Tutorials and documentation for FacetModeller

Peter G. Lelièvre

May 17, 2021

This document is a work-in-progress. Please contact me at plelievre@mun.ca if something needs updating, you need something explained further, or for any suggestions or requests.

Contents

I Preliminaries 4

1 Policies 5
1.1 (Dis)claimer . . . 5
1.2 Conditions of use . . . 5

II FacetModeller (model building application) 6

2 FacetModeller preliminaries 7
2.1 About FacetModeller . . . 7
2.2 Policies and conditions of use . . . 7
2.3 Obtaining, installing and running FacetModeller . . . 7
2.3.1 Obtaining Facetmodeller . . . 7
2.3.2 Installing Facetmodeller . . . 7
2.3.3 Running FacetModeller . . . 8
2.4 About these tutorials . . . 8
2.5 The model building philosophy of FacetModeller . . . 8

3 Tutorial - Building 2D models with FacetModeller 10
3.1 Launching FacetModeller . . . 10
3.2 Opening your image file . . . 10
3.3 Defining groups . . . 11
3.4 Saving your progress . . . 12
3.5 Calibrating the image . . . 13
3.6 Adding nodes . . . 13
3.7 Defining facets . . . 15
3.8 Defining regions . . . 16
3.9 Exporting your model . . . 17
3.10 Quitting FacetModeller . . . 18

4 Tutorial - Building 3D models: I 19
4.1 Running FacetModeller . . . 19
4.2 Opening your image files . . . 19
4.3 Defining the volume of interest (VOI) . . . 19
4.4 Defining groups . . . 20
4.5 Calibrating the images . . . 21
4.6 Adding nodes . . . 22
4.7 Defining facets between parallel sections . . . 23
4.8 Defining facets between non-parallel sections . . . 25
4.9 Defining facets on the boundary . . . 27
4.10 Defining regions . . . 29
4.10.1 Defining regions on sections . . . 29
4.10.2 Defining regions on facets . . . 29

5 Tutorial - Building 3D models: II 30
5.1 Requirements . . . 30
5.2 Preliminary topography processing . . . 30
5.3 Defining the padded VOI in 2D . . . 31
5.4 Defining the padded VOI in 3D . . . 36

6 Tutorial - Building 3D models: III **41**
6.1 Preliminary instructions 41
6.2 Opening topography files 42
6.3 Adding nodes 43
6.4 Changing node groups 44
6.5 Defining facets 44
6.6 Finishing the model 47
6.7 Wrap-up 48

Part I

Preliminaries

Chapter 1

Policies

The following general polices pertain to use of the PODIUM and MAGNUM program suites, the FacetModeller utility and any other software described in this documentation, from herein referred to as the "software". These policies should not discourage use of the software but, rather, are meant to clarify the conditions of use and to protect the developers' significant investment in the software code.

In the following, "I" refers to the primary developer, Peter Lelièvre, and "the developers" refers to those owners currently administrating the distribution and licensing of the software (currently Peter Lelièvre and Colin Farquharson).

The disclaimers and conditions below are revocable by the developers at any time and may be replaced by a formal license agreement.

1.1 (Dis)claimer

While the software described in this documentation can not be guaranteed in any way, I provide the following assurances:

- I will keep this document up-to-date with the current version of the software.
- Unless otherwise stated in this document, these programs have been thoroughly tested and see frequent use. I am happy to run further tests with any specified parameters as requested.
- I value direct and constructive feedback: please contact me at plelievre@mun.ca for any questions, suggestions or requests with respect the software.
- I will deal with any reported bugs or issues in a timely manner, as my affairs allow.
- I will provide basic user support for the software, provided users have first reviewed this documentation thoroughly.

1.2 Conditions of use

The software is freely available for academic or other non-commercial use. Other uses require expressed agreement with the developers. Use of the software requires that the developers will not be held responsible for any damages, monetary or otherwise, to the user caused directly or in part through the use of the software.

User responsibilities:

1. properly cite usage of the software in any publications resulting from its use (suggested citations are supplied in the individual chapters of this document)
2. inform the developers of any such publications
3. inform the developers of any bugs or issues encountered while using the software
4. inform the developers if the user changes institutional affiliation or email address
5. do not make available nor distribute all or part of the software, by any means, to anyone
6. do not redistribute username or password details for the repository that houses the software
7. users understand that the username or password details for the repository may change periodically for security reasons
8. completely and immediately delete all components of the software if requested by the developers
9. update the software frequently (as instructed in this documentation)
10. users agree that their name and institution or company position may appear on the developers' academic websites in a list of users of the software.

Part II

FacetModeller

Chapter 2

FacetModeller preliminaries

2.1 About FacetModeller

FacetModeller helps you build models comprising surfaces that represent contacts (interfaces) between rock units. The surfaces are represented by nodes (vertices) and connections between them (facets). These models are stored as planar straight line graphs (PSLGs) and piecewise linear complexes (PLCs) in 2D and 3D respectively. These are for use with meshing programs like Triangle and TetGen and are explained in more detail on these webpages:

- Triangle: 2D triangular meshing program
  http://www.cs.cmu.edu/~quake/triangle.defs.html#pslg
- TetGen: 3D tetrahedral meshing program
  http://wias-berlin.de/software/tetgen/1.5/doc/manual/manual002.html#sec7

Before using FacetModeller you should read those pages and understand the concept of a PSLG and PLC, and what other information is required for use of Triangle and TetGen.

FacetModeller was initially created to help students manually build 3D models from stacks of vertical cross-sections. It has grown considerably since then to help others with their modelling needs. FacetModeller is not intended to be a replacement for GOCAD (or any other 3D modelling programs) and it will never do some of the fancy automatic model building tasks that GOCAD can handle: FacetModeller is intended to be a simple-to-use, manual model-building program. Given that FacetModeller has grown from the needs of a small set of researchers, it may not meet all your needs. Therefore, **we encourage users to suggest additional functionality or other general improvements that would help you with your model building needs:** please contact the primary developer at plelievre@mun.ca.

2.2 Policies and conditions of use

Refer to Section 1 for general disclaimers and conditions of use.

FacetModeller has been submitted to the journal SoftwareX for publication. Until then, the suggested citation is:

- P. G. Lelièvre and C. G. Farquharson, 2015, FacetModeller: software for manual creation, manipulation and analysis of 3D surface-based models

2.3 Obtaining, installing and running FacetModeller

2.3.1 Obtaining Facetmodeller

Refer to Section **??** for general instructions on obtaining the software described in this documentation.

For those in my research group, please check our Wiki for more instructions for your particular platform:

http://www.esd.mun.ca/mediawiki-1.25.3/index.php/FacetModeller

2.3.2 Installing Facetmodeller

FacetModeller is a Java application. To run the application you must make sure you have the latest version of Java installed on your machine. Because the FacetModeller application is written in Java, it should in theory run on any platform that can run Java. We make efforts to test the application on various operating systems but without direct access to users' computing systems we can not guarantee that the application will execute flawlessly on your system.

2.3.3 Running FacetModeller

To run FacetModeller you must first find the `FacetModeller.jar` file: for those in my research group, use the version here:

```
[repos]/peter/NetBeansProjects/FacetModeller/dist/
```

where `[repos]` is the directory on your local machine that holds my repository directory.

For public users, use the version here:

```
[repos2]/facetmodeller/dist/
```

where `[repos2]` is the directory on your local machine that holds the public repository directory.

To launch FacetModeller,

- Run the `FacetModeller.jar` file by double clicking it, or from a terminal window using
  `java -jar FacetModeller.jar`
  or, for those in my research group, run it from the NetBeans IDE if you went that route.

2.4 About these tutorials

The best way to learn what you can do with the FacetModeller utility is to run through some quick tutorials. The following chapters provide tutorials that introduce the basic working concepts of the utility and some essential functionality. I have tried to design the application to be as intuitive as possible so hopefully you can figure out the functionality not explained here through your own exploration.

Although I try to update this documentation frequently, my code changes more frequently, so there may be differences between what you see in these tutorials and what you see when running a program. Please let me know if there are major inconsistencies that cause confusion. Also let me know if you have any suggestions that could improve the utility of these programs.

A paper on FacetModeller has been submitted to the journal SoftwareX. Once it has been accepted, new and improved tutorials will be developed based on the examples in that paper. Please stay tuned.

Throughout these tutorials you'll see some "asides", which look like ... well, like this! These can more-or-less be safely skipped if you want a faster introduction to the programs. However, I suggest you at least read them for the information provided, even if you don't follow the additional steps.

2.5 The model building philosophy of FacetModeller

FacetModeller uses a section-based model building philosophy. A section is simply a planar slice through a 3D model. For 2D models, there is only a single section. A section may be a vertical cross section, a horizontal depth section, or any planar rectangular section of arbitrary orientation.

Each node in a PSLG or PLC is attached to a single section. In a 2D model there is only a single section and all nodes are attached to that section. In a 3D model there are multiple sections and there will be different nodes attached to different sections. Nodes usually lie exactly on a section but in some cases they may lie off of the section and are projected onto it during visualization.

Facets are collections of nodes. In 2D models, facets are line elements that connect two nodes. In 3D models, facets are triangles or other planar polygonal shapes. If a facet's nodes are all attached to the same section then that facet will belong to a single section. If a facet's nodes are attached to different sections then that facet will belong to several sections.

Each watertight part of a model is a different "region" and use of some features of Triangle and TetGen requires that you provide a spatial location inside each of those different regions. FacetModeller stores each of those region specification points as a special kind of node and defining the regions is similar to defining nodes: each region is attached to a single section.

FacetModeller distinguishes different types of nodes, facets and regions by assigning them to different "groups". Each group has user-defined plotting colours associated with it. You can use the concept of groups to help you organize the different components in your model. For regions, the groups work like unique rock unit identifier integers. Actually, they are more like unique integers for each separate watertight region in a geological model (there may be multiple watertight regions that contain the same rock unit). Therefore, each group can only have one region.

Furthermore, each node, facet and region can only be assigned to a single group. In practice, this limitation means that you often have to define many groups. For example, consider the case where your model contains two intersecting surfaces: a vertical surface which we'll call "Surface V" and a horizontal surface which we'll call "Surface H". You will want some nodes associated with a group named "Surface V" and others associated with a group named "Surface H". You may also want the nodes along the intersection to be associated with a group named "Intersection H-V". That is simple enough to achieve in FacetModeller with three groups. However, now consider the same situation but you now want to be able to differentiate between nodes on "Surface V" that lie above the intersection, and nodes on "Surface V" that lie above the intersection, and similarly for the two sets of nodes on "Surface H" that lie on either side of the intersection. In FacetModeller you now need five different groups. This sometimes makes for a less-than-optimal organization of the model components. However, there are ways to merge groups, and to change node and facet group membership, which both help the process. Future development will look at ways to ease this limitation.

Chapter 3

Tutorial - Building 2D models with FacetModeller

This 2D tutorial provides basic information about FacetModeller. You should follow through this tutorial even if you are only interested in building 3D models, because the 3D tutorials rely on lessons learned here. **You should also make sure that you have read and understood the model building philosophy of FacetModeller in Section 2.5**.

3.1 Launching FacetModeller

To launch FacetModeller,

11. Run the `FacetModeller.jar` file by double clicking it, or from a terminal window using
    `java -jar FacetModeller.jar`
    or, for those in my research group, run it from the NetBeans IDE if you went that route.
12. Choose `2D` when asked what kind of model you want to create.

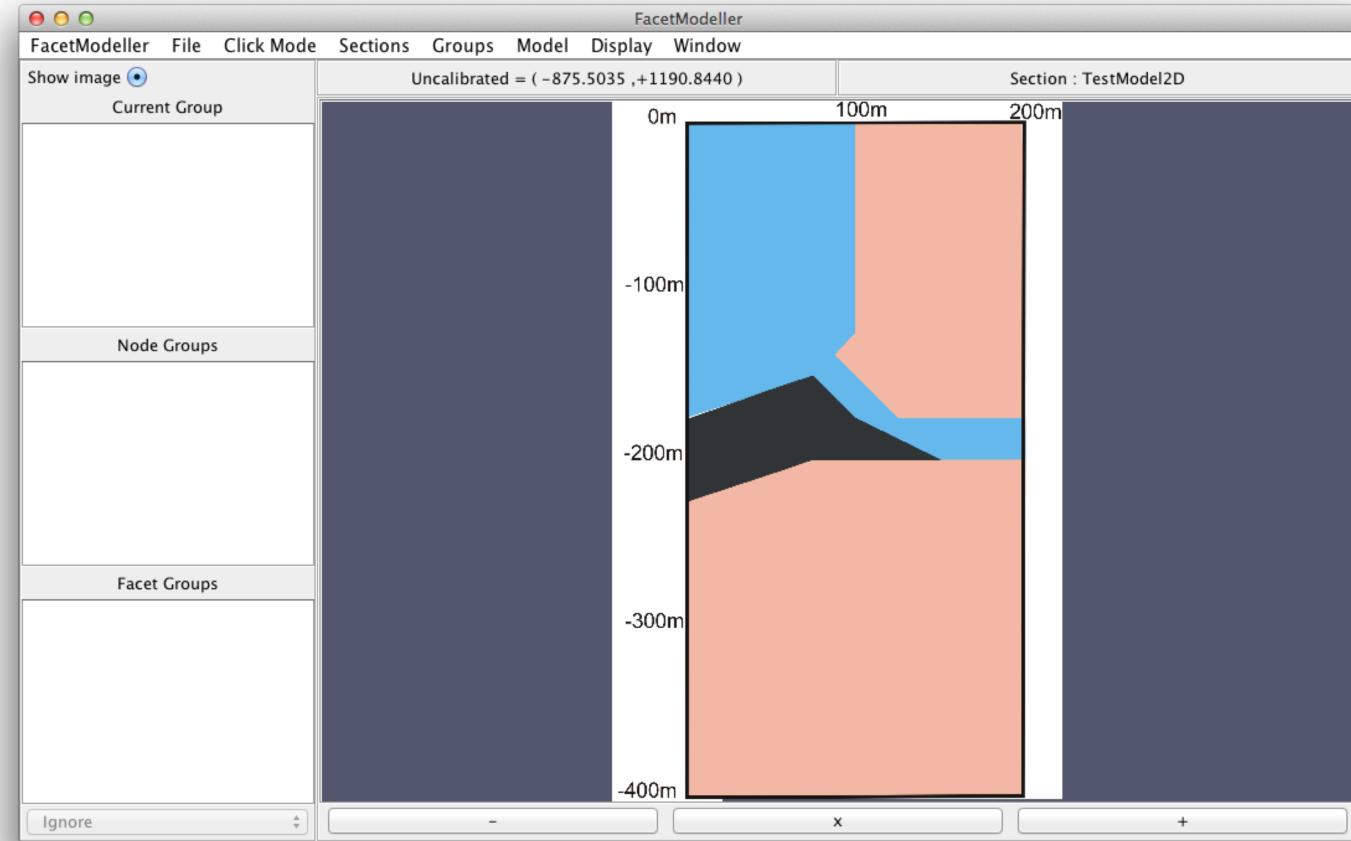
3.2 Opening your image file

The first step is to open the image file you want to discretize.

1. Select the menu item
   `File > Load cross section image file`
2. Select the file named `TestModel2D.png`. Those in my research group can find it here
   `[repos]/peter/NetBeansProjects/examples/FacetModeller2D/`
   and public users here
   `[repos2]/facetmodeller/Tutorial_2D/`
3. Close the confirmation that the file was loaded successfully by clicking `OK`.

> If the image represented a horizontal depth section, rather than a vertical cross section, then you would use the menu item
>
> `File > Load cross section image file`

You should now see something like this (being a Java application, it may look slightly different on different systems; I'm on a Mac):



There are 3 units in this model:

1. gneiss (the pink part of the image)
2. troctolite (the blue part of the image)
3. sulphides (the dark grey, almost black, part of the image)

3.3 Defining groups

You first need to define some groups so the nodes and facets that you define later can be associated with different groups. This isn't critical: you could just define some default group and use that for all nodes and facets. However, different groups of nodes and facets are coloured differently and this will help keep things clear. To define a group:

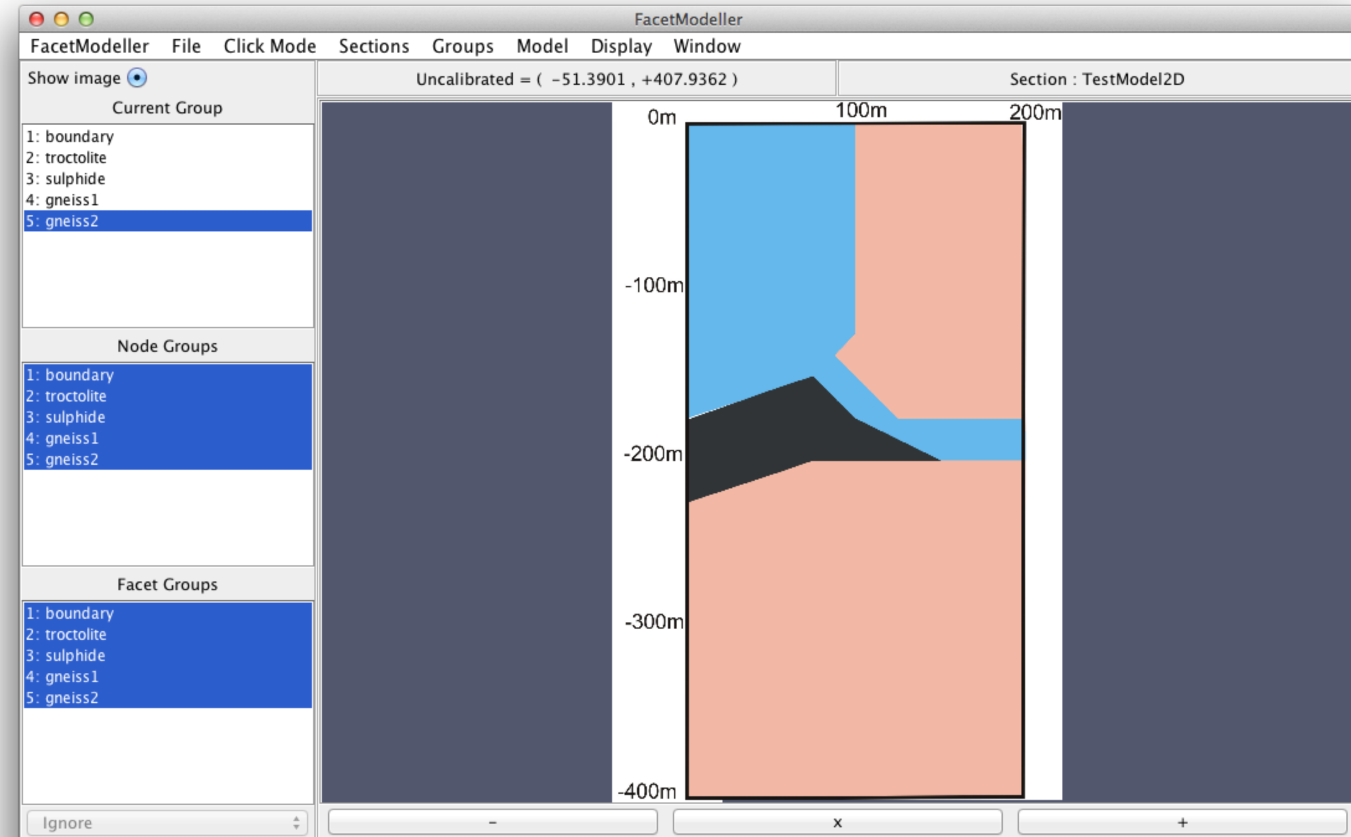
1. Select the menu item
   `Groups > New group`
2. Enter `boundary` as the name for this first group.
3. Select red as the drawing colour for the group.

> The group colour is used when drawing the nodes, facets and regions associated with the group. Initially, all three colours are set the same but they can be chosen separately using the various options in the `Groups` menu. Because you will be using this `boundary` group to define the boundary nodes, and the boundary in this image is black, red is a good choice because it will stand out against black. Of course you can use any colour you like in general.

Repeat those steps until you have defined the 5 groups with the names and colours indicated in the table below.

Name	Colour
boundary	red
troctolite	green
sulphide	white
gneiss1	black
gneiss2	black

You should now see those group names filled into the group selection boxes on the left. The group selection box at the top (labelled `Current Group`) allows you to specify which group you are currently working with, e.g. adding nodes or facets to. Only one group can be selected as the `Current Group`. The other two selection boxes (labelled `Node Groups` and `Facet Groups`) allow you to select which groups of nodes and facets are overlaid on the image. Multiple selections are allowed for the `Node Groups` and `Facet Groups` selection boxes. To select multiple items, use your shift or control/command keys while selecting the items.



3.4 Saving your progress

This is research code, still susceptible to bugs (but what isn't) so you should save early and often:

1. Select the menu item
   `File > Save session`
2. Select the location to save to and enter the name of the file to save. The file extension will be `.fms`.

> You may want to save different versions of your model as you build it. To change the name of the session file that you save to:
>
> 3. Select the menu item
>   `File > Save session as`
> 4. Select the location to save to and enter the new name of the file to save.

3.5 Calibrating the image

As you move your cursor around on the image you will see the location of the cursor indicated in a text box at the top of the FacetModeller window. Initially you will see "Uncalibrated" coordinates displayed in that text box, which are pixel coordinates for the image.

Before you can define nodes and facets, you must first calibrate the image. You will be instructed to click on two calibration points and enter their spatial coordinates in 3D.

Yes, I said 3D, and I mean 3D, despite this being a 2D model. I have done this to allow a 2D section to be registered in 3D space, in case you want to display it in a 3D viewer (such as ParaView) along with other spatially registered items. However, if you want a true 2D model, e.g. for performing 2D forward modelling, then you can simply choose to set the easting or northing values for the calibration points to equal values and FacetModeller will write a true 2D model when exporting files.

1. Select the menu item
   `Sections > Calibrate current section`
2. Close the first instruction dialog (instructing you to click on the first calibration point).
3. Click on the **top left** corner of the black boundary of the model.
4. Enter `"0 0 0"` (without the quotation marks) when asked for the easting, northing and elevation coordinates of that first calibration point.
5. Close the second instruction dialog.
6. Click on the **bottom right** corner of the black boundary of the model.
7. Enter `"200 0 -400"` as the coordinates of that second calibration point.

The image is now spatially calibrated. As you move your cursor around on the image you will see the calibrated spatial (easting, northing, elevation) location of the cursor indicated in the text box at the top of the FacetModeller window.

Here you used the axis labels present in the image to help with the calibration. Of course you won't always have that information on your image but you will at least need to know the coordinates of two landmarks in order to perform the calibration. Those landmarks must not lie at the same horizontal or vertical position!

3.6 Adding nodes

You are now ready to define the nodes in your model PSLG.

Before doing so, you may like to change some drawing defaults to make the node and facet overlays show up better:

1. Select the menu item
   `Display > Point size`
2. Enter `10`.
3. Select the menu item
   `Display > Line width`
4. Enter `2`.

You will first define some nodes around the boundary of the model:

5. Select the menu item
   `Click Mode > Add nodes`
6. Select the group named `boundary` in the `Current Group` selector box (any new nodes added will be assigned to the currently selected group).

7. Click around the boundary in the locations where you want new nodes. Make sure you add a node at the 4 corners and anywhere else on the boundary that intersects with rock unit contacts in the model. You should have a total of 9 nodes on the boundary (refer to the next screen-shot for guidance).

> As you click to add the boundary nodes, red node points will be overlaid on the image in the locations that you click. It may be a little fiddly: you have to press and release the mouse without the cursor moving (this is a safety precaution that I can remove if it annoys you).

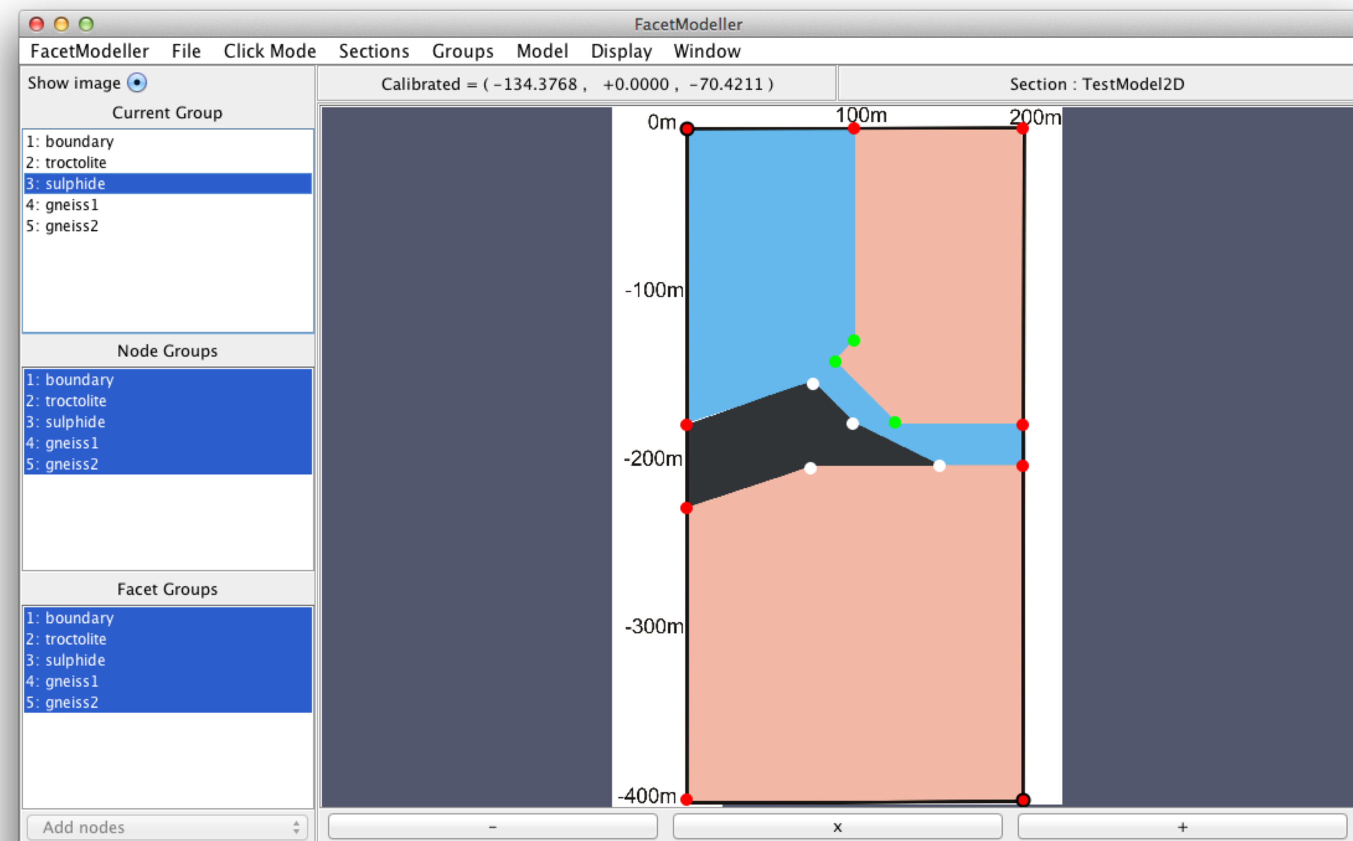
Now snap those boundary nodes to the calibration points, so that the boundary nodes will be exactly at the calibrated boundary:

8. Select the menu item
   `Model > Snap nodes to calibration points`

To finish the node definitions:

9. Select the `troctolite` group in the `Current Group` selector box.
10. Click to create the 3 nodes required to define the contact between the blue troctolite unit and the pink gneiss unit at the top right of the image (these nodes will be drawn green).
11. Select the group named `sulphide`.
12. Click to create the 4 nodes required to define the contact between the dark grey sulphide unit and the other units (these nodes will be drawn white).

Your model should now look like this:



> In the `Click Mode` menu you will see many different options that determine how to process left mouse button clicks. You will be introduced to many click modes in this tutorial. Be sure not to click in the image viewing panel by accident. You

can always select the `Ignore` click mode as a safety precaution (all mouse clicks are then ignored). The current click mode is displayed at the bottom left of the FacetModeller window.

3.7 Defining facets

You can now define facets. In 2D models, facets are line elements that connect two nodes. You will be using two different approaches to define new facets.

1. Select the menu item
   `Click Mode > Define facets node-by-node`
2. Select the group named `boundary` as the `Current Group` (as for nodes, any new facets will be added to the currently selected group).
3. Click on the first node on a facet, followed by the second. The new facet will now have been created and drawn red. Continue to define all 9 facets around the boundary of the model (refer to the next screen-shot for guidance). Remember that each facet in this 2D model contains only two nodes! Hence, for each facet you need to click on two different nodes. This means that, as you move around the boundary of the model, you may be clicking on the same node twice because it represents the second node in one facet and the first node in the next.

In this node-by-node click mode, you must have your cursor fairly close to the nodes in order to select them. When close enough, the node will be indicated by a white circle. The first node you click on will turn black, although that black circle will be hidden by the white "closest node" circle until you move your cursor away from the node. Once you click on the second node, the facet will be created and drawn.

Note the smaller black dots drawn in the centre of the facets: these are used as handles to the facets if you want to perform other processing with them, e.g. if you want to delete a facet or change the group membership of a facet.

If you make a mistake, just delete the facets you don't want (menu item `Click Mode > Delete facets`) and continue.

You will now define the facets for the contact between the blue troctolite and pink gneiss units. You will do so using a different click mode:

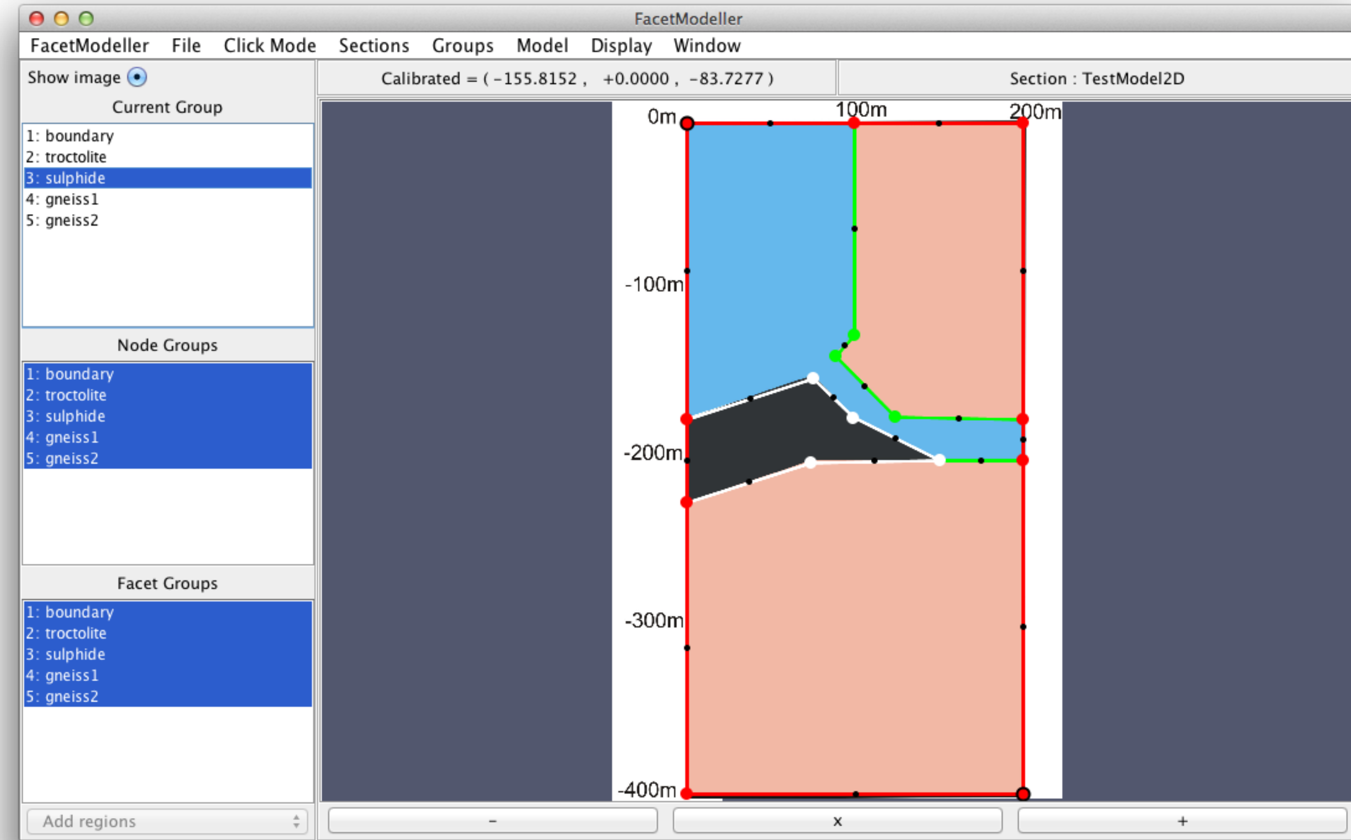
4. Select the menu item
   `Click Mode > Define linear edge facets`
   In this click mode, candidate facets are indicated as you move your cursor around on the image.
5. Select the group named `troctolite`.
6. Click when you see a facet that you wish to create. The new facet will be drawn green.
7. Continue defining the 5 facets for the troctolite-gneiss contact (there are 4 facets around the upper part of the pink gneiss unit and one facet below that connects to the dark grey sulphide unit).

As you move your cursor around in the linear edge facet click mode, candidate facets are indicated: black circles are drawn around a pair of nodes and a white dashed line is drawn between them.

The approach that the code takes to find a candidate node is a little complicated. It looks for the nodes within a particular distance from the cursor location and calculates all the possible node pairs; it then displays the facet with centroid closest to the cursor location. Hence, the candidate facet may jump around a lot as you move your cursor. The menu item `Display > Facet selection factor` allows you to change a special factor that affects the distance from the cursor location within which to search for nodes. You can increase or decrease this factor to include more or fewer nodes near the cursor location. You shouldn't decrease this factor below 1.0, and the higher it gets, the more nodes you'll catch. If the factor is set too high it may suggest more candidate facets than you would like. In fact, I have limited the factor to a maximum value of 2.0, but I can change that if needed.

Clearly, defining facets using this click mode is a little fiddly behind the scenes and it can be difficult to get exactly the facets that you want. The node-by-node approach is then available to you, but of course defining facets with that approach takes longer so I suggest use of the linear edge approach whenever possible.

To finish the model PSLG you need to define the 5 facets surrounding the dark grey sulphide body. Follow the steps above, using whatever click mode you prefer, but use the `sulphide` group. Your completed model PSLG should look like this:



> To view your model without the image displayed, unselect the radio button at the top left of the window labelled `Show image`. The outline of the image is then displayed instead of the image itself. This can be helpful when you have a more complicated image that might hinder you when defining the facets in your model.

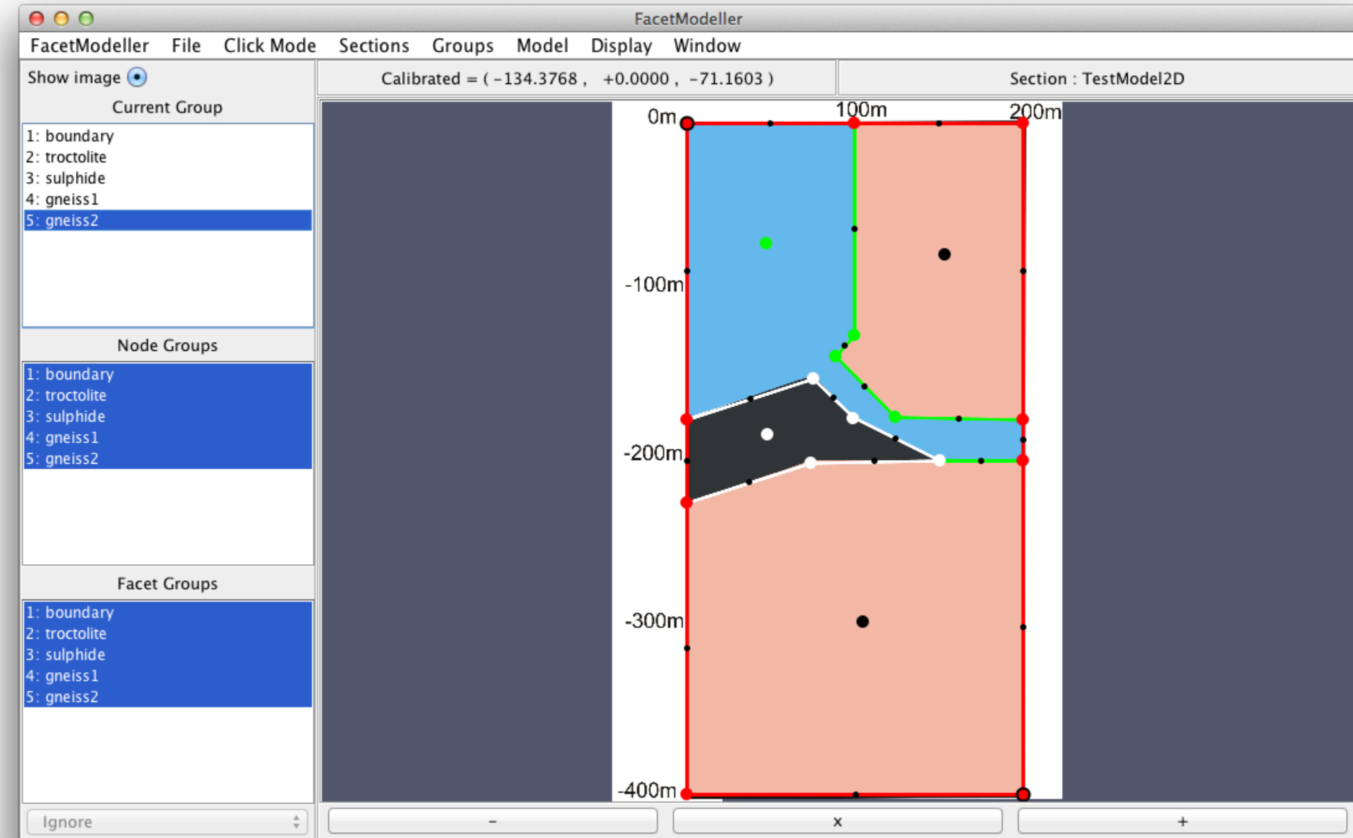
3.8 Defining regions

Now that the PSLG is defined, you need to define the different regions within it so that different attributes, e.g. physical properties, can be associated with the different parts of the model. Each watertight part of the model is a different region and you must provide a spatial location inside each of those different regions. FacetModeller stores each of those region specification points as a special kind of node and defining the regions is similar to defining nodes.

1. Select the menu item
   `Click Mode > Add regions`
2. Select the group named `troctolite` as the `Current Group`.
3. Click anywhere inside the blue troctolite unit.
4. When asked what type of point it is, choose the `Region` option. Always select the "Region" option (the "Control" option is something I need for my research).
5. Select the group named `sulphide`.
6. Click anywhere inside the dark grey sulphide unit (again, always select the "Region" option when prompted).
7. Select the group named `gneiss1`.
8. Click anywhere inside the **top** part of the pink gneiss unit.

9. Select the group named `gneiss2`.
10. Click anywhere inside the **bottom** part of the pink gneiss unit.
11. Select the menu item
    `Click Mode > Ignore`
    to tell FacetModeller to ignore any further mouse clicks (a safety precaution).

Your model should now look like this:



3.9 Exporting your model

The `File` menu provides options to export your model to various formats (`.poly`, `.node`, `.ele` and `.vtu`). Those file formats are explained on these webpages:

- Triangle: 2D triangular meshing program
  `http://www.cs.cmu.edu/~quake/triangle.html`
- TetGen: 3D tetrahedral meshing program
  `http://wias-berlin.de/software/tetgen/1.5/doc/manual/manual006.html#sec69`
- VTK File Formats
  `http://www.vtk.org/VTK/img/file-formats.pdf`

Once those files are exported, you will be able to use them with the Triangle meshing program and the ParaView visualization program.

A typical coordinate system for geological or geophysical data is

(X,Y,Z) = (easting, northing, elevation)

which is what FacetModeller uses. Others prefer to use

(X,Y,Z) = (north, east, down)

and if that is your preference then you are able to convert when exporting your model by choosing `Yes` when asked if you want to flip the z-axis: (X,Y,Z) is mapped to (X',Y',Z')=(Y,X,-Z).

3.10 Quitting FacetModeller

You can now quit your FacetModeller session by selecting the menu item

```
FacetModeller > Exit
```

or simply close the window as you normally would on whatever operating system you are using.

FacetModeller works in fundamentally different ways depending on whether you are working with a 2D or 3D model. Hence, you need to exit and relaunch the application if you want to start working with a new model of a different dimensionality.

Chapter 4

Tutorial - Building 3D models: I

A single isolated body within a boundary, digitized using cross-section images

This 3D tutorial relies on lessons learned in the 2D tutorial. You should also refer to the 2D tutorial for basic information about FacetModeller, including installation and running the program.

This first 3D tutorial does not contain any topography, nor do any of the anomalous rock units outcrop. Dealing with topography and outcropping is covered in the second and third 3D tutorials.

4.1 Running FacetModeller

Launch FacetModeller and choose `3D` when asked what kind of model you want to create. You will notice that the 3D version of FacetModeller looks a little different from the 2D version, with more panels, buttons and selection boxes (these will be explained throughout the tutorial).

4.2 Opening your image files

When building a 3D model, all the nodes in the model are associated with a particular depth- or cross-section (currently only vertical cross-sections are supported). The basic idea is to define all these sections, then add nodes to them, then create facets that connect the nodes.

The sections can be created by digitizing images or can be defined by the user without the need for an image. Here you will do the former. Select the menu item

`File > Load cross section image files`

and select the image files you want to digitize. For this example, you still start with 5 files named `x*.png`. Those in my research group can find these files here

`[repos]/peter/NetBeansProjects/examples/FacetModeller3Da/`

and public users here

`[repos2]/facetmodeller/Tutorial_3D_Part1/`

The other image files will be used later in the tutorial. Select all 5 files at once named `x*.png` and click `Open` (or whatever button your operating system gives you on the file open dialog window).

4.3 Defining the volume of interest (VOI)

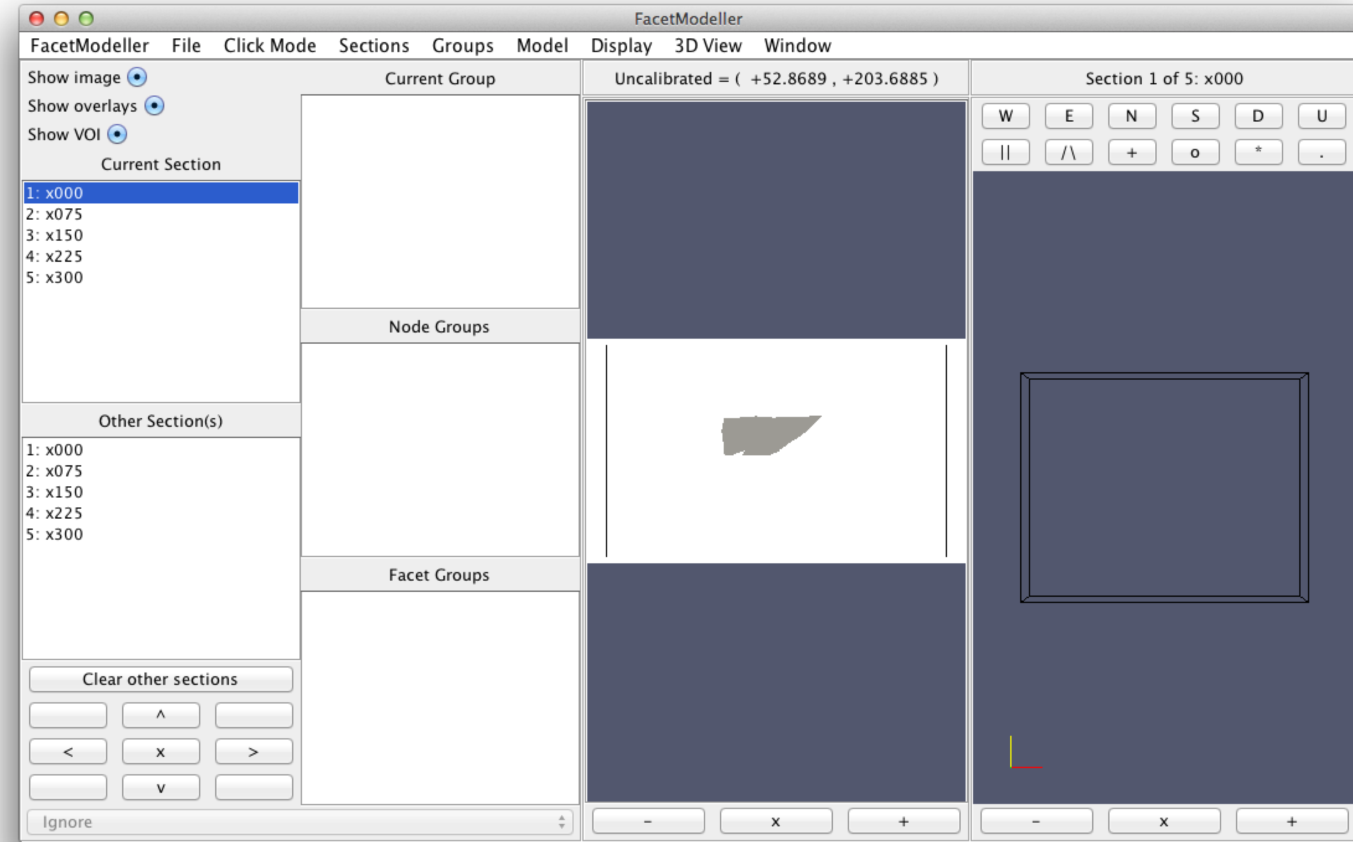
Although this step isn't required for general model building, defining the VOI can be helpful for 3D visualization, e.g. to help you make sure all of the parts of the model are inside your VOI. There are file menu items that help you make sure that all the nodes and facets you create are inside the VOI. To define the VOI, select the menu item

`Model > Define volume of interest (VOI)`

and enter the following information in the dialogs as they pop up:

Easting limits:	`-100 400`
Northing limits:	`-200 200`
Elevation limits:	`-100 150`

You should now see the following:



Remember to save your session early and often! I won't be reminding you to do so.

Again, keep in mind that this program is under development and this is a Java program that may look different on different platforms, so what you see may be slightly different than what is shown here.

The 3D version of the FacetModeller window is split into three main panels. On the left are various buttons and selection boxes that allow you to select which objects you wish to display and work with. On the right are two viewing panels; the panel on the left is a 2D viewing panel and on the right is a 3D viewing panel. The 2D viewing panel displays the currently selected section (in the `Current Section` selection box) and any nodes or facets defined on that section or between it any of the other sections (selected in the `Other Section(s)` selection box). The 2D viewing panel is where you will define nodes and facets (as in the 2D tutorial).

Only one section can be selected as the `Current Section`. Multiple selection is allowed for the `Other Section(s)` selection box. The selected `Current Section` defines the view limits and projection used for the 2D viewing panel. The selected `Other section(s)` defines what to plot on the 2D view. Nodes, facets and regions in the selected `Other section(s)` are projected onto the `Current Section` in the 2D viewing panel.

The 3D viewing panel displays the model in either parallel or perspective (the default) projection. You should currently see the VOI displayed in the 3D viewing panel. You can move the model around in the 3D viewing panel by dragging it with your mouse, but you can not interact further with the model in the 3D viewing panel, e.g. to add nodes or define facets. I will explain various features of the FacetModeller window throughout this tutorial, but may not cover everything: hopefully anything I miss will be self explanatory, or you can just play with it to see what it does.

4.4 Defining groups

There are two units in this example:

1. host rock or “background” (white in the images)
2. anomalous body or “blob” (grey in the images)

You’ll need to define new groups (menu item `Groups > New group`) with names and colours indicated in the table below (you can use whatever colours you like but I’ll be referring to them in the text so I suggest you stick with these):

Name	Colour
boundary	green
blob	red

4.5 Calibrating the images

The next step is to calibrate your images. You will do this in the 2D viewing panel and the process is similar to that in the 2D tutorial. To make this a little easier, you should hide the 3D viewing panel:

1. Select the menu item
   `3D View > Show/hide 3D view`

To perform the calibration of the first section:

2. Select the `x000` section in the `Current Section` selection box.
3. Select the menu item
   `Sections > Calibrate current section`
4. Click on the **top left** of the black axes in the image.
5. Enter `"0 200 150"` when prompted for the coordinates of that first calibration point.
6. Click on the **bottom right** of the black axes in the image.
7. Enter `"0 -200 -100"` when prompted for the coordinates of that second calibration point.

This vertical section is normal to the x (easting) direction, looking east (with north to the left, south to the right).

> When calibrating the first section, you should see a black outline in the image indicating some axes. If you don’t see that black outline, you may need to maximize the window or use the zoom button below the 2D viewing panel to zoom in (sorry, this black outline is very thin and may not be visible in the screen-shots shown here).

This calibration procedure would normally be required for each section. However, in this example the sections are stacked along the x (easting) direction so you can copy the calibration information from the current section to all the others below it in the list:

8. Select the menu item
   `Sections > Copy calibration`
9. Enter `75` when prompted for the step length defining the x separation between the sections.

If you now select any of the other `x*` sections in the `Current Section` selection box, you will see calibration points drawn on each image as small black unfilled circles. To check the locations of the sections, move your cursor around on the 2D image viewing panel: the coordinates at the cursor location are displayed above that panel.

If you now

10. open the 3D viewing panel and
11. select all the sections in the `Other Section(s)` selection box,

you will see the outlines of the 5 sections drawn black in the 3D view. Before continuing:

12. Make sure the 3D viewing panel is closed.
13. Click the `Clear other sections` button (near the bottom left of the FacetModeller window).

4.6 Adding nodes

You may first want to change the point size and line width display options in the `Display` menu. As for the 2D tutorial, you will define new nodes in the 2D viewing panel, and you will start by defining nodes on the boundary:

1. Select the menu item
   `Click Mode > Add nodes`
2. Select the `boundary` group in the `Current Group` selection box.
3. Select the `x000` section in the `Current Section` selection box.
4. Click on the four corners around the boundary on the image in the 2D viewing panel.
5. Repeat for every section.

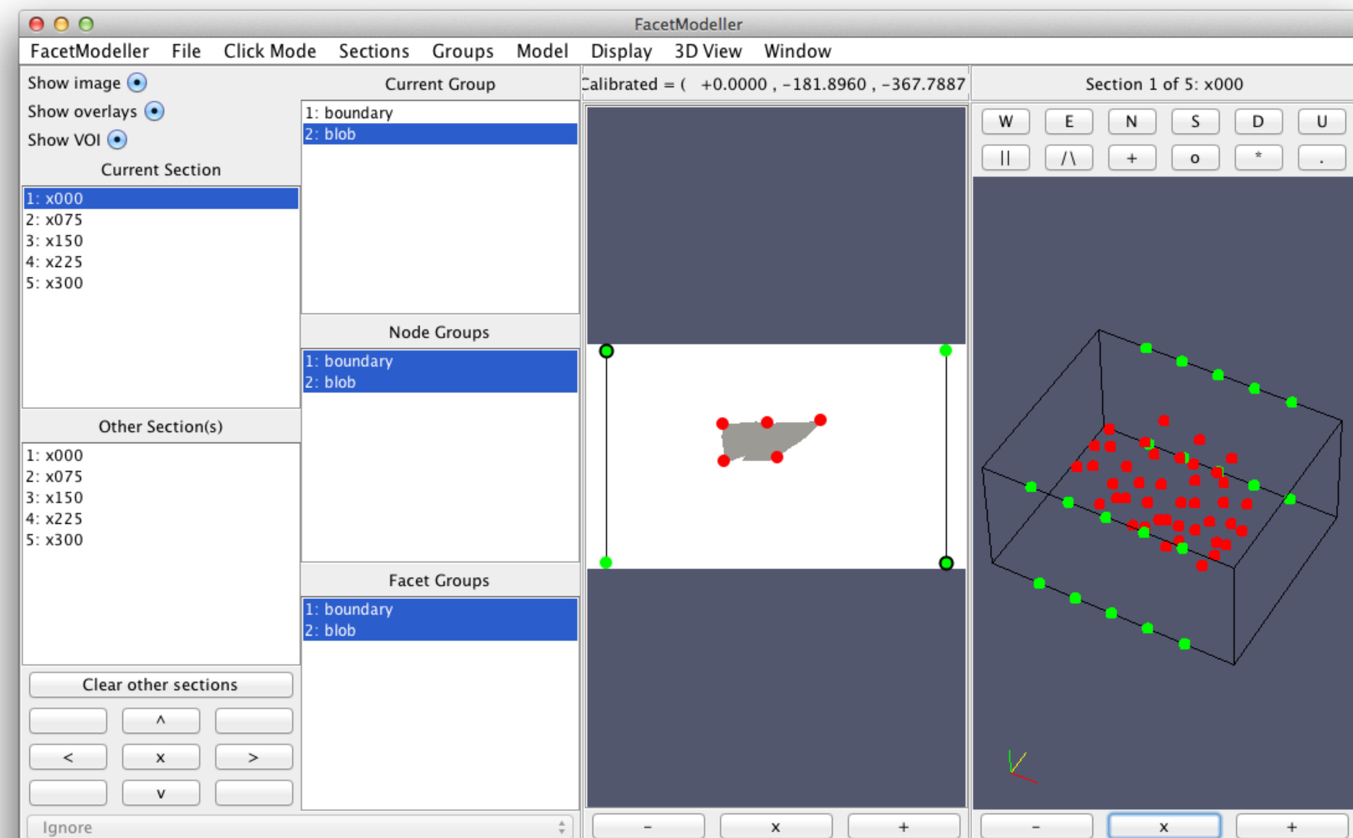
You need to snap those boundary nodes to the calibration points so that the polygonal boundary facets you create later are all planar.

6. Select the menu item
   `Model > Snap nodes to calibration points`

Now you will create nodes to define the outline of the blob:

7. Select the `blob` group as the `Current Group`
8. Select the `x000` section as the `Current Section`.
9. Click around the outline of the blob. Do this fairly coarsely (refer to the next screen-shot for guidance). You don't need to be too exact.
10. Repeat for every section.

Show the 3D view again and your model should now look like this:



If you haven't investigated the 3D viewing panel yet, now would be a good time to do so. There are zoom buttons on the bottom of the 3D viewing panel (as well as on the 2D viewing panel). The buttons on the top of the 3D viewing panel allow you to change the view direction, viewing options, etc. If you hover your mouse over those buttons, some text will pop up telling you what they do.

4.7 Defining facets between parallel sections

Remember that in 3D models, facets are triangles or other planar polygonal shapes. The first facets you will create are those on the surface of the blob:

1. Select the menu item
   `Click Mode > Define triangular facets`
2. Select the `blob` group as the `Current Group`.
3. Make sure that only the `blob` group is selected in the `Node Groups` selection box (this isn't required, but it is nice to hide any nodes that you don't need to work with).
4. Select the `x000` section as the `Current Section`.
5. Select the `x075` section in the `Other Section(s)` selection box.

In the 2D viewing panel, you should now see the nodes for the `x000` section drawn as filled circles and those for the `x075` section drawn as empty circles (nodes on the `Current Section` are always drawn filled and those on the `Other Section(s)` are always drawn empty). You will need to hide the image to help you continue:

6. Unselect the radio button `Show image` (at the top left of the FacetModeller window).
7. Move your cursor around on the 2D viewing panel to see candidate triangular facets. The candidate facets are also displayed in the 3D viewing panel so rotate and zoom the 3D view as required to help you with this task. You will also need to shift the overlaid nodes on the 2D view (see instructions below).
8. Click once you see a facet you want to create.
9. Continue clicking to define new facets between the sections.

The nodes for the current and other sections overlap along the top of the blob and it is difficult to figure out which to connect. To help with this, use the 3-by-3 grid of buttons at the bottom left of the window (under the `Clear other sections` button). These 9 buttons allow you to shift the overlaid nodes from the other section: click on any of the outer 8 buttons to move in that direction (up $\boxed{\wedge}$, down $\boxed{\vee}$, left $\boxed{<}$, right $\boxed{>}$, diagonally $\boxed{}$) and click the central $\boxed{\times}$ button to reset (no shifting).

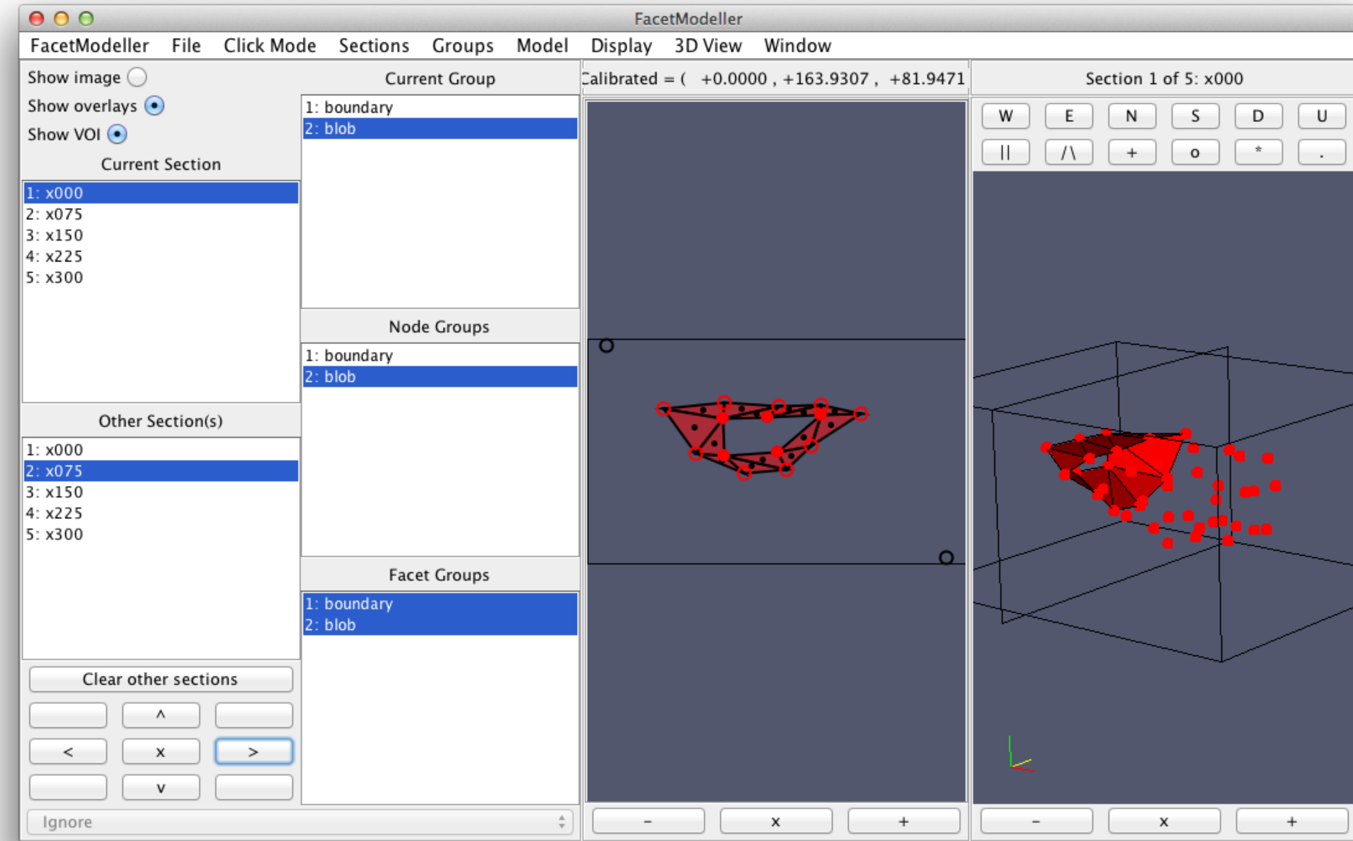
If you make a mistake, like creating overlapping facets, just delete the ones you don't want (menu item `Click Mode > Delete facets`) and continue.

Another helpful feature is the menu item

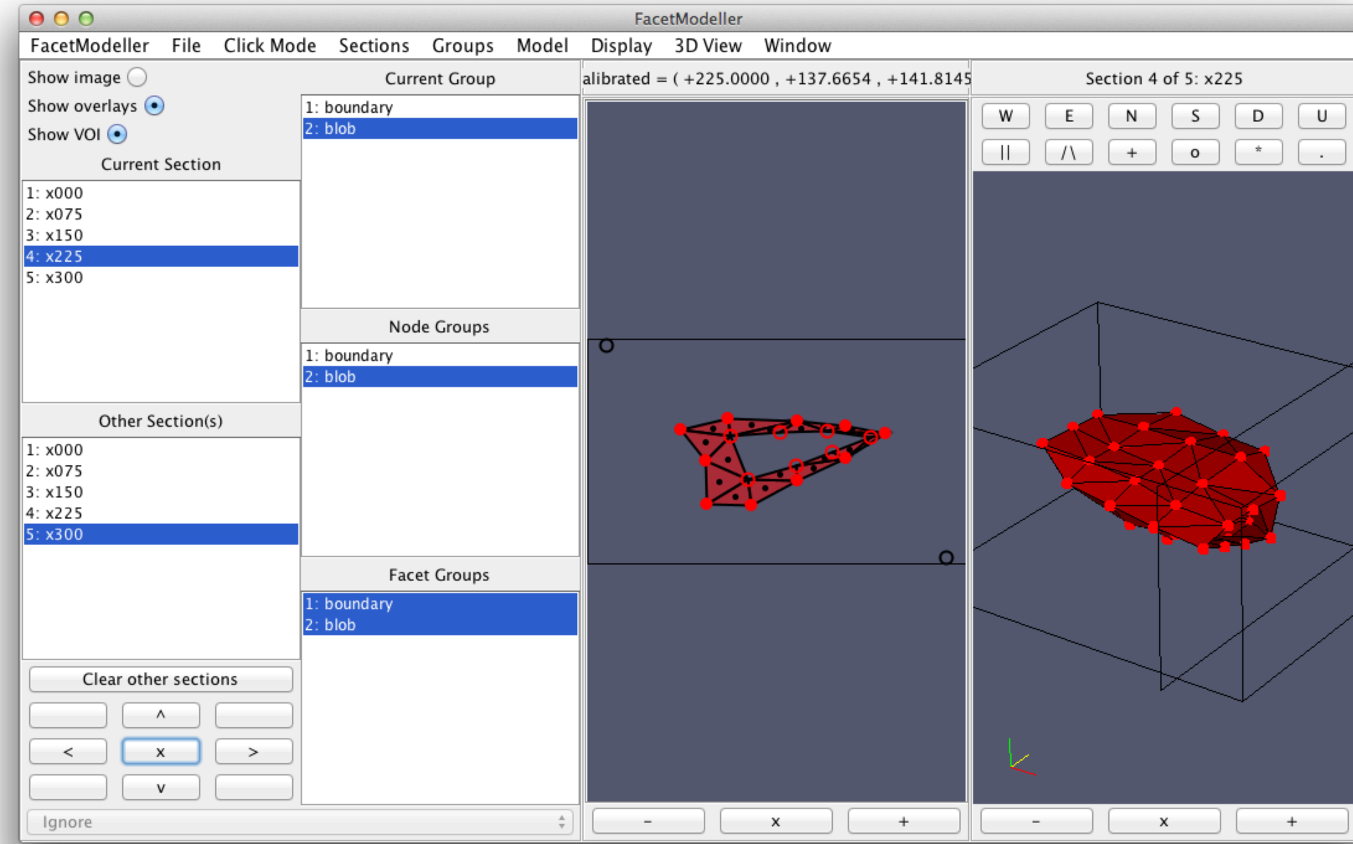
    3D View > Select 3D origin node

which allows you to select a node to become the centre of the 3D viewing panel so that you can then zoom in on the appropriate part of the model.

Here is what you should see once you have finished this step with the first pair of sections:



You can now move on to the next pair of sections. This process takes a long time. I'd suggest you practice defining a few facets and then move on, but if you do manage to get through it, your model should now look like this:



You have yet to close the body at the far western and eastern ends. To do so:

10. Select the menu item
    `Click Mode > Define facets node-by-node`
11. Select the `x000` section as the current section.
12. Clear the overlaid other sections (the `Clear other sections` button).
13. Click to define new facets around the outline you see in the 2D view.
14. Repeat with the `x300` section to close the other end of the blob.

In the node-by-node click mode, the facet is defined once you loop back to the same node you started with. Because all of these nodes are on the same section, the resulting facet will be planar. This is an important restriction for use in most meshing programs.

Currently, FacetModeller does not check that polygonal facets with more than three nodes are planar. If you create such a facet and try running TetGen on the resulting PLC, TetGen will fail.

4.8 Defining facets between non-parallel sections

You will now load a section that is not parallel to the those already loaded:

1. Select the menu item
   `File > Load cross section image files`
2. Choose `Insert` when prompted.
3. Open the file `y00.png` (in the same location as you found the other image files).
4. Make sure the `Show image` radio button is selected.

Follow through the same calibration steps as before with this new section:

5. Make sure the `y00` section is the `Current Section`.
6. Select the menu item
   `Sections > Calibrate current section`
7. Click on the **top left** of the black axes in the image.
8. Enter `"-100 0 150"` when prompted for the coordinates of the first calibration point.
9. Click on the **bottom right** of the black axes in the image.
10. Enter `"400 0 -100"` when prompted for the coordinates of the second calibration point.

This vertical section is normal to the y (northing) direction, looking north (with west to the left, east to the right).

11. Define a new group (menu item `Groups > New group`) named `blob_y` with blue as the group colour.
12. Select the menu item
    `Click Mode > Add nodes`
13. Add nodes around this new blob outline.
14. Investigate your model in the 3D viewing panel. You should see the new blue nodes you added on the new section.

You'll now define new facets between these new blue nodes and the pre-existing red nodes. You'll focus only on those near the western extent. First you need to delete the facet you created recently on the western extent:

15. Select the `x000` section as the `Current Section`.
16. Select the menu item
    `Click Mode > Delete facets`
17. Click on the centroid handle of the displayed polygonal facet to delete it.

You now want to sew up the western extent of the blob, using the nodes on sections `x000` and `y00`.

18. Select the `x000` section as the `Current Section`.
19. Select the `y00` section as the `Other Section`.

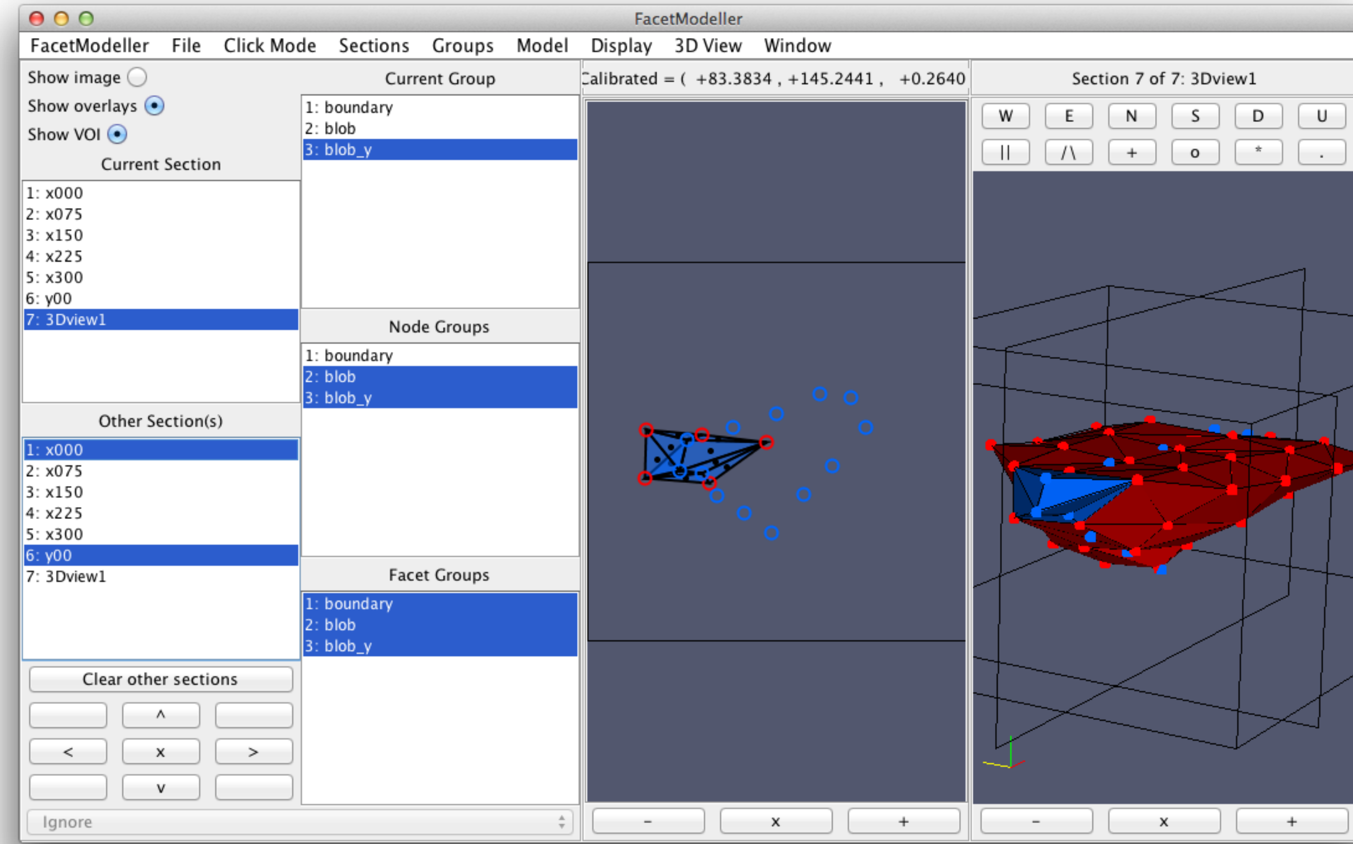
The projection in the 2D viewing panel will makes it practically impossible to define the facets required! You need another view in the 2D viewing panel. To get one:

20. Rotate the 3D viewing panel to the desired view and zoom as desired (refer to the next screen-shot for guidance).
21. Select the menu item
    `Sections > New section from current 3D view`
22. Name the section `3Dview1` when prompted.
23. Use the default white as the colour for the section when prompted.
24. Make sure this new section is selected as the `CurrentSection`.
25. Select the `x000` and `y00` sections as the `OtherSections`.
26. Make sure you have cleared any previous shifting by clicking the $\boxed{\times}$ button in the centre of the 3-by-3 grid of shift buttons at the bottom left of the window.

The 2D view should now allow you to sew up the western end of the blob with triangular facets. If that view is still not helpful, change the 3D view and use the menu item `Sections > Reset snapshot section` to reset the view for the `3Dview1` section.

27. Select the `blob_y` group as the `Current Group` (the facets will be drawn blue to help distinguish them from those already created).
28. Select the menu item
    `Click Mode > Define facets node-by-node` (you'll probably find this simpler than the other approach for defining triangular facets but you can try both if you like).
29. Click to define triangular (and therefore planar) facets.

Your model should now look like this:



4.9 Defining facets on the boundary

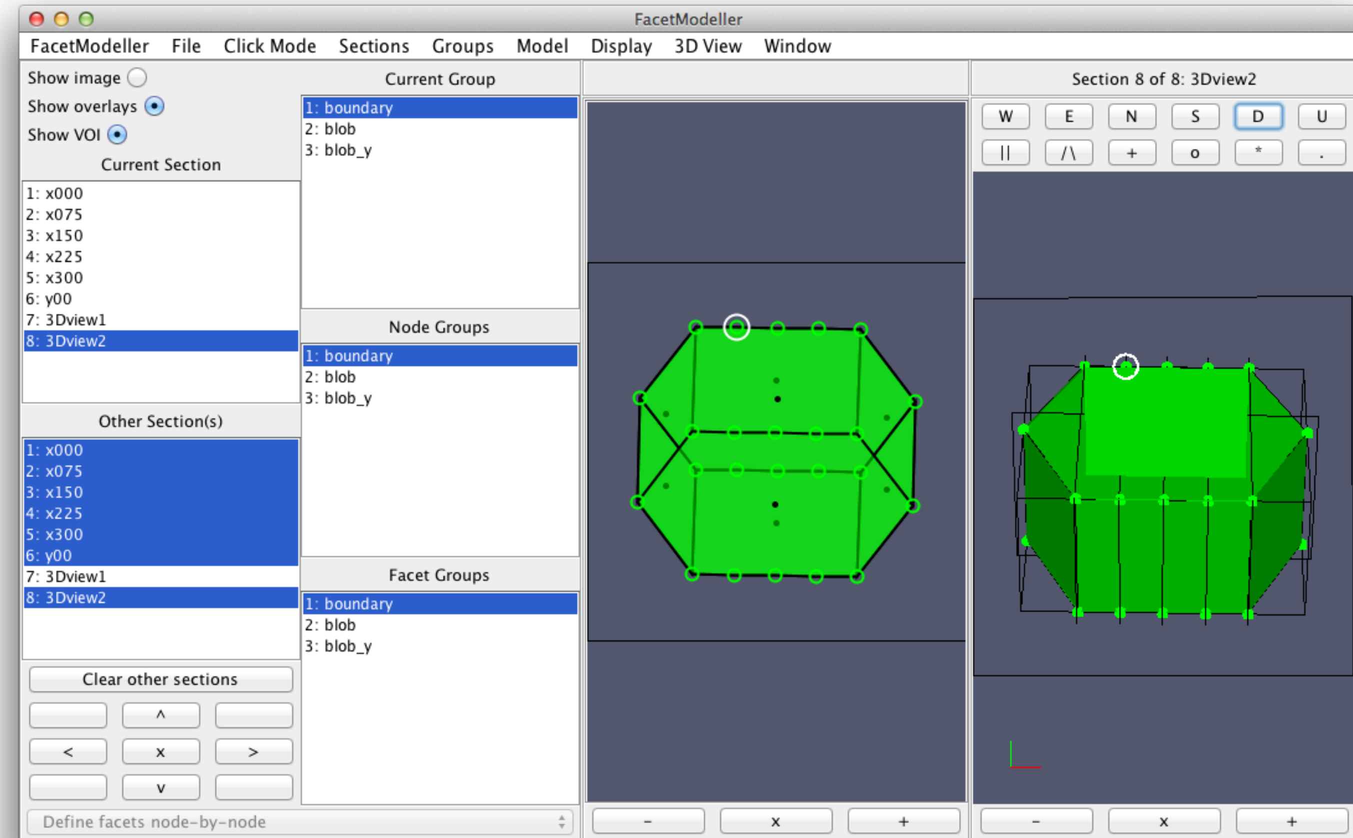
The last thing you need to do is create facets around the boundary of the model. You will need some new nodes and another view to help you:

1. Select the menu item
   `Click Mode > Add nodes`
2. Select the `boundary` group as the `Current Group`.
3. Select the new `y00` section as the `Current Section`.
4. Add four new boundary nodes at the corners of the `y00` section. You will have to click very close to the lateral extents of the 2D viewer panel.
5. Select the menu item
   `Model > Snap nodes to calibration points`
6. Select the menu item
   `Click Mode > Define facets node-by-node`
   (the node-by-node approach is the only way to define polygonal facets with more than 3 nodes).
7. Display only the nodes and facets in the `boundary` group using the `Node Groups` and `Facet Groups` selection boxes (initially there are no such facets and you are just hiding all the others).
8. Clear the zoom on the 3D view and rotate the model so that none of the boundary nodes are overlapping (refer to the next screen-shot for guidance).
9. Create a new section from the current 3D view and name it `3Dview2`.
10. Make sure that new section is selected as the `Current Section`.
11. Select all the $x*$ sections and the $y00$ section as the `Other Sections`.

12. Define eight planar polygonal facets to define a boat-like boundary surface:
    - there are rectangular facets on the southern and northern sides,
    - rectangular facets form the bow and stern of the boat-like shape, 2 at each of the western and eastern extents,
    - and hexagonal facets define the top and bottom.

    Refer to the next screen-shot for guidance.

Your model should now look like this:



Perhaps you are uncomfortable with a boundary that is not a rectangular prism? Here is how to make one. First delete all the facets on the boundary that you just created:

13. Select the `boundary` group ias the `Current Group`.
14. Select the menu item
    `Model > Delete group of facets` and poof, they are gone. Actually, you don't even need any of the existing boundary nodes ...
15. Select the menu item
    `Model > Delete group of nodes`

You now need 2 more sections to add nodes to the western and eastern extents of the VOI. You will generate 2 sections without images. They will be drawn as rectangles with some user-defined colours and they can be treated in the same way as the sections used above, e.g. they need to be spatially calibrated and you can add nodes on them. To create these new sections:

16. Select the menu item
    `Sections > New cross section without image`
17. When prompted, name the section `south` and choose the default white as the colour.
18. Enter `"-100 -200 150"` when prompted for the top left corner of the image.
19. Enter `"400 -200 -100"` when prompted for the bottom right corner of the image.

20. Repeat those steps for a new section named `north`, with calibration points `"-100 200 150"` and `"400 200 -100"`.

Now all you need to do is define 4 new `boundary` nodes on the corners of the `south` and `north` sections (8 new nodes in total), snap them to the VOI, then define the 6 `boundary` facets node-by-node.

> Alternatively, you could have used sections on the southern and northern extents of the VOI to create this rectangular prism boundary. You could also define two triangular facets on each edge of the VOI instead of rectangular facets. Since this is easily automated, I plan to provide a menu item to automate the process.

4.10 Defining regions

Once a PLC is been built, destined for input into TetGen, you may or may not want to define regions. TetGen's `-A` flag can be used to assign cell attributes to indicate the cells in different watertight volumes in the mesh generated by TetGen. If you want to specify region-based constraints on the mesh, for example maximum cell volumes, then you will want to define your own regions to which the constraint parameters can be assigned.

There are two approaches you can take to define regions. The first takes the fewest steps but may not be as effective for more complicated models where it may be difficult visualizing exactly where your region points lie in relation to the PLC surfaces.

4.10.1 Defining regions on sections

In this approach, you create new sections on which to specify the regions. The steps are as follows:

1. Create as many new sections without images as are required to slice through the different watertight volumes in the PLC.
2. Add region points on those new sections.

4.10.2 Defining regions on facets

In this approach, you create new nodes that will function as the region points. You are not using the FacetModeller "region" objects at all; instead, you are using the FacetModeller "node" objects, which are essentially the same thing anyway. Those nodes are added on the facets of the PLC and are then shifted as required to move them into the watertight volumes in the PLC. For this strategy, care must be taken when exporting the different model components. The steps are as follows:

1. Create a new node group that will hold your region coordinates. For the purposes of these instructions I'll call that new node group `region_nodes`.
2. Select that new group as the current group and choose the "Add nodes in facets" click mode.
3. Define new nodes on facets that lie on the boundary of each watertight volume in the PLC.
4. Select the menu item
   `Model > Translate nodes`
5. Translate the `region_nodes` slightly, as required to move them into the different watertight volumes in the PLC.
6. It may be impossible to move all the `region_nodes` at once such that they all enter the different watertight volumes in the PLC. In this case, you would have to define new node groups, assign the region nodes into the various new node groups, and translate each group of region nodes separately.
7. Export the group(s) of region nodes to a `.node` file.
8. At this point, I like to delete all the region nodes in FacetModeller once I'm happy with them so that I don't mistakenly export them with the entire model into the `.poly` file in the next step. You can always load them back in from the `.node` file that you saved in the previous step.
9. Export all components of the model except for those region nodes to a `.poly` file.
10. Using whatever ASCII text file editor you like, manually paste the region coordinates from the `.node` file to the `.poly` file (the files that were exported in the previous steps).
11. Add any additional meshing constraint information in the list of regions in the `.poly` file, for example maximum cell volumes.

Chapter 5

Tutorial - Building 3D models: II

A padded VOI cut by topography

This 3D tutorial example relies on lessons learned in the 2D tutorial and first 3D tutorial. You should also refer to the 2D tutorial for basic information about FacetModeller, including installation and running the program.

In this tutorial, you will learn how to generate a simple model containing a padded VOI cut by a topography surface. I won't be providing explicit step-by-step instructions like in the other tutorials so I hope you remember how to do some of the basic tasks.

5.1 Requirements

This tutorial relies on two free utility codes. You will need to use the Triangle meshing program, which can be downloaded here:

`http://www.cs.cmu.edu/~quake/triangle.html`

Instructions for compiling the Triangle c++ source code are provided on that page.

You will also need to use my program `interpolate_data`.

5.2 Preliminary topography processing

You will need to perform the following processing on your topography data before working with it in FacetModeller. First you need to do some basic processing and get it into the correct file format. **In this tutorial, I have already processed the topography data as required so you can skip this section, but read it anyway!**

1. Remove points outside of the VOI (this can be done in FacetModeller or you can use the PODIUM program `remove_range`; see Section 5.3 below for VOI information).
2. Decimate the points so the spacing is nicer to work with (see aside below).
3. Get your topography data into the column-based ascii file format below (if you have been using the PODIUM programs to perform the topography processing then your file will already be in the required format).

> If the data is too dense then the resulting model may yield more cells than is computationally feasible for forward modelling or inversion. If the data is too sparse then the model may lead to unacceptably large errors in the forward or inverse solutions. Decimation is not currently possible in FacetModeller but you can use the PODIUM utility program `decimate`.

Your topography data will need to be in the column-based ascii file format below:

```
n [...]
1 x1 y1 z1 [...]
2 x2 y2 z2 [...]
...
n xn yn zn [...]
```

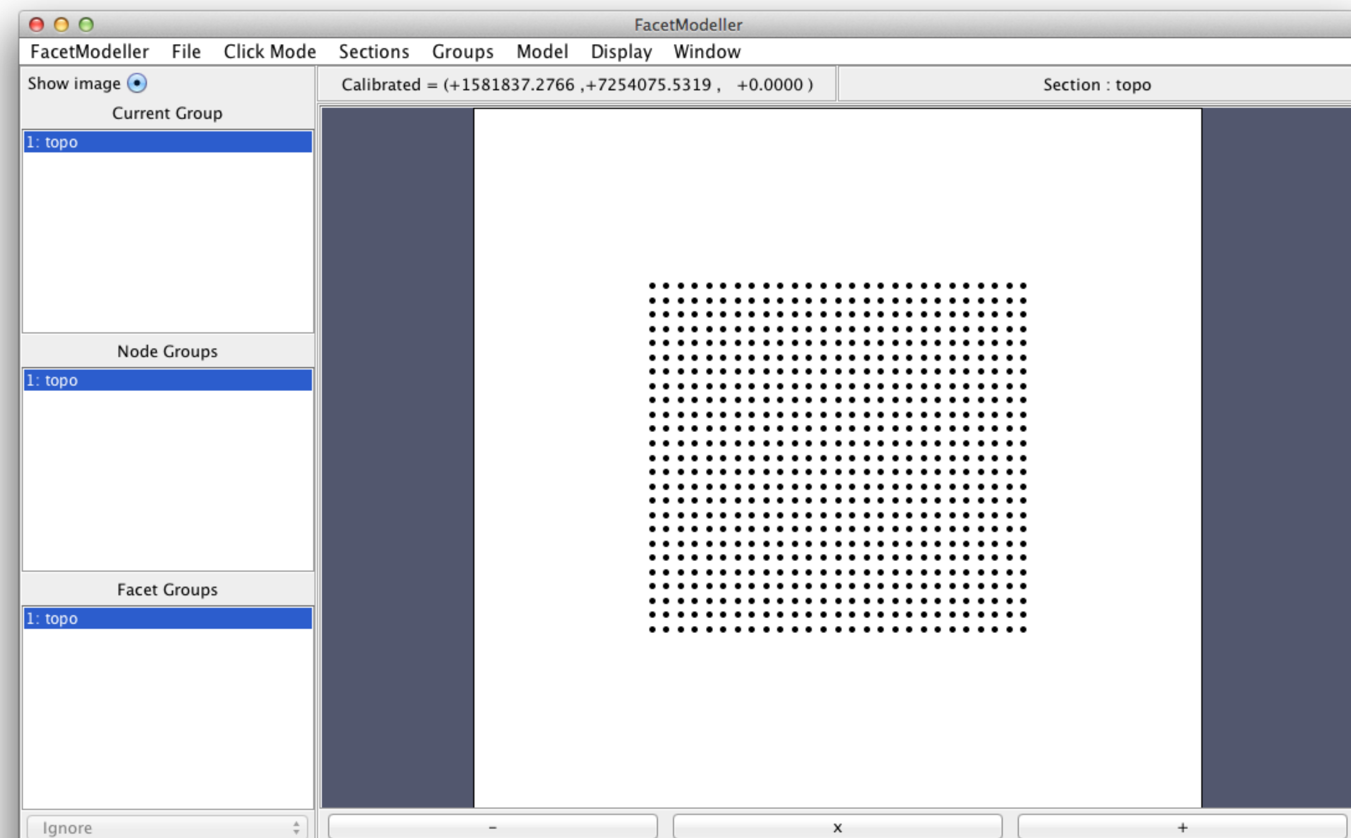
where `[...]` indicates optional additional values on a line of the file. This format is consistent with `.poly` and `.node` files in both 2D and 3D (see the file format pages for Triangle and TetGen).

5.3 Defining the padded VOI in 2D

You will be using Triangle to mesh your topography surface. First however, you need to define your padded VOI in 2D. Why 2D? That will become clear as you go through this tutorial, but the idea here is to use Triangle to generate a topography surface that conforms to the facets in your padded 3D VOI, and to do so you need to move into 2D.

1. Launch FacetModeller for building a 2D model.
2. Create a new group named `topo`, coloured the default black.
3. Select the menu item
   `Sections > New depth section without image`
   and call the section `topo`, coloured the default white.
4. When prompted for the coordinates of the bottom left corner of the image, enter `"1588400 7184850 0"`.
5. When prompted for the coordinates of the top right corner of the image, enter `"1670000 7263250 0"`.
6. Select the menu item
   `File > Load from .node/.ele files`
7. When prompted for the `.node` file, select the file `topography_decimated32.node`.
   Those in my research group can find this file here
   `[repos]/peter/NetBeansProjects/examples/FacetModeller3Db/`
   and public users here
   `[repos2]/facetmodeller/Tutorial_3D_Part2/`
8. When prompted for the `.ele` file, click `Cancel` (or whatever button your operating system gives you on the file open dialog window). This does not cancel the loading, rather it indicates that there is no `.ele` file associated with the `.node` file).

Don't forget to save early and often! You should now see the following:



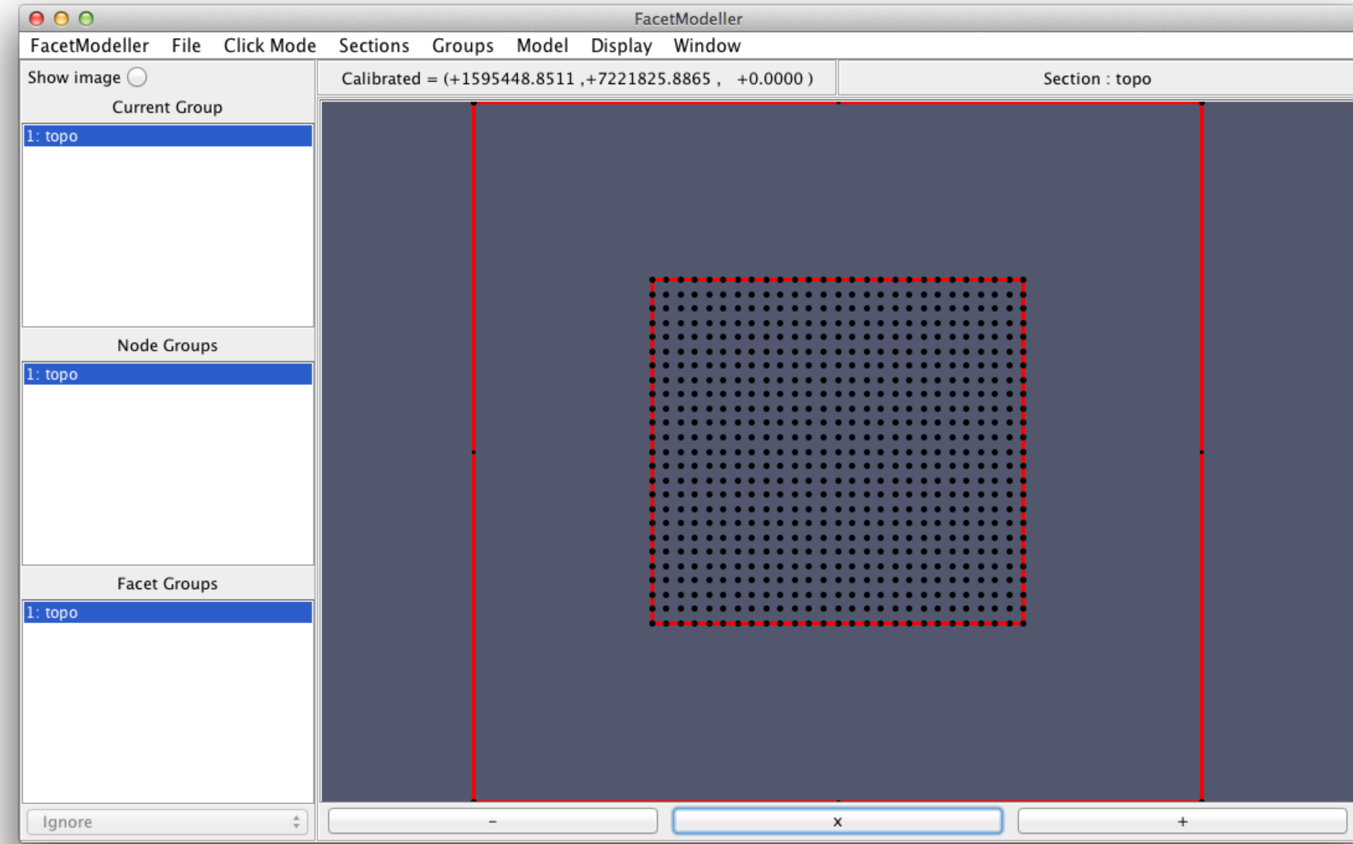
The black dots are the topography points, projected into map view. These points are gridded: you do not need gridded topography in general. You will now define the facets in the boundary of our padded VOI:

9. Add 4 new nodes on the corners of the section.
10. Snap those nodes to the calibration points.
11. Select the menu item
    `Groups > Change facet colour`
    and select red. This will make the facets show up more easily on the black-and-white backdrop.
12. Hide the section image (unselect the `Show image` radio button at top left) to help the facet overlays show up more easily in the following steps.
13. Select the menu item
    `Interaction > Picking/snapping distance` and change from the default `10.0` to `500.0`. This spatial distance (in whatever calibration units you are using; metres here for the UTM coordinates) defines how close the curser must be to a node before it can be selected, for example to attach it to a facet.

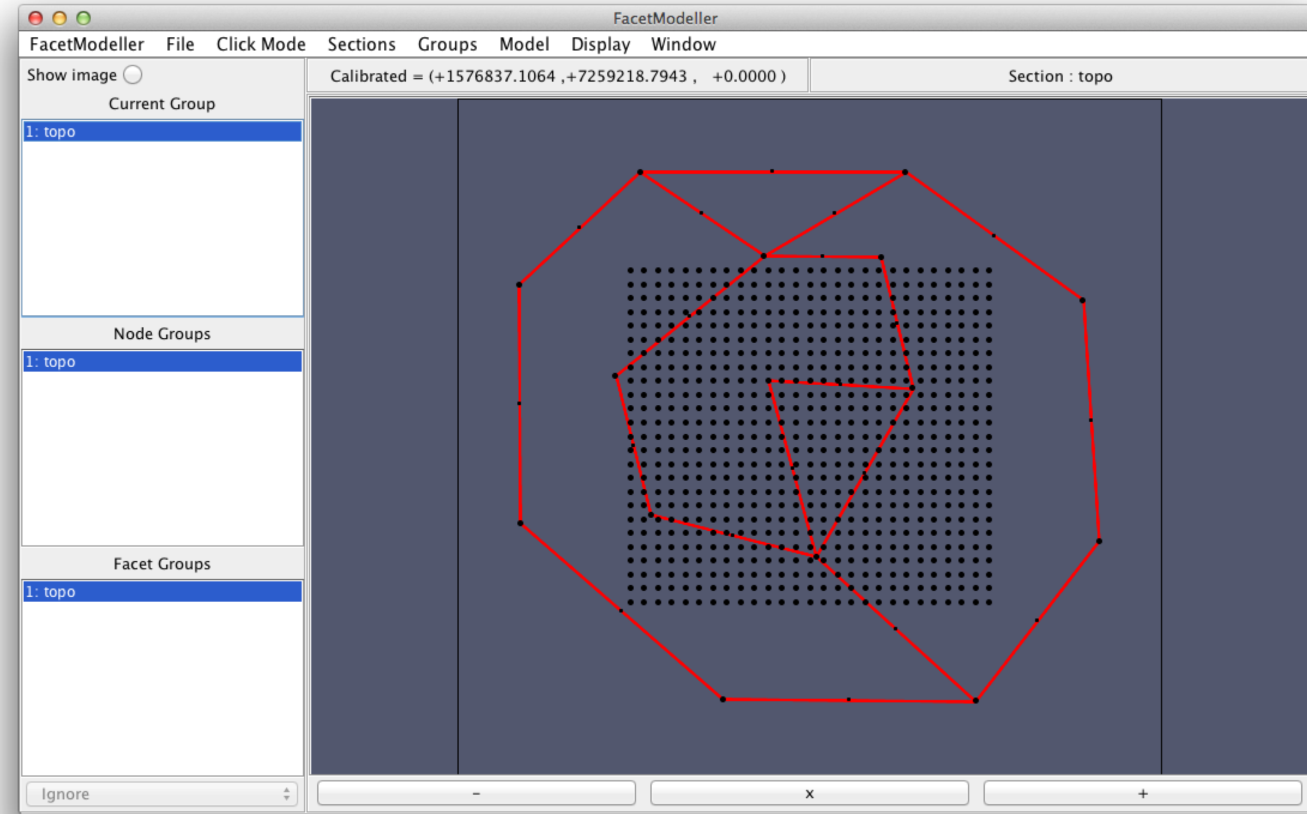
> This distance is also used when snapping nodes to, for example, the extents of the VOI. Having a smaller picking distance is advisable so that you are only selecting nodes within some small proximity of the cursor. You can tell when a node is selectable when a white circle is drawn around it, which will only happy once your cursor is close enough to it. Also, if the picking/snapping distance is too large then you may inadvertently snap nodes that you did not wish to, e.g. nodes that are not close to the VOI. Conversely, if the picking/snapping distance is too small compared to the spatial range of your model then the spatial distance corresponding to an image pixel may be larger than the picking/snapping distance. In that situation, FacetModeller will not respond to your mouse clicks and nothing will happen why you try to snap nodes.

14. Using the `Define facets node-by-node` click mode, define 4 new facets around the boundary of the section.
15. Define 4 new facets that connect the corners of the topography grid.

You should now see the following (I have changed the line display width here):



The outer red rectangle defines the boundary of the modelling region. The inner red rectangle defines a central region of interest where we want smaller cells in our modelling mesh. The space between the two red rectangles defines the padding region where we can allow larger cells. Generally, you will not have gridded topography, in which case, you would simply add new nodes anywhere you wish to define the inner rectangle. Of course, you could follow that strategy with gridded data too, and there is no reason that the central region of interest needs to conform to the topography. Also, there is no reason these need to be rectangles, and you could have facets that define as many different regions as you like. For example, you could also have done something like this:

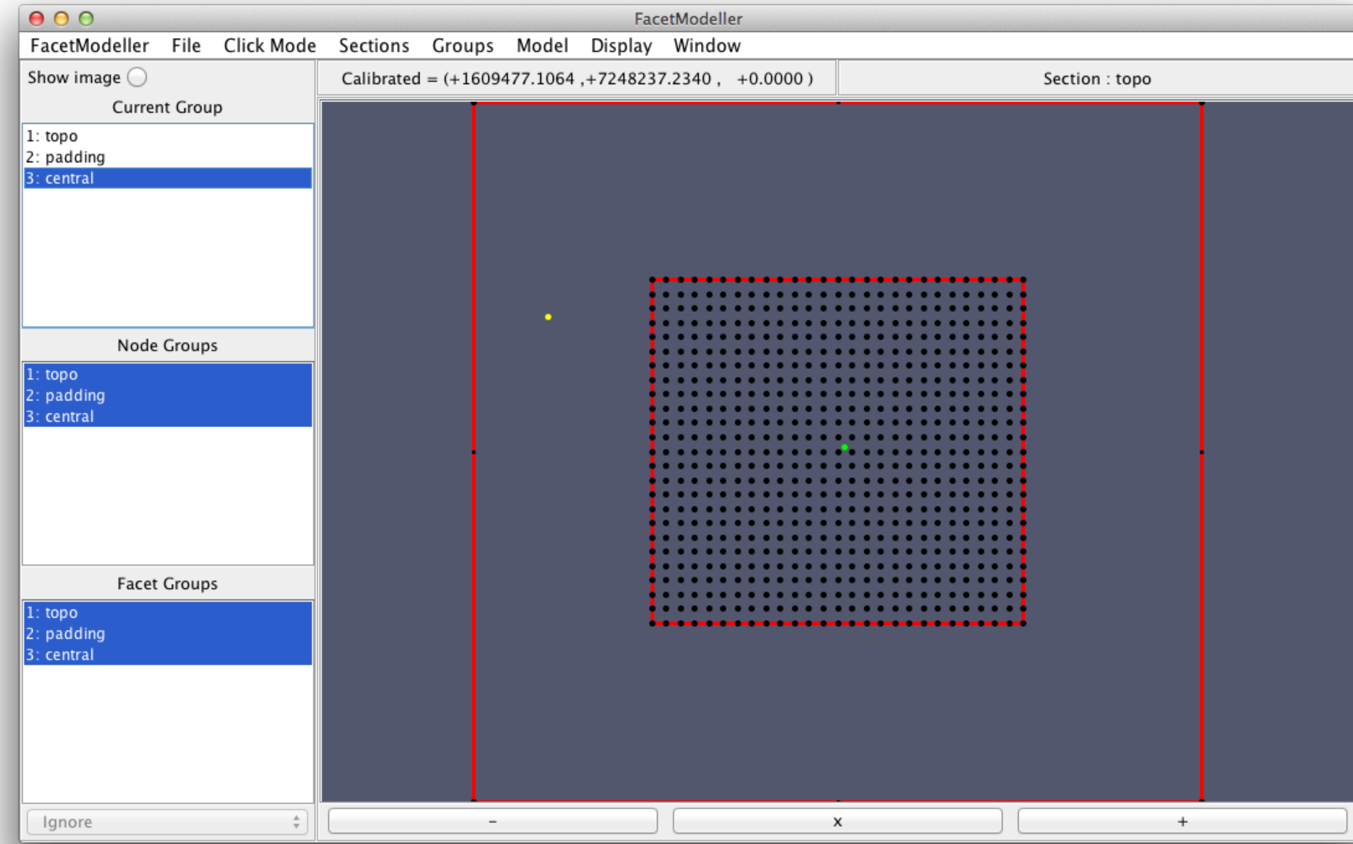


But let's stick to the simpler situation for now!

The final requirement is to add two regions to the model (if you forget how to define regions you can refer to Section 3.8):

16. Define a new group named `padding` and define a new region point anywhere inside the padding region (between the two red rectangles).
17. Define a new group named `central` and define a new region point anywhere inside the central region of interest (inside the inner red rectangle).

You should now see the following (I have coloured the two regions yellow and green):



The rest of the 2D processing goes as follows:

18. Export your model to a `.poly` file and call it `topo2d.poly`.
19. Quit FacetModeller.
20. Open the `topo2d.poly` file in your favourite text editor.
21. Scroll to the bottom and add area constraints for Triangle to use when meshing. The last two lines of the file should read something like this:
    ```
    1 1596698.8936170214 7239201.773049645 1 1000000000
    2 1629894.4680851065 7224606.028368794 2 2560000
    ```
22. Mesh the `.poly` file with the following commands:
    `triangle -pqDajPCV topo2d.poly`
    The result will be files named `topo2d.1.node` and `topo2d.1.ele` holding the triangulated 2D mesh.

> In this example you have specified an area constraint of 2560000 for the central region of interest. This corresponds to the area of the topography grid squares for this example (the grid spacing is $1600 = \sqrt{2560000}$). The area constraint of 1000000000 is set so large that the constraint is essentially ignored by Triangle.

The final task is to interpolate the topography at the new nodes generated by Triangle. I have provided the output file `interpolated.node` in the tutorial directory if you want to skip these steps.

23. The header of the topo2d.1.node file needs to be changed to turn it into a 3D file, which you can do manually or us my program `node3d`:
    `node3d topo2d.1`
24. Then you interpolate the topography using: `interpolate_data topo_decimated32 null topo2d.1 interpolated.node 1 0`
    The output file is named `interpolated.node`.

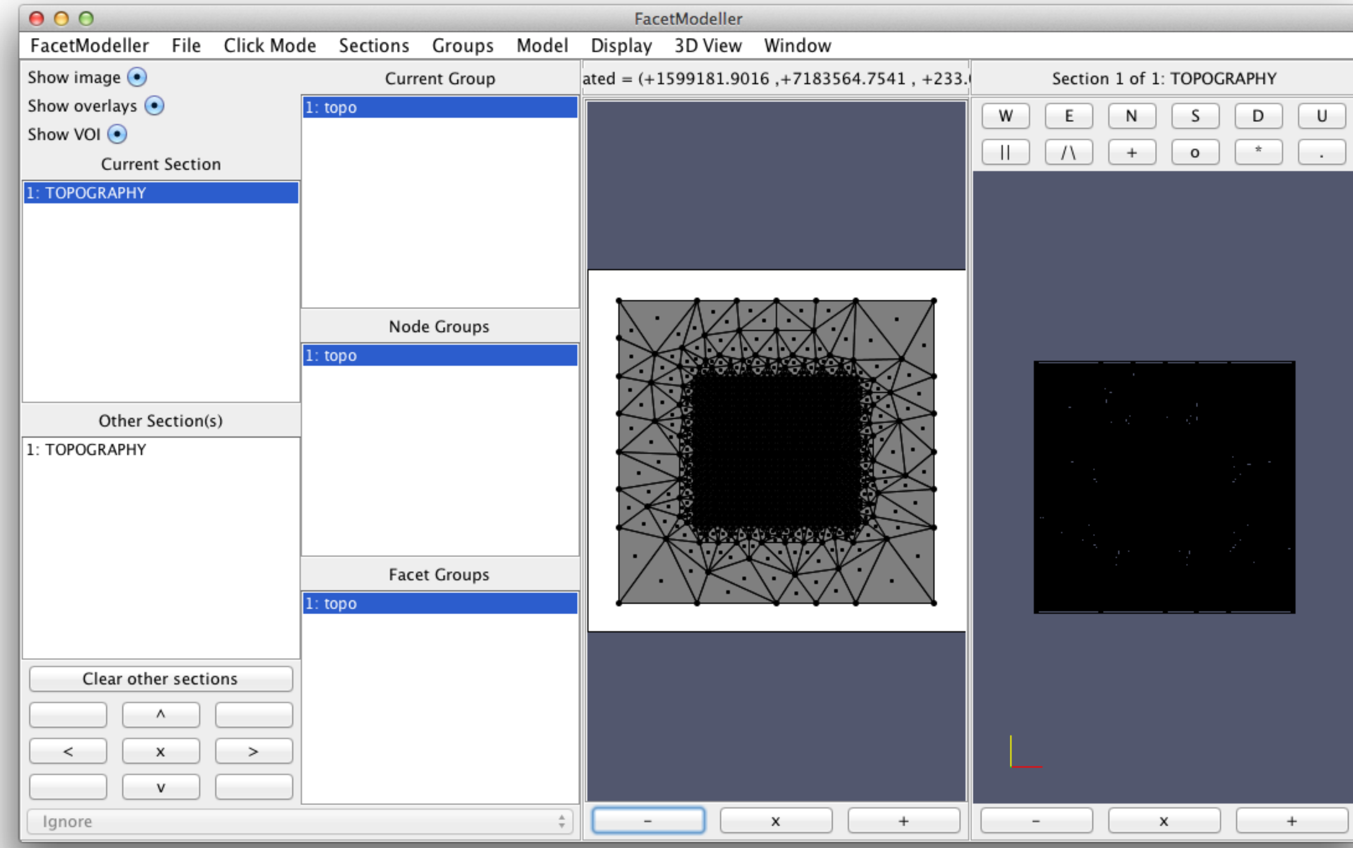
5.4 Defining the padded VOI in 3D

You now have a topography surface that conforms to the boundary facets that you will be creating in your 3D model.

1. Launch FacetModeller for building a 3D model.
2. Define the following VOI:
   Easting limits: `1588400 1670000`
   Northing limits: `7184850 7263250`
   Elevation limits: `-40000 10000`
   Note you have used the same easting and northing limits here as when calibrating the 2D section.
3. Define a new group named `topo`.
4. Create a new **depth** section without image named `TOPOGRAPHY`:
   the first calibration point is `"1580240 7177010 233"`
   and the second is `"1678160 7271090 233"`.
5. Select the menu item
   `File > Load from .node/.ele files`
6. Select and open the file `interpolated.node` when prompted.
7. Select and open the file `topo2d.1.ele` when prompted. Although this file came from the 2D meshing program Triangle, it defines triangular facets that can also be used in a 3D model.

> What you have done here is generated a depth section without image (named `TOPOGRAPHY`) onto which you have then loaded the nodes and facets from the `.node` and `.ele` files. The nodes are held in memory in 3D. When you select the `TOPOGRAPHY` section as the `Current Section`, those nodes are projected onto the section in the 2D viewing panel. You can use this approach for loading any other set of nodes and facets that need not be associated with the topography surface. For example, you may have `.node` and `.ele` files that define the surface of a more-or-less vertical fault. You could then define a vertical cross section without image and load your files onto that section. Of course, you need to be careful to calibrate the section so that any nodes you load from file will be projected onto the section.

Again, don't forget to save early and often! You should now see the following:



The 3D viewer in FacetModeller is fairly rudimentary and can be quite slow if you are displaying a large model. We hope to develop a more robust and more efficient 3D viewer in the future. For now, to avoid speed issues for this example, hide the facets:

8. Unselect the `topo` facets in the `Facet Groups` selection box (control- or command-click, depending on your operating system).

You now need to define some new sections and groups to add new nodes and facets to.

9. Create a new depth section (DEPTH section!) without image and name it `bottom`, for the bottom of the VOI:
   the first calibration point is `"1588400 7184850 -40000"`
   and the second is `"1670000 7263250 -40000"`.

10. Create another new depth section without image and name it `centre`, for the bottom of the central block (where you want finer cells):
    the first calibration point is `"1608400 7204850 -20000"`
    and the second is `"1650000 7243250 -20000"`.

11. Define new groups named `outer` and `inner` coloured green and red respectively.

12. Oh, how annoying, the topography facets came back. Hide them again.

You now need to define nodes on those two new sections.

13. Select the `outer` group as the `Current Group`.

14. Select the `bottom` section as the `Current Section`.

15. Add nodes to the corners of that section.

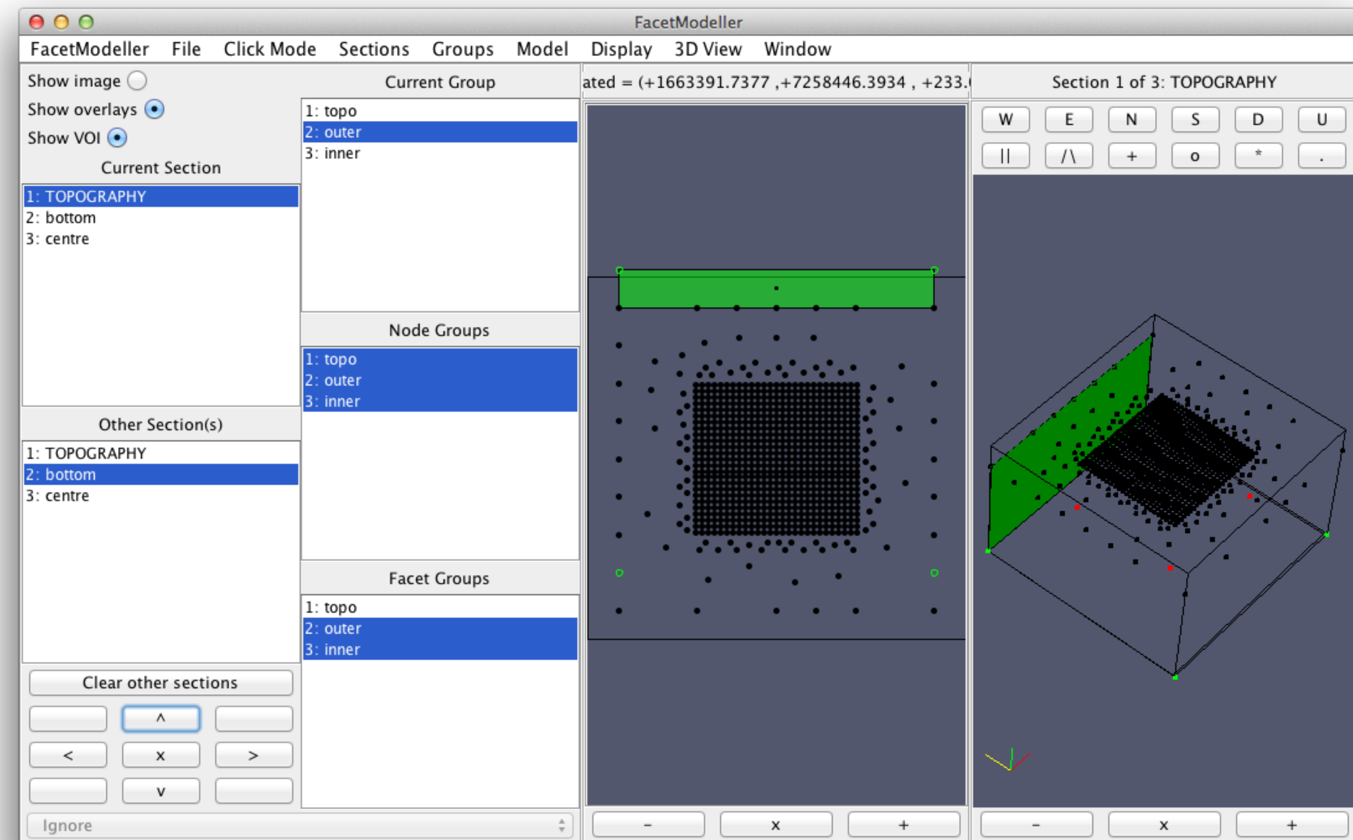
16. Snap nodes to calibration points.

17. Repeat steps 13 through 16 above but with group `inner` and section `centre`.

Wiggle your 3D model around in the 3D viewing panel and you should see those new nodes drawn green and red. The final step is to define the facets on the outer VOI boundary and on the inner central section.

18. Hide the section image (unselect the `Show image` radio button at top left).
19. Select the `outer` group as the `Current Group`.
20. Select the `TOPOGRAPHY` section as the `Current Section`.
21. Select the `bottom` section as the `Other Section`.
22. Shift the other section overlays up one step (using the top middle button in the 3-by-3 grid on the bottom left of the FacetModeller window).
23. Using the node-by-node approach, start defining a polygonal facet that contains all the black topography nodes along the northern side of the VOI, from left to right in the 2D viewer, then moves to the top right green node in the `bottom` section, then the top left green node, then back to the first black topography node you started with. In this facet definition click mode, you can actually drag across the nodes you want to add to the facet, or simply click as usual. However, when dragging, if you come close to a node it will include that node in the facet, and you may end up selecting nodes you didn't want. If that happens, change the click mode to ignore and then back to the node-by-node facet definition click mode and start again.

You should now have created a facet that defines one side of the outer VOI boundary. Your model should now look like this:



In this example, the sides of the outer VOI boundary are all aligned with the Cartesian axes. This ensures that the facet you just created is planar. Recall that in a previous aside I said you could define your 2D VOI boundary however you liked. Although that is true, using boundary facets that are not aligned with the Cartesian axes makes it far more difficult to ensure that the facets on the sides of the boundary are planar. Well, it shouldn't be too difficult given the right tools, but I haven't developed them yet. Therefore, I highly suggest you always use a VOI that is aligned with the Cartesian axes!

To finish defining the facets in your model:

24. Utilizing the overlay shift buttons as required, define the other 3 facets on the remaining sides of the outer VOI boundary.

25. Select the `bottom` section as the `Current Section` and define a rectangular facet on the bottom of the VOI boundary.
26. Repeat steps 19 through 25 above to define the side and bottom facets for the central block (you will use the `inner` group instead of `outer`, and the `center` section instead of `bottom`). You may have to hide the 3D viewing panel, or zoom in on the 2D viewing panel, to help you define these facets.

To check that you have done this correctly:

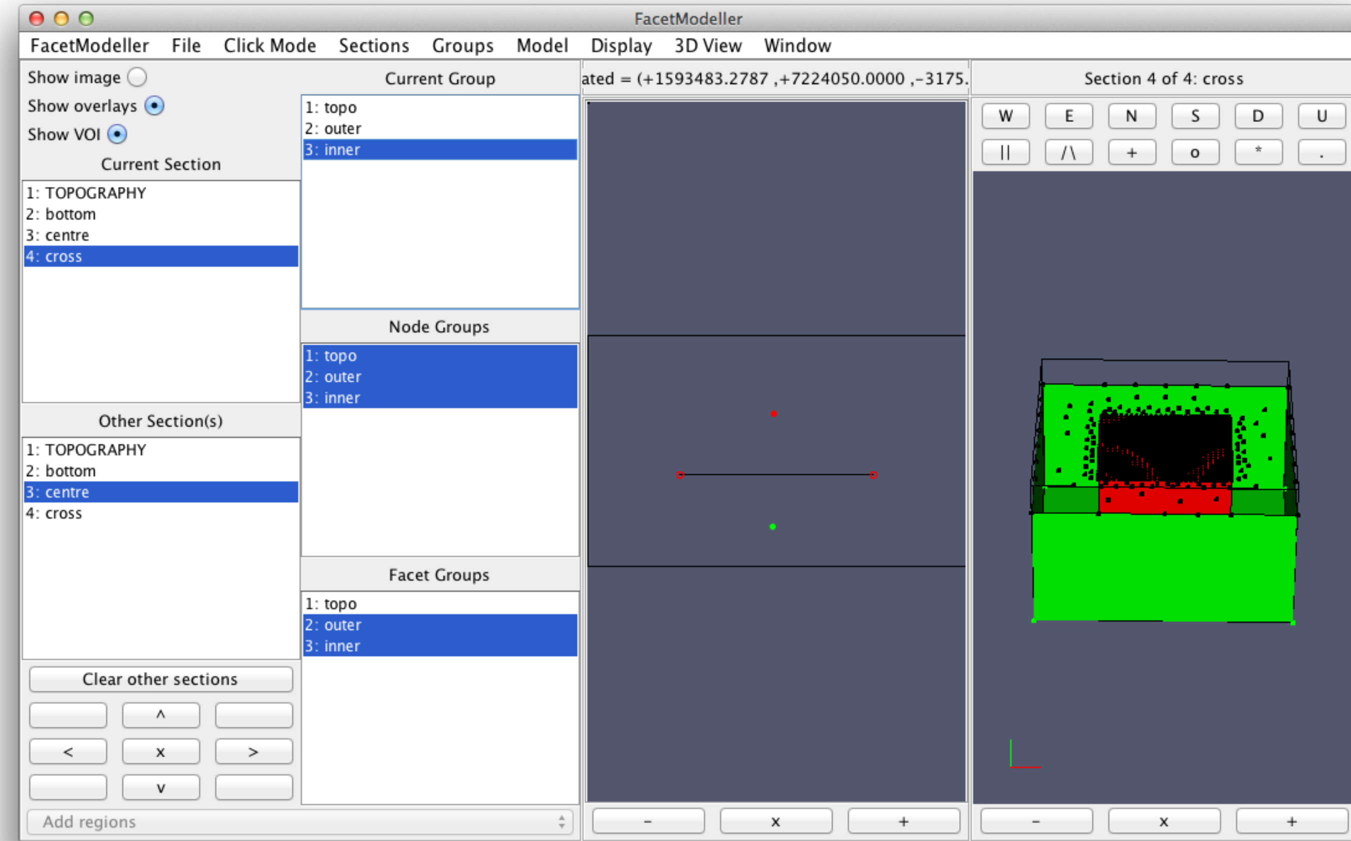
27. Select the menu item
`Model > Model information`

This will provide a dialog box showing you information about the number of nodes and facets in the model. The total number of facets should now be 1556, 10 more than the number of topography facets loaded from file. Those 10 facets include 4 on the sides of the boundary, 4 on the sides of the inner region, and 2 to close the bottoms of the outer boundary and inner region.

Finally, you need to define the two regions in your model. Defining regions in 3D is identical to the process in 2D but of course you have to now think in 3D: each region point is still attached to a specific section and you have to make sure that the region points are within the correct parts of the model volume. For this example, you will create a new vertical cross section that will slice the model in half. The region points will be attached to that new vertical cross section. You will view the model projected onto that section (in the 2D viewing panel) and you will overlay some of the other model objects such that it will become clear where you have to define the region points.

28. Create a new vertical cross section without image and name it `cross`:
the first calibration point is `"1588400 7224050 10000"`
and the second is `"1670000 7224050 -40000"`.
29. Select that `cross` section as the `Current Section`.
30. Select all the sections in the `Other Section(s)` selection box.
31. Select the `outer` group as the `Current Group`.
32. Add a new region point in the outer padding region. This can be anywhere inside the VOI and below the bottom of the inner region.
33. Select the `inner` group as the `Current Group`.
34. Add a new region point in the inner central region. Remember that the model contains topography and above the topography surface is air. Hence, the VOI contains some air and you need to make sure you add this region point somewhere below the topography surface, but still above the bottom of the inner region.

Your model should now look like this (keep in mind the topography facets are still hidden):



You can now export your model to a `.vtu` file for viewing in ParaView, or to a `.poly` file for meshing with TetGen.

Chapter 6

Tutorial - Building 3D models: III

Dealing with outcropping units

This 3D tutorial example relies on lessons learned in the 2D tutorial, 3D tutorials. You should also refer to the 2D tutorial for basic information about FacetModeller, including installation and running the program.

In this tutorial, you will learn how to sew outcropping rock unit contacts to the topography. I won't be providing explicit step-by-step instructions like in the other tutorials so I hope you remember how to do some of the basic tasks.

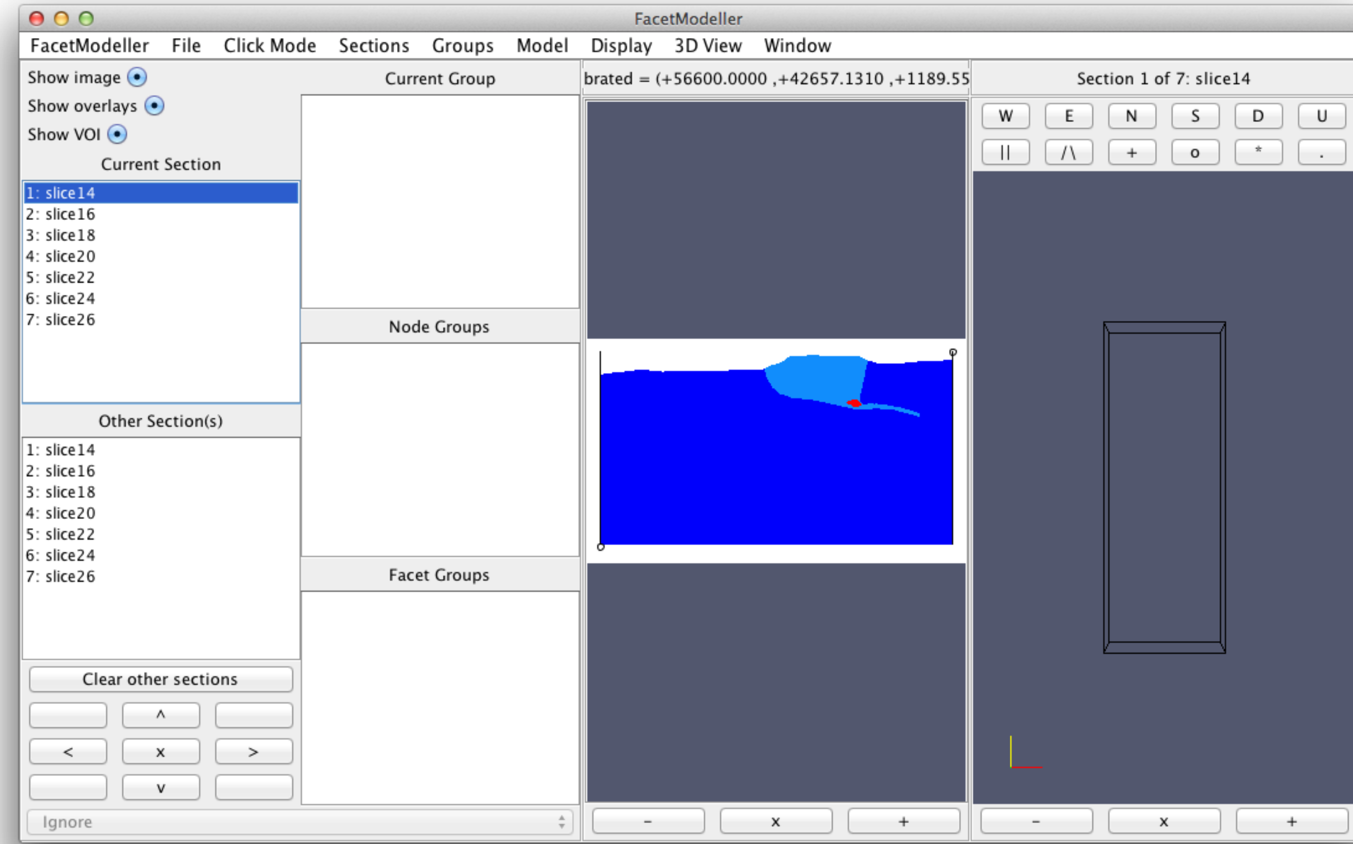
6.1 Preliminary instructions

1. Launch FacetModeller for building a 3D model.
2. Define the following VOI:
   Easting limits: `56600 57800`
   Northing limits: `40300 43600`
   Elevation limits: `-1600 225`
3. Load the image files named `slice##.png` where `##` runs from `14` to `26` (by 2). Those in my research group can find these files here
   `[repos]/peter/NetBeansProjects/examples/FacetModeller3Dc/`
   and public users here
   `[repos2]/facetmodeller/Tutorial_3D_Part3/`
   The vertical cross-sections are normal to and stacked along the x (easting) direction.
4. Calibrate section `slice14` with the following calibration information:

Click here	Corresponding spatial coordinates
bottom left of the black axes in the image	`"56600 40300 -1600"`
top right of the black axes	`"56600 43600 225"`

5. Copy the calibration to all the other sections, using a separation of `200` between sections.

Don't forget to save early and often! You should now see the following:



For this example, there are three rock units:

1. gneiss (dark blue in the images)
2. troctolite (light blue in the images)
3. sulphides (red in the images)

You'll ignore the sulphides in this tutorial and focus on modelling the troctolite body. Define new groups with names and colours indicated in the table below (you can use whatever colours you like but I find these colours helpful for this example, and I'll be referring to them in the text so I suggest you stick with these):

Name	Colour
topography	grey
troctolite	orange
outcrop	red
newTopo	green

6.2 Opening topography files

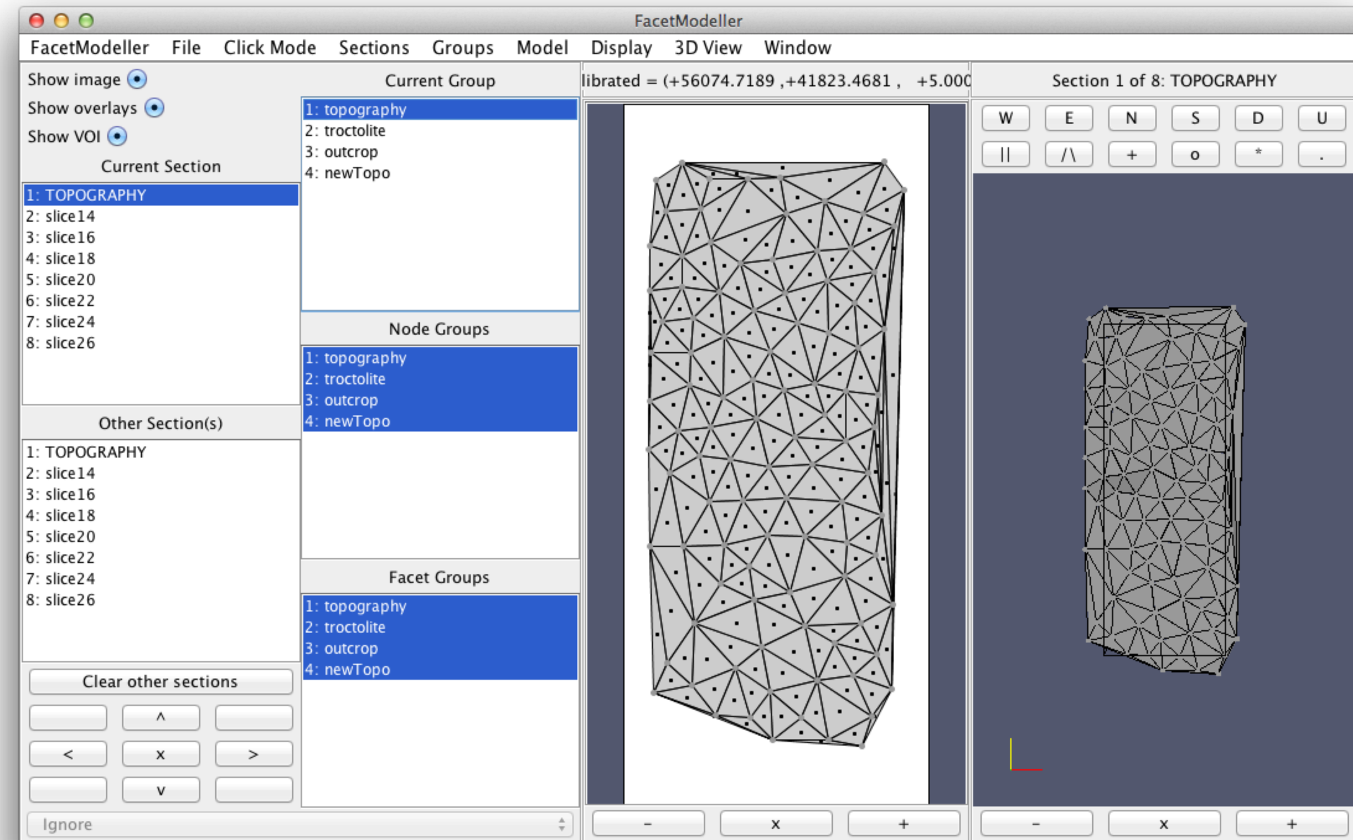
Getting your topography surface ready may take considerable effort. The process is explained in the second 3D tutorial. In this tutorial, you will not bother sewing the topography surface to the VOI.

When there are rock units that outcrop, leading to surfaces in your model that need to be sewn to the topography surface (as in this tutorial), you will want the spacing of the topography nodes to be fairly consistent with the spacing in the other surfaces to which it must be sewn. Otherwise you can end up with skinny triangles that are detrimental to mesh quality.

In this tutorial I have processed the topography surface as required (but not bothered to sew it to the VOI). To load it:

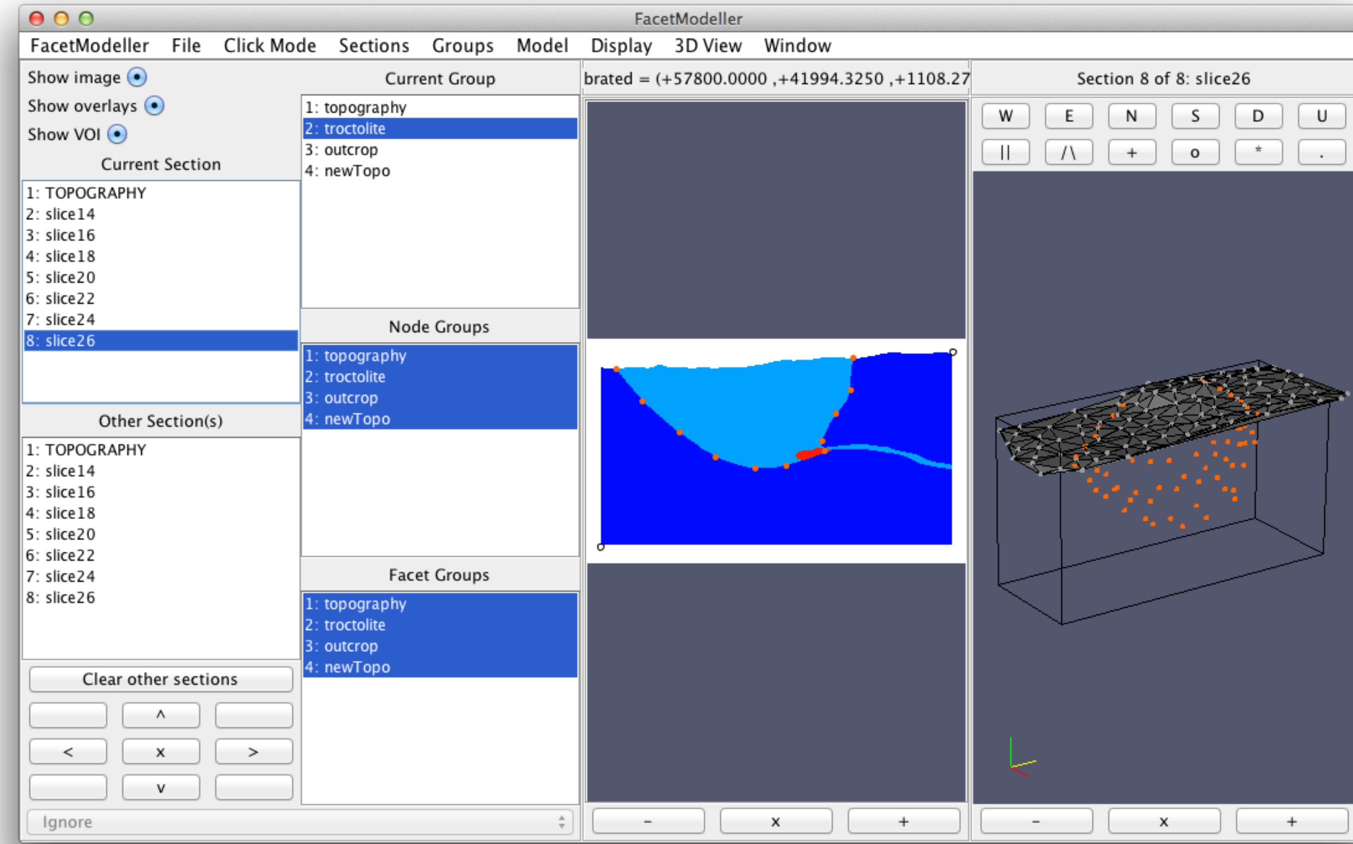
1. Make sure the `topography` group is the `Current Group`.
2. Create a new **depth** section without image named `TOPOGRAPHY`:
   the first calibration point is `"56241 39740 5"`
   and the second is `"58160 44141 5"`.
3. Select the menu item
   `File > Load from .node/.ele files`
4. Select and open the file `topo_dec3d.node` when prompted (in the same directory as the section images).
5. Select and open the file `topo_dec3d.ele` when prompted (in the same directory).
6. When asked for a colour, select the default white.

You should now see the following:



6.3 Adding nodes

As in the first 3D tutorial, you will define some nodes around the anomalous body (the troctolite in this example). Select the `troctolite` group and, for every `slice##` section, add nodes to define the gneiss-troctolite contact (between the dark and light blue colours in the images). Make sure you add nodes where this contact outcrops (connects with the topography). You can ignore the long wing-like feature at depth. Your model should now look like this:



6.4 Changing node groups

The complication in this model building example is that the gneiss-troctolite contact outcrops and must be sewn correctly to the topography surface. Before you can start creating facets, you need to make sure you can distinguish between the nodes on the contact that are below the topography and those at the topography surface.

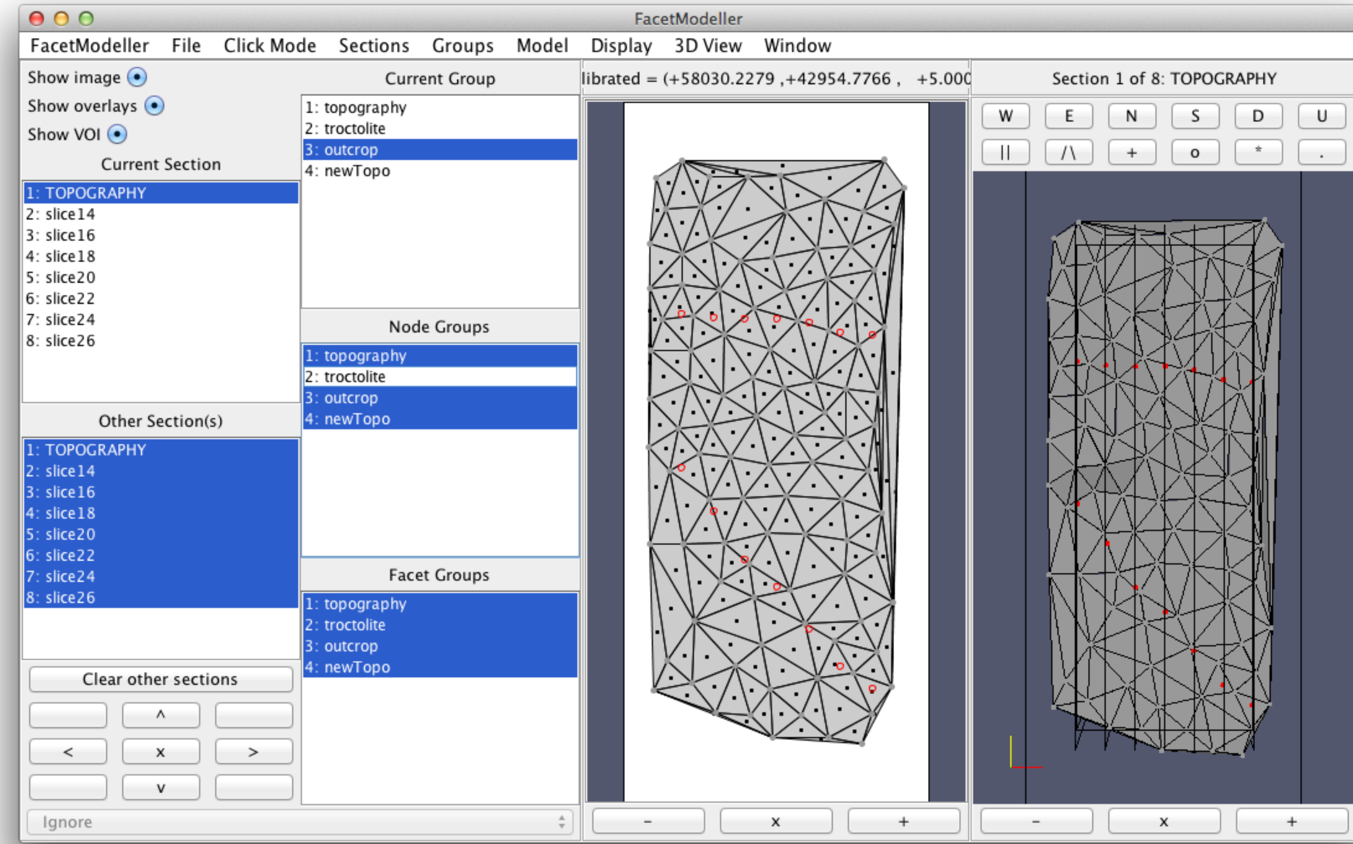
1. Select the `outcrop` group as the `Current Group`.
2. Select the menu item
   `Click Mode > Change node groups`
   Any node you now click on will be moved into the selected `outcrop` group.
3. For every `slice##` section, click on the two nodes where the gneiss-troctolite contact outcrops. You should see them change colour to red.

6.5 Defining facets

Before you continue, you will need to change the view in each view panel to an overhead view and change what is being displayed:

4. Select the `TOPOGRAPHY` section as the `Current Section`.
5. Select all the sections in the `Other Section(s)` selection box.
6. Show the 3D viewer panel and press the button labelled $\boxed{\text{D}}$ (view downward) on the top of the 3D viewer panel.
7. Zoom in a couple times on the 3D viewer.
8. Unselect the `troctolite` group in the `Node Groups` selection box.

You should now see this:

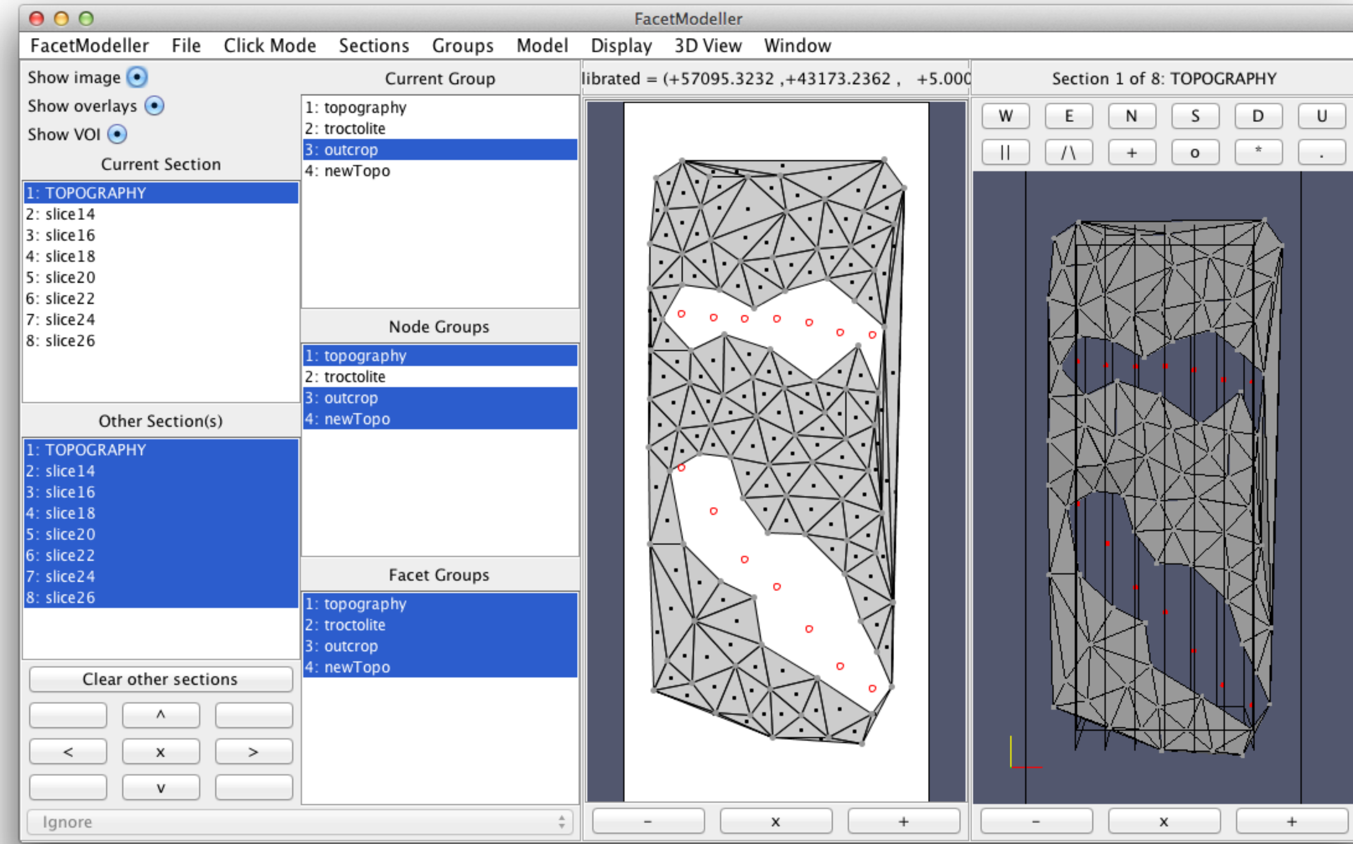


To sew the gneiss-troctolite contact to the topography surface you'll need to make sure that the nodes in the `outcrop` group are incorporated into the topography. To create the gneiss-troctolite contact surface, the red nodes will need to be connected between adjacent parallel sections. This will create two lines of red nodes along the topography surface where the gneiss-troctolite contact outcrops. Any topography facets that interfere with those connections must be removed. Refer to the next screen-shot for guidance with these steps:

9. Delete any topography nodes close to the red `outcrop` nodes or close to lines that connect those red nodes across adjacent sections.

10. Delete any remaining topography facets that enclose the red `outcrop` nodes or have edges that would intersect the lines of red nodes required to define the outcrop of the gneiss-troctolite contact.

Your model should now look like this

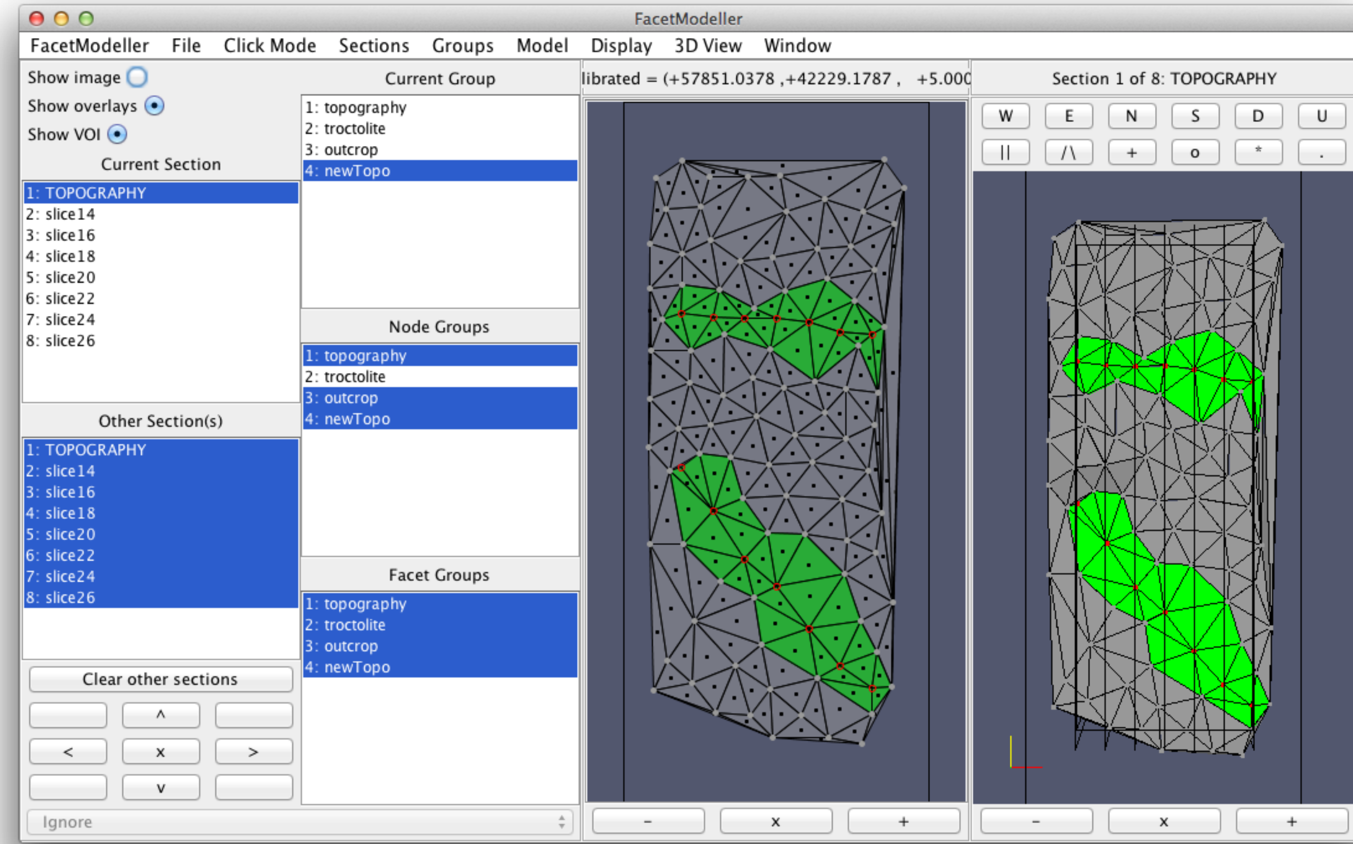


You now need to fill in the holes in the topography with new facets.

11. Select the menu item
    `Click Mode > Define triangular facets`
12. Select the `newTopo` group as the `Current Group`.
13. Unselect the `Show image` radio button radio button.
14. Define new triangular topography facets to fill in the holes. **Make sure there are facet edges linking all of the red nodes in the `outcrop` group.**

> Sometimes it may not be clear how to best connect the nodes to provide the best quality mesh (which may or may not be important to you, depending on how you intend to use your model). To help with this, the text bar above the 2D viewer panel will display the minimum vertex angle of each candidate triangular facet: for a quality mesh, you will want to avoid small vertex angles.

This task takes a while to complete. I suggest you practice defining a few facets and move on once you understand what you are doing and why you are doing it. If you do manage to get through it, your model should look like this from overhead:

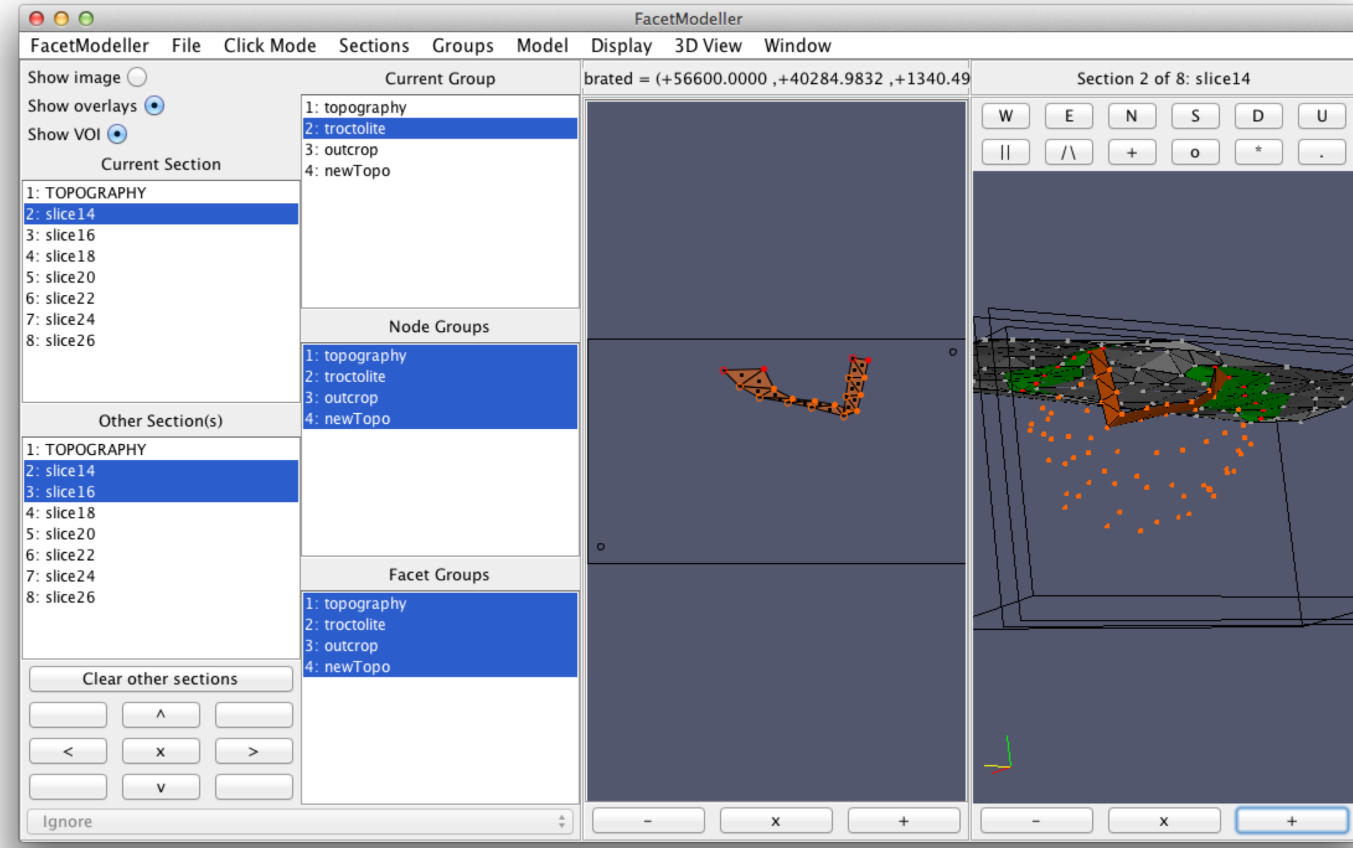


Again, the important part is that there are edges linking all of the red nodes in the `outcrop` group.

> To make this example even more complicated, you could also consider adding new nodes along the topography surface on each vertical section, then delete/define facets to merge those nodes into the existing topography surface.

6.6 Finishing the model

Now that the topography is taken care of, you can begin to create the gneiss-troctolite contact surface below the topography. To do so, follow the steps you took in the first 3D tutorial where you created facets between parallel sections. Add these facets to the `troctolite` group. Once you have finished this step with the first pair of sections, your model should look like this:



6.7 Wrap-up

You should now know how to complete this model (if you really want to). I haven't covered every possible 3D model-building task in these tutorials, nor have I shown you all that FacetModeller can do. My hope is that the application is simple enough to use (once you get the basic idea of how it works from these tutorials) that you can discover the other functionality by yourself while sifting through the menu items. I am, of course, always happy to provide support.

Future major development plans are to allow you to:

- Add new nodes to a topography surface, with the elevation of the new nodes interpolated from the existing information.
- Smoothly subdivide all the facets in a model to generate a finer smoother model, so you'll be able to create a coarse approximation of your model and then refine it automatically.

Please let me know if there are any other enhancements to FacetModeller that would help you with your model building needs. Thanks for playing!