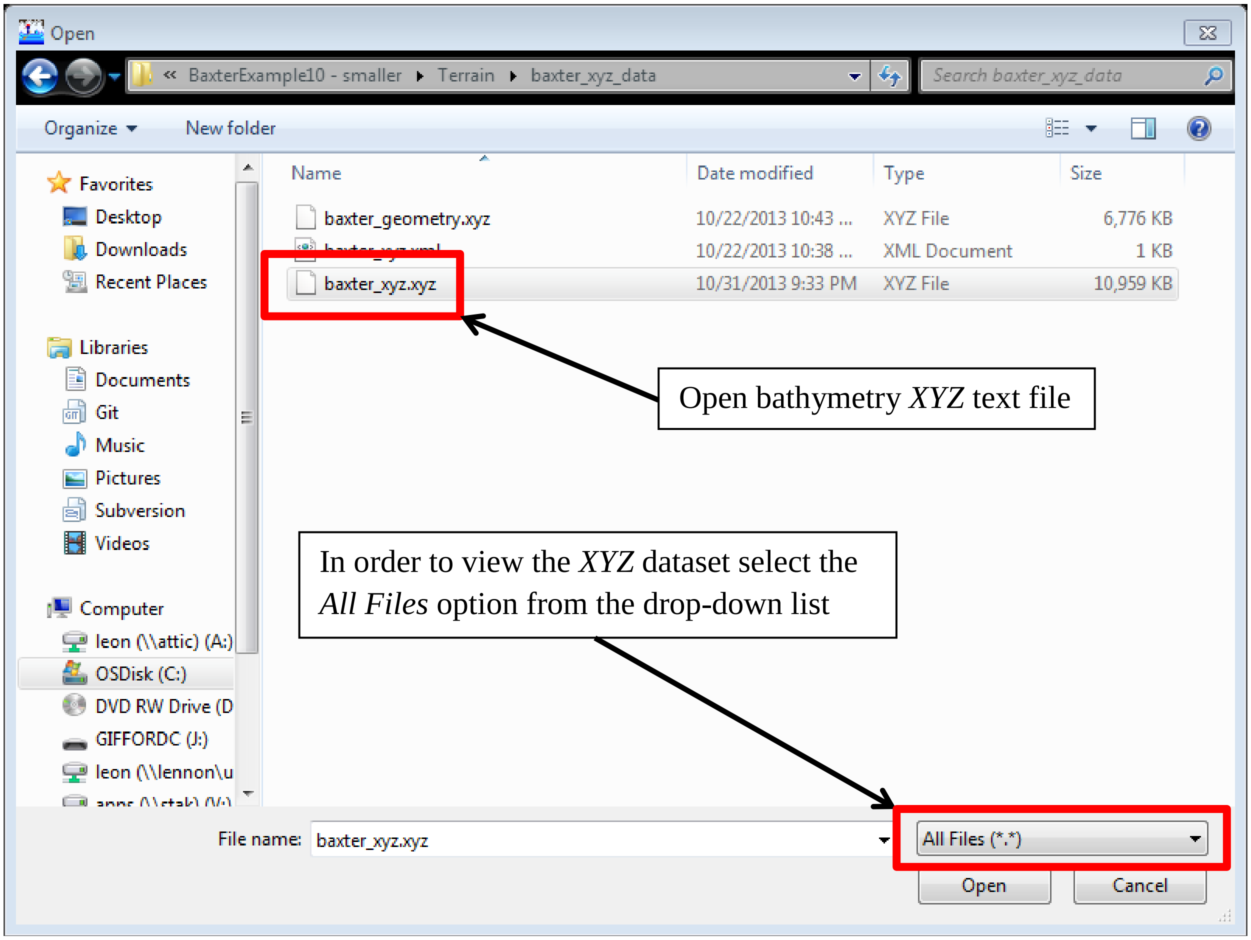


Figure 17: When importing the bathymetry text file, toggle file types to *All Files (*.*)*

3.2 Viewing data in BlueKenue

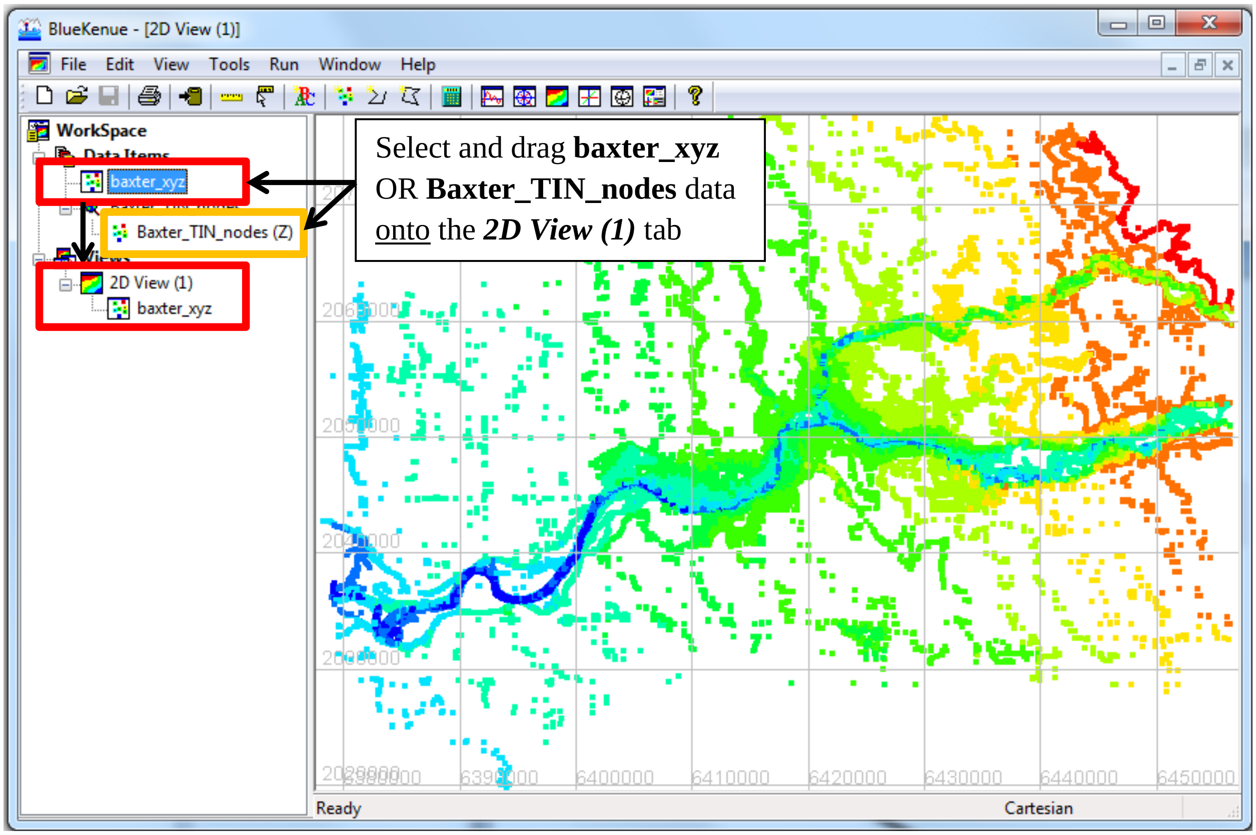


Figure 18: To view **Data Items** imported to BlueKenue, drag item of interest to a **Views** object (e.g. **2D View (1)**)

The bathymetric data can also be viewed in an iso-metric **3D View** by opening a new **3D View icon** or through **Window ¿ New 3D View** as shown in Figure 19.

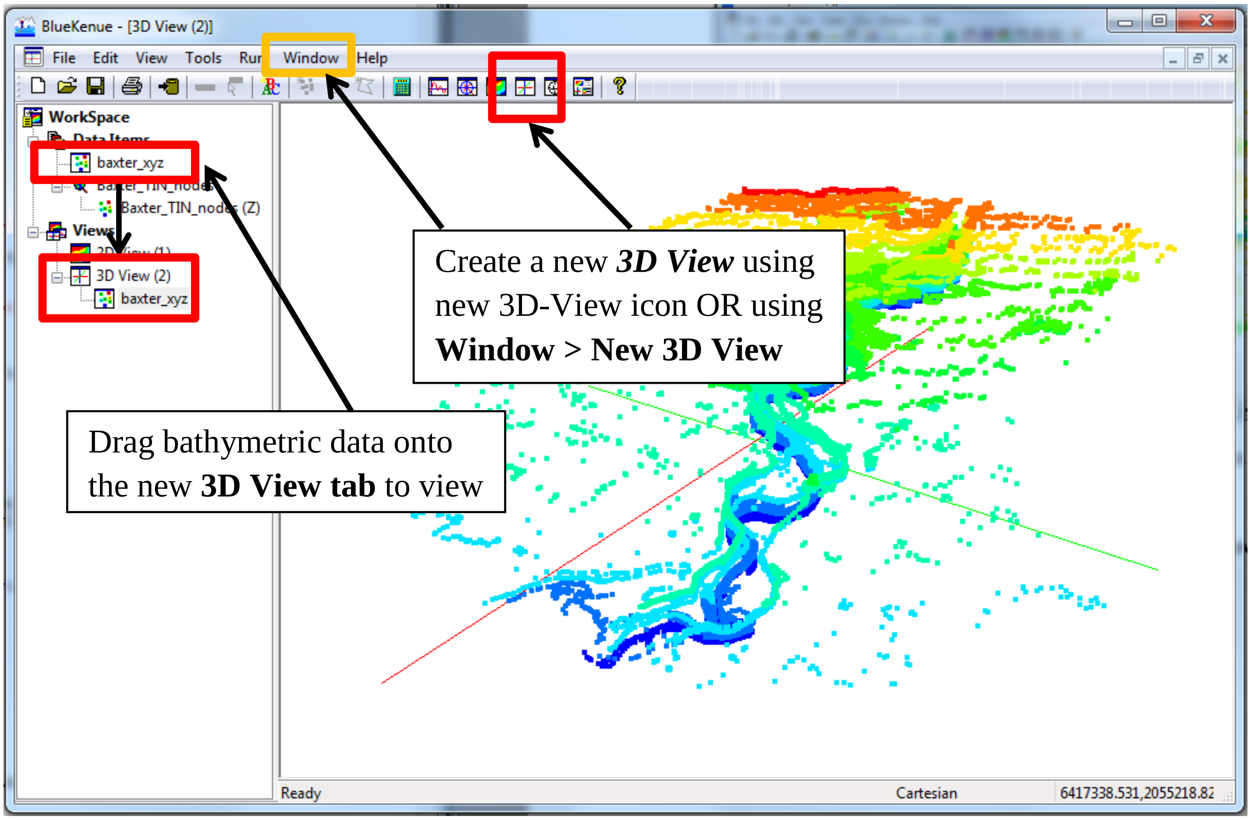


Figure 19: Create a new viewing window using the **toolbar icons** or **Windows** command and drag the data object to the new view (e.g. **3D View (2)**)

3.3 Mesh Generation

TELEMAC-2D solves the depth-averaged Navier-Stokes equation utilizing both Finite-Element (FE) and Finite-Volume (FV) formulations. All of these formulations require that a spatial representation of the domain be created using a compuatational mesh. BlueKenue has several tools for mesh generation and editing. Mesh types that BlueKenue can generate are unstructured and regular (via **T3 Channel Mesher**) triangular meshes.

The basic requirements for creating a simple mesh in BlueKenue are outlined in Appendix A. For this tutorial, the following steps are illustrated for generating a mesh incorporating the Baxter River, Tule Creek, Flood-plains, and Levee embankment components:

1. Create **New Closed Line** around computational domain of interest
2. Create **New Open Lines** outlining the river channel, and levee, right and left extents
3. Open a new Channel Mesher via **File ¿ New ¿ T3 Channel Mesher...**
   (a) Specify the number of cross-channel nodes
   (b) Specify the mesh element length along the channel
   (c) Drag and drop the right and left channel bank open-lines to the **RightBank** and **LeftBank** objects
   (d) Double click the **T3 Channel Mesher** and select **Run**
   (e) This will be performed for both the Baxter River and Tule Creek channels
4. Represent the levee utilizing a new **T3 Channel Mesher...**
   (a) Specify the number of cross-channel nodes
   (b) Specify the mesh element length along the levee
   (c) Drag and drop the right and left channel levee open-lines to the **RightBank** and **LeftBank** objects
   (d) Double click the **T3 Channel Mesher** and select **Run**
5. Open a new T3 Mesh generator via **File ¿ New ¿ T3 Mesh Generator...**
   (a) Drag domain outline to **Outline** child-object
   (b) Drag main-channel and creek mesh onto the **SubMeshes** child-object
   (c) Drag levee mesh onto the **SubMeshes** child-object
   (d) Drag **3D Line Set** representing bridge abutments to **HardLines** child-object
   (e) Double click the **T3 Mesh Generator** and specify **Default Edge Length** and **Edge Growth Ratio**

Following these steps will result in the generation of a hybrid unstructured triangular mesh that incorporates major topographic features in this flood-inundation scenario. The steps outlined above are described in additional detail, in their respective order, in the following subsections.

3.3.1 Steps 1 and 2: Create Open- and Closed-Lines in BlueKenue

Steps 1 and 2 involve creating a bounding polygon of the entire domain and the delineation of the river channel, and levee, right and left banks. Figure 20 shows the process of creating a **New Closed Line** to represent the extents of the computational domain. Once the **New Closed Line** icon is selected, you can use the cursor to define the points representing the outline. Figure 20 also shows the imported *HEC-GeoRAS* cross-section cuts as a reference for creating the outline (see Section 2.3 for details how)

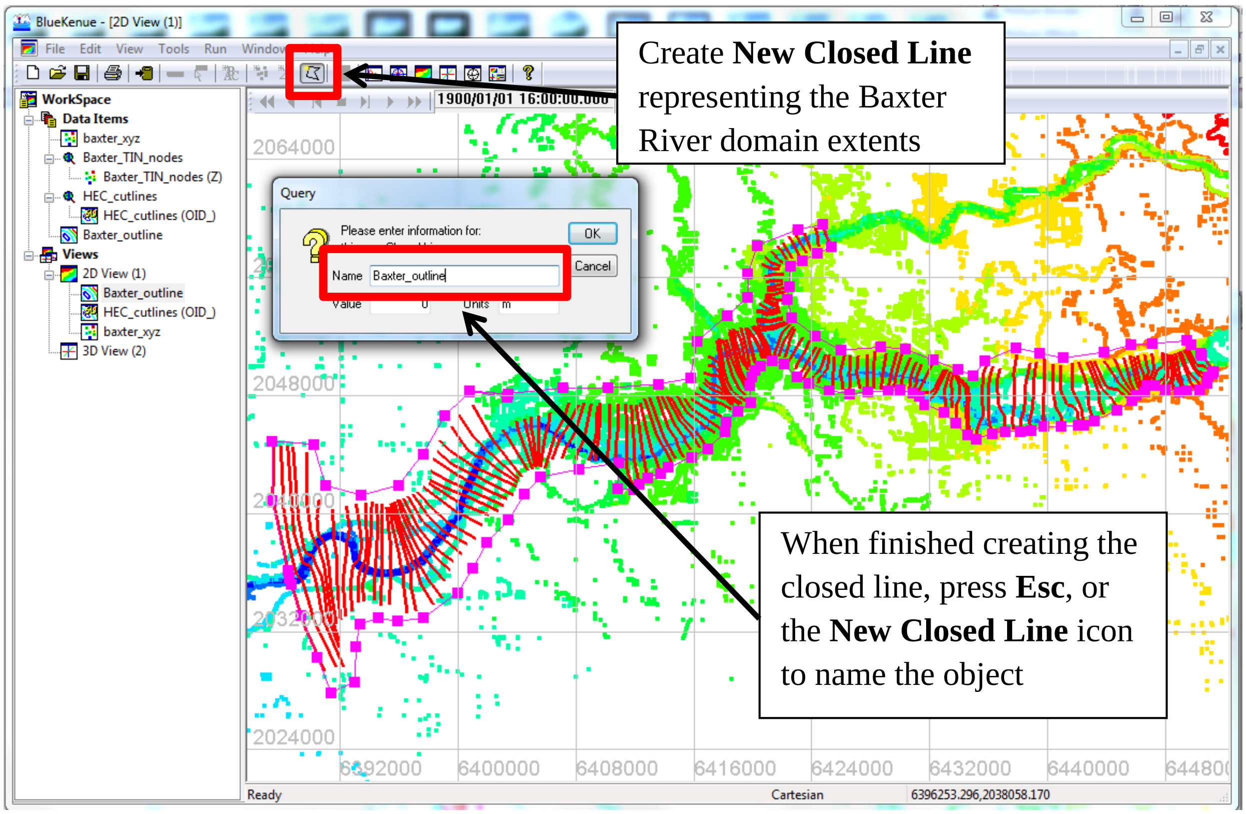


Figure 20: Step 1; create a **New Closed Line** representing the domain extents

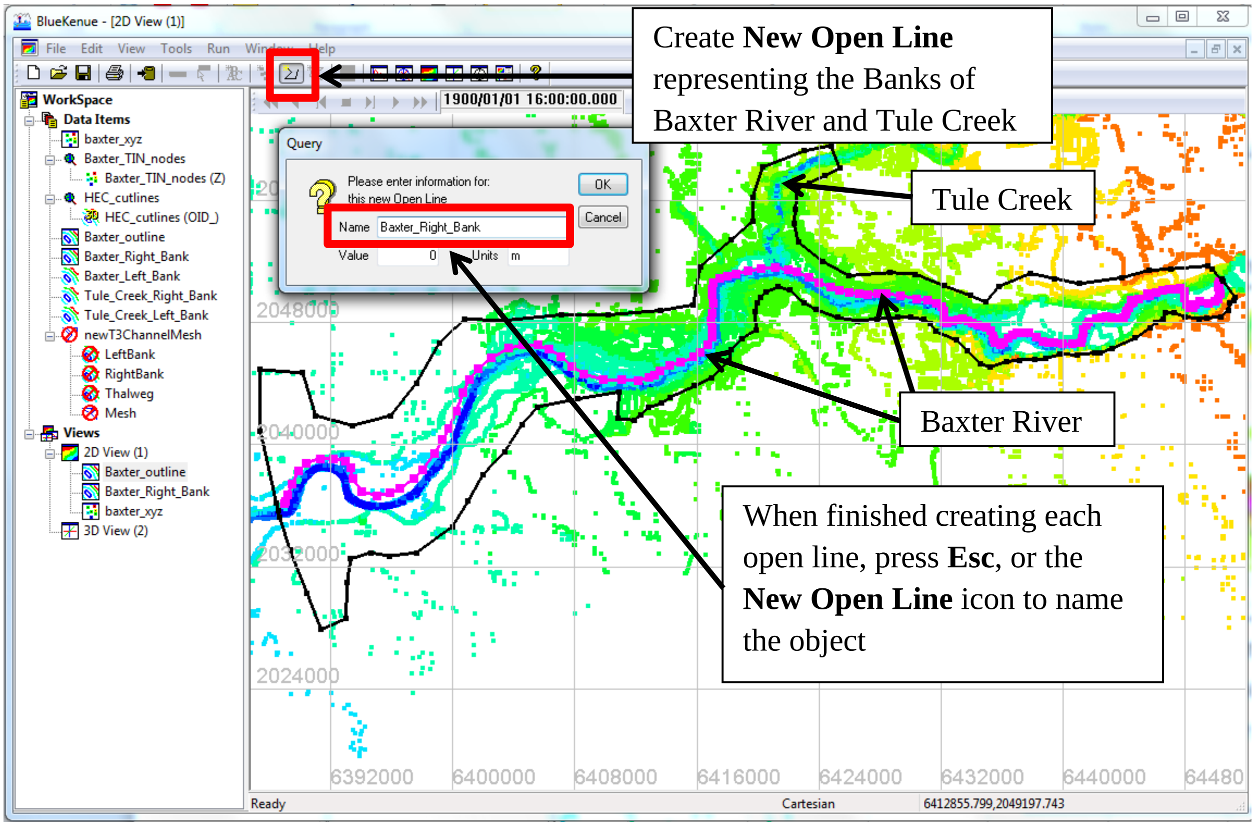


Figure 21: Step 2; Right and Left banks are necessary for all channels; either import the 2D-Lines (.i2s), or create them using **New Open Lines**

3.3.2 Steps 3 and 4: Create Channel Meshes in BlueKenue

The mesh for this tutorial includes three channel mesher components:

1. Baxter River channel mesh
2. Tule Creek channel mesh
3. Levee represented using channel mesh

Table 3 presents the values that will be used for the tutorial mesh parameters. Generating a channel mesh is illustrated in Figures 22 through 25

Table 3: BlueKenue Channel Mesher and Mesh Generator tutorial values

BlueKenue Mesh Type	Mesh Component	Cross Channel Node Count $(-)$	Channel Interval/Edge Length (M)
T3 Channel Mesher	Baxter River	10	50
T3 Channel Mesher	Tule Creek	10	30
T3 Channel Mesher	Levee Mesh	3	20
T3 Mesh Generator	Full Mesh	-	100

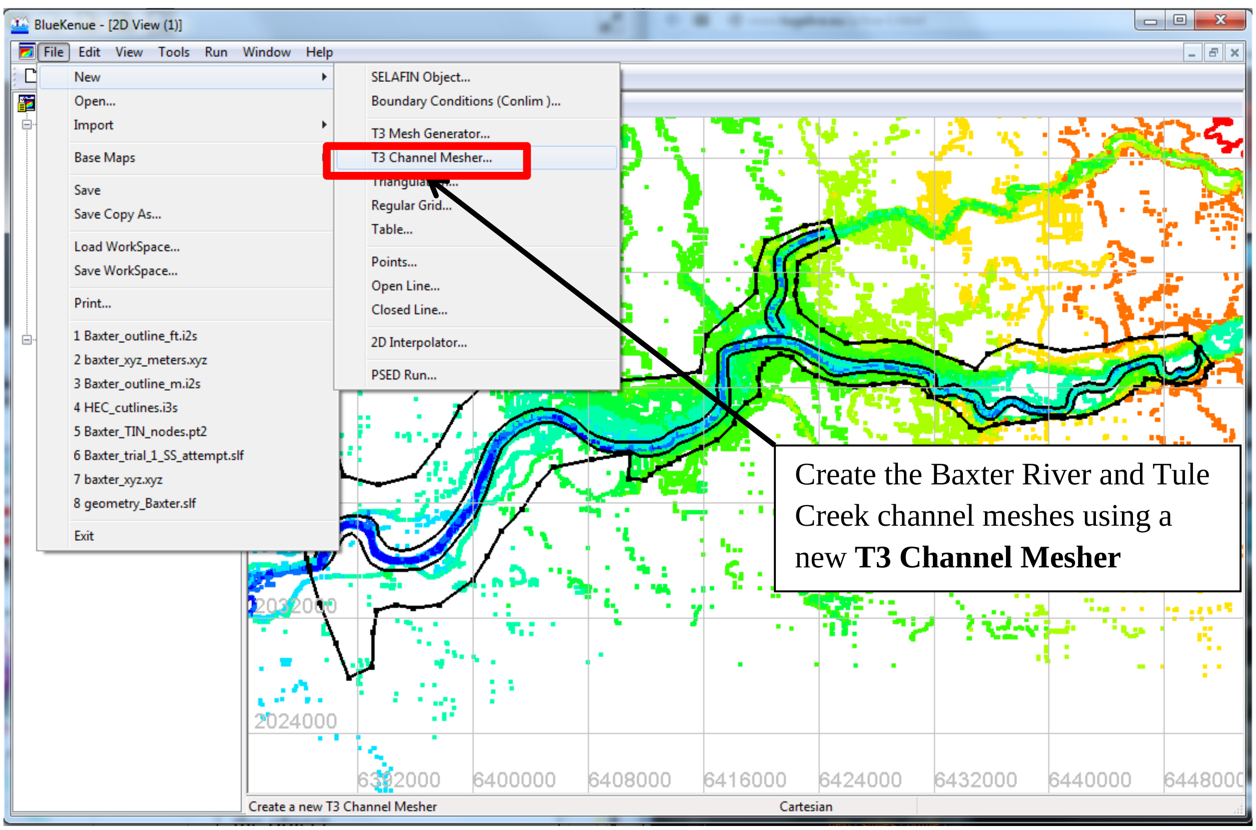


Figure 22: For each channel mesh (i.e. 3 in total) a new **Channel Mesher** object should be initialized

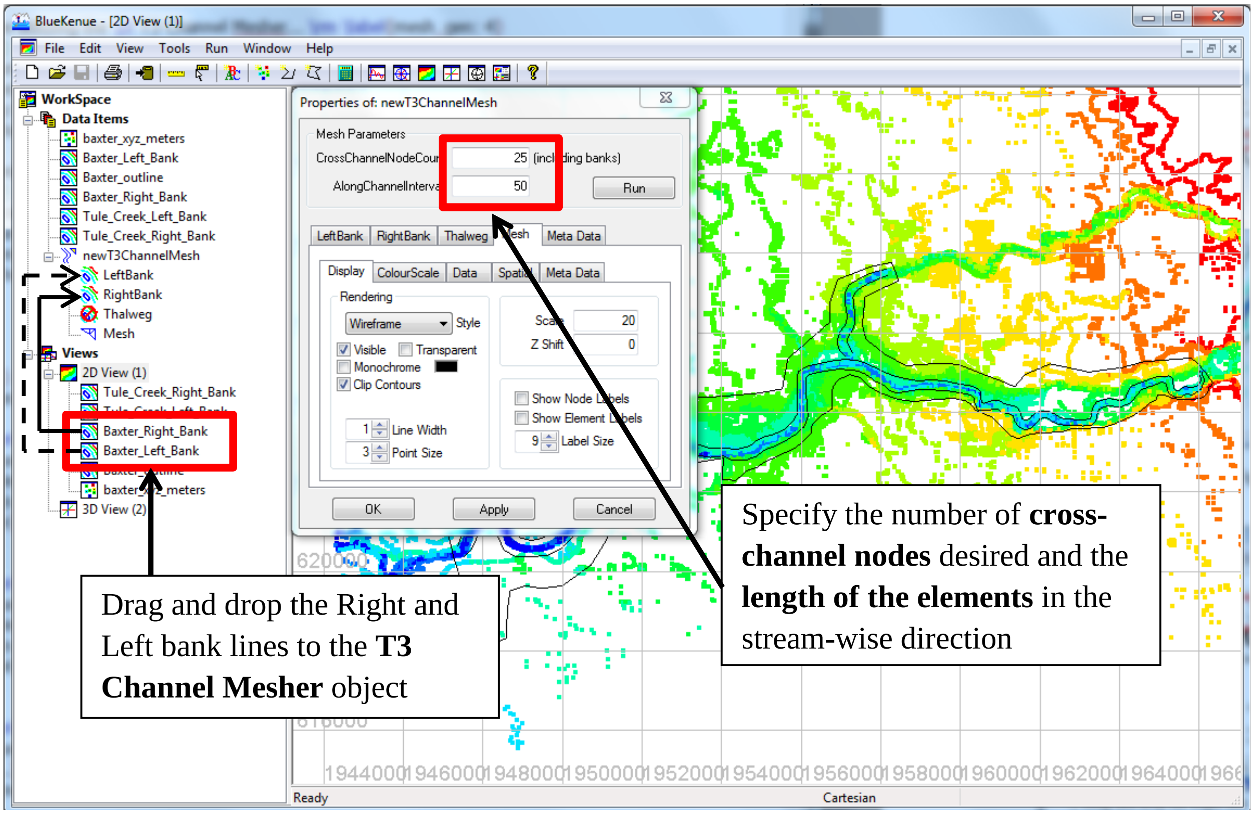


Figure 23: Specify the **Open Lines** representing banks of the channel feature of interest (e.g. Baxter River) and specify the meshing parameters (Table 3)

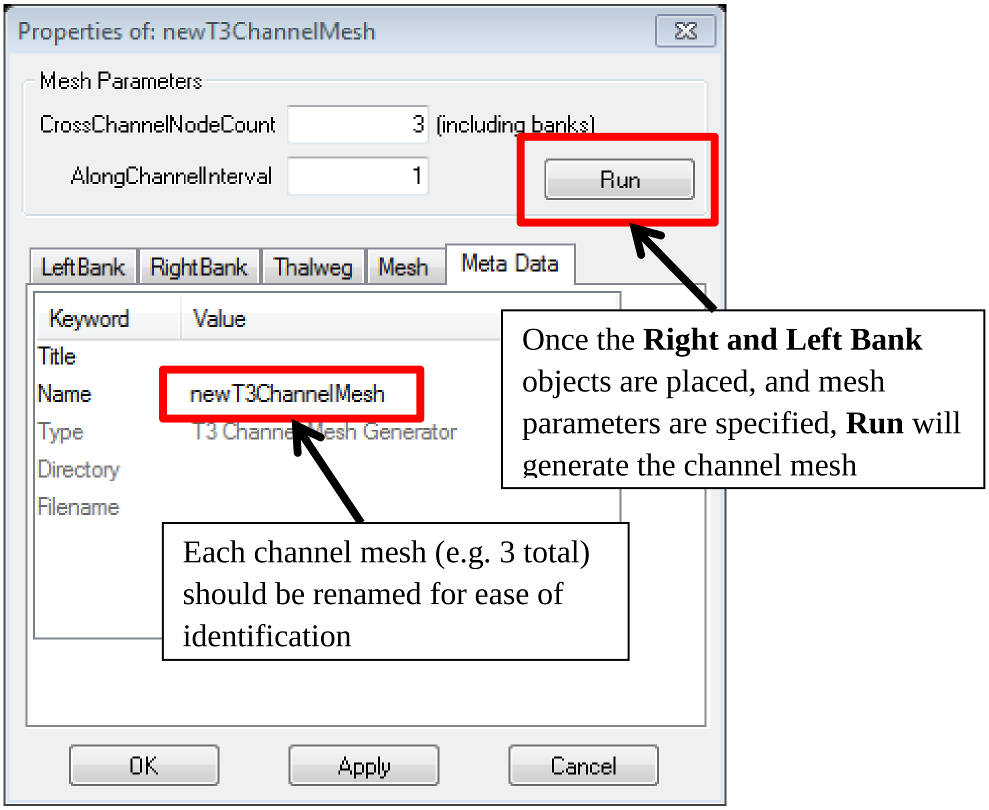


Figure 24: Rename and **Run** the Channel Mesher

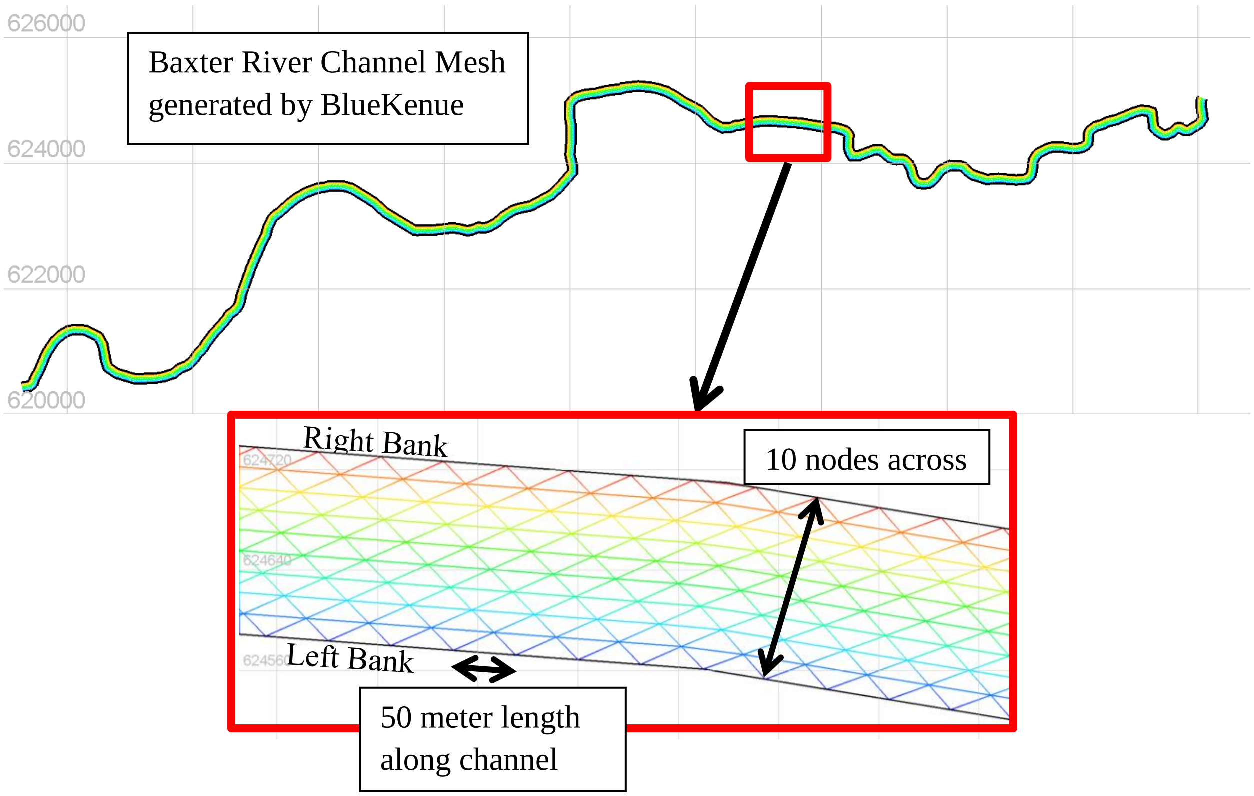


Figure 25: Resulting Baxter River Channel Mesh and corresponding parameters

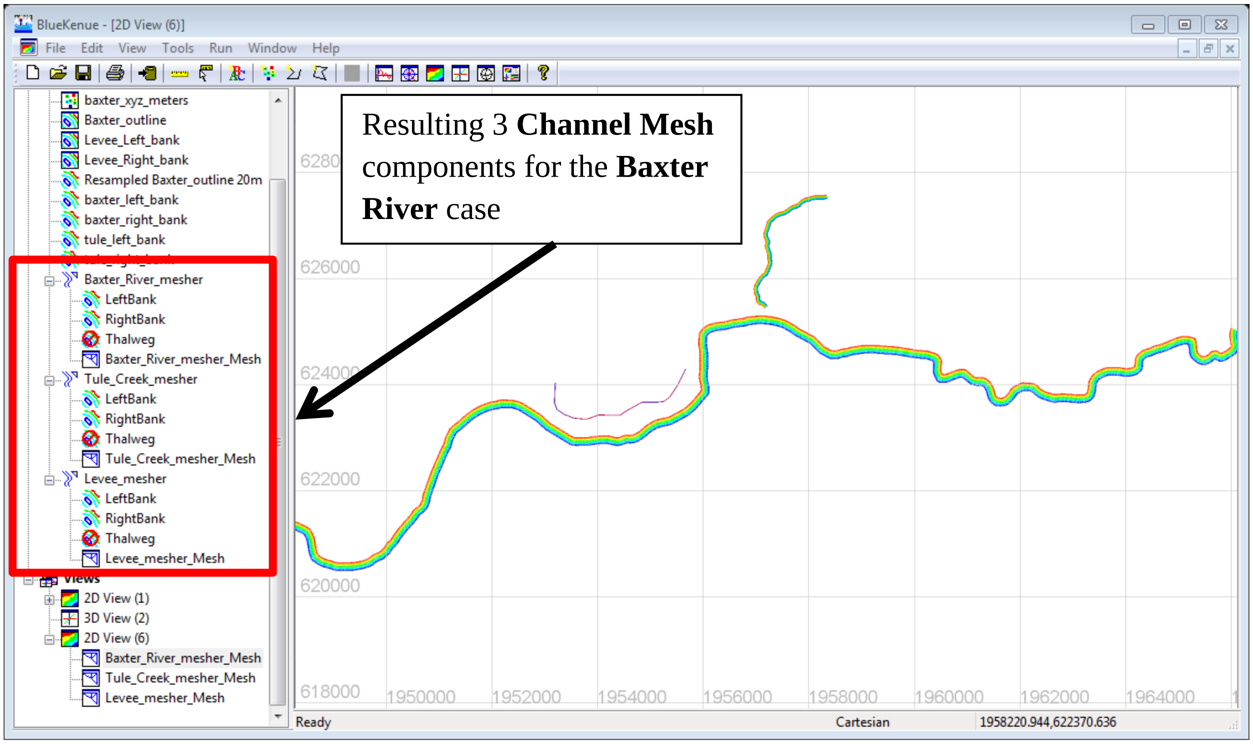


Figure 26: Resulting Baxter River Channel Meshes

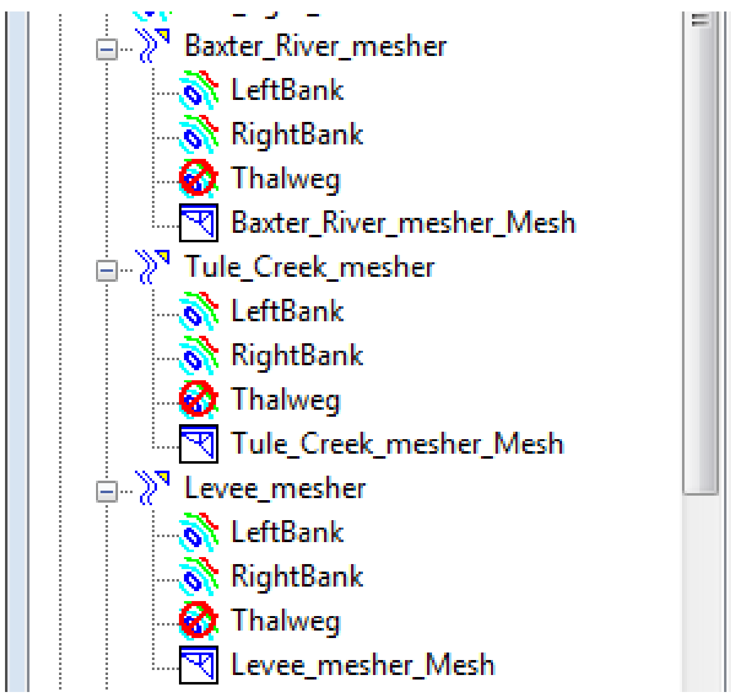


Figure 27: Baxter River Channel Meshes components

3.3.3 Steps 5: Generate Combined Mesh in BlueKenue

Once the submesh components are created, the domain mesh combining all elements can be generated. Due to the large size of the domain (~ 20KM), a coarse mesh will be generated for demonstration purposes. The objects needed for creating this mesh include:

1. Baxter River channel mesh
2. Tule Creek channel mesh
3. Levee channel mesh
4. Baxter domain outline (`Baxter_outline_20m_resampled.i2s`)
5. Bridge 3D elevation line (`Bridges3d_20m_resampled.i3s`)

All of the above items are available from the tutorial dataset directory, `BK_baxter_tutorial_files`.

Open a new T3 Mesh generator via **File ¿ New ¿ T3 Mesh Generator...** and enter the settings specified in Figure 28 and select **Apply,** then exit.

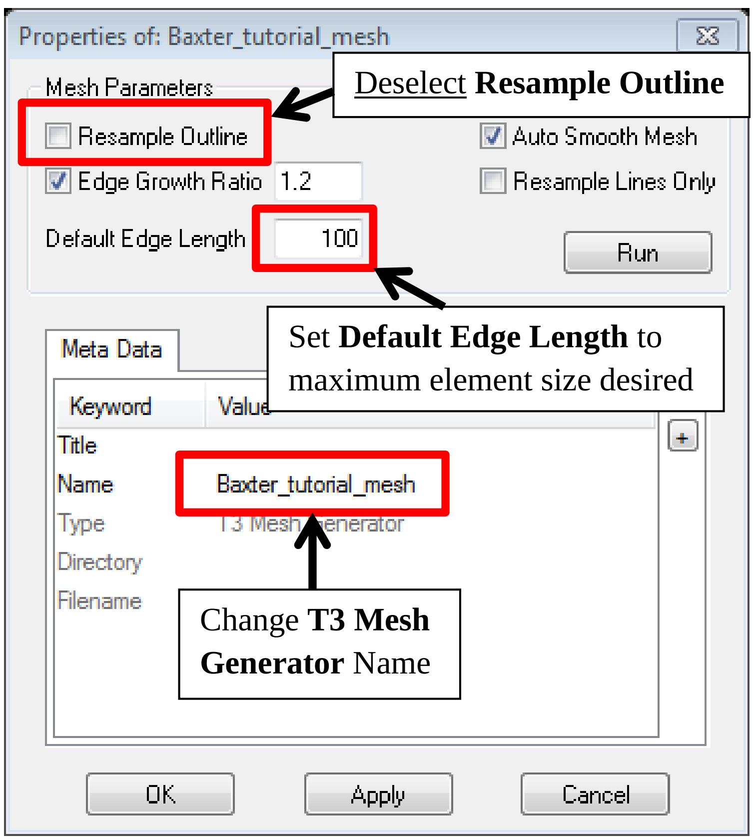


Figure 28: Settings for Baxter tutorial T3 Mesh Generator

BlueKenue mesh generator utilizes a Delaunay triangulation method and can accommodate a wide range of complex topographic and bathymetric features. Following Step 5 of the Mesh Generation process, the location of each component in the **T3 Mesh Generator** are specified. Figure 29 illustrates the component locations from Step 5.

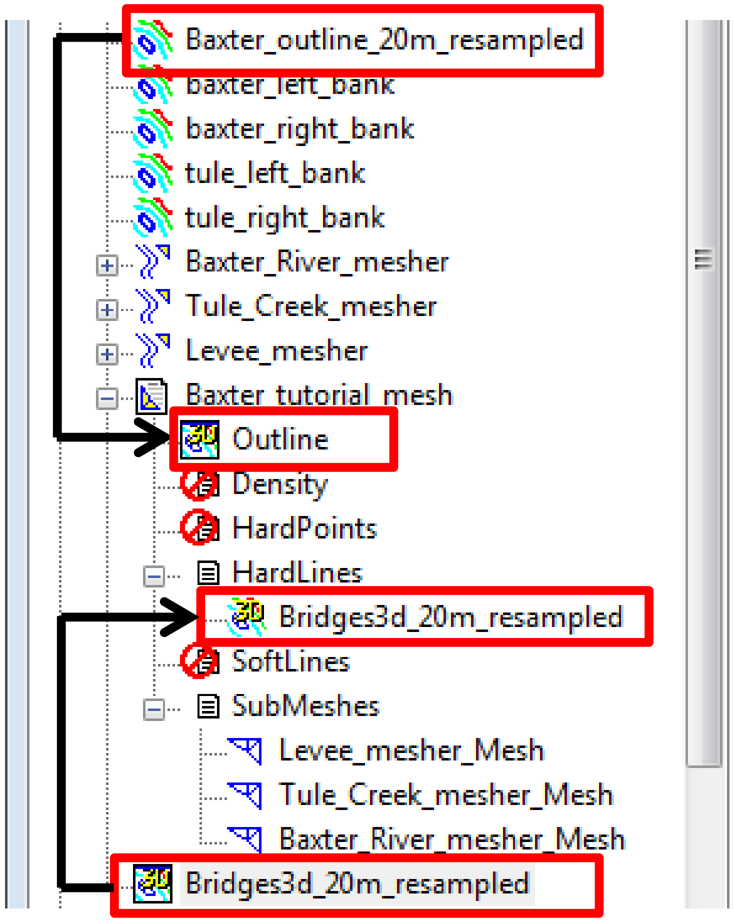


Figure 29: Mesh components added to Mesh Generator

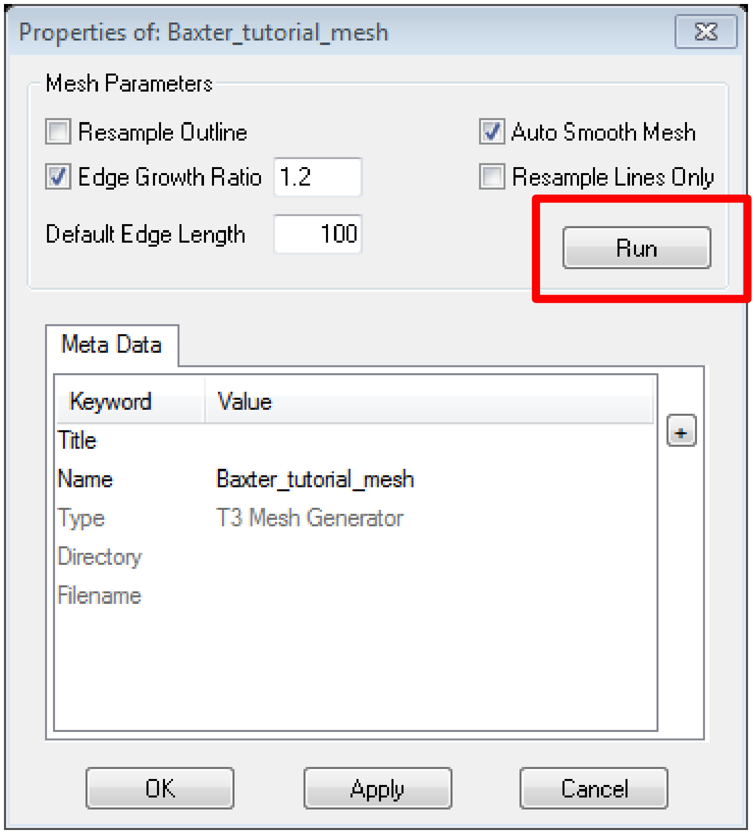


Figure 30: Execution of the mesh generator

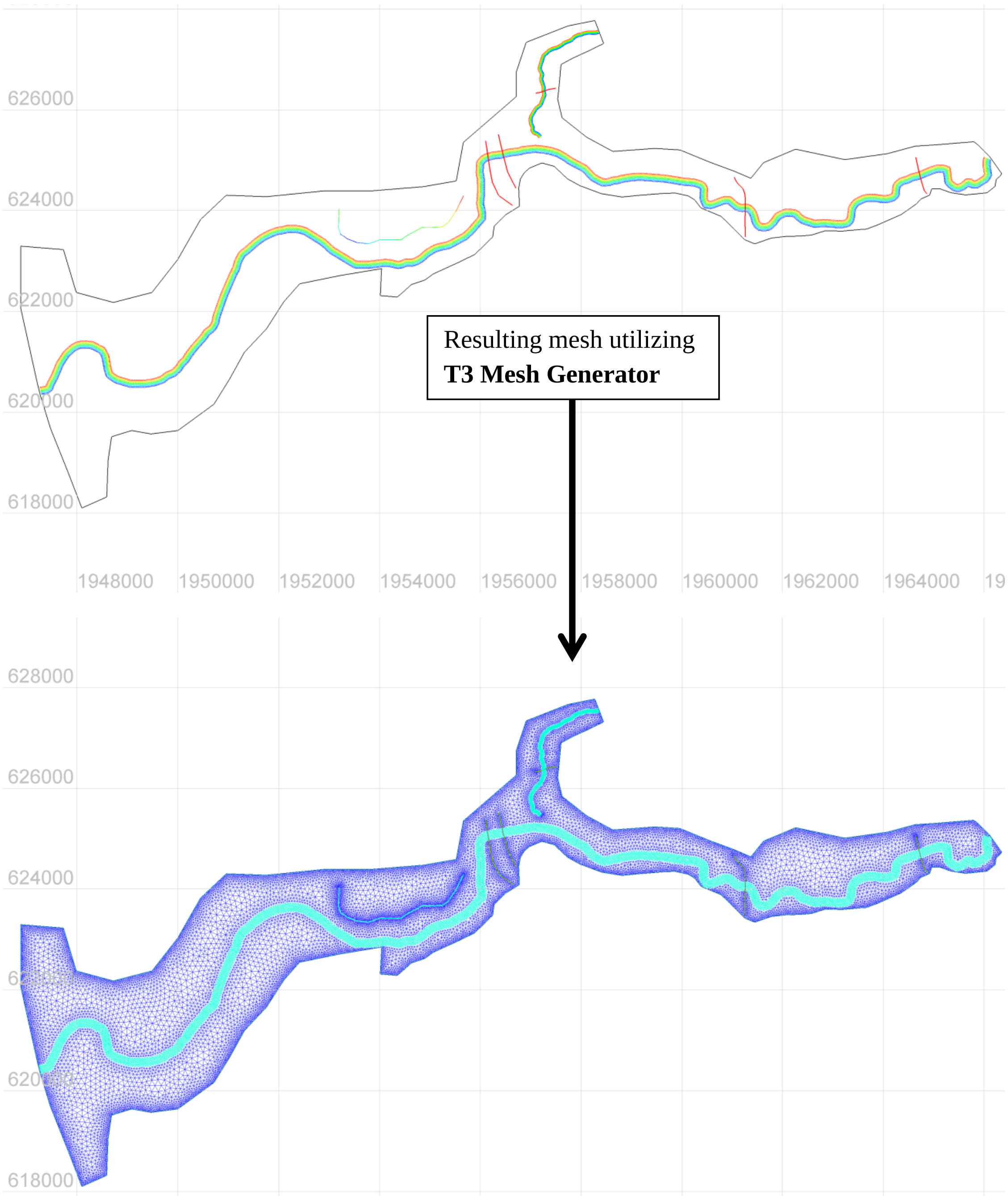


Figure 31: Resulting 2D triangular mesh generated using T3 Mesh Generator

3.4 Interpolate bathymetry to the mesh

Section 3.3 describes the process of creating a two-dimensional triangular mesh to represent the domain extents. It can be noticed that the mesh initially has a constant value assigned to each mesh node (e.g. EL. 0 m). In order to map elevation values to the nodes of a mesh, BlueKenue has a tool called **2D Interpolator...** .

A primary function of BlueKenue is to project, or interpolate the values of one dataset, onto another. In this tutorial, the bathymetric and topographic information are contained in several components, one of which is `baxter_xyz_meters.xyz` . Mapping the bathymetry to a mesh can be completed through the following steps:

1. Create a new 2D Interpolator through **File ¿ New ¿ 2D Interpolator...**
2. Drag and drop all relevant elevation datasets onto the 2D Interpolator object
3. Select the mesh of interest, and go-to **Tools ¿ Map Objects...**
4. Find the **2D Interpolator** object on the list, select and press OK
5. View the resulting interpolation in a 2D-View or 3D-View

Step 1, create a new **2D Interpolator** object through **File ¿ New ¿ 2D Interpolator...** .

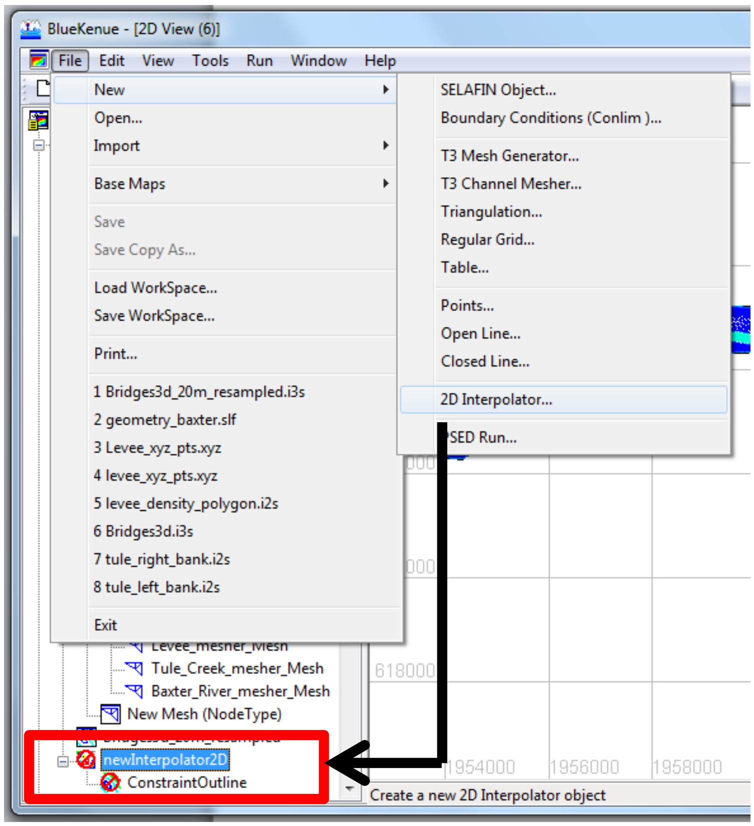


Figure 32: Create a new **2D Interpolator** object

Step 2, drag and drop all relevant elevation datasets onto the **2D Interpolator** object.

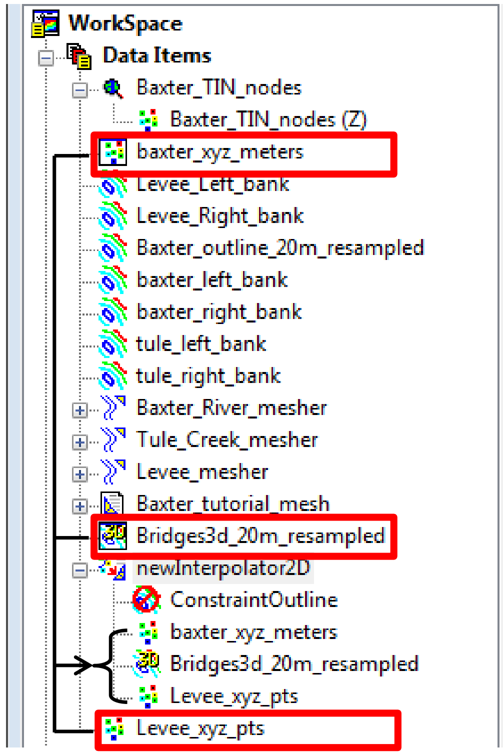


Figure 33: Drag elevation datasets to **2D Interpolator** object

Step 3, select the mesh of interest, and utilize the **Tools ¿ Map Objects...** command to interpolate elevation data to the mesh nodes.

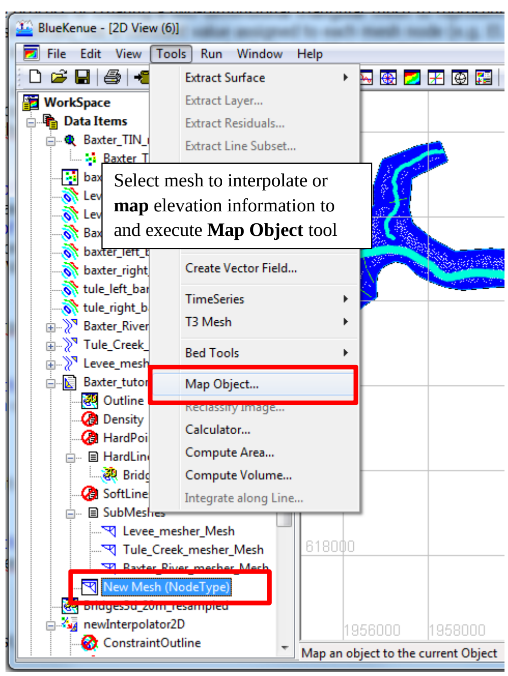


Figure 34: Select the mesh and execute the **Map Objects** command (e.g. **Tools ¿ Map Objects**)

Step 4, find the **2D Interpolator** object on the list, then select, and press OK.

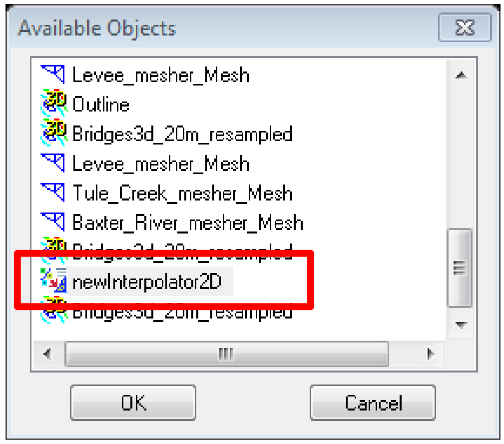


Figure 35: Select the elevation source data to map, or interpolate, onto the selected object (mesh)

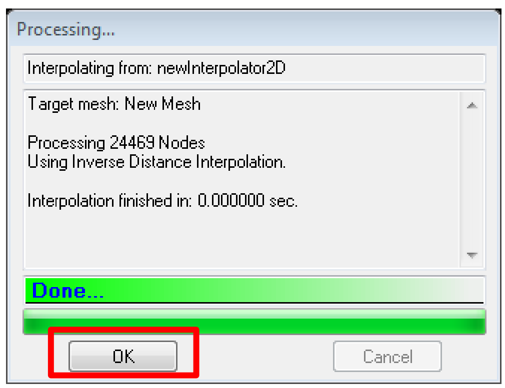


Figure 36: When **Map Object** command is finished, a progress window will appear

Step 5, once the elevation data is mapped to the mesh, drag the mesh object to a **2D-View** or **3D-View** to inspect the interpolation.

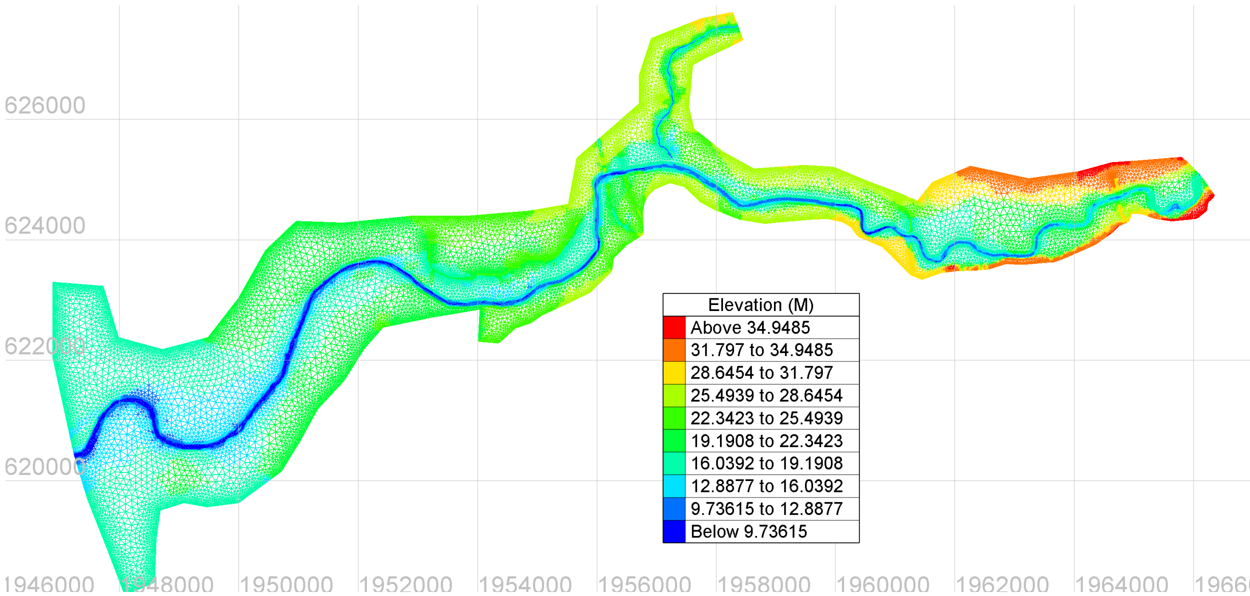


Figure 37: Drag the interpolated mesh onto a **2D View** Object to view the mapped bathymetry

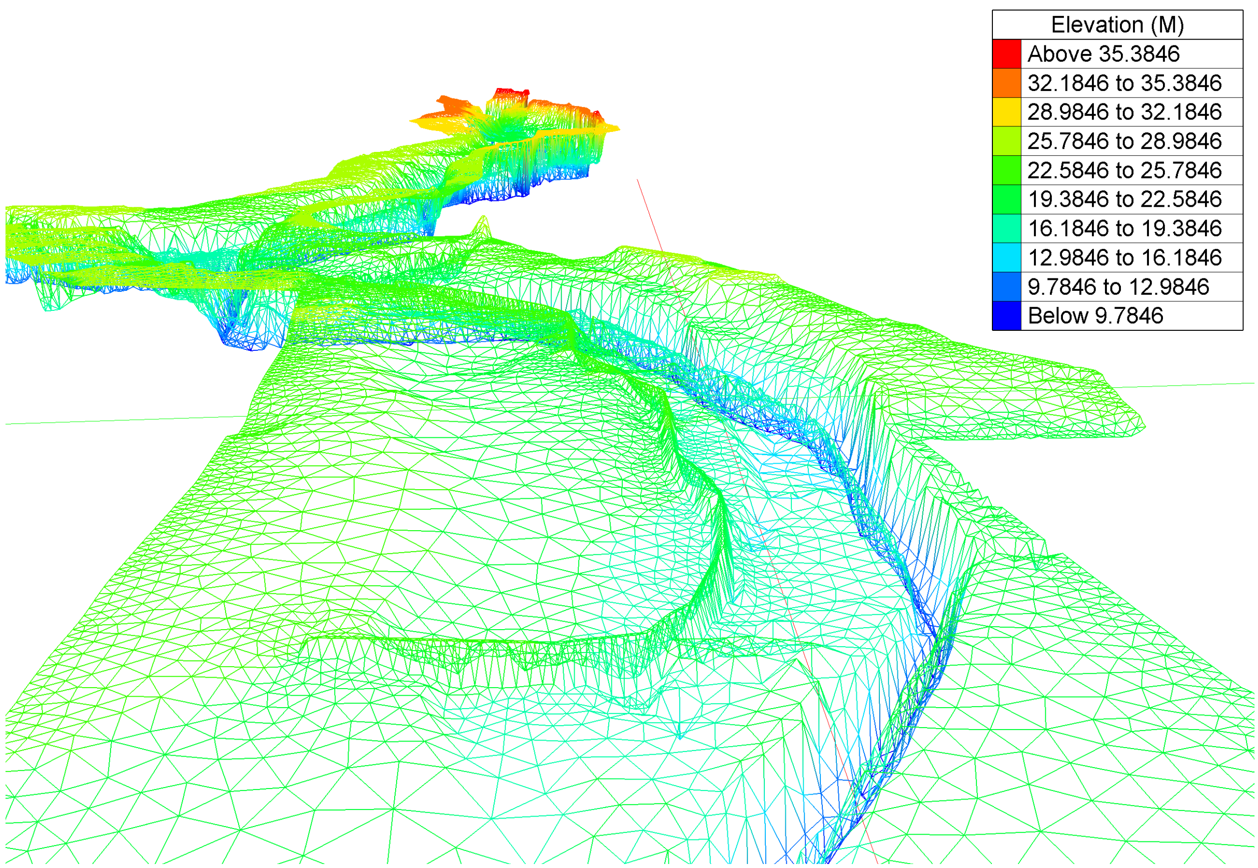


Figure 38: Drag the interpolated mesh onto a **3D View** Object to view the mapped bathymetry

3.4.1 Levee height adjustment

Utilizing the steps outlined in Section 3.4, new levee heights can be imposed and mapped, or interpolated, to the bathymetric mesh. A method to adjust the crest height of the levee involves adjusting the **xyz** point file **levee_xyz.xyz** . This is accomplished utilizing the BlueKenue calculator tool. Select the **levee_xyz.xyz** object, go-to **Tools ¿ Calculator...** , and the xyz dataset can be vertically adjusted. As the calculator directly alters the **xyz** point file, it is recommended to make copies of the levee elevation data file prior adjustment.

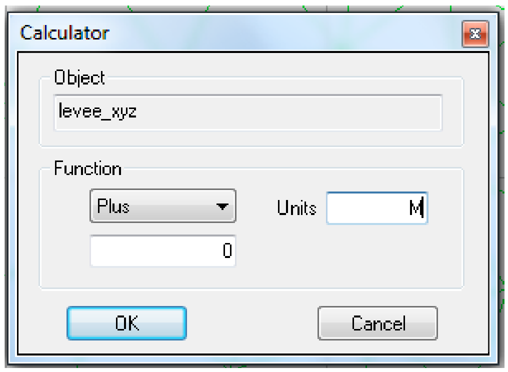


Figure 39: Adjust the levee height prior to interpolating the domain mesh by using the BlueKenue calculator

Figure 40: Apply adjusted levee heights using a **2D Interpolator**

If creating a new levee using the **T3 Channel Mesher** , the following are the steps used to generate **levee_xyz.xyz** used in Section 3.4.

1. Create **T3 Channel Mesh** representing levee
2. Use **2D Interpolator** to map original levee elevations (from Exported Shape File)
3. Save mapped levee mesh as a text file (*.t3s)
4. Open levee mesh file in text editor, delete mesh face info, leaving xyz vertex info at each levee mesh node
5. Save as a xyz file, and include in **2D Interpolator** used for combined mesh to incorporate changes exactly

3.4.2 Viewing cross-sections of the mesh

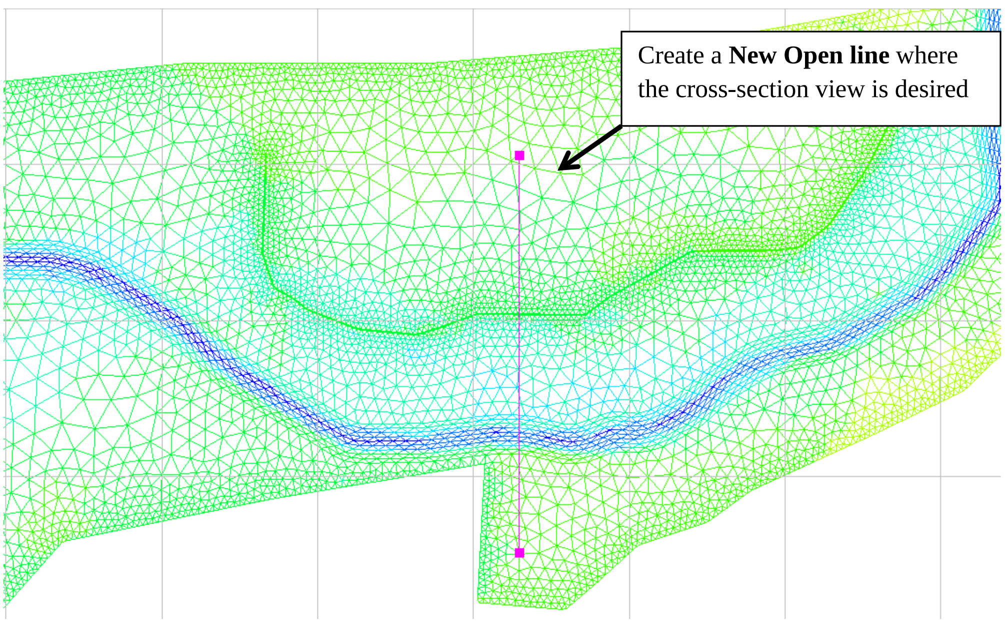


Figure 41: Create a new line where cross-section view is desired

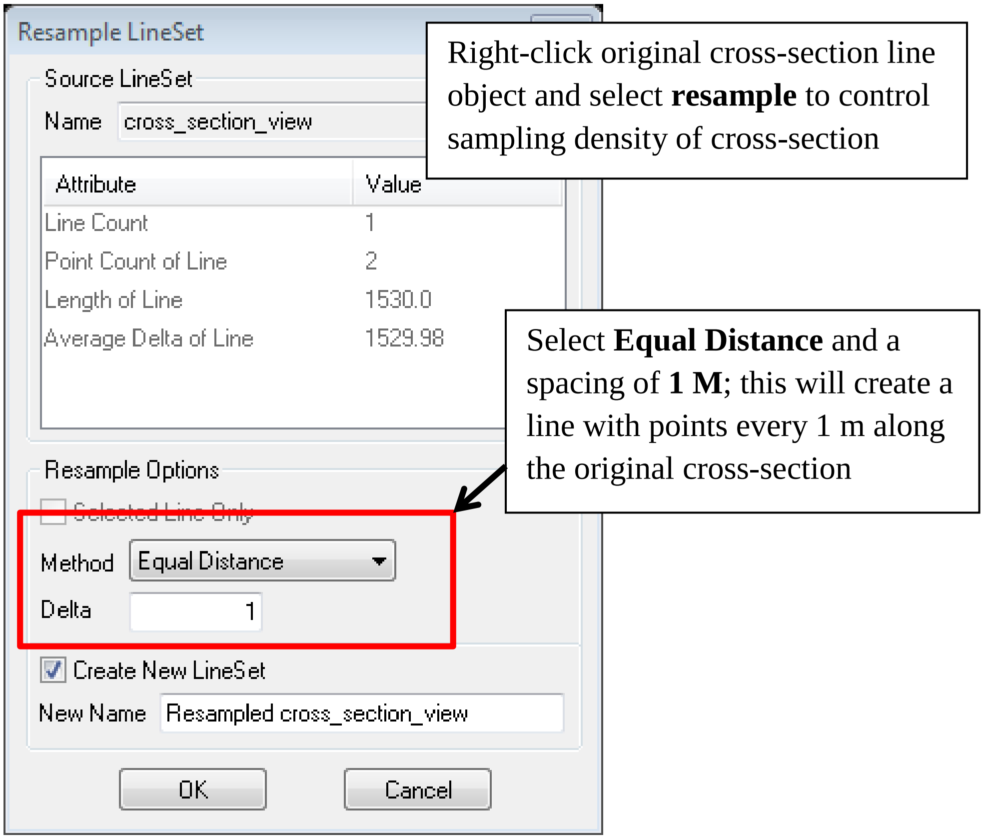


Figure 42: Resample the newline to increase number of sampling points

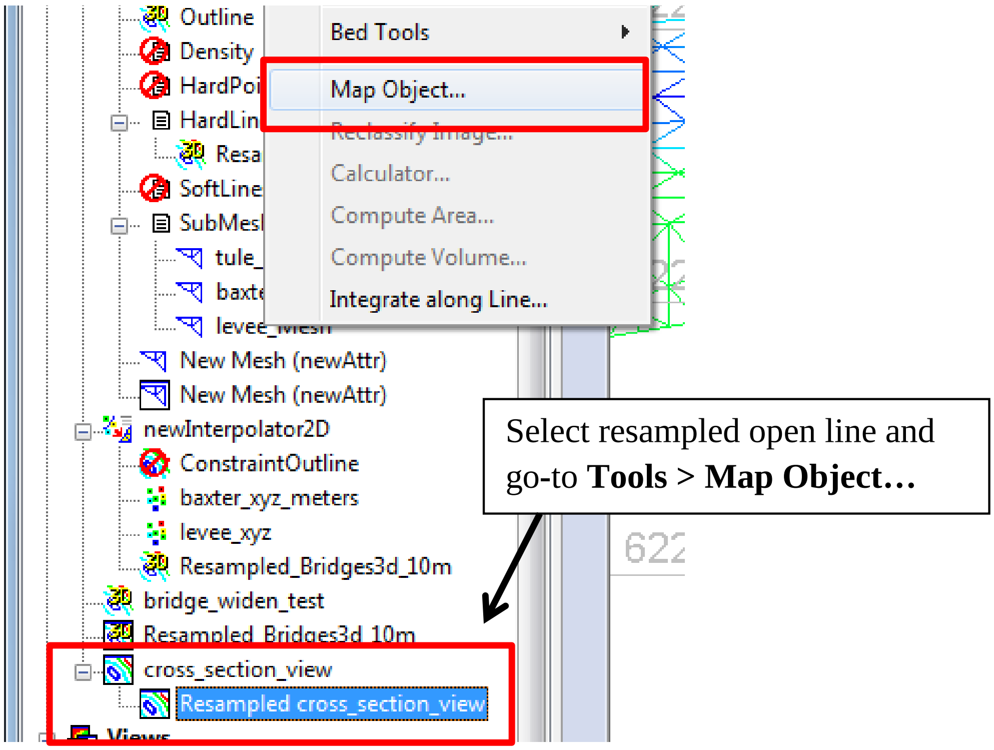


Figure 43: Once resampled, use command **Map Object** to choose the source surface to sample from

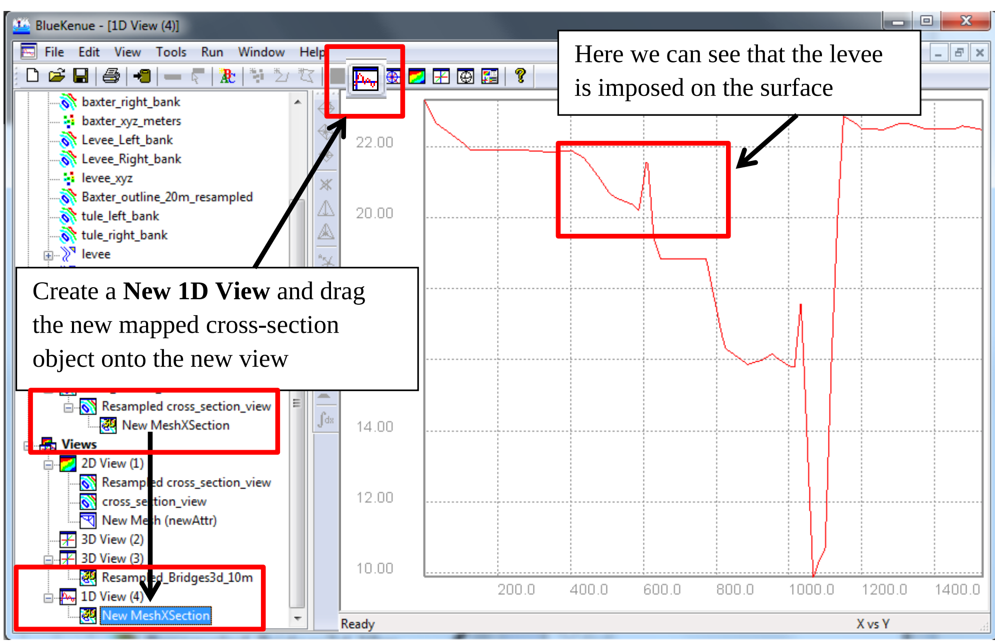


Figure 44: Drag the mapped, resampled line onto a new **1D View** to see the cross-section

4 TELEMAC-2D input file generation

A TELEMAC-2D hydrodynamics simulation requires a minimum of three input files:

1. Geometry File (*.slf)
2. Boundary Conditions File (*.cli)
3. TELEMAC-2D simulation parameters file (*.cas)

Items 1 and 2 are prepared in BlueKenue, and FUDAA Pre-Processor will be used to set up the TELEMAC-2D parameters file.

4.1 BlueKenue

4.1.1 TELEMAC-2D Geometry File

The geometry file for TELEMAC-2D can be represented in several formats. Using BlueKenue, a formatted binary file will be created that includes the bathmetric mesh. The process of creating a geometry file is achieved through the following steps:

1. Create a **New SELAFIN File** and rename (e.g. `geometry_baxter`) Step 1 - Page 47
2. Add the bathymetric mesh as a **New Variable** specified as **BOTTOM** Step 2 - Page 49
3. Map the bathymetry to the new child-object mesh, **BOTTOM** Step 3 - Page 50
4. Save geometry file to the TELEMAC-2D simulation directory Step 4 - Page 51

Step 1 , create new Selafin File utilizing either the **New SELAFIN Object** icon or through **File ¿ New ¿ SELAFIN Object** (i.e. Figure 45).

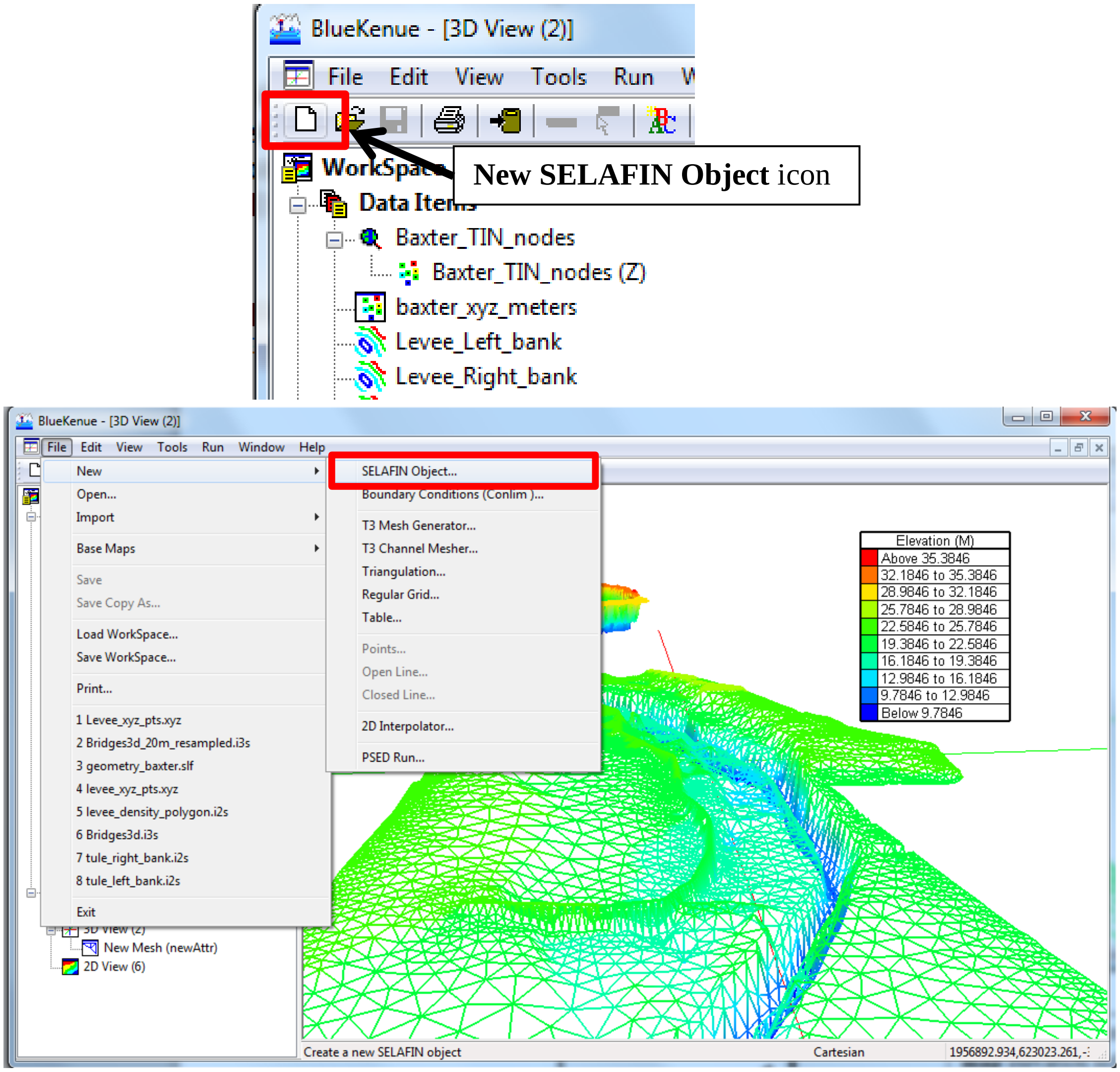


Figure 45: Create a new geometry object using either the toolbar icon or File command

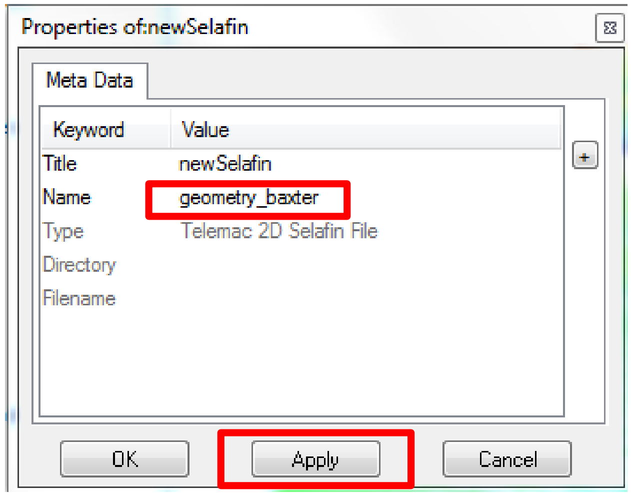


Figure 46: Double-click the **New Selafin** object, rename, and select **Apply**

Step 2 , add the bathymetric mesh as a **New Variable** specified as **BOTTOM** as shown in Figures 47 and 48.

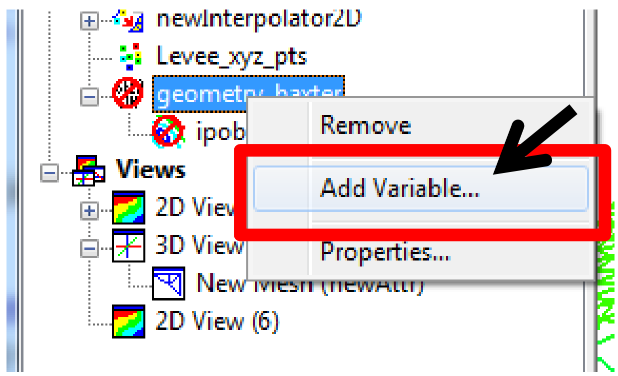


Figure 47: Right-click the geometry object and select **Add Variable**

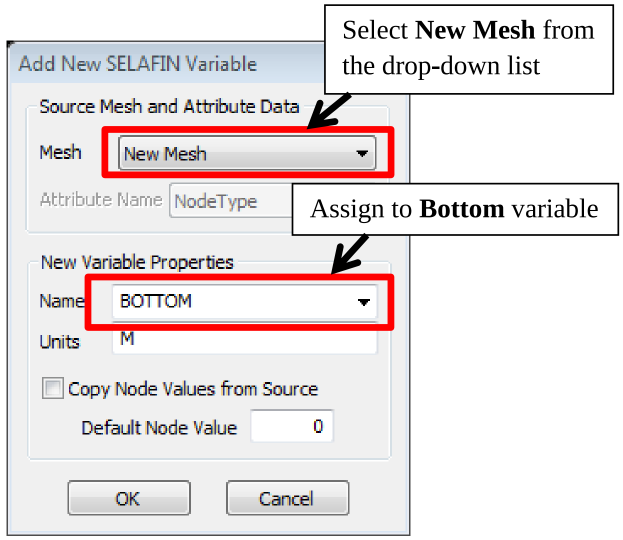


Figure 48: Select the computational domain mesh and add as a **BOTTOM** variable

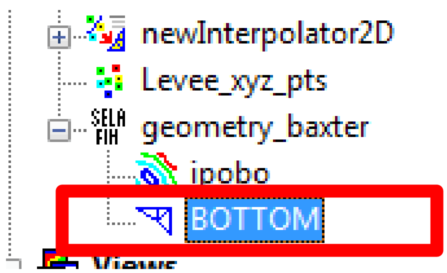


Figure 49: Geometry object with **BOTTOM** variable added

Step 3 , map the bathymetry to the new object **BOTTOM** as shown in section 3.4. Again, inspect that the elevation data is represented on the geometry **BOTTOM** child-object in a **2D View** or **3D View** window. An example of a non-interpolated mesh and correctly interpolated mesh are shown below in Figure 52.

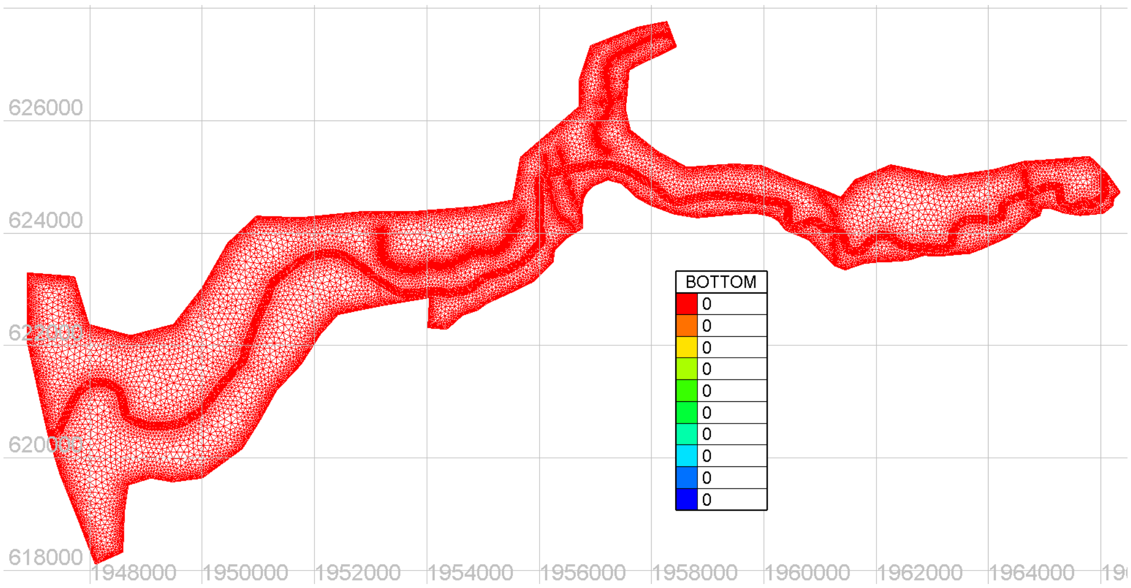


Figure 50: Non-interpolated / Mapped **BOTTOM** - BAD

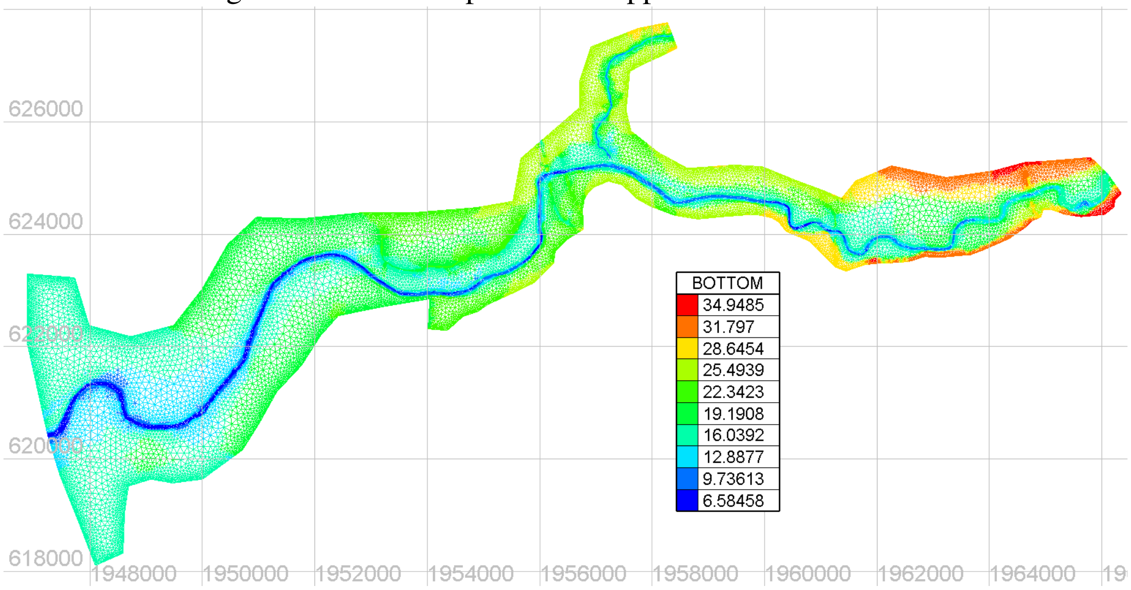


Figure 51: Correctly interpolated / Mapped **BOTTOM** - GOOD

Figure 52: Comparison of bad versus good **BOTTOM** interpolation

Step 4 , save the **geometry** file under the `TELEMAC_simulation_files` directory.

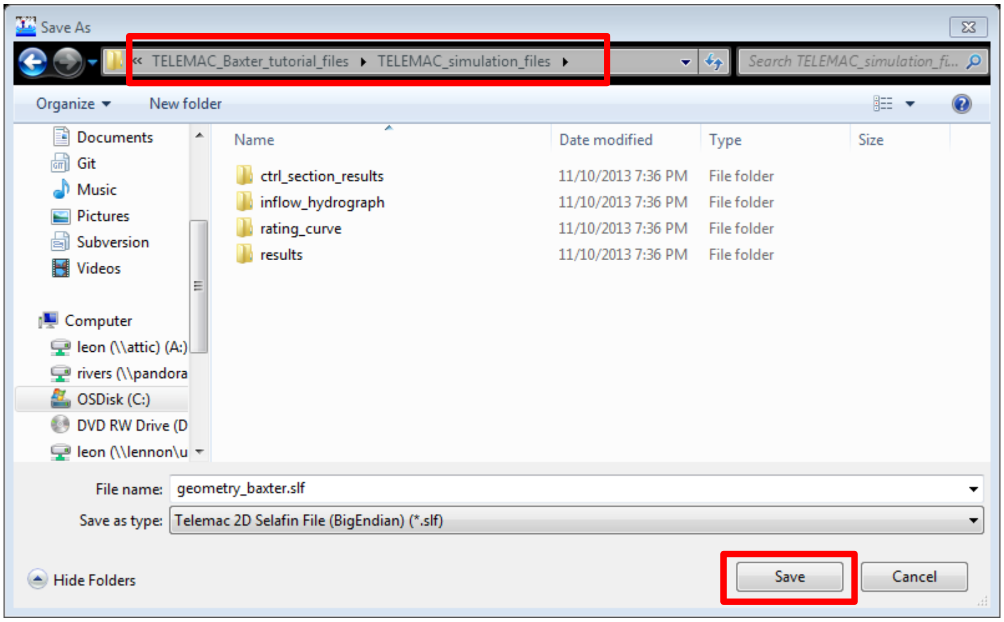


Figure 53: Save **geometry** object under the TELEMAC simulation directory

4.1.2 TELEMAC-2D Boundary Condition File

1. Create a **New ¿ Boundary conditions (conlim)** file for **BOTTOM** — Step 1 - Page 52
2. Rename the boundary conditions file (e.g. `bc_baxter`) — Step 2 - Page 54
3. Drag the boundary conditions object to a **2D-View** object — Step 3 - Page 55
4. Prescribe boundary conditions at the inflow and outflow domain locations — Step 4 - Page 56
5. Save boundary conditions file to the TELEMAC-2D simulation directory — Step 5 - Page 60
6. Save boundary conditions child-object to the T2D simulation directory — Step 6 - Page 60

Step 1 , create a **New ¿ Boundary conditions (conlim)** file for the geometry child-object **Bottom**

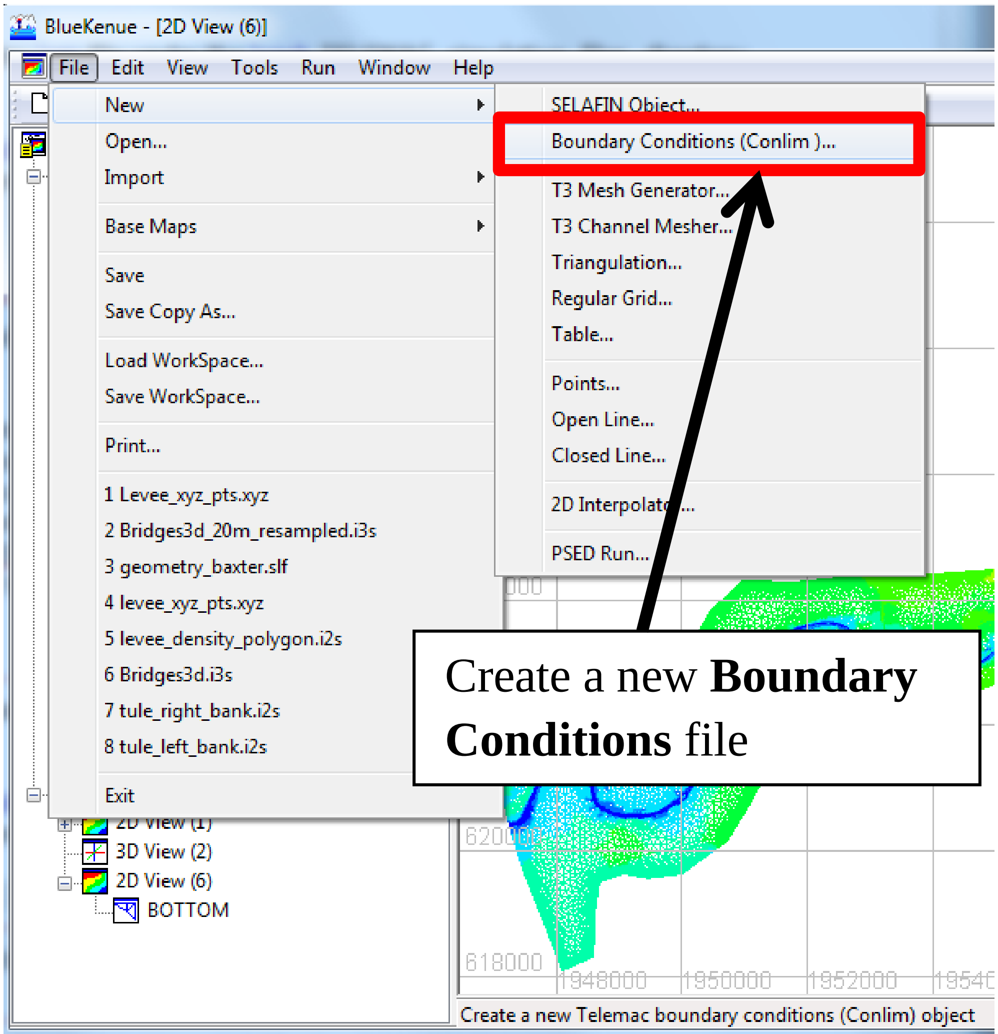


Figure 54: Create a new **Boundary Conditions** file for the **BOTTOM** object

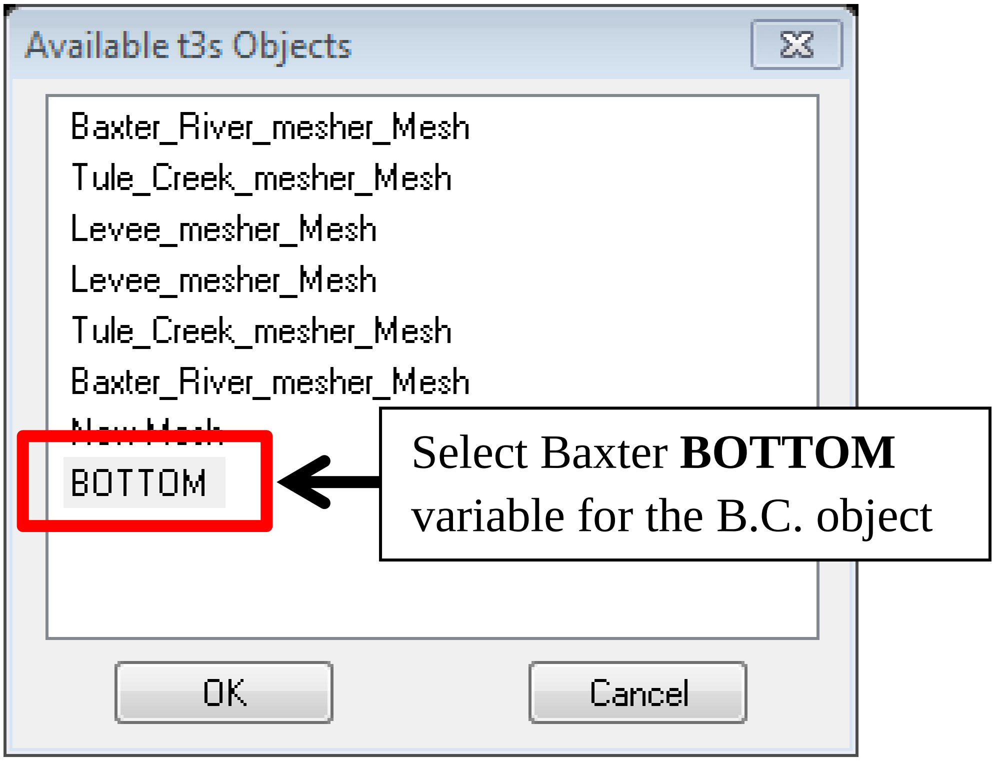


Figure 55: Assign the new **Boundary Conditions** file to the **BOTTOM** object

Step 2 , rename the boundary conditions file (e.g. `bc_baxter`). This can be done by double-clicking the **BC BOTTOM** object and changing the **Name** under the **Meta Data** tab (Figure 56).

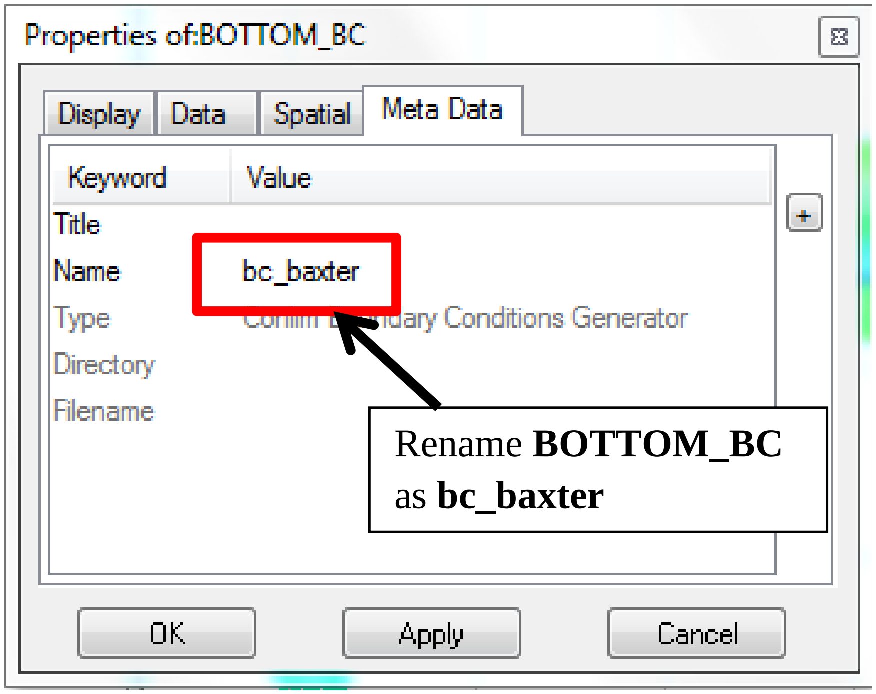


Figure 56: Rename the **BOTTOM BC** object for ease of identification

Step 3 , drag the boundary conditions object to a **2D-View** object.

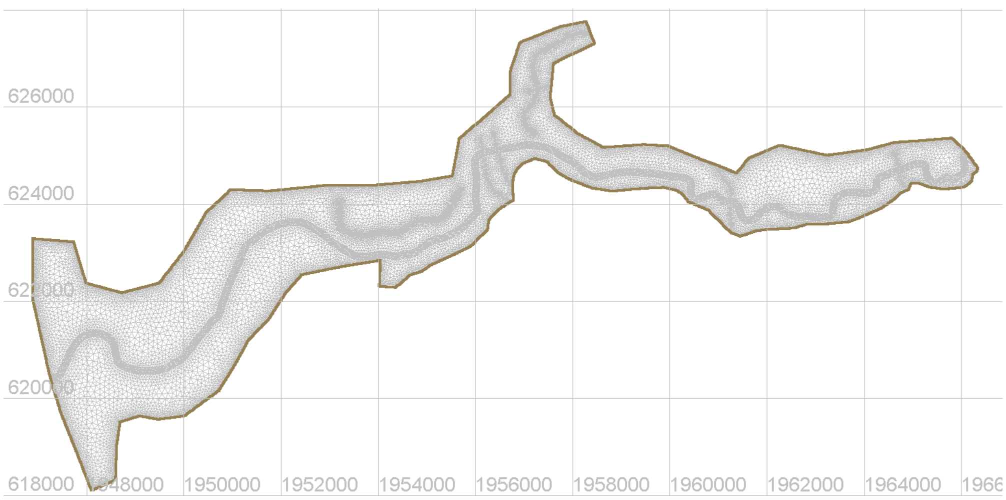


Figure 57: All boundary nodes are assigned as **Closed boundary (wall)** by default

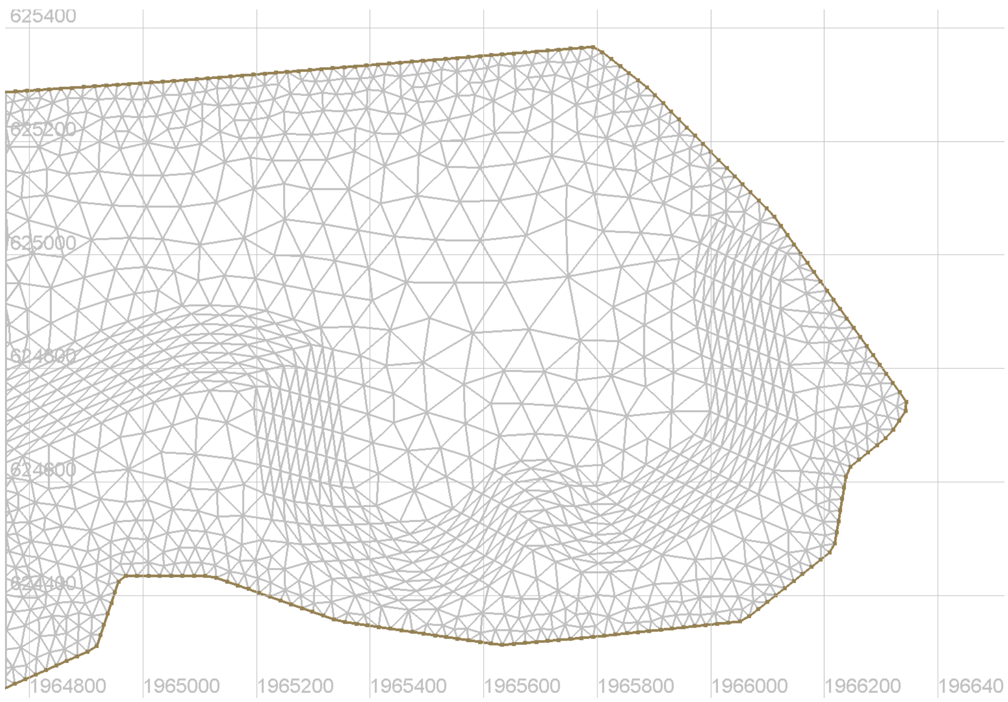


Figure 58: Zoomed in view of the default **bc baxter** object

Step 4 , prescribe boundary conditions at the inflow and outflow domain locations

Each node along the boundary has a specific code and color representing what type of boundary it is. For example, **Closed boundary (wall)** nodes are brown, **Open boundary with prescribed Q** nodes are blue, and **Open boundary with prescribed H** nodes are green, where Q is volumetric flowrate, and H is water depth. The full list of boundary nodes are shown below in Figure 59

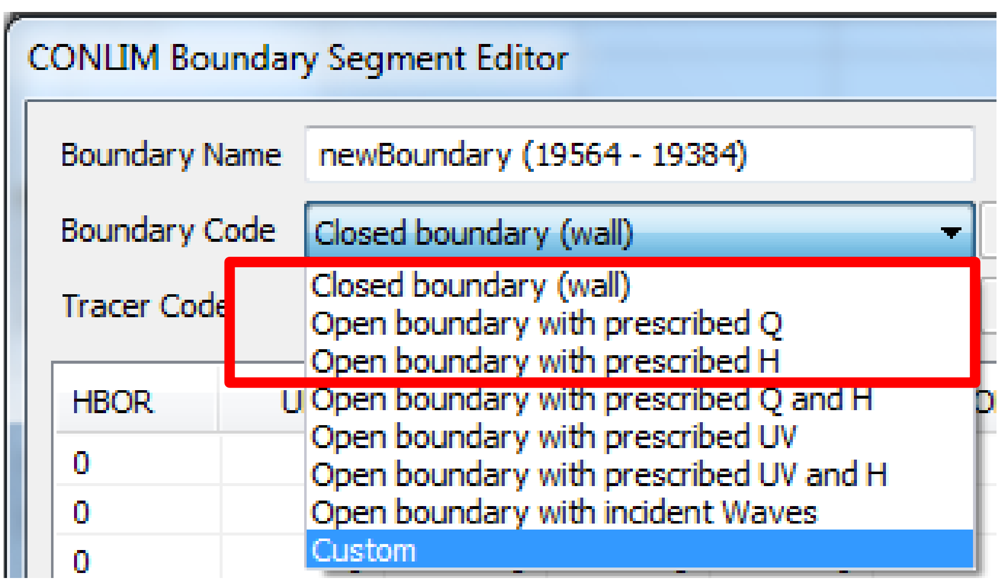


Figure 59: TELEMAC-2D Boundary condition node types

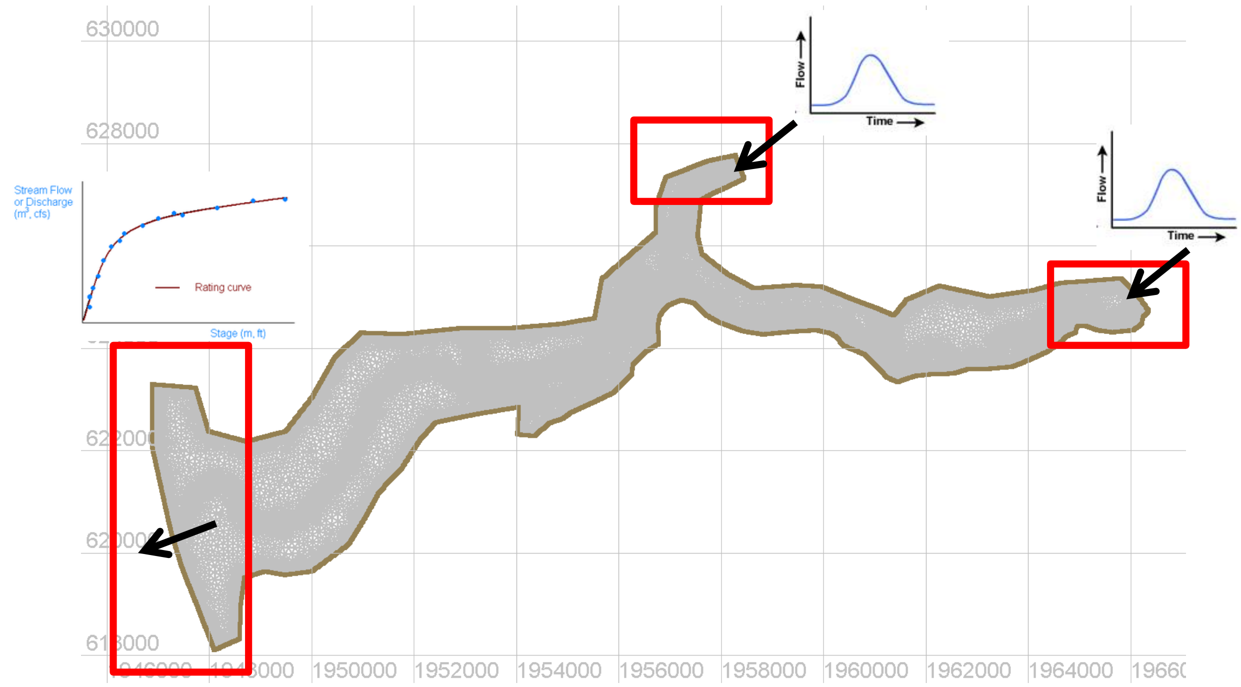


Figure 60: Boundary condition overview for the Baxter River tutorial

Updating and adding boundary condition segments is performed on the **bc baxter** object within a **2D View** window. To create a segment of nodes, representing a single boundary condition, **select the starting edge node** of the domain where the boundary condition begins, then holding the **Shift** key, **select the end node** where the boundary condition segment ends. For example, Figure 64 shows the delineation of the upstream Baxter River boundary condition as an **Open boundary with prescribed Q** . The process of prescribing a boundary condition nodes is illustrated in Figures 61 through 64 below.

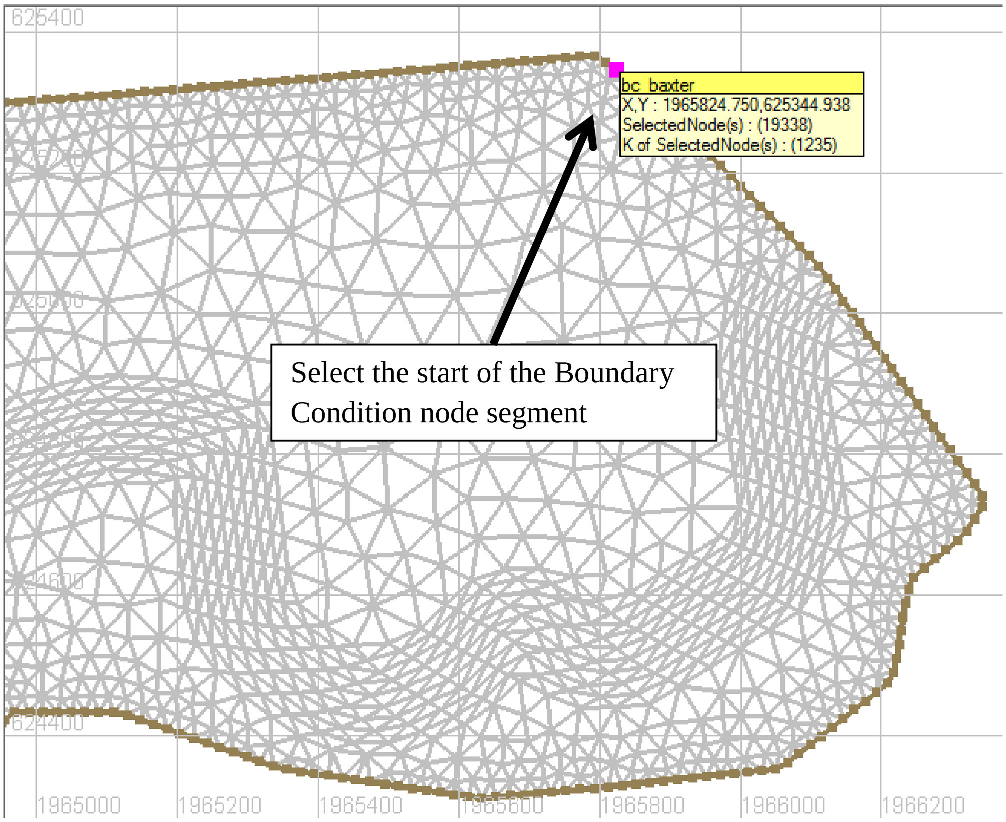


Figure 61: Select the starting edge node where the boundary condition segment begins

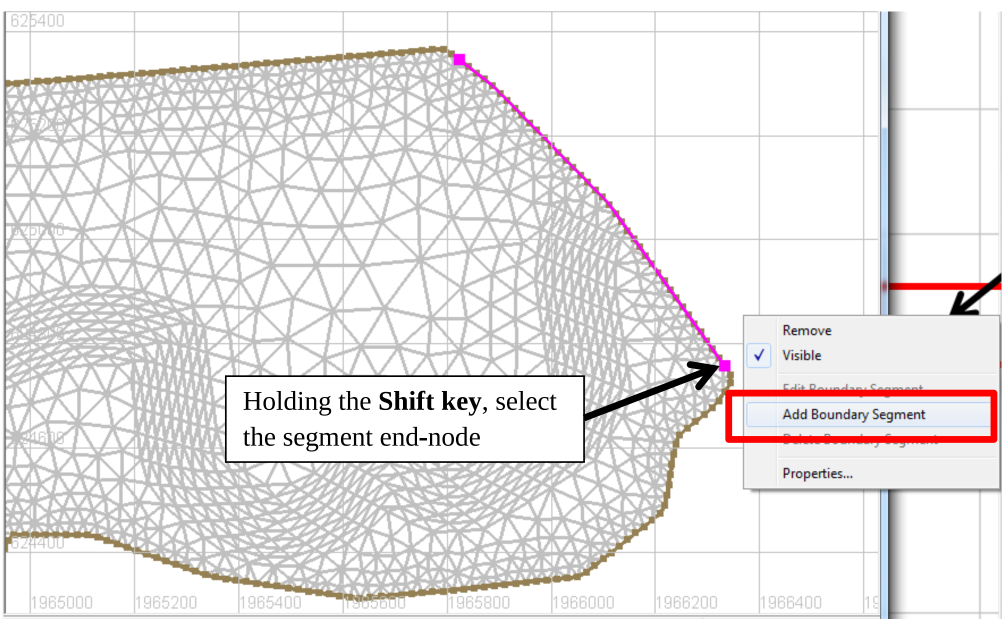


Figure 62: Select the end node of the segment and right-click to **Add Boundary Segment**

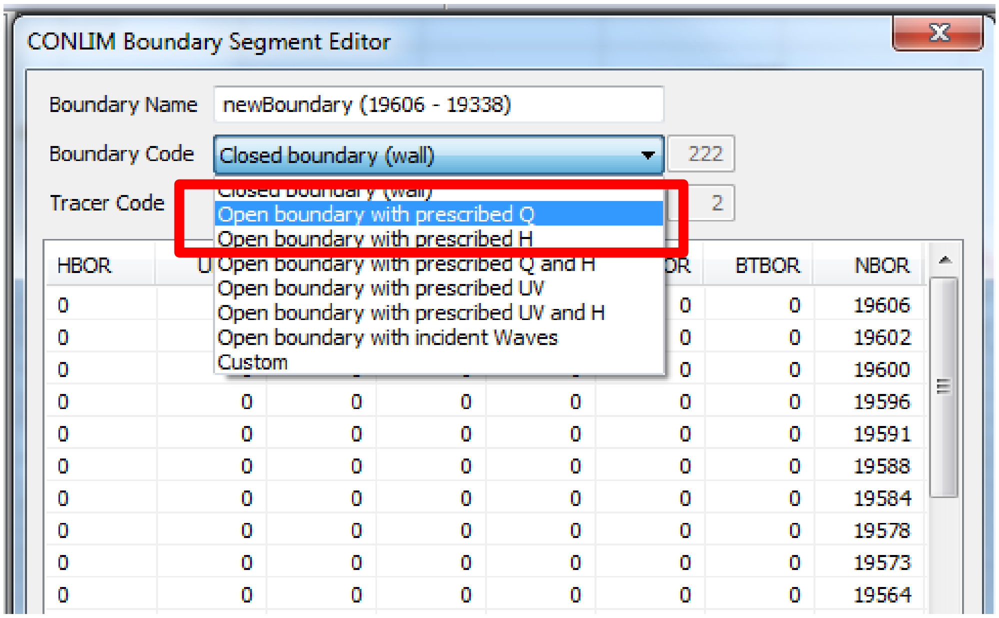


Figure 63: Apply an **Open boundary with prescribed Q** BC code to the Baxter River upstream reach

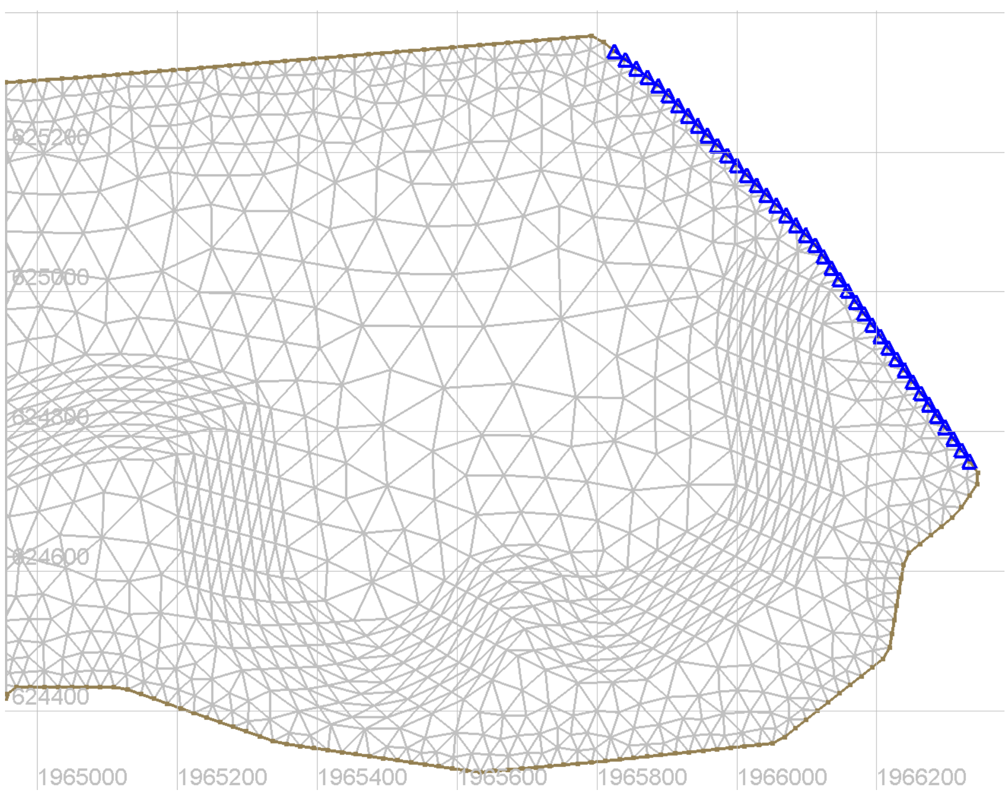


Figure 64: Boundary condition applied to **bc baxter** object

Step 5 and Step 6 , save boundary conditions file and boundary conditions child-object (.cli) to the TELEMAC-2D simulation directory

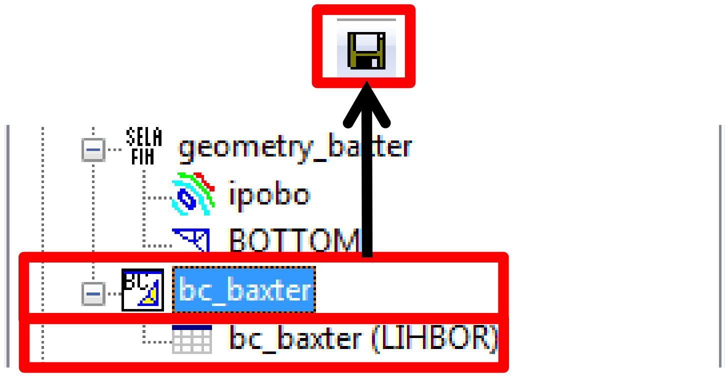


Figure 65: Save **both** boundary condition objects to the simulation directory

4.2 FUDAA Pre-pro

For this tutorial, the TELEMAC-2D parameter files will be provided, however creating a new hydraulic project is detailed below.

4.2.1 TELEMAC-2D parameters file (.cas)

1. Open FUDAA Pre-Processor using executable (*.jar)
2. Launch the hydraulic project editor,
3. Create a new TELEMAC-2D hydraulic project,
4. Specify the TELEMAC-2D files to be used:
   (a) boundary conditions
   (b) geometry
   (c) results filename, etc.
5. Save the hydraulic project,

Creating a new TELEMAC-2D project is illustrated below in Figures 66 through 74

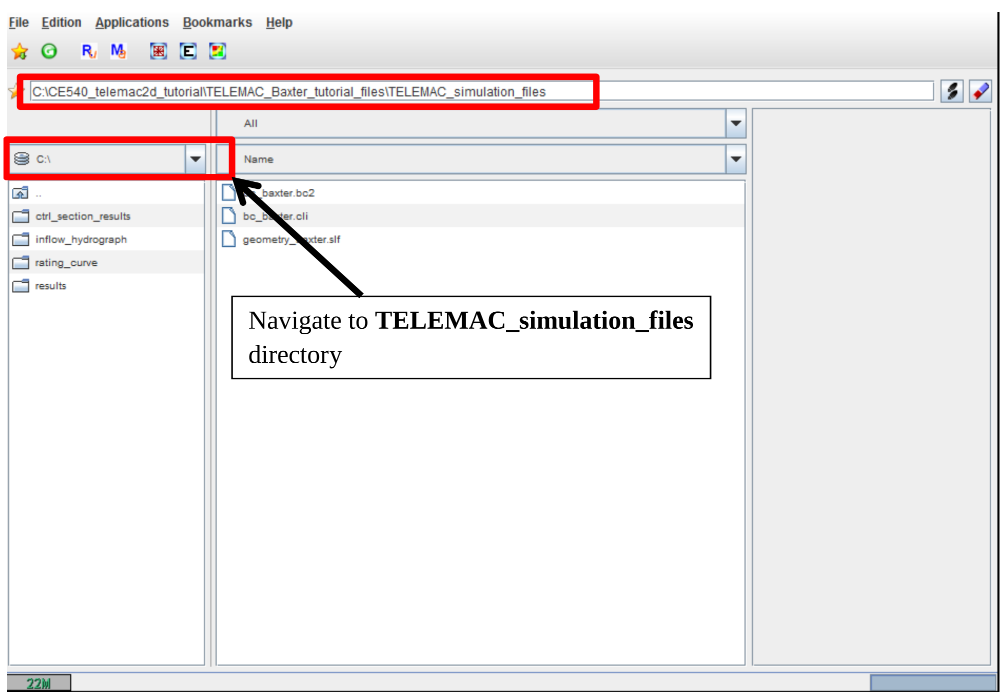
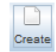


Figure 66: FUDAA Supervisor initialized screen

If creating a new hydraulic project, launch the **hydraulic project editor** by selecting the icon, . Create a new TELEMAC project using the **Create...** icon, 

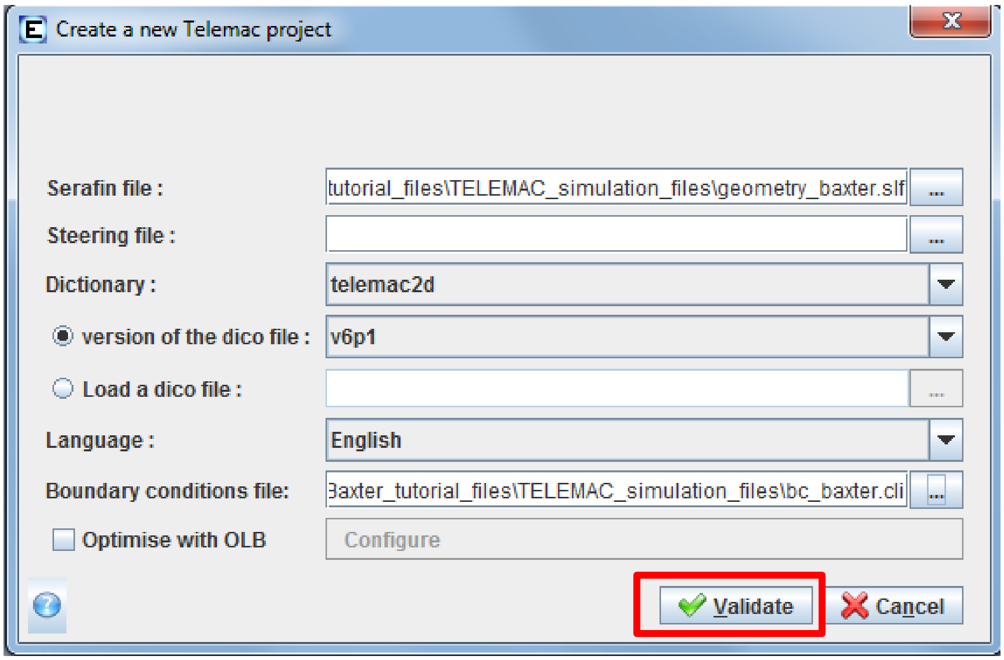


Figure 67: Populate this window with necessary hydraulic project files

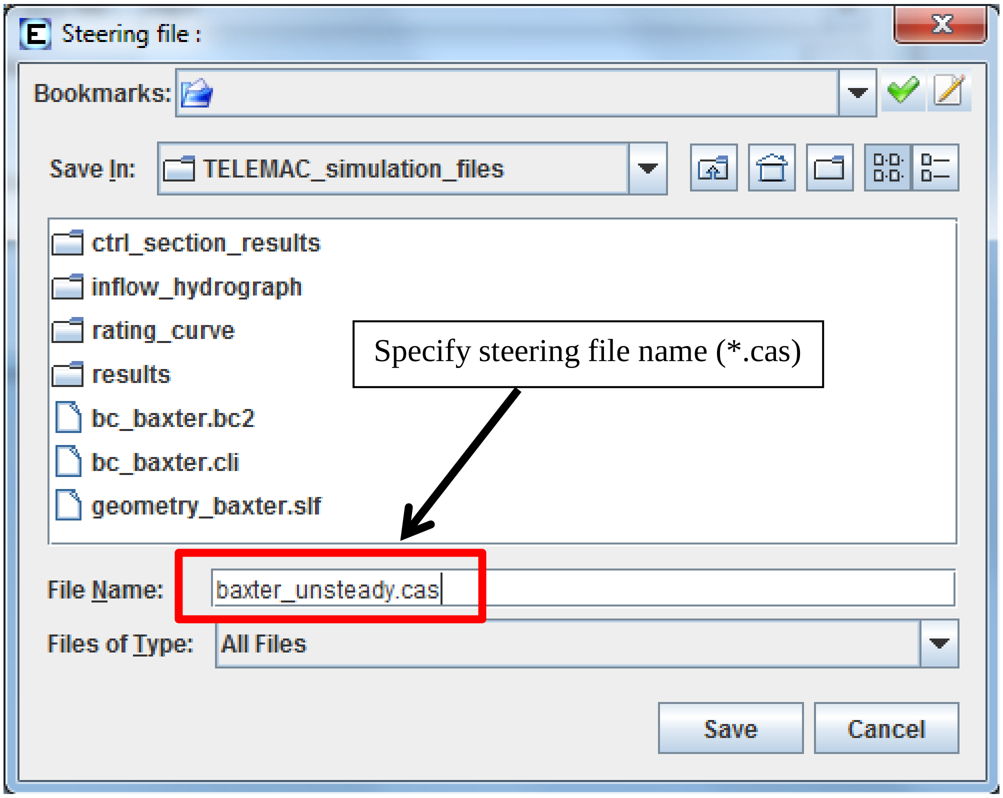


Figure 68: Under the **Steering file** field, specify the name of your parameters file (e.g. **baxter_unsteady.cas**

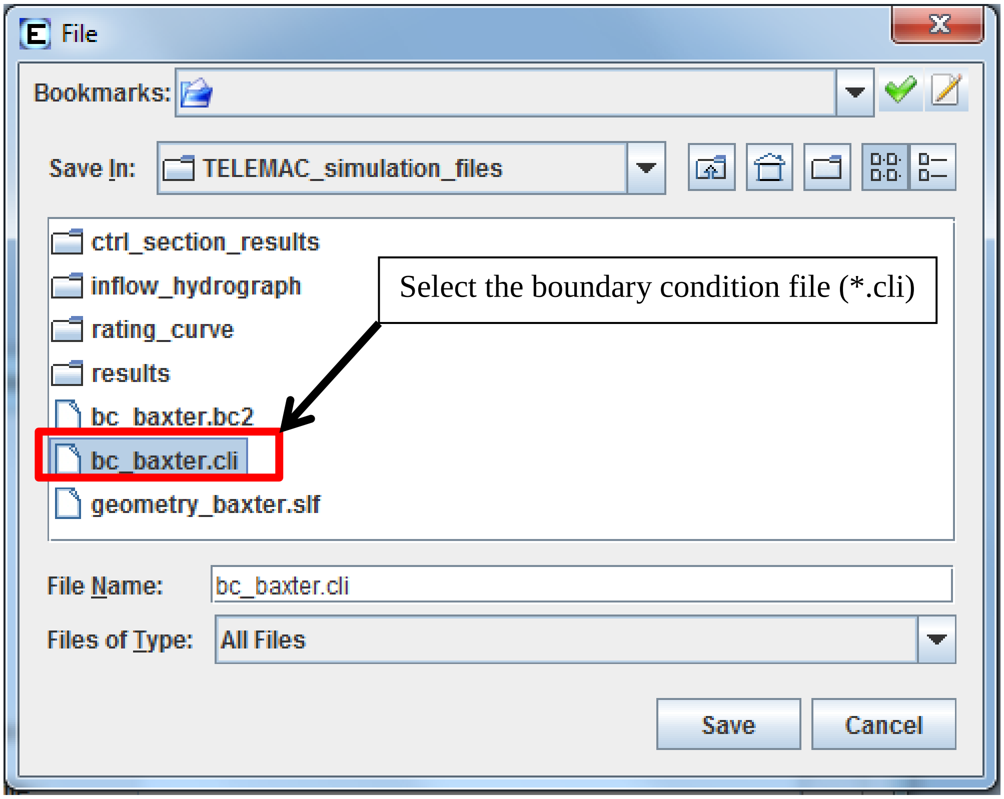


Figure 69: Under the **Boundary conditions** field, specify the file location of bounary conditions file, **bc_baxter.cli**

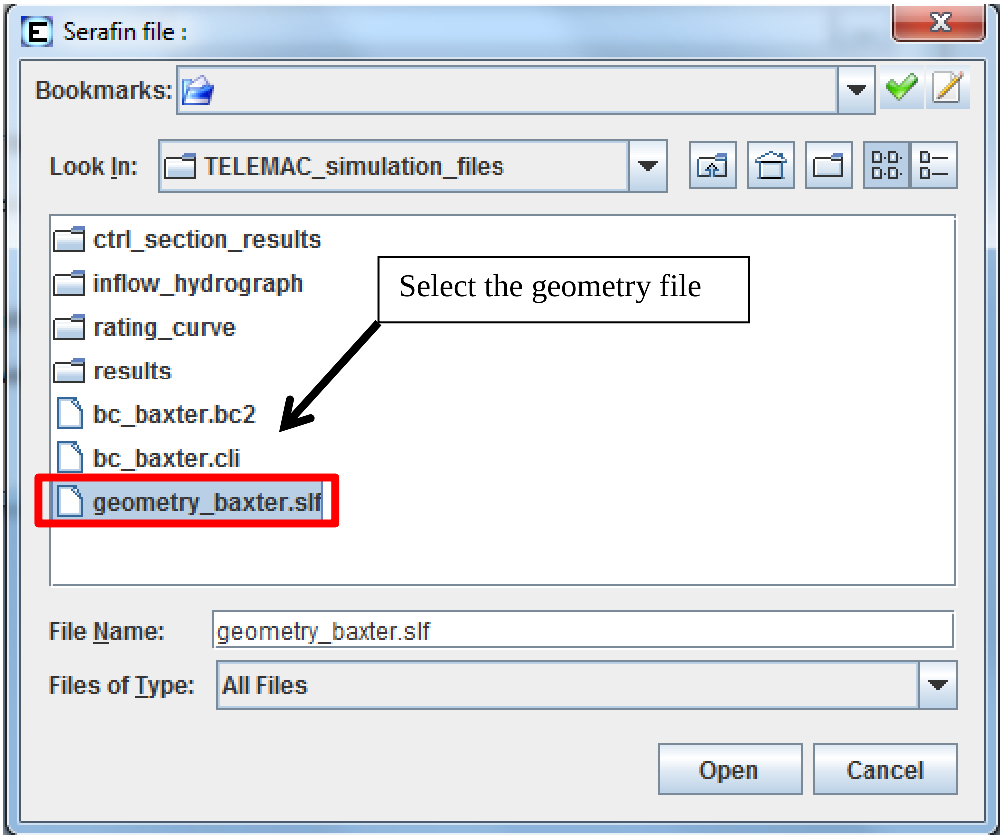


Figure 70: Under the **Serafin file** field (i.e. TELEMAC geometry file format), specify the location of **geometry_baxter.slf**

Once the core files have been specified, select **Validate** (e.g. Figure 67) to begin specifying TELEMAC-2D parameters.

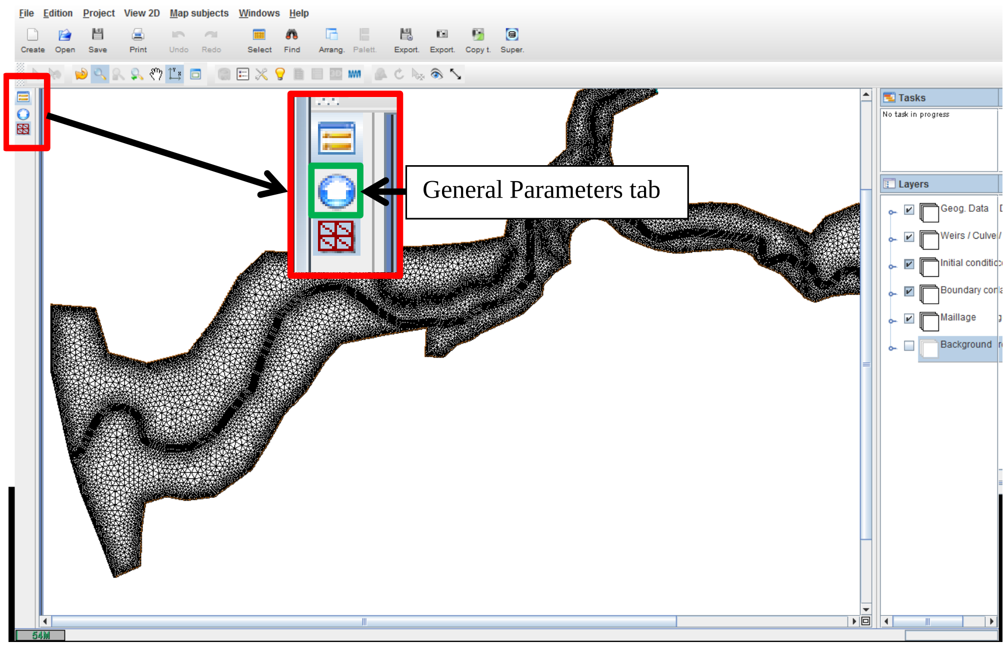


Figure 71: Select the **General Parameters** tab to access the project parameters

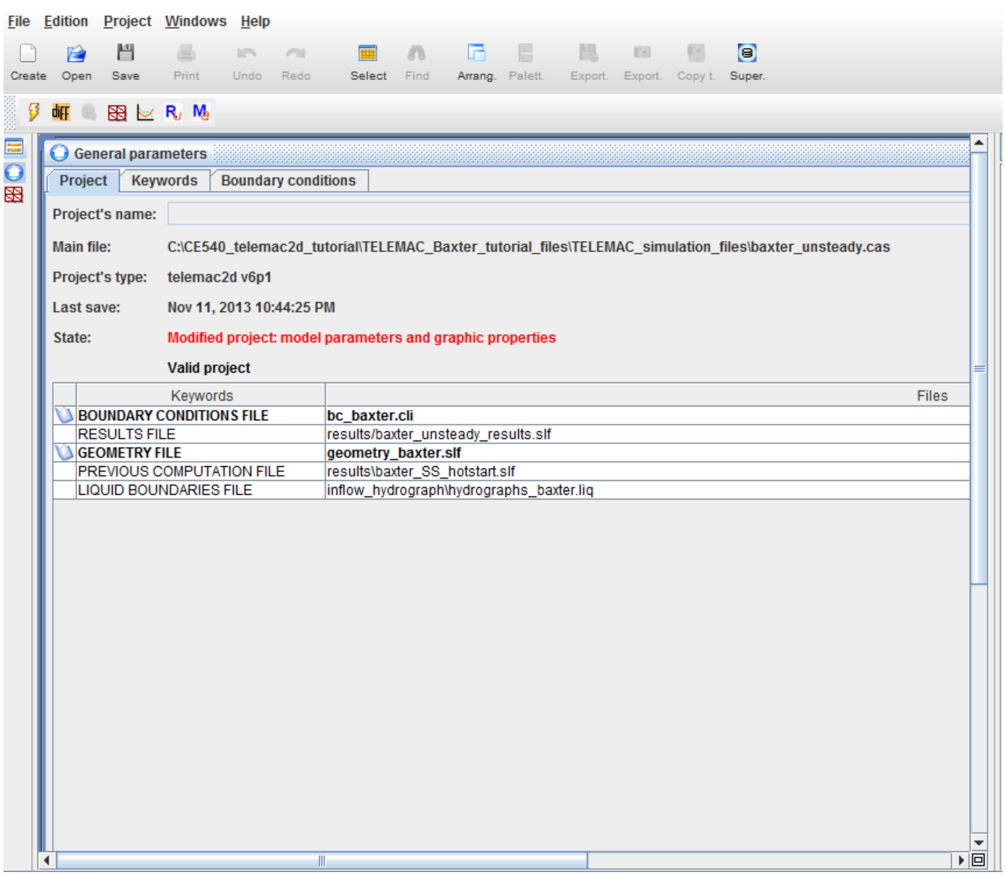


Figure 72: Under **General Parameters** the files associated with the project are shown

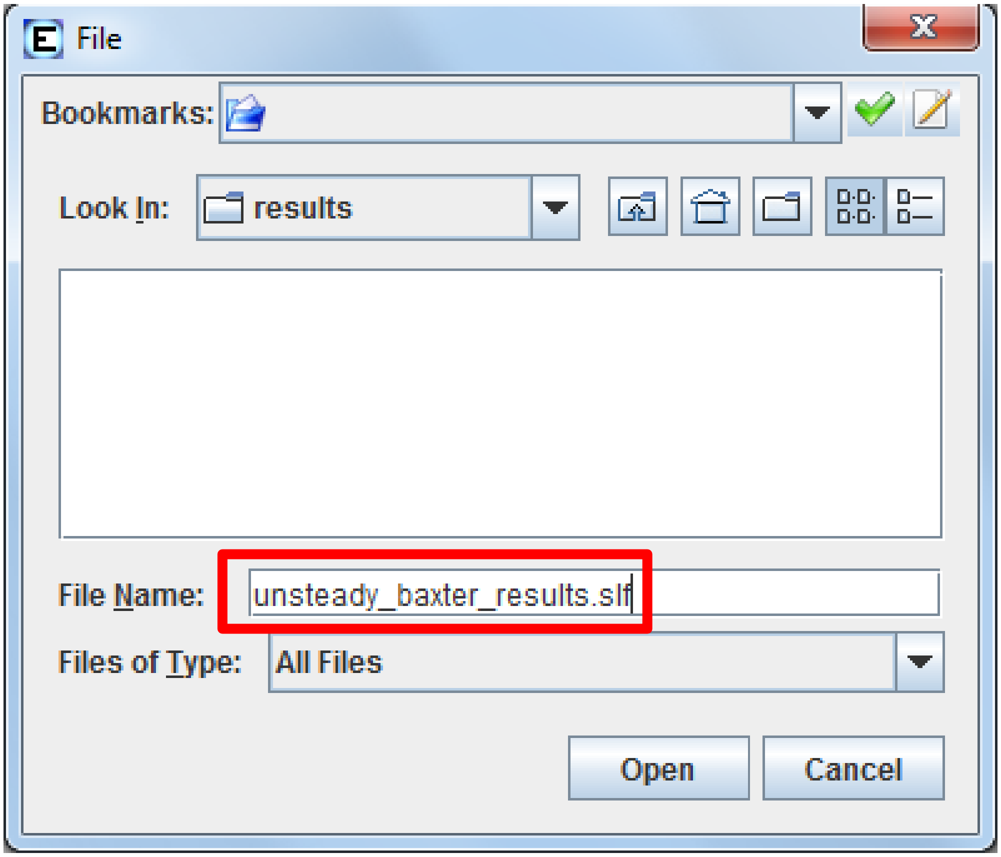


Figure 73: Select the **Results File** field to specify where to write the T2D results

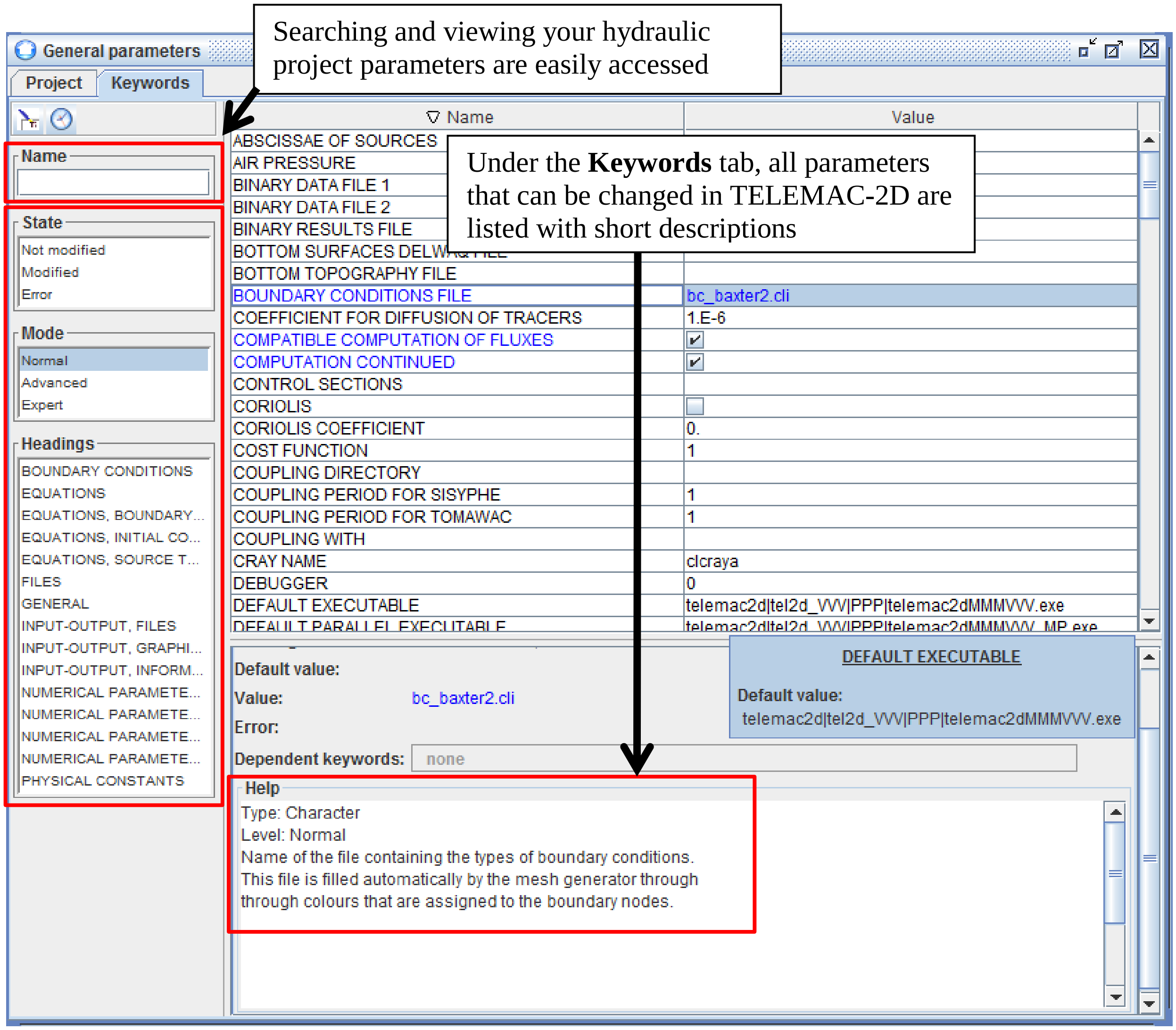


Figure 74: The **Keywords** tab is used for viewing and understanding the hydraulic project parameters

4.2.2 Example unsteady parameters file (.cas)

To view the example TELEMAC-2D parameters file, initialize FUDAA Pre-pro and launch the **hydraulic project editor** , E . Next, open the **baxter_unsteady.cas** file from the **TELEMAC_simulation_files** directory to load and view the project parameters.

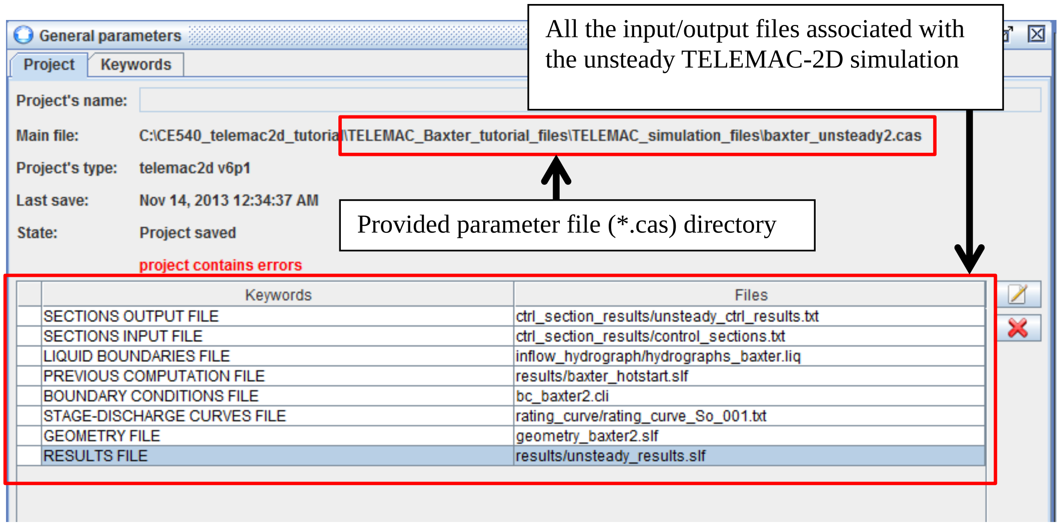


Figure 75: Project files for example TELEMAC-2D unsteady parameters file

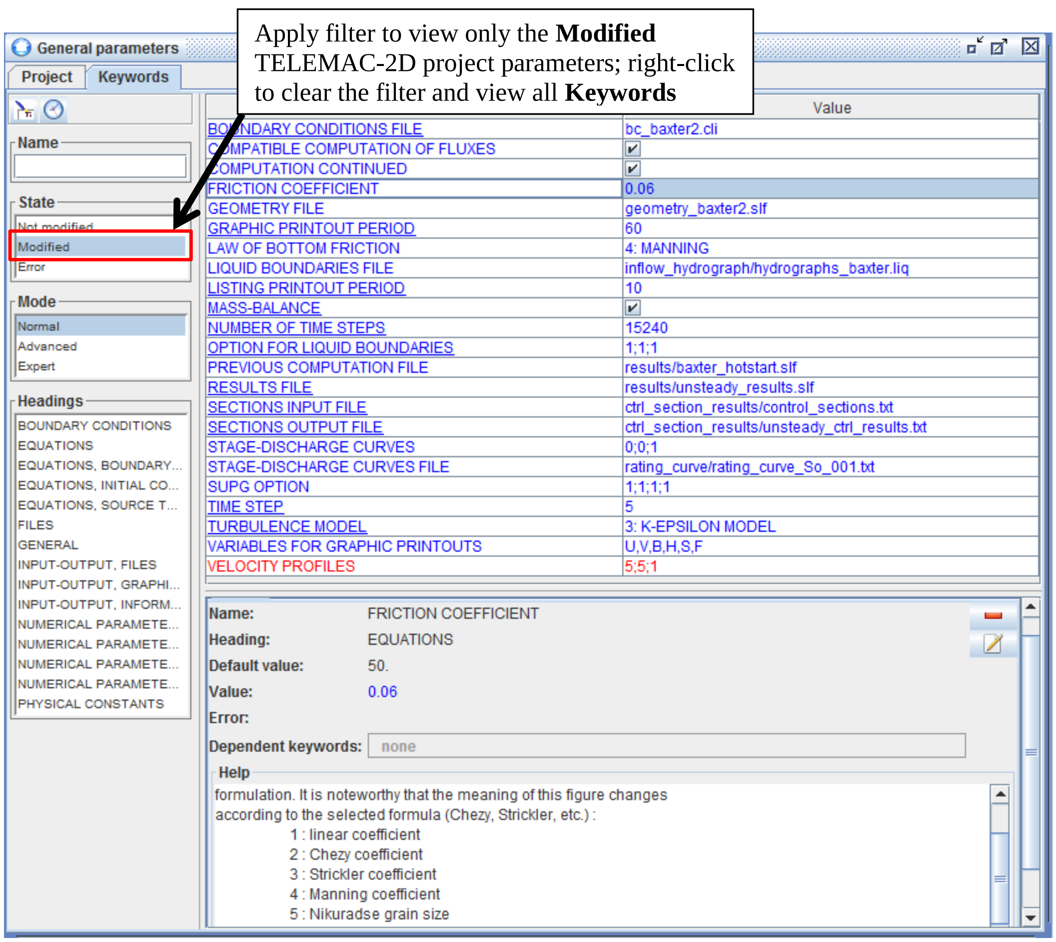
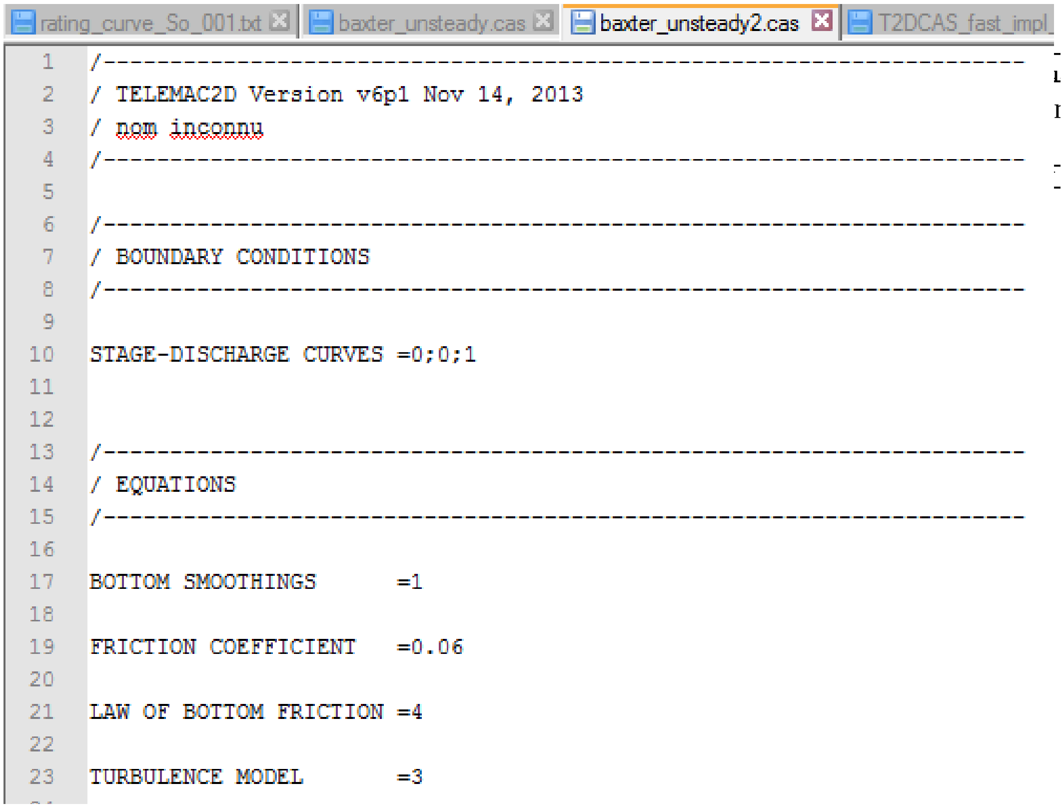


Figure 76: Using the filters under the **Keywords** tab can help view pertinent parameters to your project quickly




```
/---------------------------------------------------------------------
/ TELEMAC2D Version v6p1 Nov 14, 2013
/ nom inconnu
/---------------------------------------------------------------------

/---------------------------------------------------------------------
/ BOUNDARY CONDITIONS
/---------------------------------------------------------------------

STAGE-DISCHARGE CURVES =0;0;1


/---------------------------------------------------------------------
/ EQUATIONS
/---------------------------------------------------------------------

BOTTOM SMOOTHINGS      =1

FRICTION COEFFICIENT   =0.06

LAW OF BOTTOM FRICTION =4

TURBULENCE MODEL       =3
```

Figure 77: Once changes are saved from FUDAA, the parameters file is well organized and ready for the TELEMAC-2D simulation

5 Running TELEMAC-2D simulation

1. Open DOS command window
2. Test that **Python** and **gfortran** is working
3. Change working directory to TELEMAC-2D simulation directory
4. Execute TELEMAC-2D simulation

5.1 TELEMAC-2D from the DOS Command Prompt

The following figures (78 to 85) illustrate the process of opening a new DOS command prompt, testing Python and gfortran installations, and executing TELEMAC-2D.

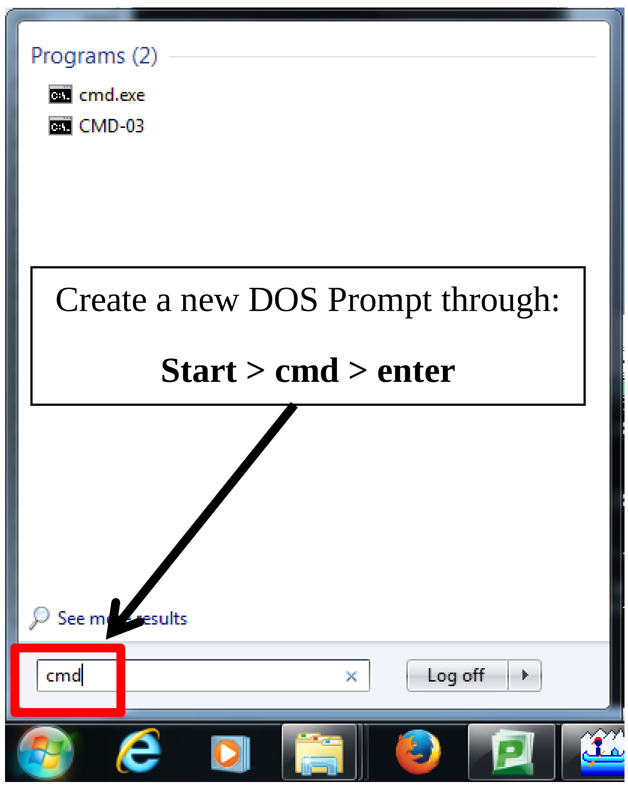


Figure 78: initialize new DOS command prompt

Figure 79: Test Python installation

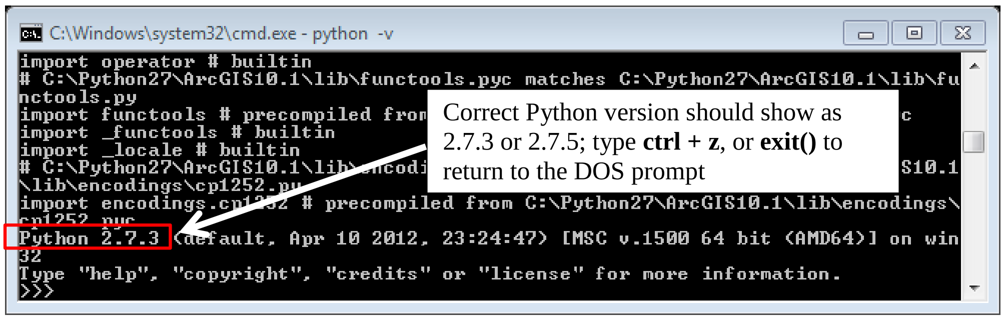


Figure 80: Check Python version, should be 2.7.3 or 2.7.5

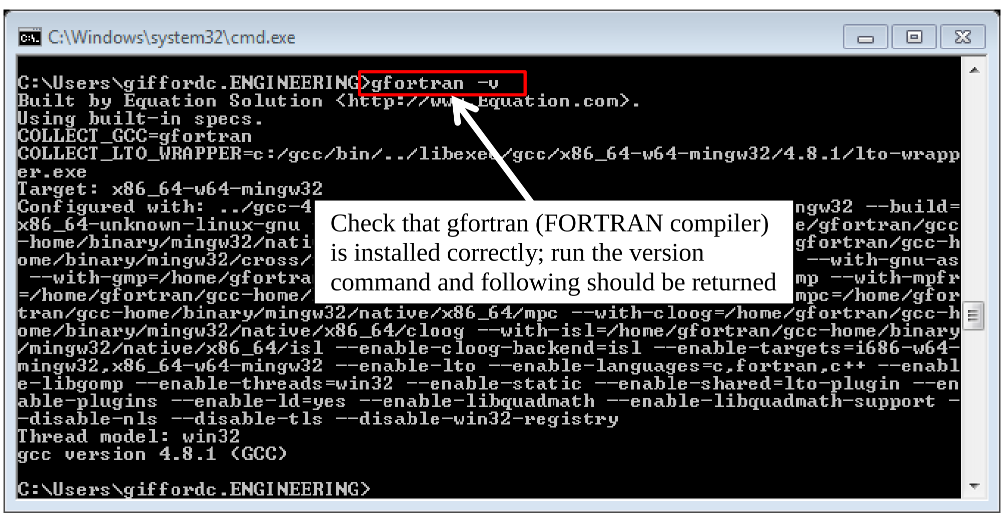


Figure 81: Test gfortran installation

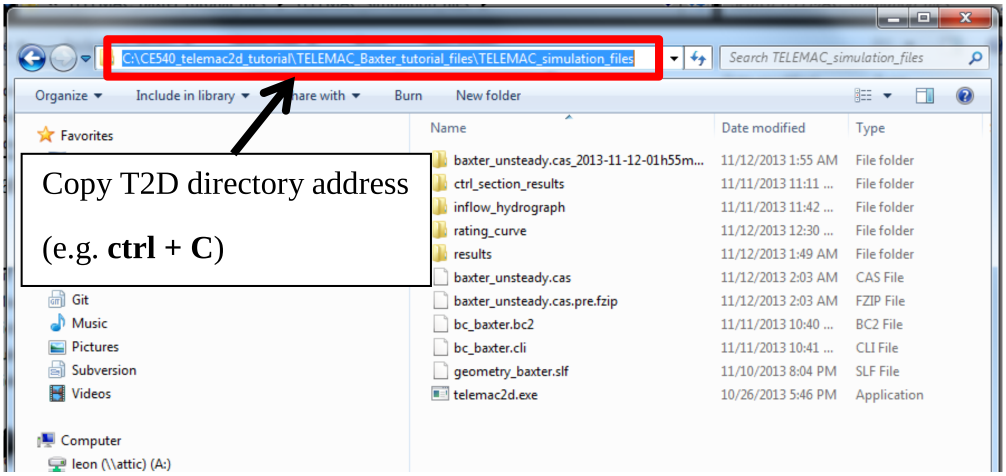


Figure 82: Copy the full-path directory address to the **TELEMAC_simulation_files** folder

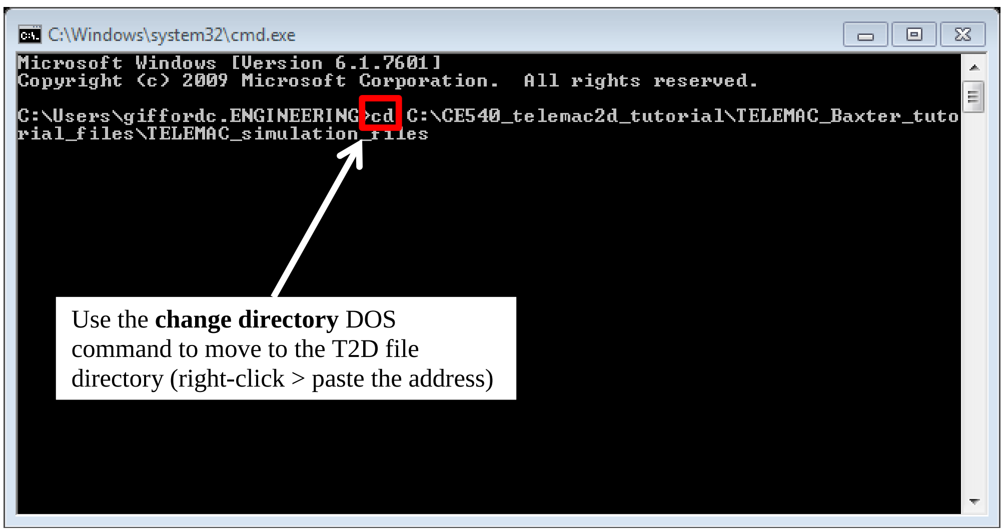


Figure 83: Using the change-directory DOS command, **cd** , change the directory to the location of the **TELEMAC_simulation_files** folder

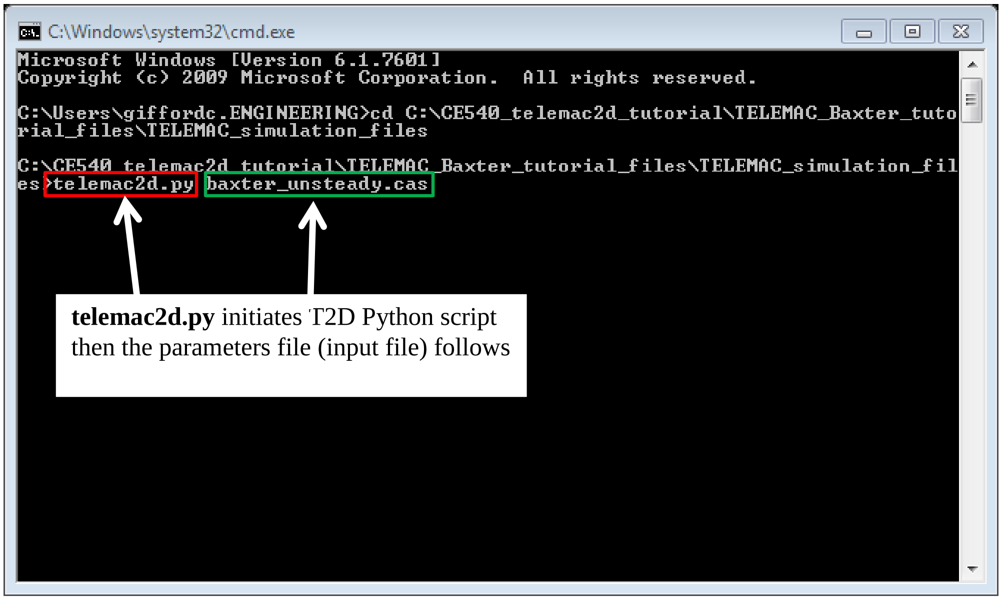


Figure 84: Within the input file working directory, execute TELEMAC-2D using the command, **telemac2d.py name_of_input_file.cas**

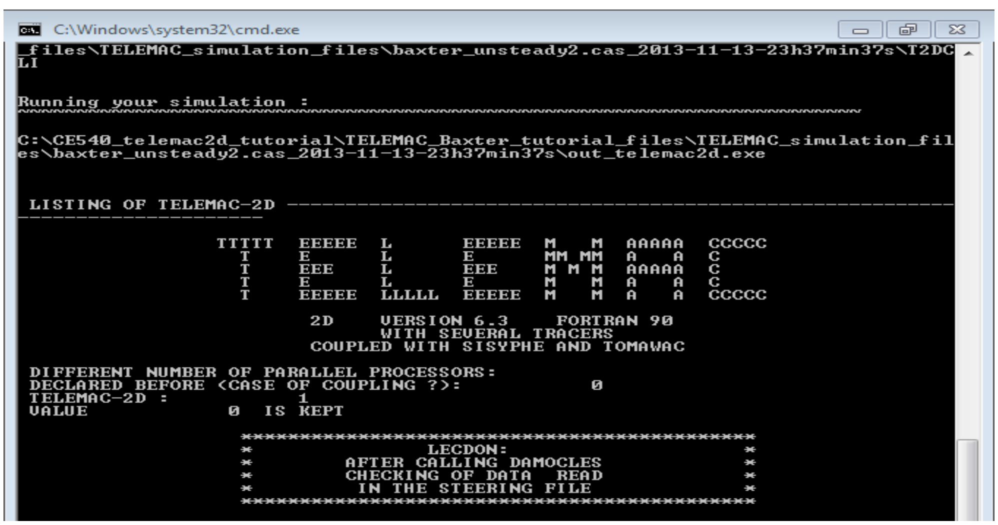


Figure 85: If there are no errors, the simulation will execute until finished

5.2 TELEMAC-2D Steady-state simulation

Good initial conditions are imperative for reducing overall simulation times as well as ensuring that no additional factors influence the final results. In general, achieving a steady-state solution occurs when the discharge and water surface elevation are no longer changing as a function of time (i.e. $\frac{\partial Q}{\partial t} = 0, \frac{\partial H}{\partial t} = 0$). Here we can utilize the fully-implicit descritization of TELEMAC-2D to speed up the computation for achieving an initial steady state.

An example steady-state parameters file has been included in the **TELEMAC_simulation_files** directory.

5.2.1 Create a HOTSTART file from previous computation

A **HOTSTART** file in this context refers to a simulation, continuing from the final time-step of a previous simulation. For example, a steady-state solution could have been reached for a given mesh configuration. If the computational mesh were changed for the same domain (added levees, bridge piers, etc.), instead of re-simulating from an at rest condition, the values from the previous mesh/solution can be mapped to the new, updated mesh. It requires a few steps, but ultimately saves time, especially when altering meshes on larger domains where reaching a steady-state solution can take hours or days to complete.

Figures 86 through 95 illustrate the process of creating a HOTSTART file from one mesh to another, using the **BlueKenue Calculator** tool.

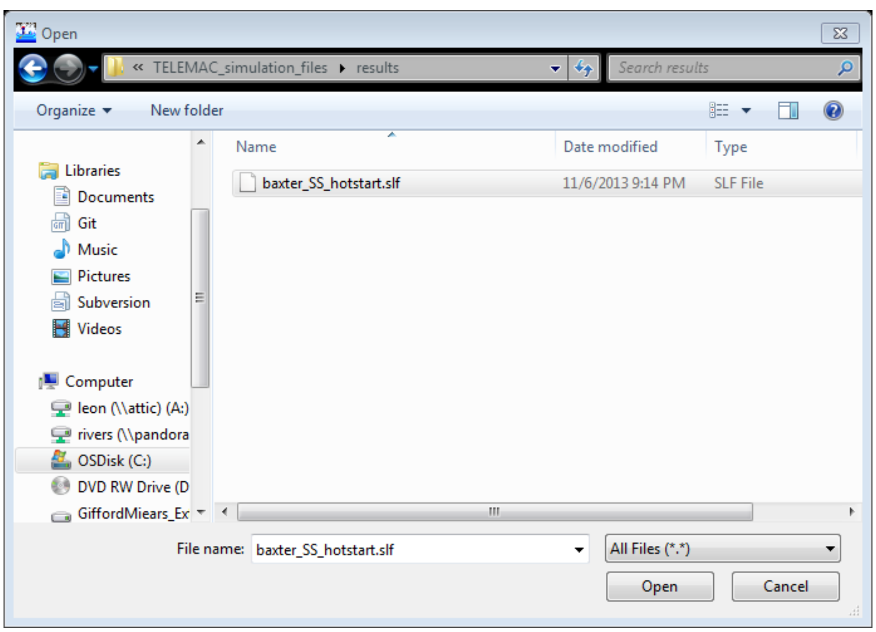


Figure 86: Open the previous results file in BlueKenue (e.g. Baxter_SS_hotstart.slf)

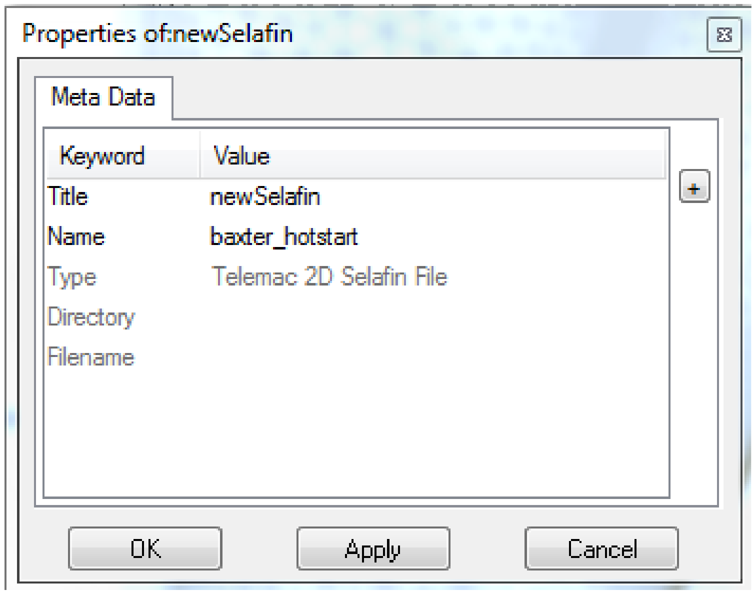


Figure 87: Create, and rename a new **Selafin** file for the HOTSTART components to be stored

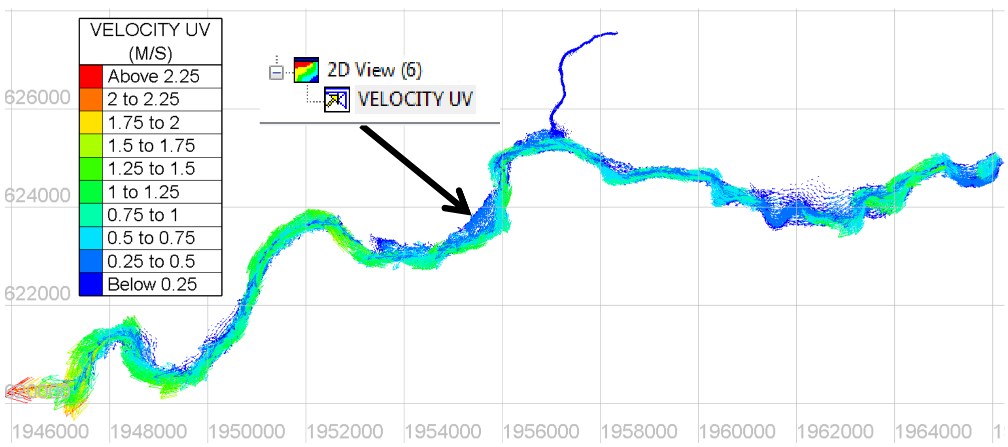


Figure 88: If the previous results file has several timesteps, be sure to **Animate** the results, and fast-forward to the final **frame**

Select the **VELOCITY UV** object from the previous results file, **go-to Tools ¿ Calculator...** and extract a copy of the **U** velocity component of the object using the calculator method shown in Figures 89 and 90. This is necessary because BlueKenue automatically combines **U** and **V** components of velocity when loading result files, where TELEMAC-2D requires that each variable component be available separately.

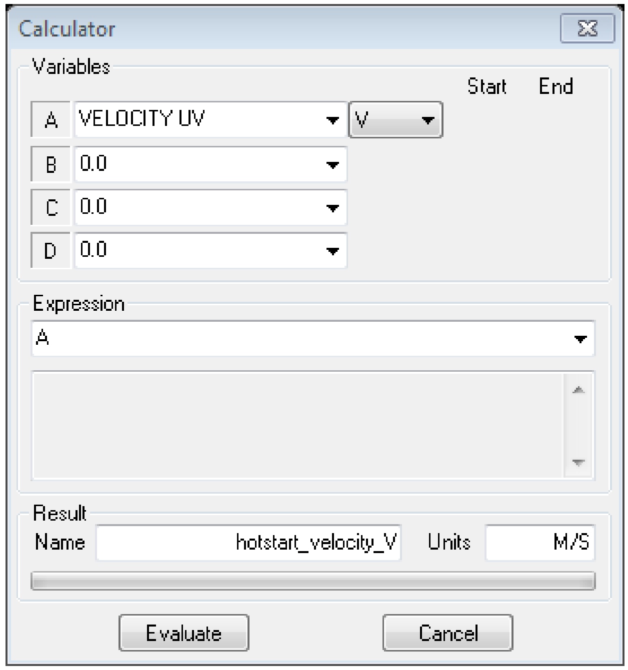


Figure 89: BlueKenue Calculator method for extracting **V** component velocities as a new object **hotstart_velocity_V**

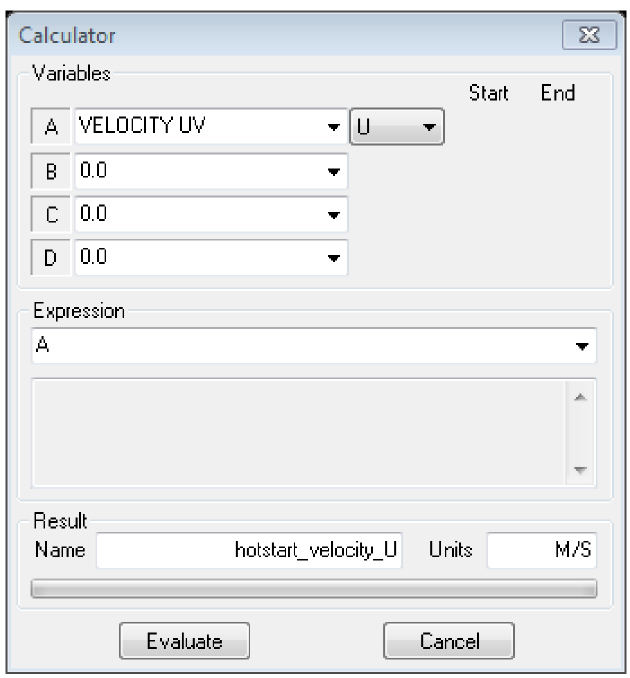


Figure 90: BlueKenue Calculator method for extracting **U** component velocities as a new object **hotstart_velocity_U**

Now that the **U** and **V** components of velocity across the domain are extracted, new objects have to be added to the **Selafin** file for interpolating. Add the current domain mesh as the **Source Mesh** and specify the **New Variable** as **VELOCITY U** , for example (Figure 91).

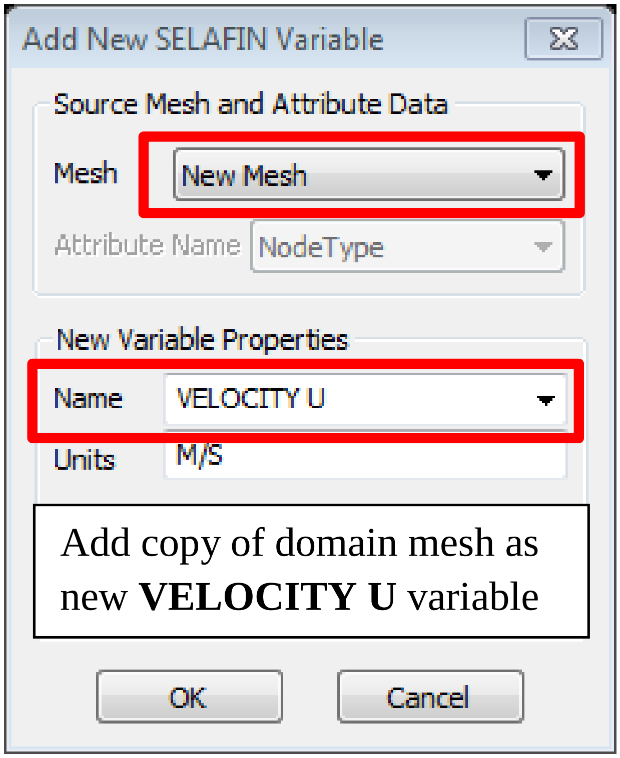


Figure 91: Add copies of the current domain mesh as new variables to the **Selafin** file

Once the **New Variables** are added, they will by default be set to **0** and appear flat as shown in Figure 92.

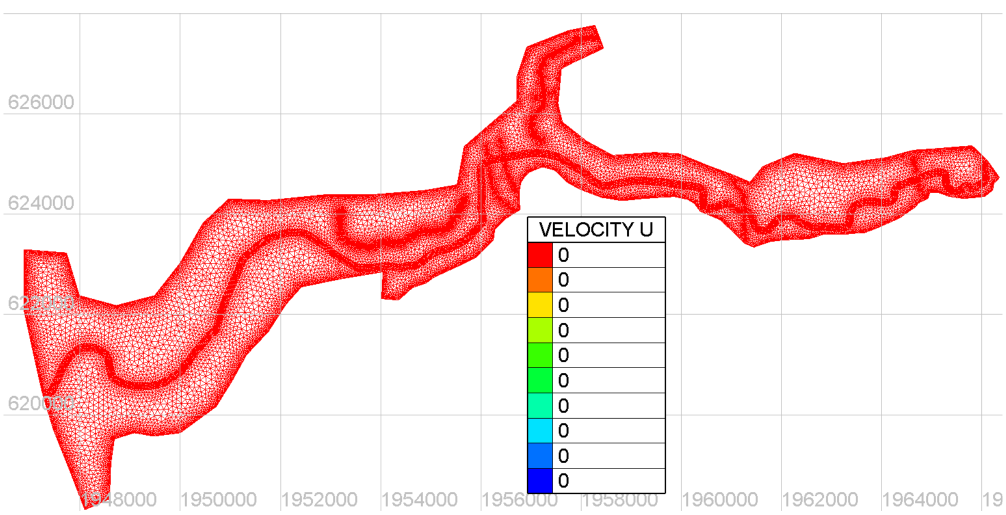


Figure 92: Unmapped new variable **VELOCITY U** , prior to **Map Objects** command

Select one of the **New Variables** of the **Selafin** file, and **go-to Tools ¿ Map Objects** and select the previous result file child-object that corresponds to the selected **New Variable** . This is shown with the child-object **VELOCITY U** in Figure 93.

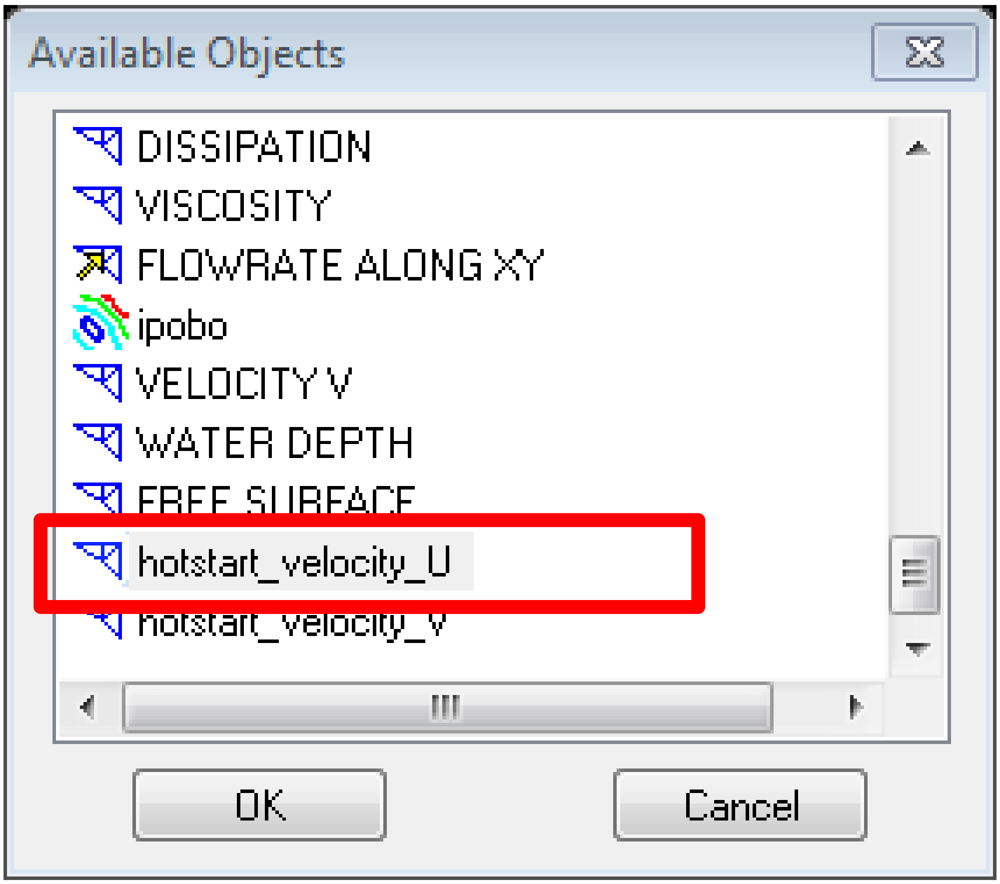


Figure 93: Map previous result file components to new components using **Map Objects** command

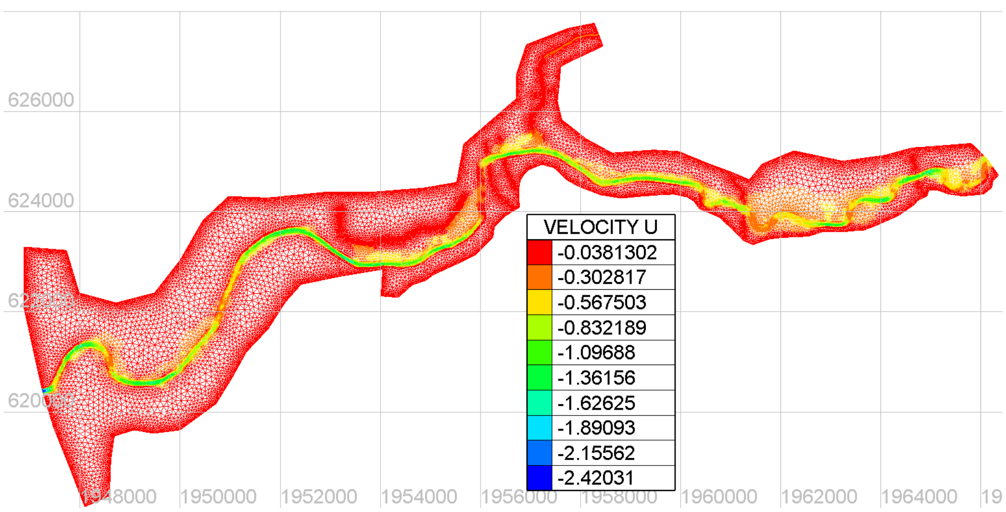


Figure 94: Successfully mapped **VELOCITY U** object

The variables required for a complete HOTSTART file to be created are VELOCITY U, VELOCITY V, WATER DEPTH, and FREE SURFACE ELEVATION.

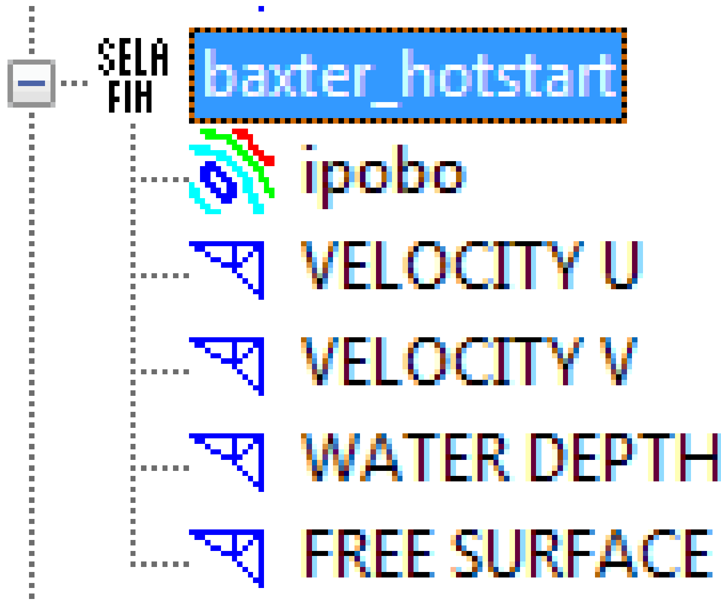


Figure 95: Successful HOTSTART file generation should contain the above four child-objects

Once the HOTSTART file is created, it can be included in the parameters file as a **PREVIOUS COMPUTATION FILE** .

5.3 TELEMAC-2D Unsteady simulation

5.3.1 Unsteady boundary conditions

Inflow hydrographs are to be specified at the **Open boundary with specified Q** nodes for both upstream reaches. The downstream boundary will be represented using a rating curve assuming **Normal Flow** and a bed-slope, S_0, equal to 0.001 m/m.

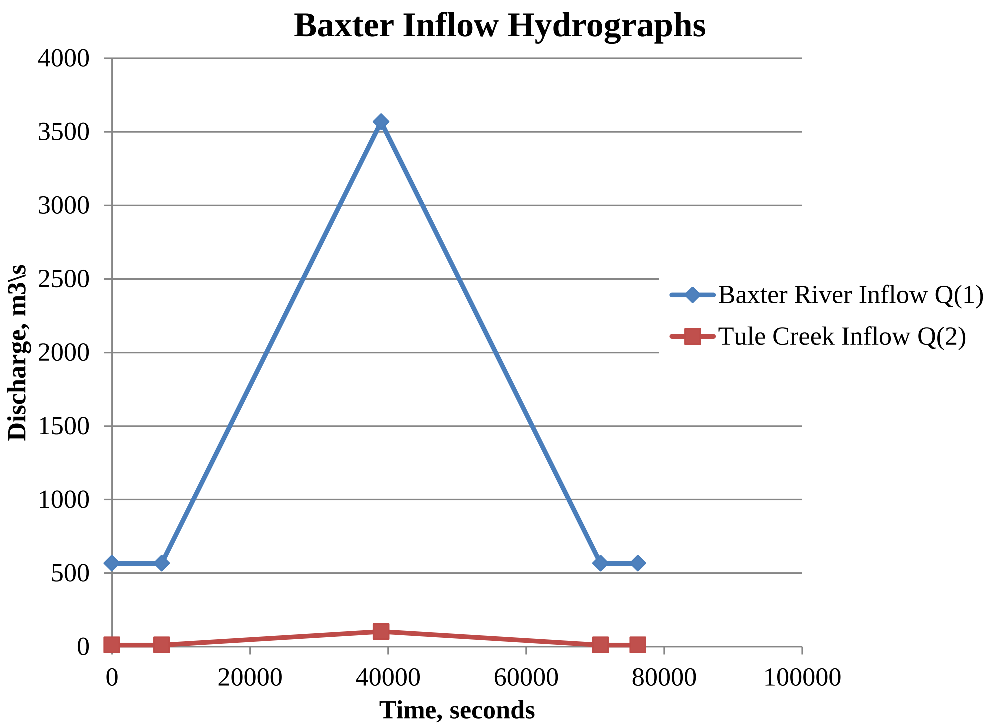


Figure 96: Inflow hydrographs applied to the Baxter River and Tule Creek

Table 4: Baxter tutorial inflow hydrograph values

T s	Q(1) m3/s	Q(2) m3/s
0	566.337	11.327
7200	566.337	11.327
39000	3567.923	102.649
70800	566.337	11.327
76200	566.337	11.327

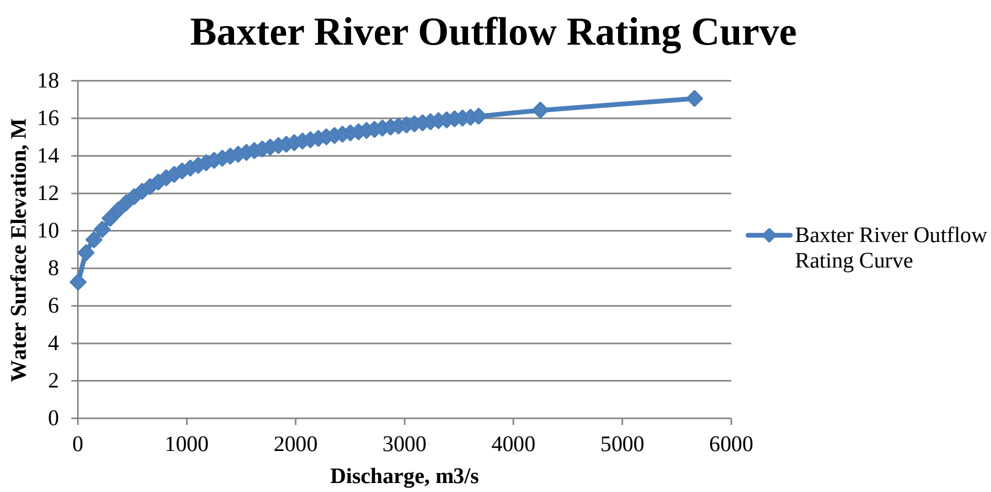


Figure 97: Outflow rating curve for this tutuorial

Table 5: Rating curve for downstream boundary condition

Z(3) m	Q(3) m3/s
7.266	2.832
8.830	76.399
9.513	149.966
.	.
.	.
.	.
16.106	3681.190
16.426	4247.527
17.057	5663.369

6 Post-processing utilizing BlueKenue

BlueKenue as a post-processing tool serves as an intuitive and capable platform for viewing and analyzing 2D- and 3D-results. Once the results are opened within the **Workspace** , the following are some guidelines for basic viewing of results.

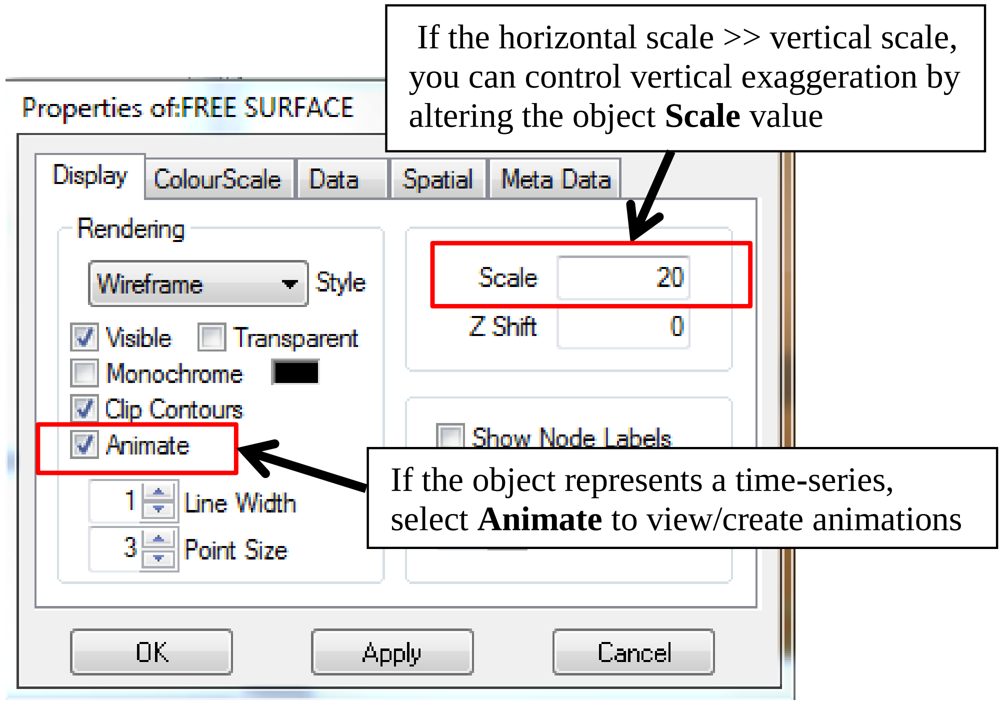


Figure 98: Applying a vertical exaggeration to datasets with horizontal scales much larger than the vertical scale (e.g. BOTTOM, FREE SURFACE, etc.) helps to visualize geometric features

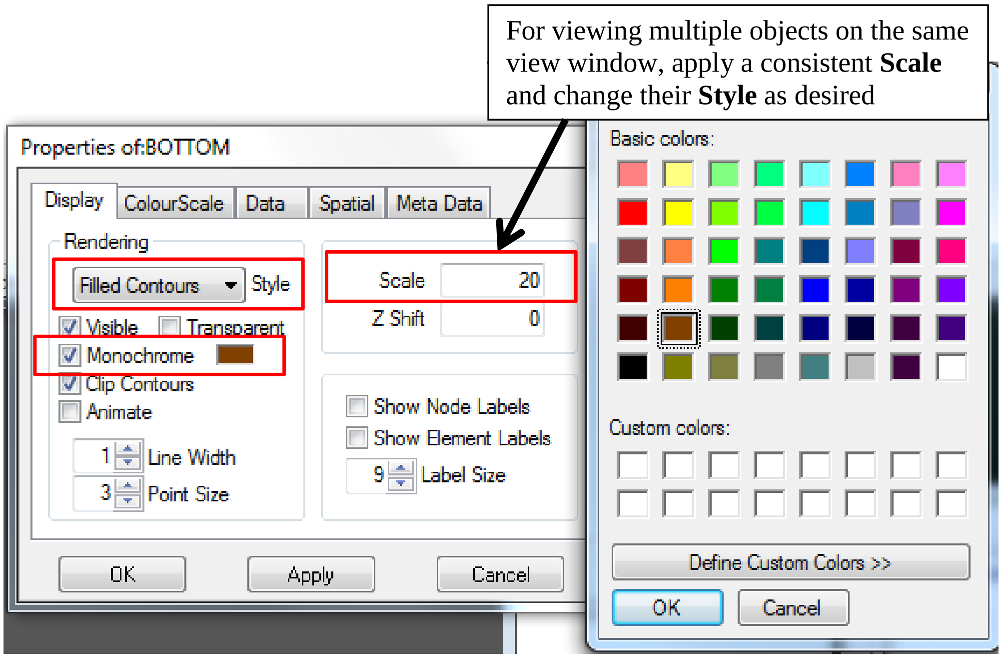


Figure 99: Changing the **Style** and opacity of the object is easily performed by double-clicking or right-clicking the object

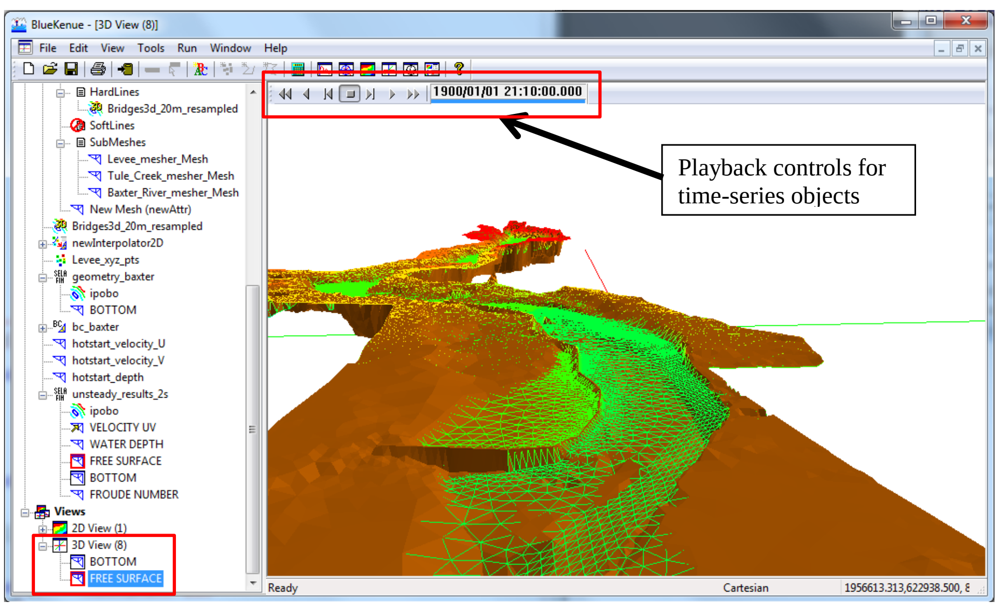


Figure 100: After initializing the **Animate** item under properties and select **Apply** , then the play-back controls initialize to view the dataset

6.1 Flood inundation view settings

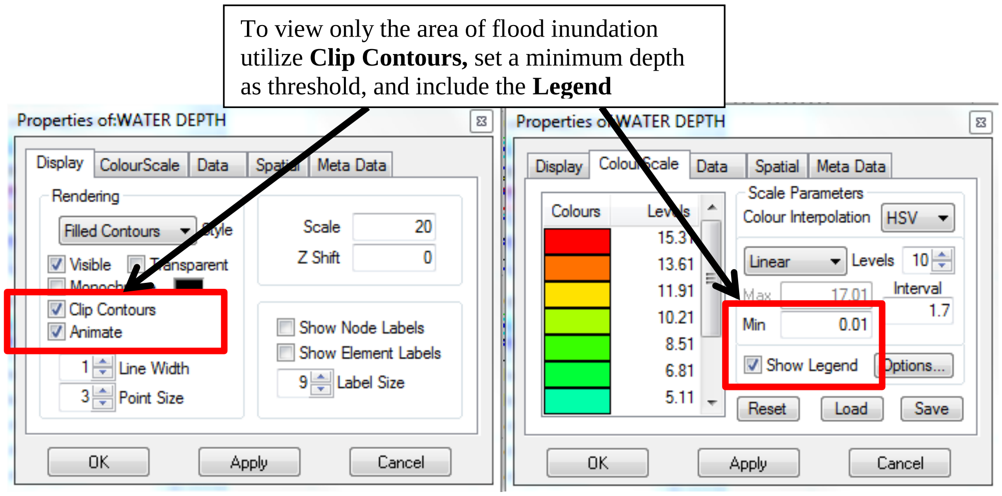


Figure 101: To view only the Flood Inundation depths, apply **Clip Contours** and make sure that **Style** is set to **Filled Contours**

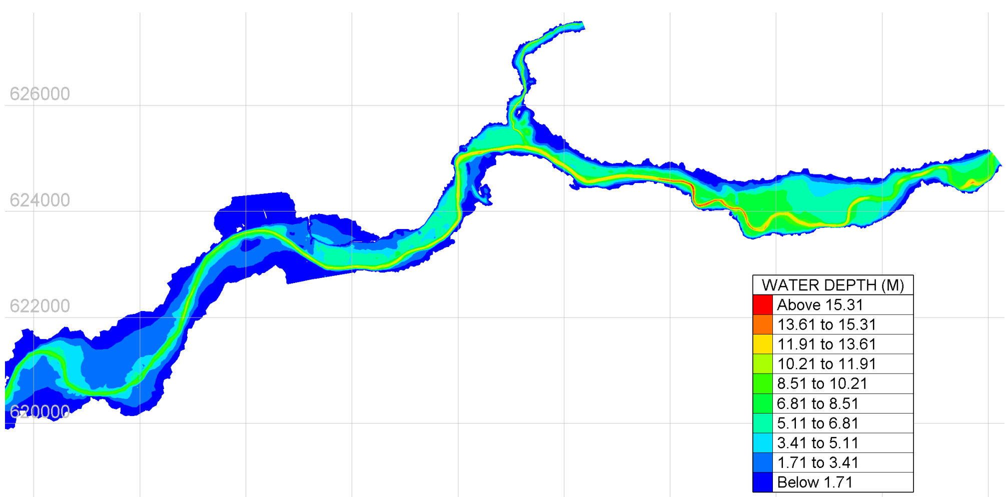
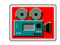


Figure 102: Resulting view from Figure 101 settings with Animate enabled

6.2 Animation of Flood-wave propagation

Animations of your TELEMAC-2D results are easily exported as video files using BlueKenue. For example, the following steps will illustrate how to export an animation of the flood-wave propagation to a video file.

1. Create a new **3D View**
2. Drag and drop TELEMAC-2D result objects to animate

3. Enable animation on the T2D object and select desired **Scale** , viewing angle, etc.
4. Under properties of the **3D View** object, select the **Recording** tab
5. Choose destination filename, number of frames, playback frame-rate, encoding method, etc.
6. Create the animation by selecting the record icon, 

6.3 Outflow hydrographs using MATLAB

Using control sections defined within the computational domain, it is possible to calculate and track discharge at specified locations. The control sections can be specified by either coordinate pairs, or by start-and-end mesh nodes. The control sections incorporated in this tutorial are shown in Figure 103 below. An example of outflow hydrographs resulting from using control sections is shown in Figure 104. The simple MATLAB plotting routine is included in the **ctrl_section_results** directory as **plot_ctrl_section_results.m** .

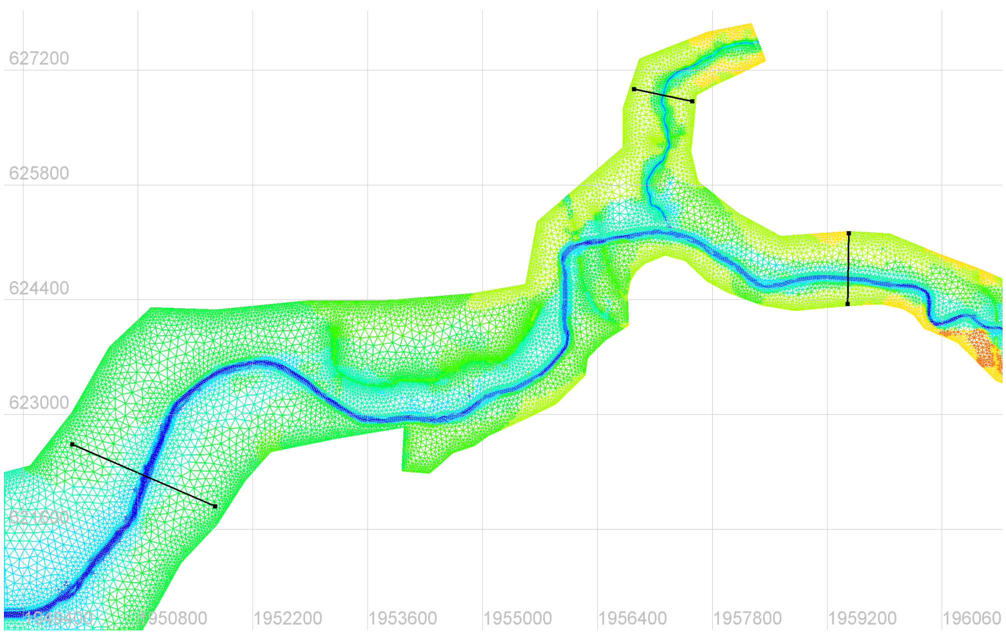


Figure 103: Control sections within the Baxter River domain

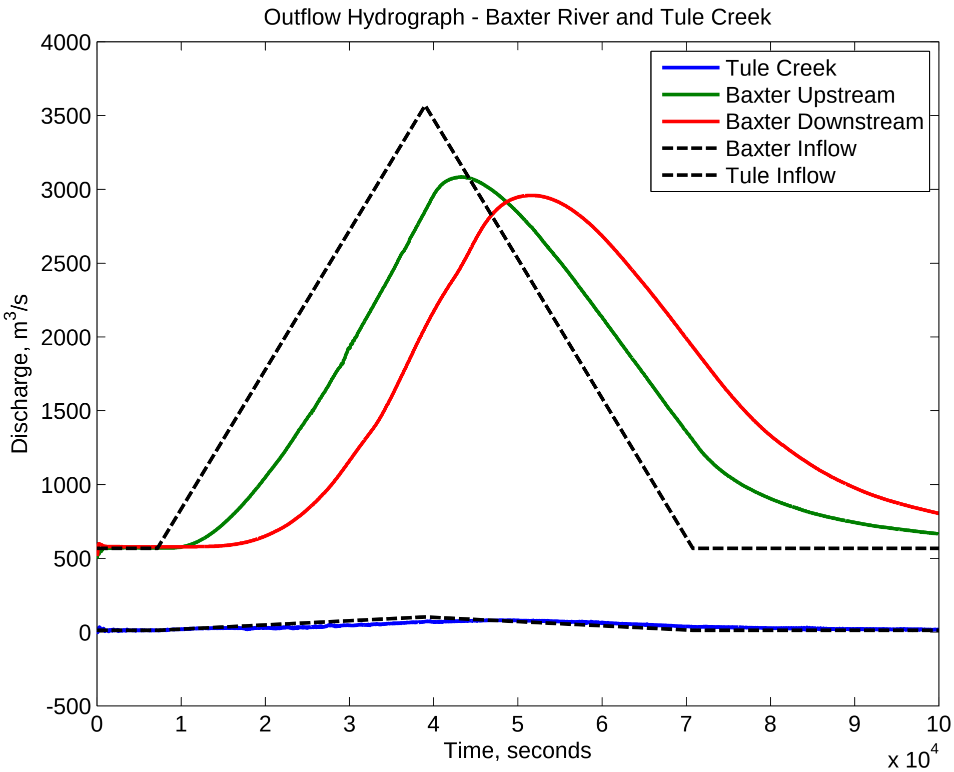


Figure 104: Outflow hydrographs resulting from unsteady TELEMAC-2D simulation

6.4 TELEMAC-2D vs HEC-GeoRAS Flood Inundation

Comparison between TELEMAC-2D and HEC-GeoRAS flood inundation can be achieved through importing the HEC-GeoRAS inundation map to BlueKenue. The process for exporting an ArcView Shape File from ArcGIS is detailed in Section 2.3. Note that the exported Shape File will need to be converted using the conversion in Section 2.5.

A quick comparison between the TELEMAC-2D and HEC-GeoRAS flood inundation extents shows loose agreement, with TELEMAC-2D showing greater inundation extents for the same flow event. Using the **Animate** playback controls, the TELEMAC-2D max inundation occurs around time 15:00 (HH:MM).

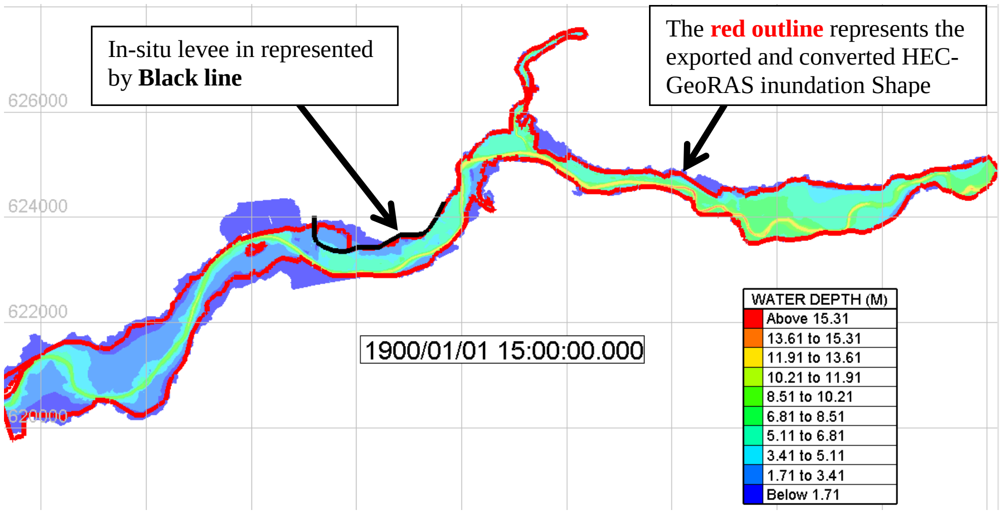


Figure 105: Example comparison of TELEMAC-2D and HEC-GeoRAS inundation extents

Appendix A Create Simple Meshes in Blue Kenue

This section gives brief step-by-step instructions on how to create a simple mesh in BlueKenue for use with the TELEMAC hydrodynamics suite. The content and figures herein are adapted from the work of **Stephen Kwan, MCS, Ph.D** . You can find the original document link at his website: `http://river2dm.wordpress.com/telemac2d/`

1. Load xyz data.
2. Create closed line (Figure A106) Page 90
3. Highlight the xyz data in $DataItems$
4. Select Files>New>T3 Mesh generator (Figure A107) Page 90
5. Give length of element side and press OK
6. Drag the new ClosedLine into Outline of newT3Mesh (Figure A108) Page 91
7. Double click on new T3 Mesh and press run (Figures A109 and A110) Pages 91 & 92
8. Select File>New 2D Interpolator (Figure A111) Page 92
9. Drag subset into newInterpolator2D (Figure A112) Page 93
10. Highlight New Mesh (NodeType)
11. Select Tools>Map object
12. Choose New2Dinterpolator (Figure A113) Page 93
13. Give name, put M for units (Figure A114) Page 94
14. Mesh shown in (Figure A115) Page 94

A.1 Supporting figures for simple mesh tutorial

The figures below correspond to the above list of steps required to create a simple mesh in BlueKenue.

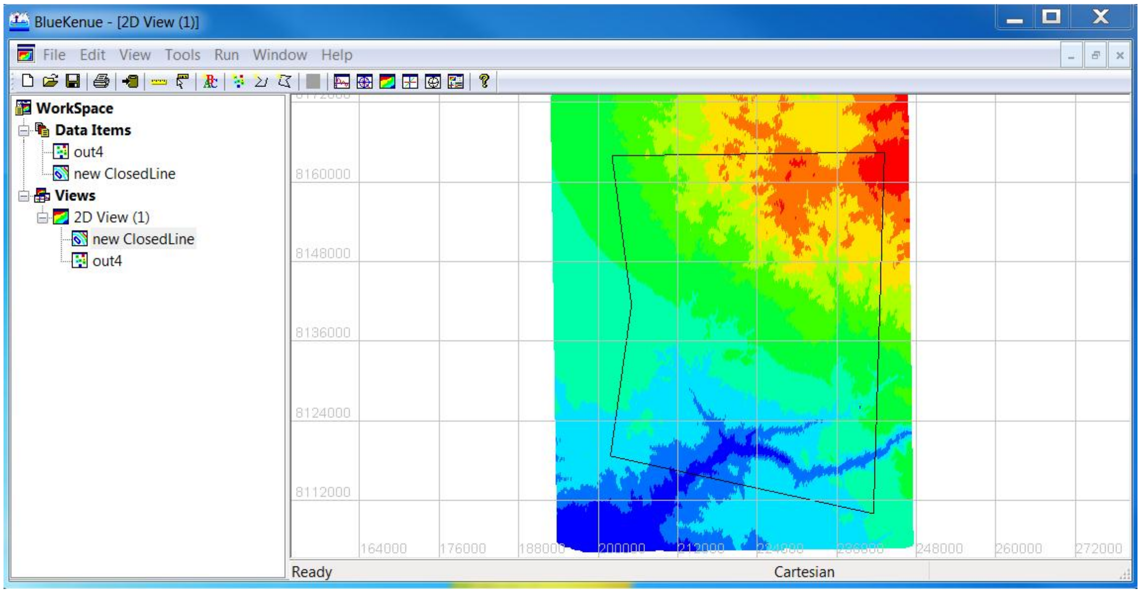


Figure A106: Create closed line for the exterior boundary of the mesh.

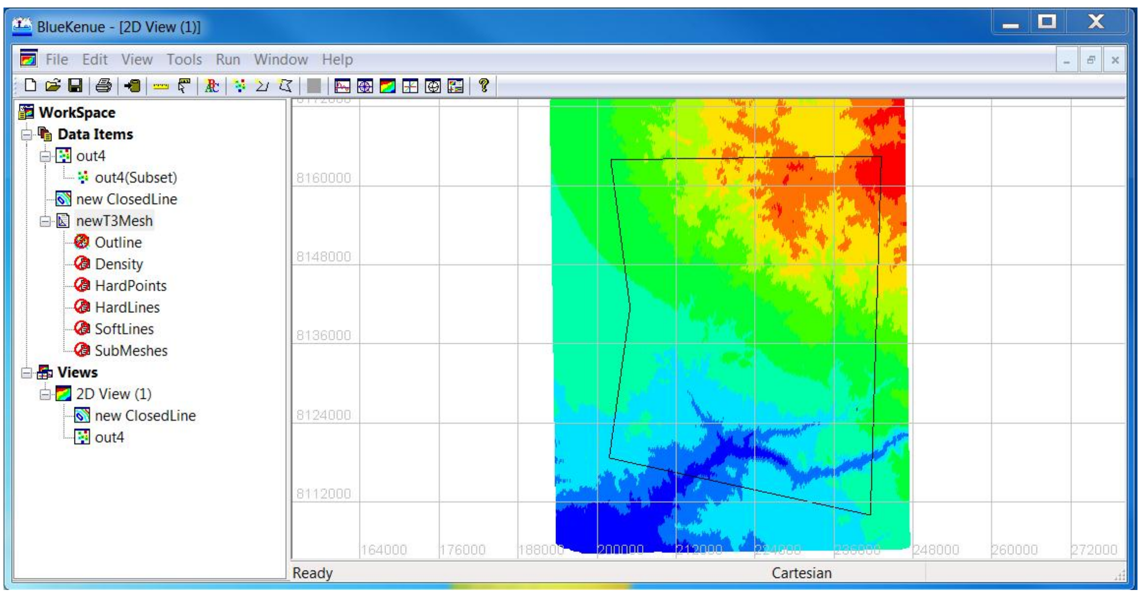


Figure A107: Create New T3 Mesh

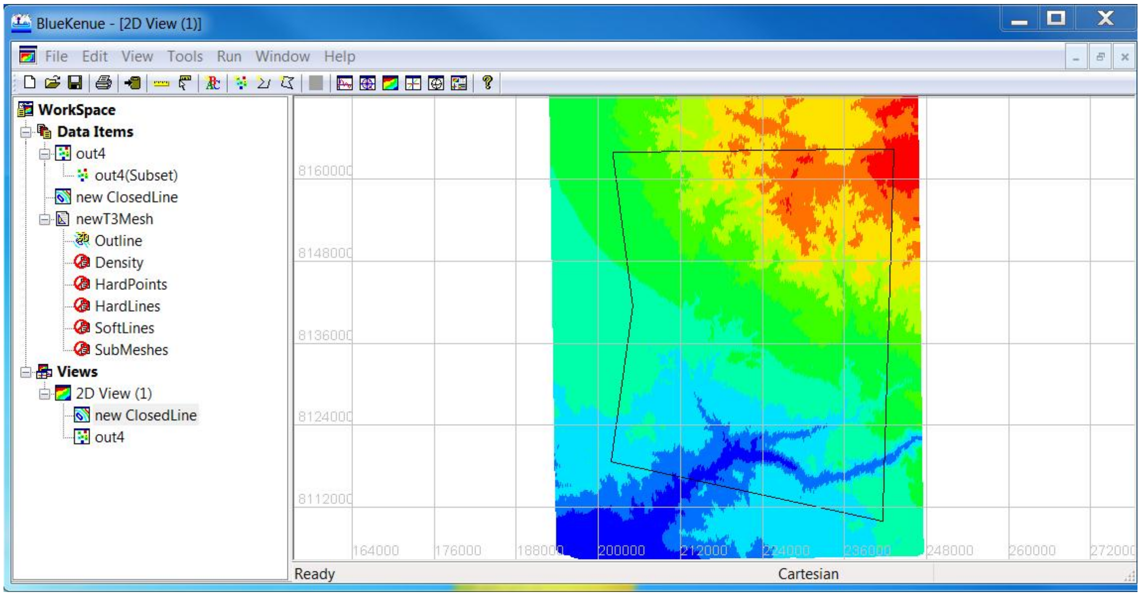


Figure A108: Drag $newClosedLine$ into newT3Mesh $Outline$

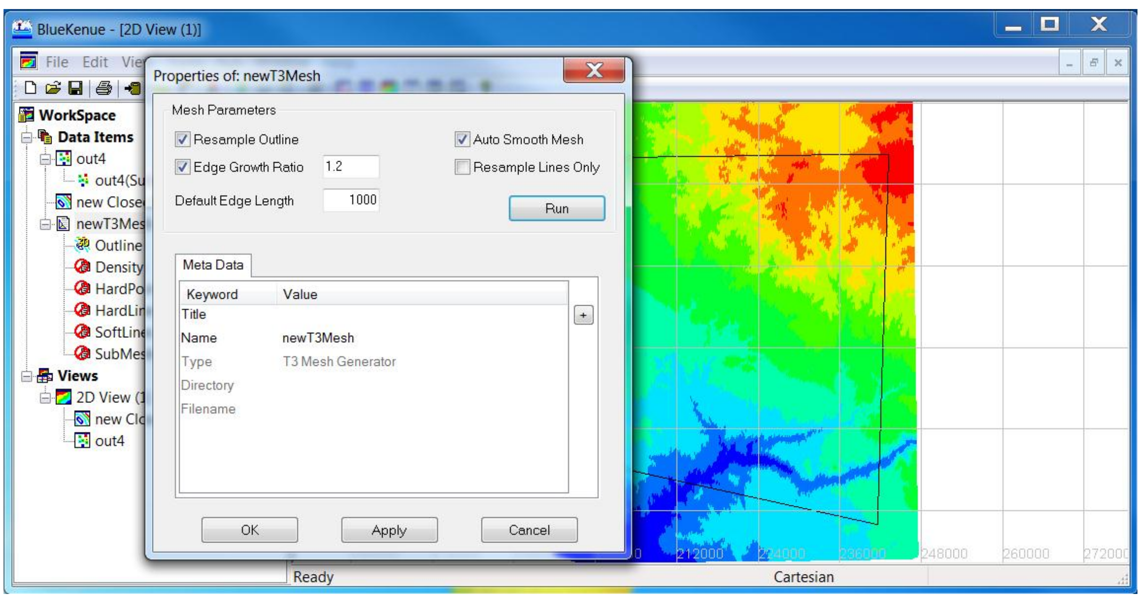


Figure A109: Select edge length for elements and run.

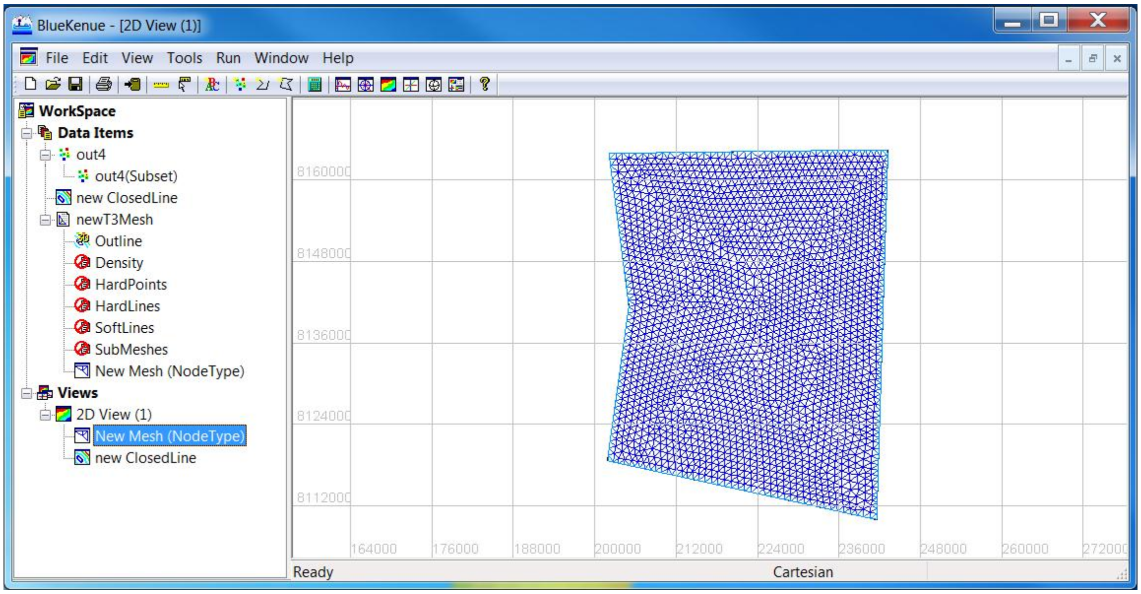


Figure A110: The new T3 mesh!

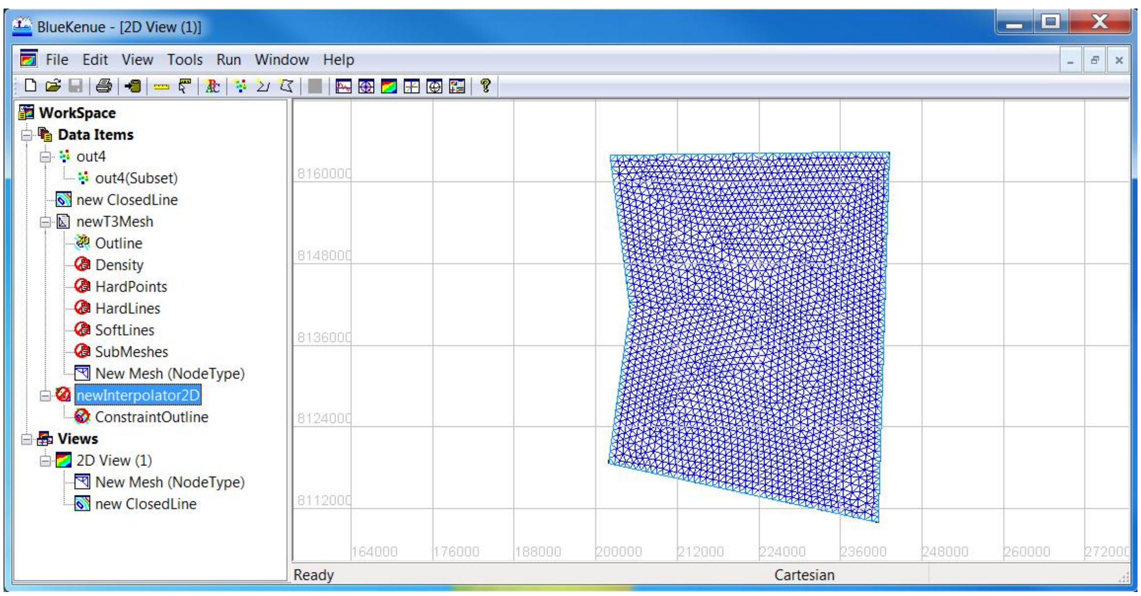


Figure A111: Create New 2D Interpolator.

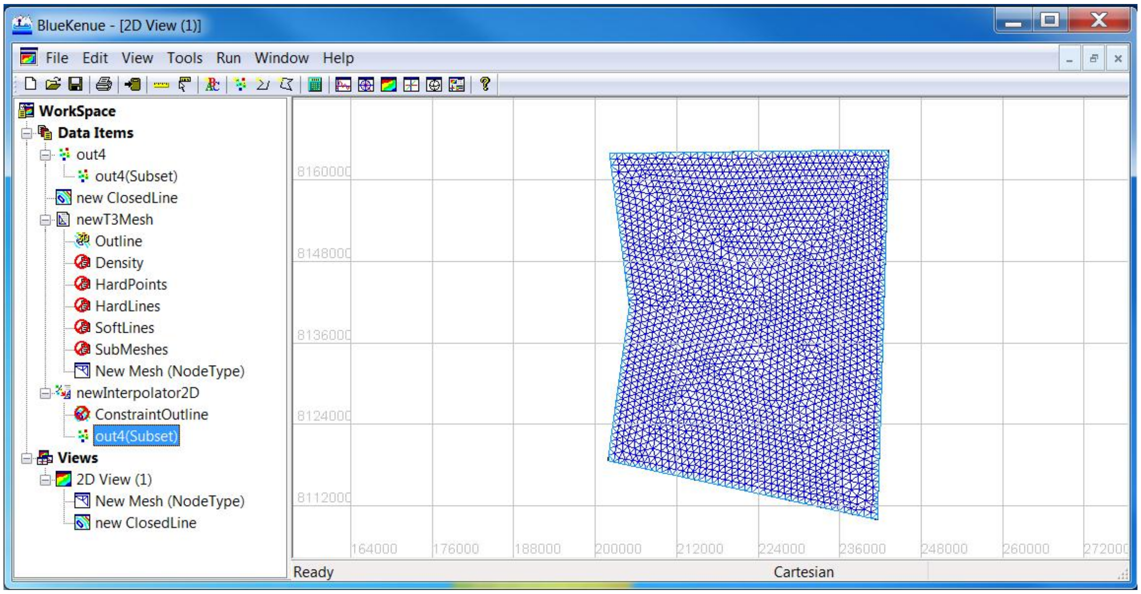


Figure A112: Drag subset of xyz data into newInterpalator2D.

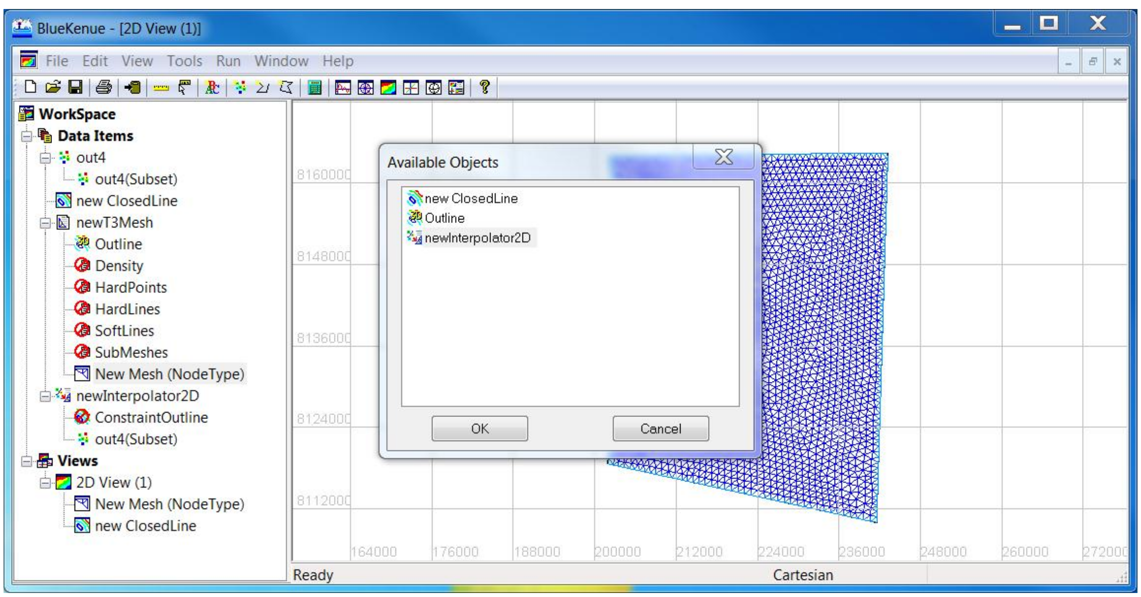


Figure A113: Choose Tools>Map Object and select newInterpolator2D.

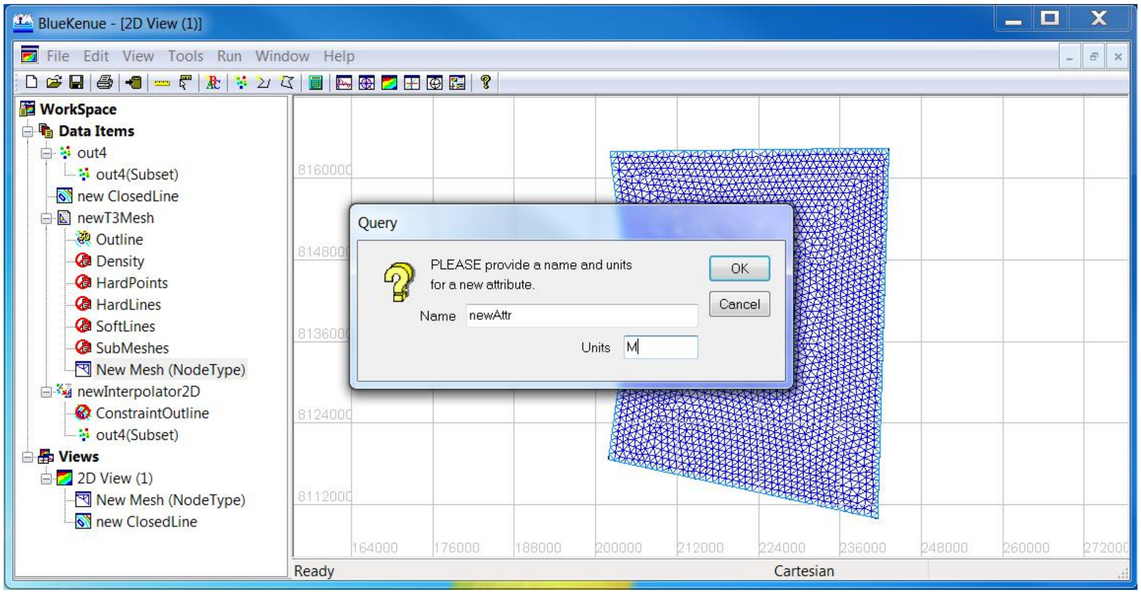


Figure A114: Give a new name and choose put M for units.

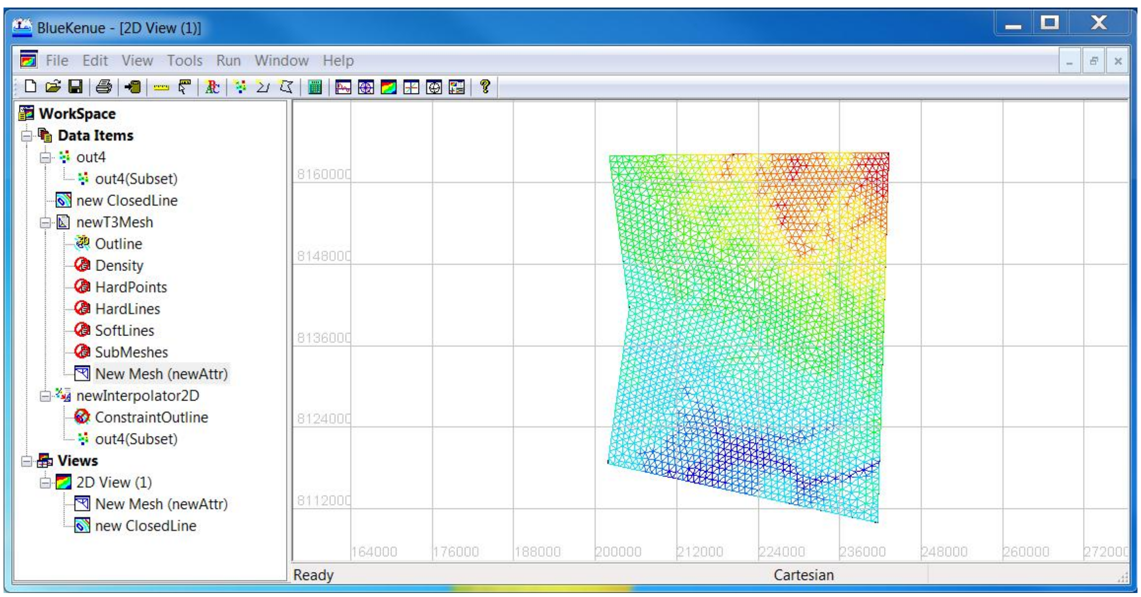


Figure A115: The new interpolated mesh.

Appendix B Self-installation of TELEMAC hydrodynamics suite

TELEMAC-MACARET Python Installation Notes Link

B.1 Pre-requisite programs for TELEMAC installation

These brief notes detail requirements for source code compilation using a Python/gfortran/windows configuration. As stated above in Section 1, the required programs to be installed include:

- - Python 2.7.3 or 2.7.5
- - gfortran (gcc) Compiler
- - SVN Tortoise subversion program

B.2 TELEMAC source code checkout using Tortoise SVN

Checking out the source code of TELEMAC is easily done using Tortoise SVN software. Create a new folder named 'opentelemac' in the desired directory (e.g. c:\... for this example), right-click the folder and select 'SVN Checkout...'. Next, you'll enter the SVN address below and then enter the username and password when prompted. Note: if there is a new version of TELEMAC available, simply change the SVN address per www.opentelemac.org.

URL of Repository (as of November 1st, 2013):
`http://svn.opentelemac.org/svn/opentelemac/tags/v6p3r1/`
username: ot-svn-public
password: telemac1*

In order for TELEMAC to communicate with the required programs from the DOS command prompt, the user/system environmental variables must direct the system to each application/required file. You can get to the environmental variable dialogue through the Control- Panel, typing 'environmental variables' into the search bar, or from the 'Start' menu. Once there, append the environmental variables to include the ones stated below. As time goes on, each system will be slightly different, however the general idea is as follows:

1. grant access to Python, scripts¿python27 folder, and gcc bin folder within PATH
2. include EQ_LIBRARY_PATH for gfortran (gcc) to reference
3. include SYSTELCFG path to the TELEMAC configuration file

B.3 Environment variables for running/compiling TELEMAC

PATH:
`c:\gcc\bin;c:\Python27\Lib;c\Python27;` ...
`c:\opentelemac\v6p3\scripts\python27;`
EQ_LIBRARY_PATH:
`c:\gcc\x86_64-w64-mingw32\lib`
SYSTELCFG:
`c:\opentelemac\v6p3\configs\systel.cfg`

See `www.opentelemac.org` for additional installation instructions if this isn't clear: `http://www.opentelemac.org/...` search for Python installation instructions.

B.4 Testing your PATH variables

To test that your path variables are working, open a DOS command prompt session and type the following:

- 'gfortran -v'
- 'python -v'

If your path variables are set-up correctly, the version information will print to your screen. (note: to exit python-mode type 'exit()' or 'CTRL+Z+Enter')

B.5 Compiling TELEMAC

Once the path variables are working and the 'systel.cfg' file is appropriately set-up (or copied from this installation repository) for your system specifics, open the command prompt. Change directory to the pytel folder directory (**not necessary**),
(e.g. `cd c:\opentelemac\v6p3\scripts\python27`) and execute the command:

'compileTELEMAC.py'

If all is set-up correctly, then the TELEMAC system will soon be installed, compiled, and ready for use on your machine.

As mentioned before, if you're having any trouble installing TELEMAC with these instructions as well as the provided guide- lines of opentelemac.org, feel free to contact me.

Cheers!

Christopher Gifford-Miears
cgiffordmiears@gmail.com