

## Adding geological information to a simulation-ready mesh

**Parent topic:**  
**SECTION 6: 2D and 3D meshes and grids in 3D GeoModeller**

3D GeoModeller provides four workflows: two for adding geological information to a simulation-ready mesh, and two for direct export of meshes from GeoModeller:

- Meshing workflow 1 (MW1)—Fill FEFLOW centroids from a GeoModeller model
- Meshing workflow 2 (MW2)—Prism a FEFLOW Triangulation for creating a layered mesh
- Meshing workflow 3 (MW3)—Direct export of a prismaed triangulation of topography from GeoModeller
- Meshing workflow 4 (MW4)—Direct export a GeoModeller fully unstructured mesh

### Meshing workflow purposes and advantages

3D GeoModeller implicit 3D model export workflows were designed for numerical modellers wanting to:

- 1 Build their 3D geology in 3D GeoModeller, and then
- 2 Export the model including faults and geology to a 3rd party software for performing numerical simulations.

Possible applications might be to perform groundwater flow modelling, geothermal flow modelling, or reservoir simulation.

The advantages of using 3D GeoModeller as a 'front-end' geology model builder are that 3D GeoModeller is intuitive, fast and easy to use. 3D GeoModeller is capable of building complex geology including faults, folds, overturned strata, intrusions and thin bodies etc. using constraints from raw geology observations such as 'contacts' on geology formation boundaries, and 'dips' as structural measurements (including a dip-angle and dip-direction or azimuth).

## Meshing workflow 1 (MW1)—Fill FEFLOW centroids from a 3D GeoModeller model

**Parent topic:**  
Adding geological information to a simulation-ready mesh

### MW1 action summary

MW1 fills *FEFLOW* Centroids of a layered finite element mesh with geology from 3D GeoModeller

### About MW1

You need pre-created data & model files to perform this example workflow.

MW1 is for filling *FEFLOW* centroids of a layered finite element mesh with geology from 3D GeoModeller.

MW1 is specifically designed for coupling 3D GeoModeller with *FEFLOW* software (developed by DHI-WASY, Berlin). MW1 has been available in 3D GeoModeller since 2013, and can use much older versions of *FEFLOW* (well before *FEFLOW* 7.0).

### MW1 advantages

MW1 is important for groundwater modellers and other users. It delivers geological identities in 3D space to centroid locations of cells of a pre-created finite element mesh (FEM)—pre-created in *FEFLOW*.

### Videos for MW1

You can view a demonstration of MW1 on the Intrepid Geophysics YouTube channel in two short videos. Intrepid Geophysics and DHI-WASY team members produced them jointly.

The videos are as follows:

- 1 How to fill *FEFLOW* Centroids of a layered finite element mesh from a 3D GeoModeller model

[https://youtu.be/cn-4zAXYU\\_Y](https://youtu.be/cn-4zAXYU_Y)

- 2 How to build a layered fem mesh suiting the extents & topography of your 3D GeoModeller model

[https://youtu.be/jWNGb\\_oGqdw](https://youtu.be/jWNGb_oGqdw)

### Steps for MW1

Follow these steps:

#### Stage 1—using *FEFLOW*

**>> In *FEFLOW* [Aim: to create a text file containing centrepoints of elements of a pre-existing *FEFLOW* mesh called *TutorialA.fem*]**

- 1 In *FEFLOW* Open *TutorialA.fem* File, Open, *TutorialA.fem*
- 2 In the Data Panel/Auxiliary Data > User Data > (right click) Add Elemental Distribution and call it “Geology”
- 3 From the Geology shortcut (right click) menu choose Export Data > All Elements > As Center... (exports the centroid coordinates and number of elements of the mesh as a text (.dat) file)
- 4 Call it “ExportCentroids” (and choose No to adding new map). Choose OK.

## Stage 2—using 3D GeoModeller

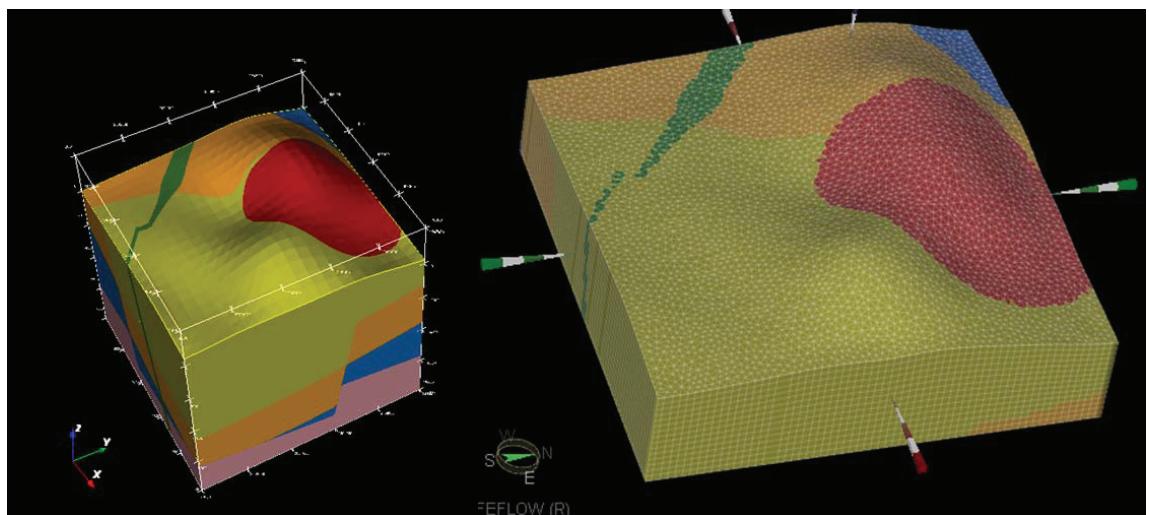
**>> In 3D GeoModeller [Aim: to fill FEFLOW centroids or centrepoints, with geological information]**

- 1 In 3D GeoModeller, **Open TutorialA.xml** (3D GeoModeller project: TutorialA) and ensure that the Compute is done
- 2 Choose **Export > 3D Model > Fill FEFLOW Centroids**
- 3 Browse to the text file just exported from FEFLOW (**ExportCentroids.dat**), and > **Open**
- 4 3D GeoModeller creates two files, **ExportCentroids\_filled.txt** and **ExportCentroids\_legend.txt**.
- 5 Rename the file extensions to .dat, and hence **ExportCentroids\_filled.dat** and **ExportCentroids\_legend.dat**.

## Stage 3—using FEFLOW

**>> In FEFLOW again [Aim: to add the geological information to the element centrepoints of the infinite element mesh file .fem]**

- 1 In the **Map** panel background, shortcut (right click) menu, > **Add Map(s)...** Browse and select **ExportCentroids\_filled.dat** and > **Open**
- 2 In the **Map** panel, select the new ASCII Table File **ExportCentroids\_filled** (right click) > **Define Coordinate Fields**
- 3 Feflow parses the map first, and then you should set the co-ordinates:
  - X is **centre\_X**
  - Y is **centre\_Y**
  - Z is **centre\_Z**
 Choose **OK**
- 4 Again in the **Map** panel, select the ASCII Table File **ExportCentroids\_filled** (right click) > **Link to Parameter(s)...**
- 5 In Parameter Association (left) select **GEOL**, and for User Data (right) select **Geology**. Double-click or choose (from the lower left) **Add Link**.
- 6 (Below) Observe that:
  - (In **Properties/Link type** define how to link data) For **Element/Layer Selection** choose **Field Containing element number**, and hence field **ELEMENT**. **OK** to close.
- 7 In **Maps**, select the new link **GEOL > Geology**, and **Double-click** to activate (And ensure the 3D Viewer is active, so more Tools/Toolbars will be available.)
- 8 From the upper Toolbars (Cyan-square tool) Choose **Select All**.
- 9 From the upper Toolbars (Green check-tick tool) Check the green **TICK** to import the data from the map, into the geology parameter of the mesh.
- 10 From the upper Toolbars (cross, left of the Cyan-square tool) Choose **Clear**

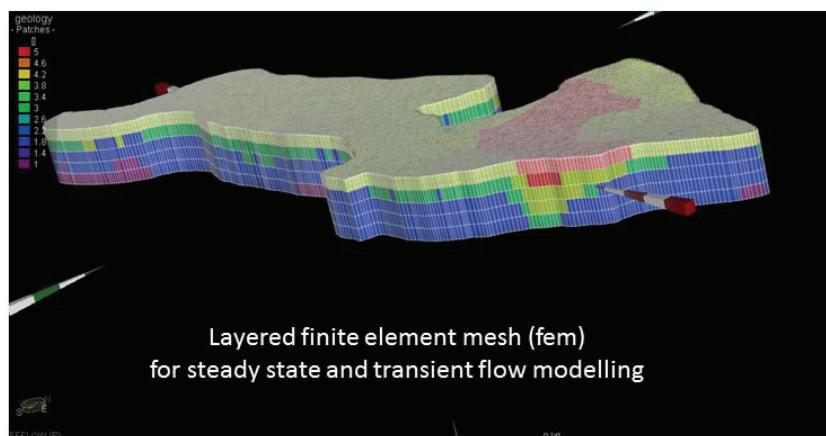
**Selection.**

Extensions in FEFLOW. Next in FEFLOW (using tutorial material from that Application) you can learn to, for example:

- 11 (a) define all the parameters of the model, or
- 12 (b) Store the geology selection, and name it for later assignments

## Coupling GeoModeller and FEFLOW

The hydrological model



By pre-creation of a FEFLOW layered finite element mesh, then  
introduction of the geology-identity to each cell  
GeoModeller (Menu called fill FEFLOW centroids)



## Meshing workflow 2 (MW2)— Prism a FEFLOW Triangulation for creating a layered mesh

**Parent topic:**

[Adding geological information to a simulation-ready mesh](#)

### MW2 action summary

Prism a 2D *FEFLOW* Triangulation (pre-created) for generating and exporting a layered mesh, including geological information, in GeoModeller.

### About MW2

**You need pre-created data & model files to perform this example workflow.**

MW2 is specifically designed for coupling 3D GeoModeller with *FEFLOW* software (developed by DHI-WASY, Berlin).

This option allows you to load a pre-created 2D triangulation of topography from *FEFLOW* (.fem file) and then to prism it in GeoModeller according to geology.

Meshing Workflow 2 is new in 3D GeoModeller v4.0.

### MW2 advantages

MW2 is important to groundwater modellers among other users. It delivers a 3D layered finite element mesh including geological information (.fem file) which is ready to use in *FEFLOW*.

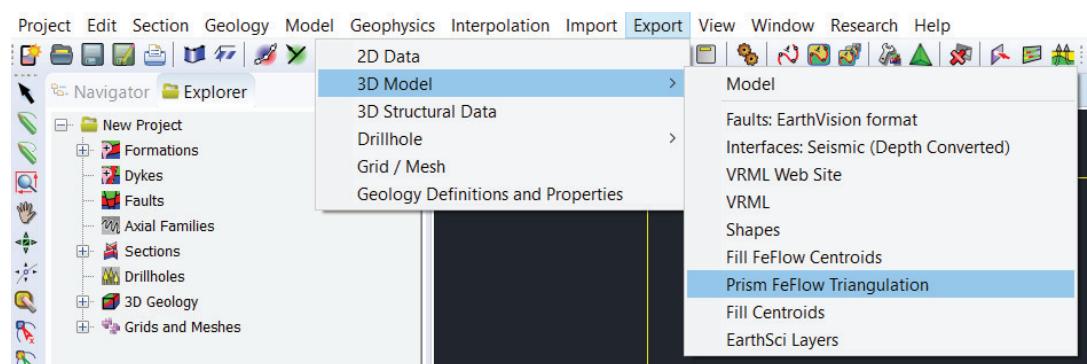
### Steps for MW2

You need to do this from the workspace of your 3D GeoModeller model (the model with matching geo-location to your starting 2D .fem file created in *FEFLOW*).

Ensure that you have the required 2D .fem file (triangulated topography) with geo-location matching the 3D GeoModeller model that contains our geological data.

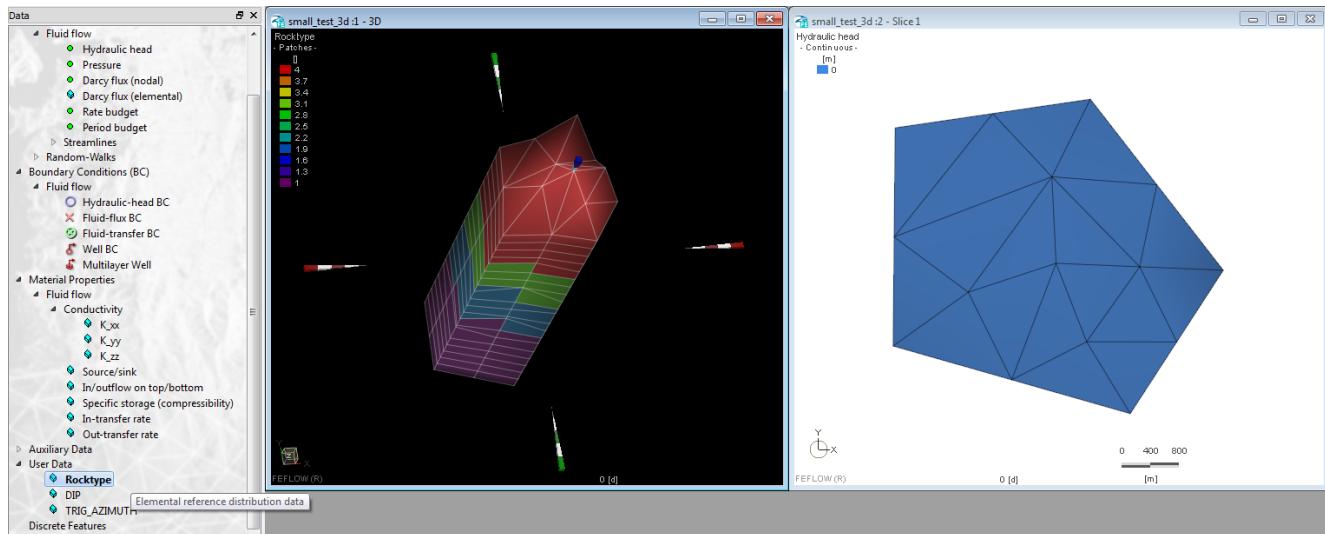
Follow these steps (with example data for the Wet River Valley model):

- 1 In 3D GeoModeller, Open **WetRiverValley.xml** (3D GeoModeller project: WetRiverValley)
- 2 Ensure that the **Compute** is done.
- 3 Choose **Export > 3DModel > Prism FEFLOW Triangulation.**

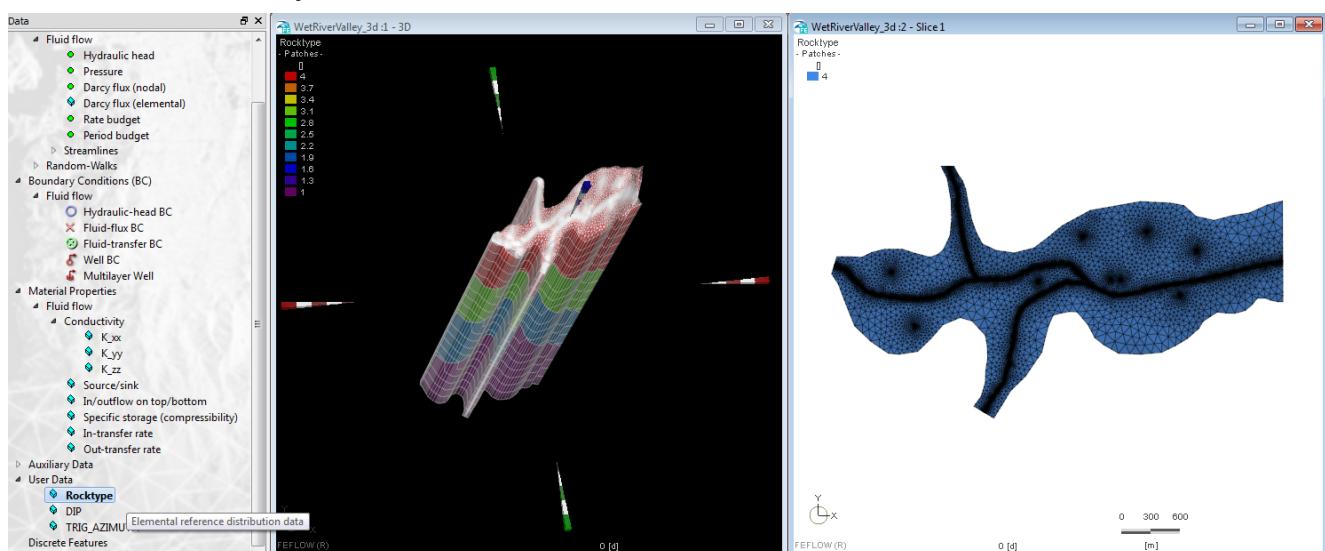


An **Open** dialog box appears

- 4 Locate, select and **Open** file called **small\_test.fem** (also: WetRiverValley.fem). 3D GeoModeller inserts the geological data, and saves the new .fem file in the same folder as the geolocated 2D .fem file with \_3d appended to its name.



(also: WetRiverValley.fem).



**In FEFLOW [Aim: display the geological information in the element mesh]**

## 5 Open the .fem file, and activate the 3D viewer

Double-click the User Data called Rock type

## Meshing workflow 3 (MW3)—Export a prismaed triangulation of topography

**Parent topic:**  
Adding geological information to a simulation-ready mesh

### MW3 action summary

3D GeoModeller layered mesh direct-export of a prismaed triangulation of topography

### About MW3

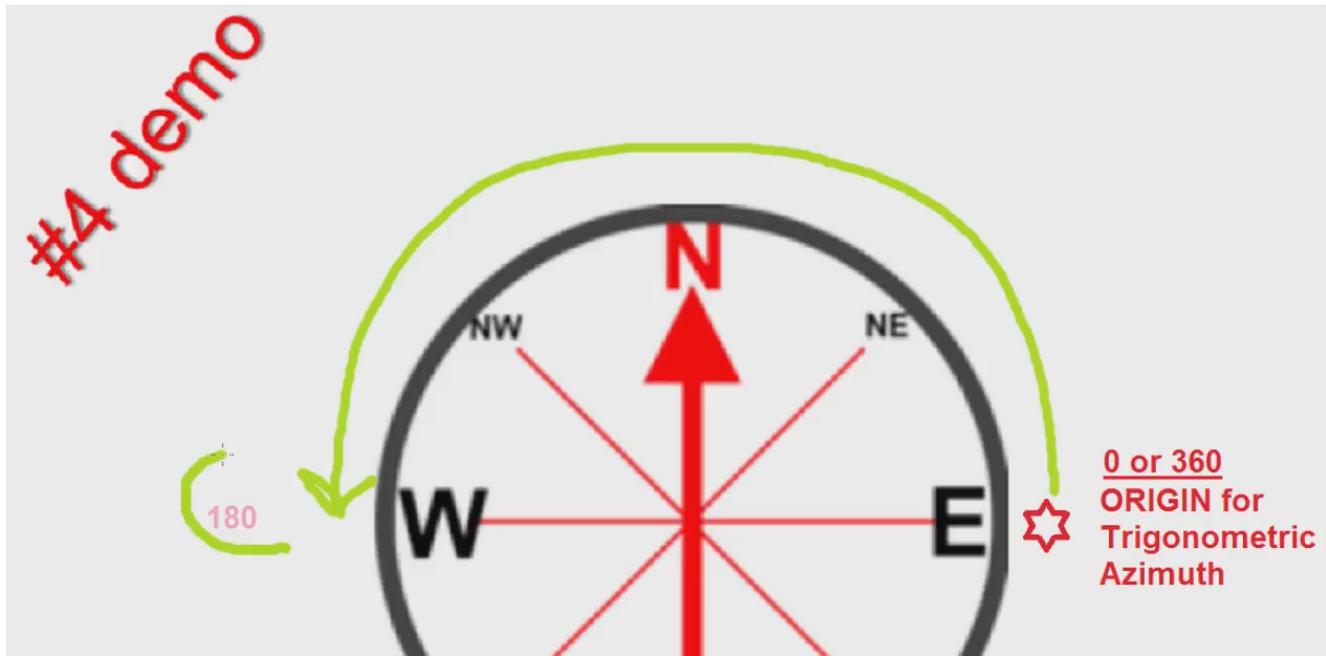
**This option does not require any pre-existing file from FEFLOW.** It may be performed on any GeoModeller project / model, as the export is generated entirely from the GeoModeller side.

This workflow instructs 3D GeoModeller to triangulate the topography surface of the current model and then to prism it according to geology shapes.

The image below shows the user interface of *FEFLOW* 7.0 with three 3D Viewer windows activated. These three relate to the 3 fields of attributes which are exported with the 3D GeoModeller layered mesh direct-export:

- **Rocktype** reports directly the Stratigraphic pile from the GeoModeller model
- "Dip" and "Trigonometric-Azimuth" define a vector in 3D space which parallels the 'field lines' generated within each geological volume, as solved in 3D in GeoModeller by co-kriging (the coupling of primary data: contacts & dip/dip-direction data), using the potential field method of interpolation, an implicit 3D method.

**Trigonometric Azimuth definition: Origin is at EAST, and the sense is clockwise, through 360 degrees.** (This is different from geological azimuth, as used in GeoModeller.)



```

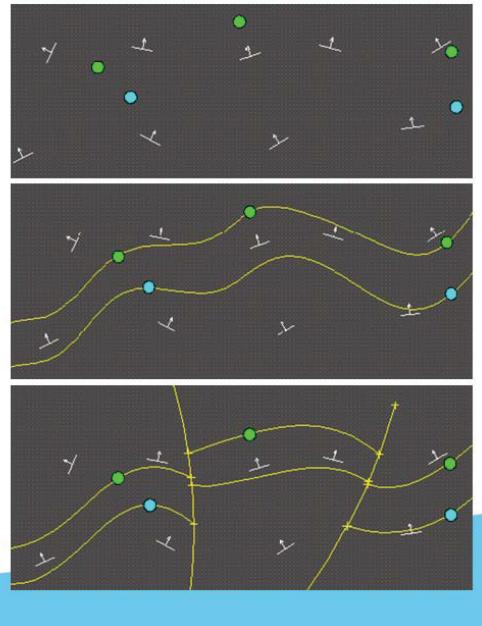
// find phi and theta from vec.
// get azimuth and dip from Geomodeller convention and then transform to phi and theta
Handle<Structural_PlaneOrientation> plo = new Structural_PlaneOrientation(gp_Dir(vec));
theta = plo->Dip();
phi = (360 - plo->Azimuth() + 90);
if(phi >= 360) phi -= 360;
}
else {
theta = phi = 0

```

- These are made available for use as metrics to guide anisotropic flow during simulations - Especially useful for sedimentary geology, hence a proxy for bedding. They are uniquely available from 3D GeoModeller because of the way that 3D GeoModeller performs the calculation of 3D implicit surfaces for geology boundaries.
- The 2D image below illustrates the potential field method of interpolation. 3D GeoModeller performs the calculation of 3D implicit surfaces in this way, by coupling together contacts and dip/dip-direction data during interpolation. Re-use of vectors from this approach gives us the ability to supply Phi and Theta angles as user data for *FEFLOW* users.
- MW3 supplies these as attributes of the exported 3D GeoModeller layered mesh.

### GeoModeller - Potential field method of interpolation

- 3D implicit surfaces constrained by contacts & structural data **together**
- “co-kriging” Lajaunie et al. 1997
- a mathematical model
- contacts belong to iso-potential surfaces of a 3D scalar field
- dips are treated as gradients of the field
- 3D fault surfaces built same way  
(add discontinuous drift functions)



### MW3 advantages

MW3 provides an export mesh option which is important and of interest to all generic users of meshes, regardless of their chosen 3rd party simulation code.

It contains geological information and vector data as a proxy for sedimentary bedding, and delivers a 3D layered finite element mesh .fem file product.

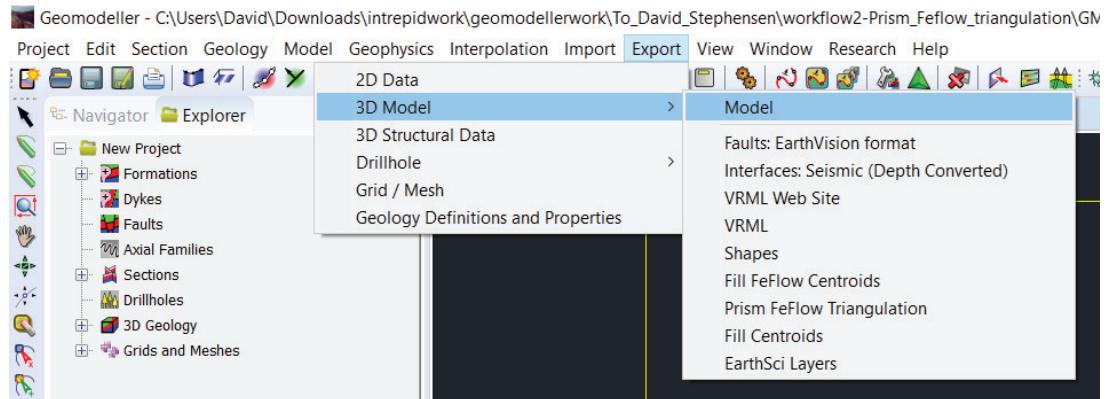
### Steps for MW3

You need to do this from the open workspace of your 3D GeoModeller model. No

external files are required to perform this meshing, it is a direct-export option from 3D GeoModeller.

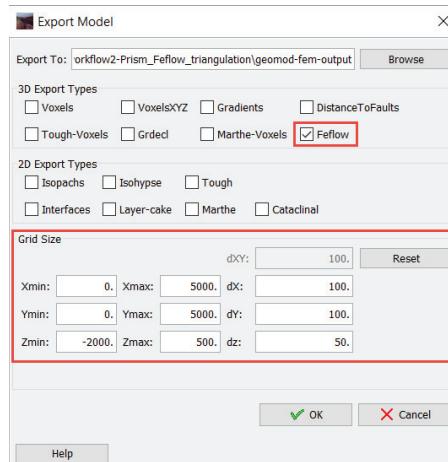
Follow these steps:

- 1 Open your 3D GeoModeller model (any model)
- 2 Compute your model
- 3 Select from the main menu: **Export > 3DModel > Model**



The **Export Model** dialog box appears

- 4 Browse to a suitable destination folder path, and Name your required output file
- 5 Check (tick) **FEFLOW** export and set your output resolution required.



- 6 Choose **OK**.
- 7 3D GeoModeller creates the mesh and stores the data in two files in the 3D GeoModeller project folder:
  - **geomod-fem-output\_3d.fem**
  - **geomod-fem-output\_triangles.data**

Name	Date modified	Type	Size
GM_Project	18/07/2017 6:03 PM	File folder	
geomod-fem-output_3d.fem	21/07/2017 4:32 PM	FEM File	44,912 KB
geomod-fem-output_triangles.data	21/07/2017 4:32 PM	DATA File	507 KB
small_test.fem	4/05/2017 3:56 PM	FEM File	35 KB
WetRiverValley.fem	4/05/2017 3:56 PM	FEM File	2,280 KB
WetRiverValley_3d.fem	21/07/2017 3:32 PM	FEM File	158,230 KB

**8** Output file is called \*\_3D.fem. Ready for simulation is a code of your choice.

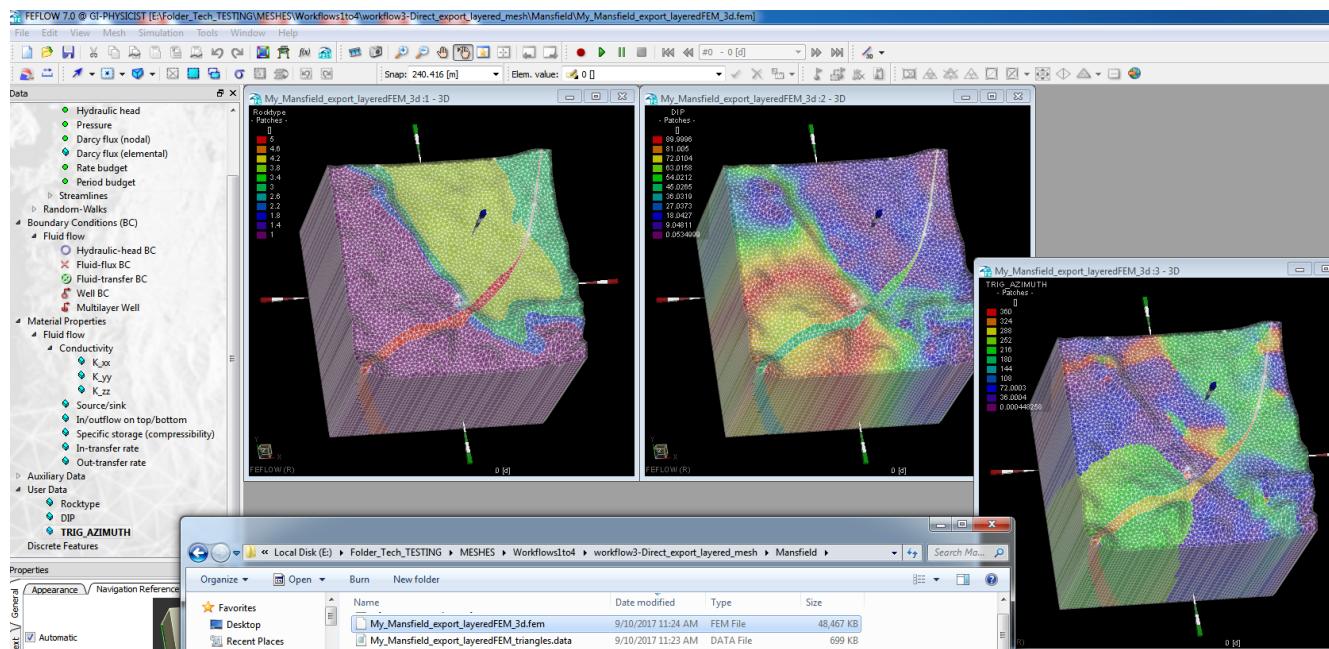
A video demonstration of this workflow (MW-3) can be found on YouTube:

Direct export of a prismsed triangulation of topography from GeoModeller  
(Layered finite elements mesh - fem)

<https://www.youtube.com/watch?v=sJpu-0nkgI4>

### GeoModeller Direct Export of layered fem files

In FEFLOW: Activate a 3D viewer and Double-click each User Data type in turn:  
"Rocktype" and "Dip" and "Trig-Azimuth"



## Meshing workflow 4 (MW4)—Directly export a 3D GeoModeller fully unstructured mesh

[Parent topic:](#)

[Adding geological information to a simulation-ready mesh](#)

### MW4 action summary

Directly export a 3D GeoModeller model to a fully unstructured mesh

#### About MW4

This workflow produces a fully unstructured mesh by direct export

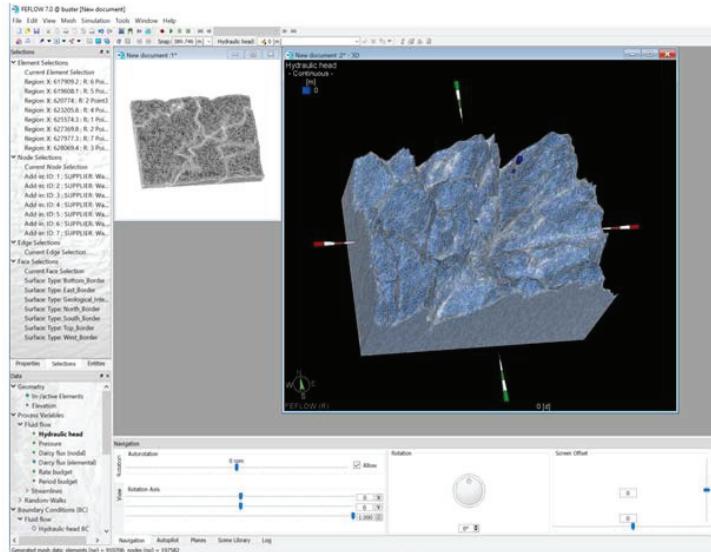
MW4 provides an export mesh option which is important and of interest to all generic users of meshes regardless of their chosen 3rd party simulation code.

It delivers a fully unstructured finite element mesh. The file types currently supported are:

- **.VTK** files—a generic option
- **.medit** files—a generic option
- **.GMOD** with **.dat** auxillary files—a *FEFLOW* option

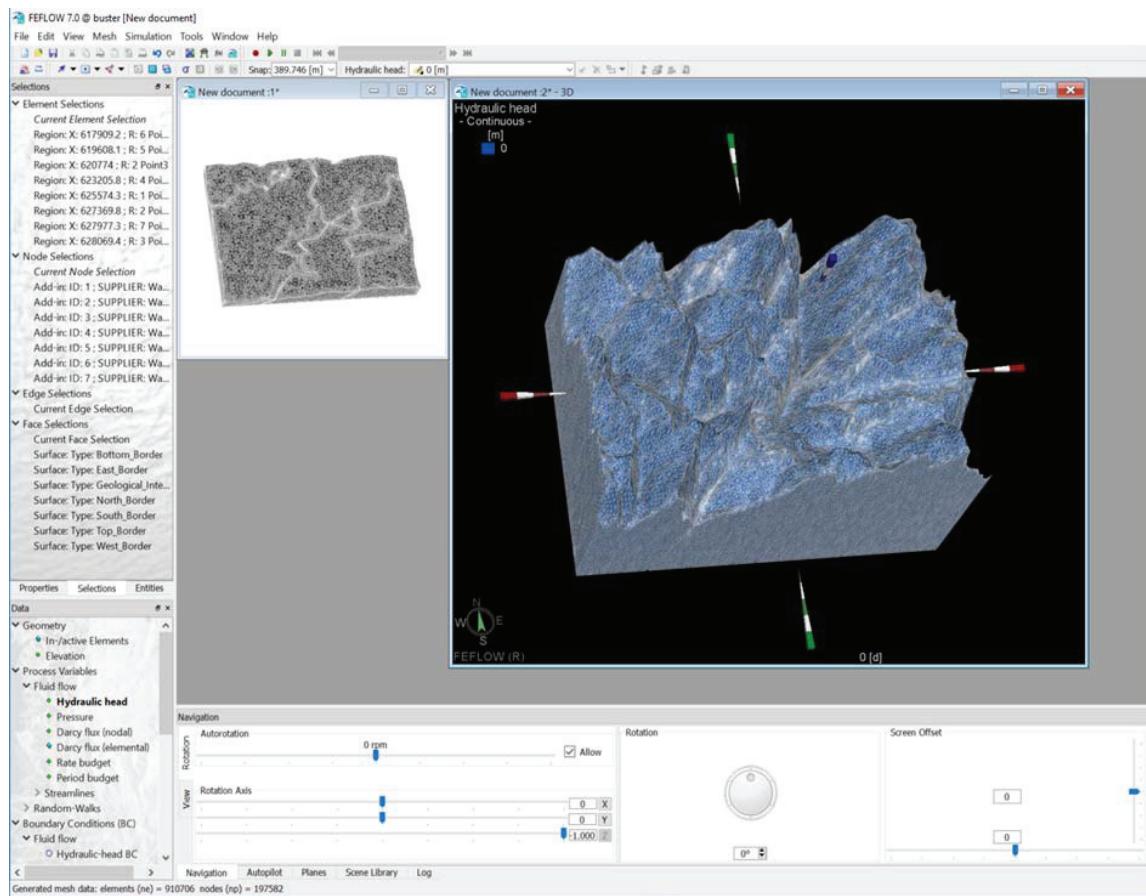
MW4 instructs 3D GeoModeller to access CGAL libraries (available as an add-on to 3D GeoModeller-Base version) to perform tetrahedral & triangulated meshing of the current implicit 3D geology and structural model.

### Fully unstructured meshes export from GeoModeller



- With CGAL libraries for tetrahedron meshing
- user- controls for adaptive mesh (coarse or fine per geology unit)
- Water tight & manifold
- thin bodies, pinch outs, dipping faults (spatially limited or infinite faults)
- 30x reduction in the computational simulation problem: By much reduced # of equations for elements & nodes – detail where you want it





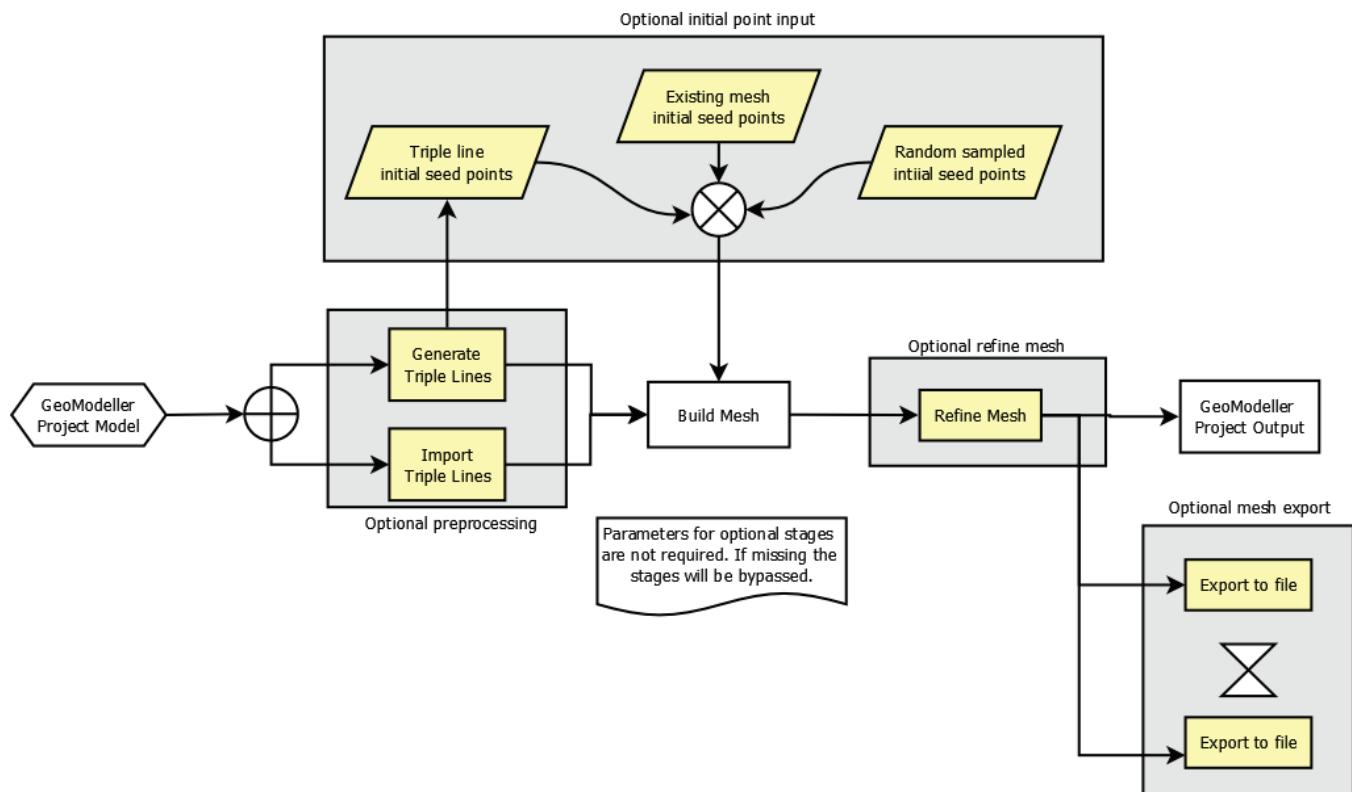
For further detailed information about the specifics of the 3D GeoModeller approach to creating fully unstructured meshes, see [Addendum—flow diagram and parameters of MW4](#).

### MW4 advantages

Advantages of the 3D GeoModeller fully unstructured mesh outputs are:

- CGAL libraries facilitate tetrahedron meshing (naturally more adaptive).
- Unstructured meshing facilitates a 30x reduction in the computational simulation problem - by enabling much reduced numbers of equations for elements & nodes – detail where you want it !
- User- controls are available in the graphic user interface for further adaptive meshing options (coarse or fine elements can be set per geology unit)
- Meshes are water tight & manifold
- Thin bodies, pinch outs, dipping faults (and spatially limited or infinite faults) - all are being supported, with the existing (and ongoing development) of mesh repair routines in the GeoModeller code.

## Flow diagram for MW4



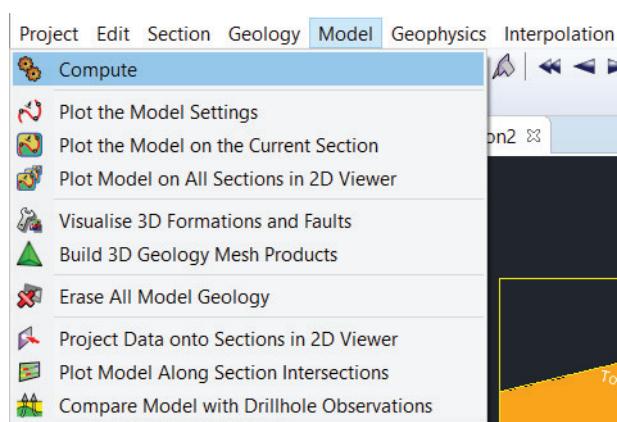
### Steps for MW4

You need to carry out MW4 from the open workspace of your 3D GeoModeller model. **No external files are required to perform this meshing, it is a direct-export option from 3D GeoModeller.**

The diagram below illustrates the stages of the meshing process.

Follow these steps:

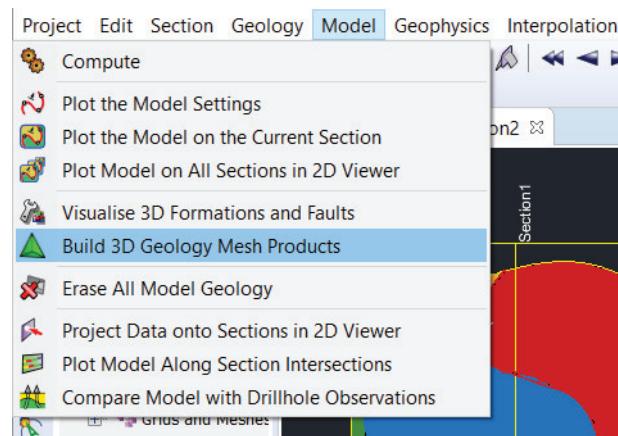
- 1 Open a 3D GeoModeller project
- 2 Make sure that the model is computed:
  - 1 Choose the gears icon or Main menu: **Model > Compute**



- 2 Select all of the relevant features and choose **OK** to compute

### 3 Start the CGAL Meshing Wizard.

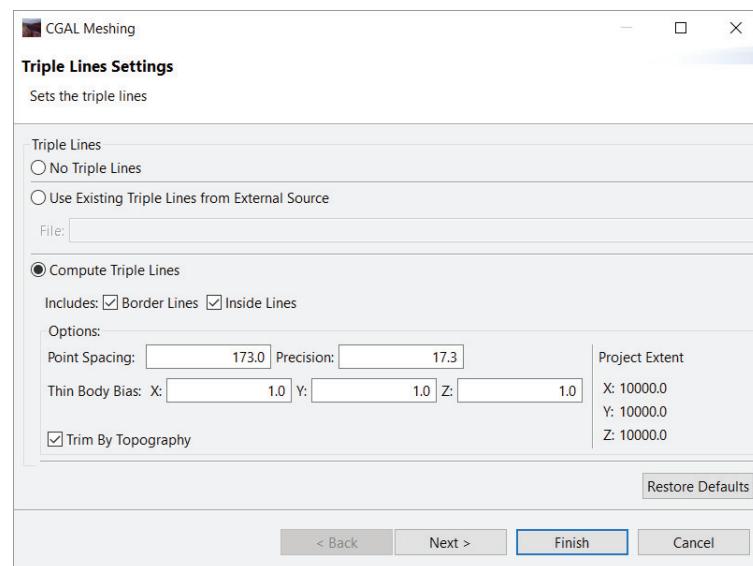
Click the green tetrahedron icon on the main toolbar or choose **Model > Build 3D Geology Mesh Products**



The **CGAL Meshing Wizard** appears.

Go through each page of the **CGAL Meshing Wizard** and enter the desired settings as described in the following steps.

### 4 Enter the **Triple Lines** settings.

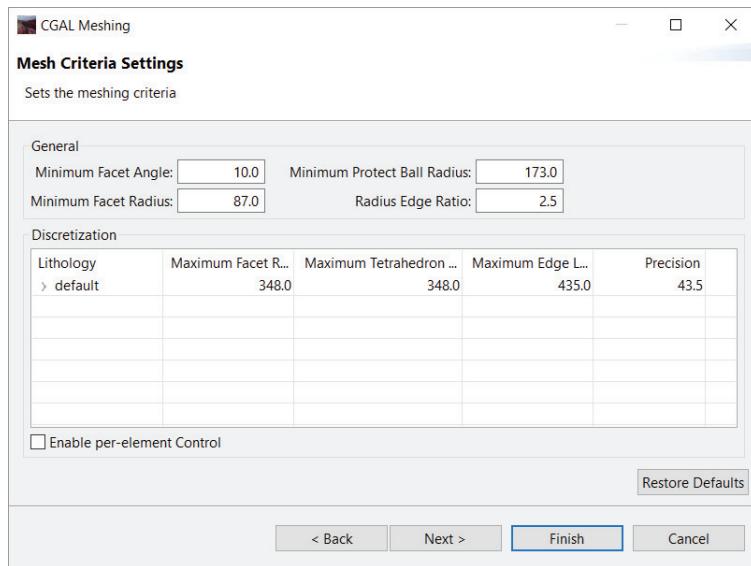


You can:

- Use existing triple lines from a file OR
- Compute the triple lines during meshing

Choose **Next**

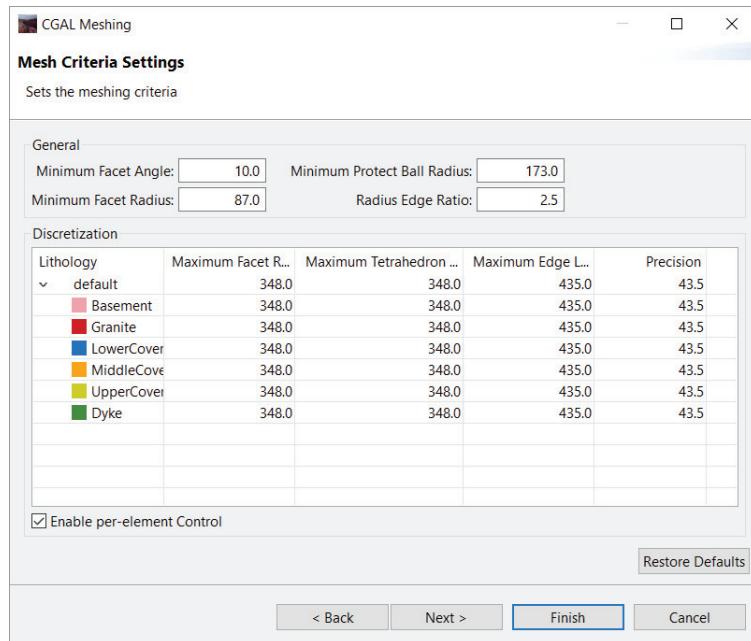
## 5 Enter the Mesh Criteria settings.



Set your **General meshing** parameters. These are applied to all lithologies.

Select your **Default discretisation**. These are applied to all lithologies.

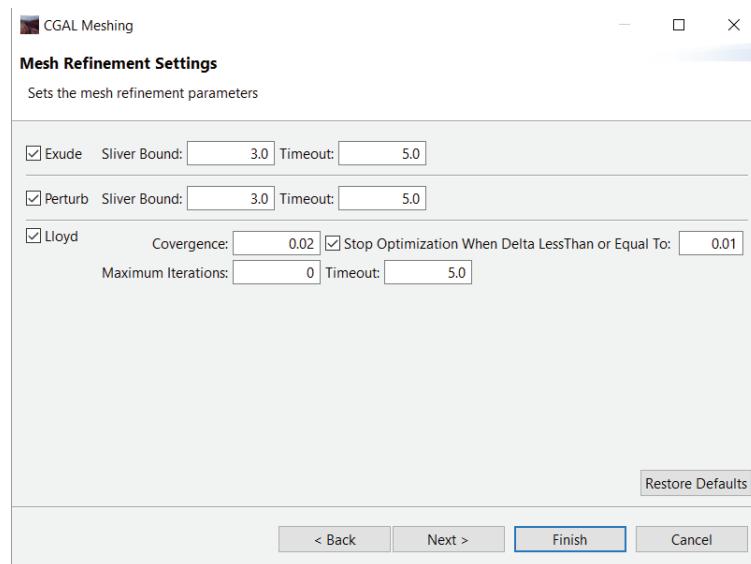
If you want to change the discretisation meshing parameters for individual lithologies check **Enable per-element Control**



## Choose Next

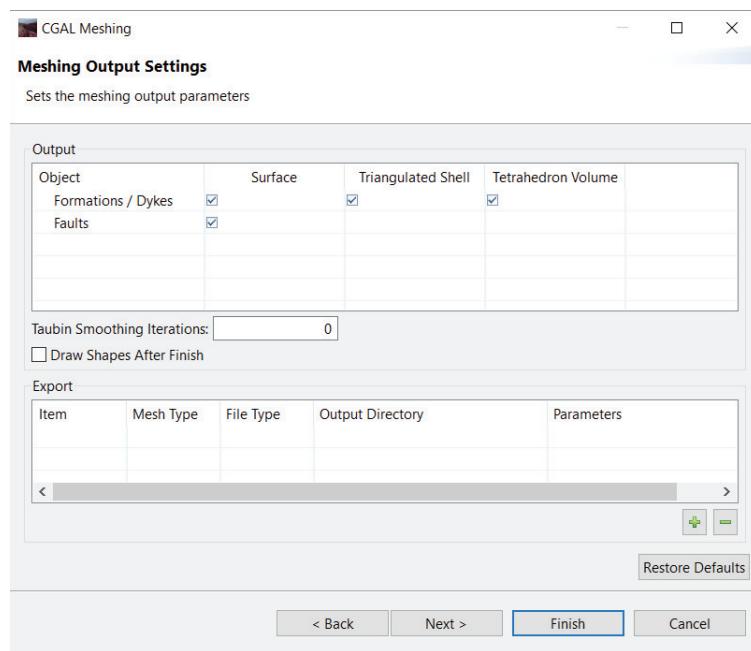
## 6 Enter the **Mesh Refinement** settings.

**Refinement options** are disabled by default. Check the corresponding boxes to enable them and then edit settings as required.



Click **Next**

## 7 Enter the **Meshing Output** settings.



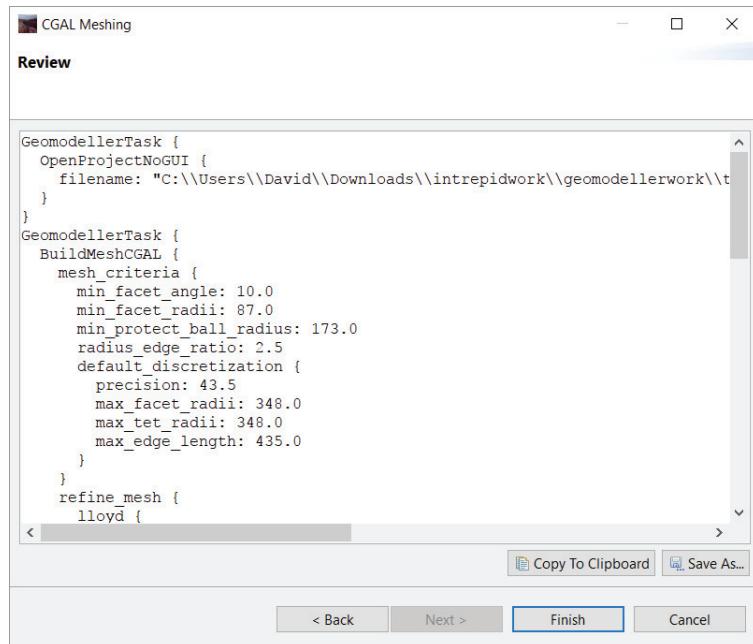
Select the required objects in output field.

Add meshes to the **Export** table by clicking the green **+** button in the lower right corner. **Note:** You must set at least one.

Change the mesh file type as desired (for example, **feflow**, **vtu**, **medit**)

Enter an output directory for each mesh export. **Note:** Do not leave it as the default setting.

- 8** If you want to review the automatically generated task file before export, choose **Next**.



- 9** Choose **Finish** to compute and export the mesh.

#### Addendum—flow diagram and parameters of MW4

This section provides a description of the parameters used by the Delaunay meshing algorithms available to 3D GeoModeller. The CGAL library is used by 3D GeoModeller for Delaunay based meshing. The Delaunay methods have a number of stages from pre-processing the implicit model, mesh generation, post-processing mesh refinement and finally output and export to file. Some stages are optional. The schema is organised into parameter groups that control aspects of a particular stage.

##### **build\_triple\_lines**

Triple lines are used to define 1-features, or the intersection line between three sub-domains, where a sub-domain is either a geological unit or project bounding box extent.

Parameter	Default Value	Description
<b>topo_limited</b>	true	Clip meshing at the modelled topography section. When true all points above <b>topo</b> will be treated as outside the meshing domain and no triple point will be computed. When false triple lines will be generated on the project bounding box above <b>topo</b> .
<b>border_line</b>	true	Generate 1-features where geology intersects with the project bounding box.
<b>inside_line</b>	true	Generate 1-features where three or more geology units intersect

Parameter	Default Value	Description
<code>thin_body_bias</code>	vector of x, y, z scale	When doing a near surface mesh with many thin , near horizontal bodies, increase the vertical sampling by 20
<code>point_spacing</code>	1% of the diagonal of project bounding box	The spacing between discrete sections used to sample the model.
<code>precision</code>	10% of point_spacing	The precision with which 3D GeoModeller model is sampled on sections.
<code>divide_model_using_regular_sections</code>	true	Use regular 3D GeoModeller sections when dividing and sampling the model. When false CGAL sections will be used
<code>export_triple_lines</code>	object	The message defining parameters to use when exporting triple lines to a VTK polydata or ASCII format. When not present no export will occur.

#### `build_triple_lines::export_triple_lines`

Parameter	Default Value	Description
<code>out_directory</code>	"."	The directory where exported triple line files will be saved.
<code>format</code>	<code>TripleLine_ASCII_CGAL</code>	The output file format, either VTK or ASCII. The output data will be polylines defined by the geology triple line interfaces.

#### `import_triple_lines`

Parameter	Default Value	Description
<code>filename</code>	<code>triple_lines.txt</code>	The full path to the triple line file. The file is expected to be complete, i.e. no additional triple lines will be created.

**geomodeller\_output**

This block controls the data that will be placed into the 3D GeoModeller project once meshing is complete. Note, only volume (triangulated shell) and surface can be added to the 3D shapes menu in a 3D GeoModeller project. Tetrahedral volume meshes are added as MeshGrid objects in the project tree.

Parameter	Default Value	Description
<b>output_directory</b>	n/a	<unused>
<b>smooth</b>	0	The number of Taubin smoothing iterations to apply to the mesh prior to placing into 3D GeoModeller. <b>Note</b> this may break the Delaunay properties of the mesh.
<b>draw_shapes</b>	false	When false shapes will not be displayed automatically. When true the shapes added to the project will be displayed. Generally this should be true when invoked from within 3D GeoModeller itself and displayed objects are available. In batch this should be false.
<b>formations</b>	true	Generate 3D GeoModeller output meshes of the formations. This is irrespective of the type of mesh generated.
<b>faults</b>	true	Generate mesh surfaces of all faults.
<b>surfaces</b>	true	Generated surface meshes of formations or faults, depending on the <b>formations</b> and <b>faults</b> flag settings.
<b>volumes</b>	true	Generate triangulated shell meshes of formations if the <b>formations</b> flag is set.
<b>tetrahedrons</b>	true	Generate MeshGrid tetrahedral volume meshes of formations.

**`mesh_criteria`**

The `mesh_criteria` parameters control the quality of mesh elements and the time required to generate the mesh. A higher quality will come at the expense of more time required. The mesh criteria parameters consist of global and per-formation parameters. A default discretization block is mandatory and used when any formation is not listed explicitly.

Parameter	Default Value	Description
<code>min_facet_angle</code>	10.0	Controls the shape quality of mesh facets. Specifies the minimum angle of any triangle facet in the mesh.
<code>min_facet_radii</code>	0.5% of the diagonal of the project bounding box	Controls the smallest facet accepted by the meshing algorithm. The facet radii is the radius of the facet circumcircle, which is the smallest circumscribing circle formed from the three triangle facet vertices.
<code>min_protect_ball_radius</code>	1% of the diagonal of the project bounding box	Controls the minimum distance between any non-triple line point and the triple lines, in effect setting a minimum size of elements with one or more vertices on a triple line. Note other mesh criteria have no effect within a protecting sphere radius. The image shows protecting spheres on the mesh surface.
<code>radius_edge_ratio</code>	2.5	Controls shape quality of mesh elements. This is the ratio between the tetrahedron radii and the shortest edge of the element. Large radius-edge ratio indicates a poor quality tetrahedron, except in the case of slivers which can have a low radius-edge ratio but are considered poor quality elements. The image shows the radius and shortest edge as <b>R</b> and <b>L</b> respectively. The radius-edge ratio is simple <b>R/L</b>
<code>default_discretization</code>	object	A group of parameters that provide the default mesh criteria for all formations that do not have an explicit discretization block.
<code>formation_discretization</code>	object	A group of parameters that provide the mesh criteria for a specific formation. The parameters are the same as the default discretization block.

```
mesh_criteria::default_discretization and
mesh_criteria::formation_discretization
```

The `default_discretization` is the set of parameters that will be used for all formations that do not have an explicit `discretization` block provided. The `formation_discretization` allows `mesh_criteria` to be specified for individual formations. Both parameter blocks contain the same parameters.

Parameter	Default Value	Description
<code>max_facet_radii</code>	<code>4 x min_facet_radii</code>	Controls the maximum facet size. The facet radii is the radius of the facet circumcircle, which is the smallest circumscribing circle formed from the three triangle facet vertices.
<code>max_tet_radii</code>	same as <code>max_facet_radii</code>	Controls the maximum tetrahedron size. The tetrahedron radii is the radius of the circumsphere, which is the smallest circumscribing sphere formed from the four tetrahedron vertices.
<code>precision</code>	0.5 of <code>min_facet_radii</code>	Controls the approximation error between surface facets and the sub-domain boundaries. This parameter specifies the maximum distance between a surface facet centre and its Delaunay ball circumsphere centre. The Delaunay ball centre lies on the domain boundary and the surface facet centre is an approximation to that boundary.
<code>max_edge_length</code>	<code>5 x min_facet_radii</code>	Controls the maximum distance between consecutive protecting balls on 1-feature triple line segments. In the image here protecting spheres are shown as green spheres on the mesh surface.
<code>formation</code>	na	The formation that will use these parameters. <b>Note:</b> This is ignored for the <code>default_discretization</code> .

**`refine_mesh`**

Use this with caution

The `refine_mesh odt` and `lloyd` options may result in a mesh that is non-manifold and/or violates the Delaunay guarantees. The `exude` and `perturb` options do not have this limitation.

The refine mesh parameters are used to control the post-meshing optimization algorithms. These can be used to remove slivers and poor quality mesh elements. All optimization is performed while maintaining the Delaunay properties of the mesh.

Parameter	Default Value	Description
<code>lloyd</code>	object	When present, Lloyd optimization will be performed to smooth the mesh surfaces.
<code>odt</code>	object	When present ,Lloyd optimization will be performed to smooth the mesh surfaces.
<code>exude</code>	object	When present, sliver exude will be performed to remove slivers. The image shows sliver tetrahedrons
<code>perturb</code>	object	When present, mesh perturbation will be performed to remove slivers. The image shows sliver tetrahedrons

**`refine_mesh::lloyd` and `refine_mesh::odt`**

These two blocks have the same parameters and control an energy minimisation algorithm that will move vertices to smooth the mesh surfaces.

Parameter	Default Value	Description
<code>time_limit</code>	5	The maximum time in seconds to perform the refinement. A limit of 0 is unbounded
<code>max_iter</code>	0	The maximum number of iterations. A value of 0 will be unbounded
<code>convergence</code>	0.02	The target energy to reach.
<code>freeze_bound</code>	0.01	Any consecutive iteration with a delta less than or equal to <code>freeze_bound</code> will stop the optimization.
<code>do_freeze</code>	true	Use the <code>freeze_bound</code> to halt the algorithm

```
refine_mesh::exude and refine_mesh:perturb
```

These two blocks have the same parameters and control the vertex optimization.

Parameter	Default Value	Description
<b>time_limit</b>	5	The time in seconds to run the algorithm. A value of 0 will be unbounded.
<b>sliver_bound</b>	3 degrees	The minimum value of the smallest dihedral angle in degrees that should constitute a sliver. The smallest dihedral angle is the smallest angle between two facets of the tetrahedron.

```
initial_points
```

This block provides the parameters for generating the initial mesh points. When triple lines are available the triple line points are used as seed points for the meshing algorithm by default.

Parameter	Default Value	Description
<b>use_triple_points</b>	true	When true the triple line points will be used to seed the mesh algorithm. It is an error to set this to true when there is no <b>build_triple_lines</b> or <b>import_triple_lines</b> parameters specified.
<b>random_points</b>	object	When present the algorithm will be seeded with randomly sampled points.
<b>mesh_points</b> (repeated block)	object	When present a <b>mesh_points</b> block will specify an existing VTP mesh file to use as seed points for the meshing algorithm. Multiple <b>mesh_points</b> blocks can be defined.

```
initial_points::random_points
```

Defines the parameters to use for random initial point sampling.

Parameter	Default Value	Description
<b>count</b>	12	Defines the number of random samples.

**`initial_points::mesh_points (repeat block)`**

Defines a VTP mesh file that contains initial seed points for the meshing algorithm. Preprocessing can be applied to the input mesh to clean and decimate to reduce the number of points while optionally maintaining mesh topology.

Parameter	Default Value	Description
<code>filename</code>	empty-string	The VTP mesh file name
<code>clean_mesh</code>	object	Clean the mesh prior to extracting points. This block holds the clean mesh algorithm parameters. An empty block will use default parameters.
<code>decimate_mesh</code>	object	Decimate the mesh prior to extracting points. This block holds the decimate mesh algorithm parameters. An empty block will use default parameters.

**`VTK QuadricDecimation (initial_points::mesh_points::decimate_mesh)`**

Defines the parameters to use for mesh decimation.

Parameter	Default Value	Description
<code>target_reduction</code>	0.9	Set the desired reduction (expressed as a fraction of the original number of triangles). The actual reduction may be less depending on triangulation and topological constraints.
<code>use_attribute_error_metric</code>	false	Decide whether to include data attributes in the error metric. If off, then only geometric error is used to control the decimation. By default the attribute errors are off.
<code>use_volume_preservation</code>	false	Decide whether to activate volume preservation which greatly reduces errors in triangle normal direction. If off, volume preservation is disabled and if <b>AttributeErrorMetric</b> is active, these errors can be large. By default <b>VolumePreservation</b> is off the attribute errors are off.
<code>use_scalars_attribute</code>	true	If attribute errors are to be included in the metric (i.e., <b>AttributeErrorMetric</b> is on), then the following flags control which attributes are to be included in the error calculation. By default all of these are on.
<code>use_vectors_attribute</code>	true	If attribute errors are to be included in the metric (i.e., <b>AttributeErrorMetric</b> is on), then the following flags control which attributes are to be included in the error calculation. By default all of these are on.
<code>use_normals_attribute</code>	true	If attribute errors are to be included in the metric (i.e., <b>AttributeErrorMetric</b> is on), then the following flags control which attributes are to be included in the error calculation. By default all of these are on.

Parameter	Default Value	Description
<code>use_tcoords_attribute</code>	true	If attribute errors are to be included in the metric (i.e., <code>AttributeErrorMetric</code> is on), then the following flags control which attributes are to be included in the error calculation. By default all of these are on. The tcoords are the texture coordinates at each vertex.
<code>use_tensors_attriubte</code>	true	If attribute errors are to be included in the metric (i.e., <code>AttributeErrorMetric</code> is on), then the following flags control which attributes are to be included in the error calculation. By default all of these are on.
<code>scalars_weight</code>	1.0	Set the scaling weight contribution of the attribute. These values are used to weight the contribution of the attributes towards the error metric.
<code>vectors_weight</code>	0.0	Set the scaling weight contribution of the attribute. These values are used to weight the contribution of the attributes towards the error metric.
<code>normals_weight</code>	0.0	Set the scaling weight contribution of the attribute. These values are used to weight the contribution of the attributes towards the error metric.
<code>tcoords_weight</code>	0.0	Set the scaling weight contribution of the attribute. These values are used to weight the contribution of the attributes towards the error metric.
<code>tensors_weight</code>	0.0	Set the scaling weight contribution of the attribute. These values are used to weight the contribution of the attributes towards the error metric.
<code>use_boundary_constraints</code>	true	Apply weighting to free boundary edges.

**VTK CleanPolyData (initial\_points::mesh\_points::clean\_mesh)**

Defines the parameters available for the mesh clean option

Parameter	Default Value	Description
<b>tolerance_is_absolute</b>	false	By default <b>tolerance_is_absolute</b> is false and <b>tolerance</b> is a fraction of Bounding box diagonal, if true, <b>AbsoluteTolerance</b> is used when adding points to locator (merging)
<b>tolerance</b>	0.0	The tolerance in terms of fraction of bounding box length or absolute, depending the <b>tolerance_is_absolute</b> flag. This value defines the distance between elements to determine the degenerate case, e.g. a line less than tolerance will be considered degenerate and converted to a point, depending on the control flags values.
<b>convert_lines_to_points</b>	true	Turn on/off conversion of degenerate lines to points
<b>convert_polys_to_lines</b>	true	Turn on/off conversion of degenerate polygons to lines
<b>convert_strips_to_polys</b>	true	Turn on/off conversion of polygon strips to a single polygon.
<b>point_merging</b>	true	If on, a locator will be used, and points laying within the appropriate tolerance may be merged. If off, points are never merged. By default, merging is on.

**export\_mesh**

There can be repeated **export\_mesh** blocks in a parameter set, each for a different output.

Parameter	Default Value	Description
<b>output_directory</b>	" <b>export_mesh</b> "	The output directory into which exported mesh files will be stored
<b>model_name</b>	n/a	Specify which shapes to write, e.g. which formation shapes. If not present all will be exported.
<b>file_type</b>	VTU	The file format to export. See table below. See the latest schema <b>intrepid_source/cxx/common/schemas/gmtaskmodel.proto</b> for supported formats.
<b>mesh_type</b>	VOLUME	The type of mesh to export. Supported values are SURFACE, SHELL and VOLUME. Here VOLUME denotes a tetrahedral volume and shell is a 3D GeoModeller 3D shape "volume" (triangulated shell mesh). See table below.

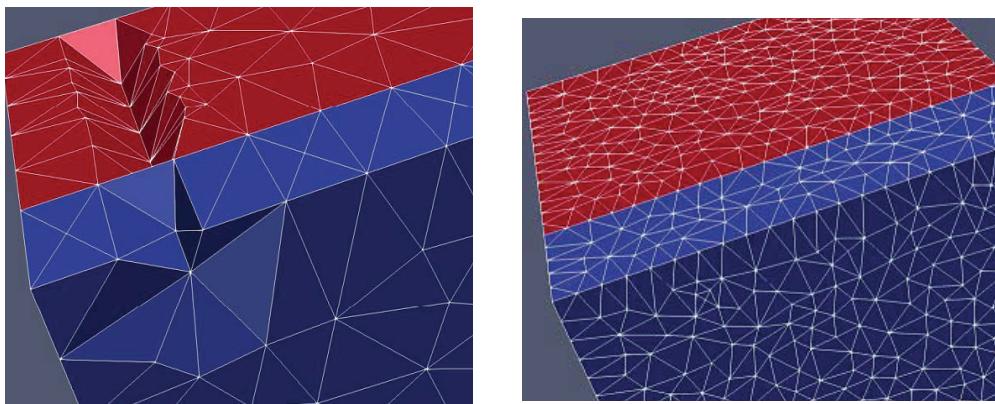
Parameter	Default Value	Description
<code>filterout_triangles_from_inner_tetrahedrons</code>	true	<i>FEFLOW</i> only: Do not explicitly list facets in the output file for inner tetrahedrons.
<code>wantTetrahedra</code>	true	<i>FEFLOW</i> only: Output tetrahedrons
<code>envelop_surfaces_only</code>	true	<i>FEFLOW</i> only: Only output surface shells
<code>Formation_IsoPotentials</code>	true	VTU only: Store the ISO potential for each point in the mesh
<code>Formation_GradientPotentials</code>	true	VTU only: Store the gradient of the potential function at each point in the mesh
<code>property</code>	repeated	VTU only: Store the formation properties listed in the repeated message.

***file\_type and mesh\_type mapping\****

File type	Surface	Shell	Volume
<b>TSURF</b>			
<b>VULCAN</b>			
<b>IGES</b>			
<b>STEP</b>			
<b>DXF12</b>			
<b>DXF13</b>			
<b>VTP</b>			
<b>STL</b>			
<b>VTU</b>			
<b>FEFLOW</b>			
<b>MEDIT</b>			
<b>C3T3</b>			
<b>COMSOL</b>			

## Factors Influencing Surface Mesh Artifacts

There can be competing influences between the various parameters above when dealing with complex geology. It is possible to have triple lines leave a pattern of divots depending on the parameters chosen. The following is a discussion and demonstration of a sensitivity analysis for a simple test project, the aim of which is to exercise all the important parameters and make a judgment call about acceptability of the resulting unstructured mesh.



Example of a bad divot in the model (left) and the fixed mesh (right)

(left) `divots_prec10_maxtet150_minball150_line100.vtu`

(right) `divots_prec10_maxtet150_minball50_line100.vtu`

### Simple Project

Box geology model with two formations, no faults.

Project bounds: 1000mx1000mx1200m (z range: -1000m to 200m)

Fake triple line introduced at -100 below topo running from x=0 to x=1000 with spacing of 50m between samples, y fixed at 200m

### NOTES

precision and min sphere radius fix divots with minimum impact on runtime performance, when both at 45 they override all other parameters w.r.t divot removal.

max\_facet\_radii provides some improvement but is more targeted towards cell quality than interface precision

max\_tet\_radii provides divot improvements but has the biggest impact on runtime performance. edge-length has some effect on divots but is outweighed.

Line distance	Min Sphere	Precision	Max Edge	Max tet radii	Max facet radii	Result (Pass/Fail)
100	500	500	500	150	150	F
100	10	500	500	150	150	F
100	500	10	500	150	150	F
100	500	500	10	150	150	F
100	250	250	250	150	150	F
100	10	250	250	150	150	F
100	250	10	250	150	150	F
100	250	250	10	150	150	F
100	125	125	125	150	150	F
100	10	125	125	150	150	F
100	125	10	125	150	150	F
100	125	125	10	150	150	F
100	75	75	75	150	150	F
100	10	75	75	150	150	F
100	75	10	75	150	150	PASS
100	75	75	10	150	150	F
100	50	50	50	150	150	F
100	10	50	50	150	150	PASS
100	50	10	50	150	150	PASS
100	50	50	10	150	150	F
100	45	45	50	150	150	PASS
100	45	45	500	150	150	PASS

Results indicate precision and min-sphere radius relative to line distance are significant divot factors.

Max Tet Radii (NOTE: for all max\_facet\_radii settings a minball & precision of 45 remove divots, overriding facet radii setting)

100	50	50	150	50	150	F
100	50	50	150	45	150	PASS
100	75	75	150	45	150	PASS
100	100	100	150	45	150	F

Max facet radii (NOTE: for all max\_facet\_radii settings a minball & precision of 45 remove divots, overriding facet radii setting)

100	50	50	50	150	50	PASS
100	50	50	150	150	75	F
100	75	75	150	150	50	PASS
100	100	100	150	150	50	F

## Summary of Successful and Failed Meshing Parameters

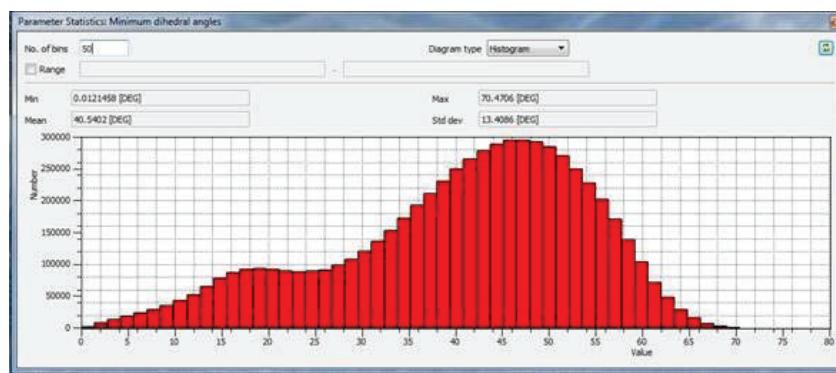
## Factors Influencing Mesh Quality

For simulation purposes, there can be restrictions on the shape of the cells when solving partial differential equations for such things as Darcy's Law. This doesn't apply when you have regular grids as all cells are orthogonal.

In the case of unstructured tetrahedra meshes, the controlling attribute is generally the dihedral angle. For instance some numeric codes become unstable if there are cells with dihedral angles less than 15 degrees.

It turns out with all the options available to manage the creation of unstructured meshes that this extra requirement cannot be achieved without sacrificing the requirement to have continuous smoothed surfaces joins as delivered by the triple line technology.

The triple lines are designed to eliminate what have been called "Shark's Teeth" on the mesh edges, by not using triple lines and having a further relaxation pass over the mesh using the "ODT" options the minimum dihedral angle achieved can be greatly improved. However the tough requirement of having manifold and water-tight meshes as well as no mesh cells less with a dihedral angle less than 15 degrees is right on the cusp of what is mathematically possible when it comes to arbitrarily complex geological models.



Histogram of minimum Dihedral Angles