# MeshMaker–PTC interface.
# User's guide

José-Luis Olivares

## Contents

**1 Introduction** **2**

**2 Brief description of MeshMaker usage** **2**

**3 The `ptc.dll` tabbed dialog** **3**
3.1 Basic . . . . . 4
3.2 General . . . . . 4
3.3 Time . . . . . 4
3.4 Graphs . . . . . 6
3.5 Output . . . . . 6
3.6 More output . . . . . 6

**4 The layers created by `ptc.dll`** **6**

**5 How to define a model for PTC** **8**
5.1 Define the domain outline . . . . . 8
5.2 Create the mesh . . . . . 10
5.3 Input the values for the different variables in your model . . . . . 10

**6 Defining a simple flow-only problem** **13**

**7 Running PTC** **14**

**8 How to import PTC results into MeshMaker** **16**
8.1 Bringing PTC data into a MeshMaker Data layer . . . . . 17
8.2 Making graphs of imported data . . . . . 17

# 1 Introduction

The MeshMaker-PTC interface is a user interface for PTC created within MeshMaker. It allows the user to define a problem to be solved by PTC using only the tools built into MeshMaker. The PTC configuration and execution is controlled by means of a MeshMaker extension (a Dynamically Linked Library, or DLL) that configures MeshMaker projects for PTC problems, presents a tabbed dialog for easy definition of different PTC options, and executes the program when the user requests it.

# 2 Brief description of MeshMaker usage

MeshMaker is a program to create finite element meshes or finite difference grids in a graphical, easy way. It has a set of utilities that allows you to import digitized maps, extract domain outlines from them, and automatically generate grids or meshes on the domains. It is also possible to associate different variables to the mesh or grid as a whole, or to particular elements or nodes, such as values for boundary or initial conditions, concentrations, etc. This association is made through the use of *Layers*. Finally, the data configured inside MeshMaker can be exported in formats suitable to any numerical code, because MeshMaker has a built-in language to export its data. This export process is made by means of an *Export Template*.

A *Layer* is the fundamental object to work with in MeshMaker. It can be pictured as a transparency sheet, where the users record the information of their projects. The analogy of a transparency sheet becomes obvious when you see the program running on screen. By drawing domains, meshes, grids, and maps on different transparencies that can be moved up or down, deleted or just hidden, the user can add or change parts of the project in a graphical way.

Different kinds of information can be stored on specific *types* of layers. The basic objects that MeshMaker works with, domain outlines, meshes and grids, are defined in layers of type Domain, Mesh and Grid, respectively. The other kinds of layers are Information, Maps and Data.

*Information* layers are used to associate variables with the meshes or grids. The name of an *Information* layer is also the name of a variable on it. (It is possible to define several variables in a single *Information* layer.) For example, you can define the permeability of the soil in a layer of this type. Then you can give values to it globally (that is, by giving it a constant value over the entire mesh or grid) or make it vary over the mesh/grid (using the contouring tools described in the manual). When you link this variable to a mesh, it can be assigned to nodes or elements,

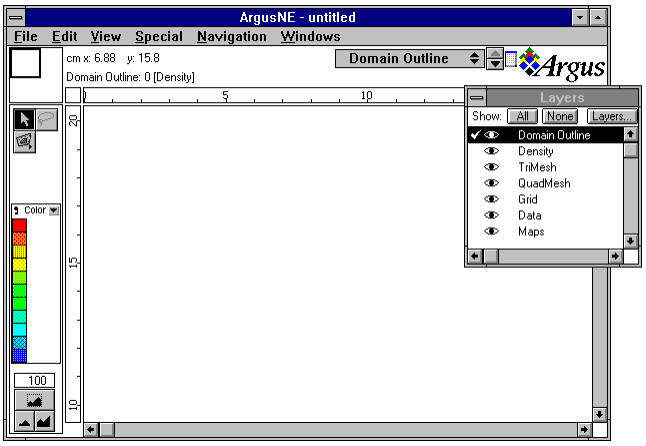


Figure 1: The main window of MeshMaker and the Layers panel.

while on a grid it can be assigned to blocks or block centers.

*Maps* layers are used to import digitized maps to MeshMaker. This permits the user to extract the Domain Outline from the map, or simply to use the map as a reference. Maps layers are also used to create graphics of different kinds, such as contours or 3d surfaces.

*Data* layers are used to import data generated by other programs (PTC in particular) into MeshMaker. Those data can be used for several things; for example, for creating graphical output of the results generated by PTC.

The main window of MeshMaker is shown in Fig. 1. Whenever the program is started, it shows up as in this figure. The program itself creates seven layers, named **Domain Outline**, **Density**, **TriMesh**, **QuadMesh**, **Grid**, **Data** and **Maps**. The control panel for moving around layers is also shown in Fig. 1. This layer structure must be modified in order to define a PTC problem. The PTC DLL takes care of this, as you will see later. To start a new PTC project, select the option New PTC Project... in the File menu, as shown in Fig. 2. This option executes `ptc.dll`, bringing to the screen the tabbed dialog shown in Fig. 3.

## 3 The `ptc.dll` tabbed dialog

In this section we describe the data to be specified in the six panes of the PTC configuration dialog.

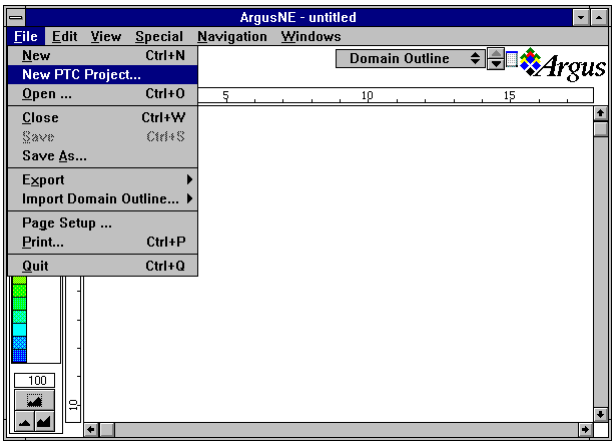


Figure 2: The “New PTC project” option in the “File” menu.

### 3.1 Basic

In this pane, you enter the Project title, the number of geological formations of your problem, the type of mesh to be used, and two physical properties: the molecular diffusion and the upstream weight. This pane is shown in Fig. 3a.

### 3.2 General

Here you specify the global strategy of your PTC run. In the General control subpanel, you specify whether or not to run flow, to do velocity calculations, to do transport, to use memory during execution, and to do mass balance calculations. You also specify the steady-state criterion for flow calculations. Finally, you select whether or not to use the water table option. This pane is shown in Fig. 3b.

### 3.3 Time

The time stepping of the PTC run is defined here. In this dialog you specify the total number of time steps of the simulation, the number of flow time steps after which the time step resets, the number of time steps after which time steps cease to change, the number of concentration time steps per flow time step, the time step multiplier (used to reset the time step), and the total time of the simulation. This pane is shown in Fig. 3c.

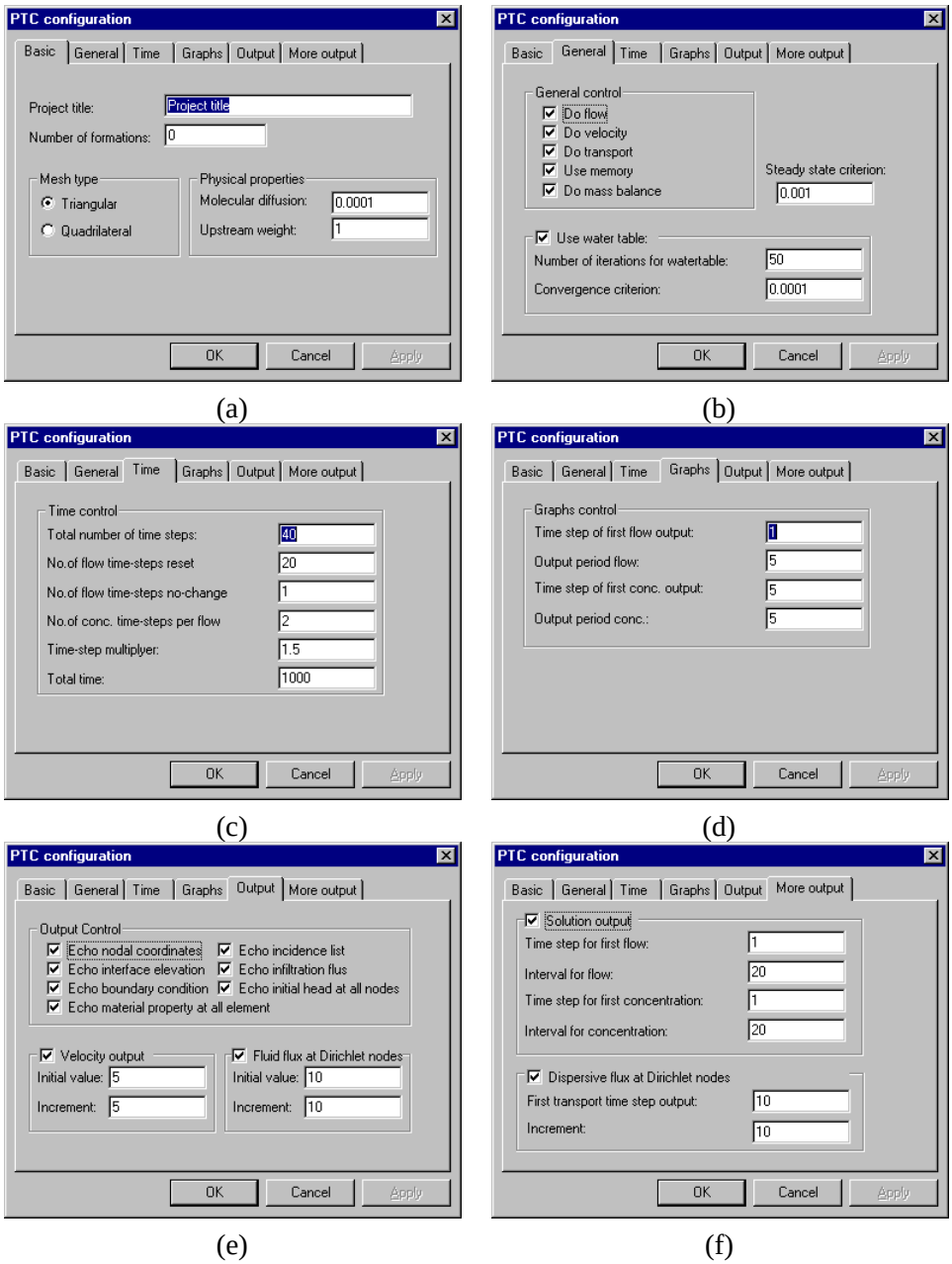


Figure 3: The “PTC configuration” tabbed dialog.

### 3.4 Graphs

The output to graphics files is controlled in this dialog. Here you specify time step number for first flow output and the output period from there on; the same is done for transport. It is shown in Fig. 3d.

### 3.5 Output

The output to the log file of the PTC session is defined in this pane. The diferent echos that can be turned on or off in the upper subpanel are written to the file named `ptc.out`. In the lower subpanels, you also specify whether to write velocities and fluid fluxes at Dirichlet nodes. This pane is shown in Fig. 3e.

### 3.6 More output

This pane is a continuation of the previous one. The output specified here also goes to `ptc.out`, the log file. Specify here the time step for first flow output, and the time intervals after the first one, and the same for concentration. Finally, select whether to write dispersive flux at Dirichlet nodes. This pane is shown in Fig. 3f.

Once the user has finished with this dialog, pressing the OK button returns control to MeshMaker. The DLL creates a new MeshMaker window for the PTC project just defined. This project has the layer structure required to configure a PTC problem inside MeshMaker. The layers created by `ptc.dll` are described next.

## 4 The layers created by `ptc.dll`

The following layers are created by the DLL:

**Domain Outline**
: Type Domain. This layer is used to define the domain to be modelled. It is a graphical representation in a plane of the surface to be studied. In general, it is a closed polygon, which approximates the real map of the area of interest.

**TriMesh** or **QuadMesh**
: Type Mesh. This layer is used to create a finite-element mesh, triangular or quadrilateral, according to the selection made in the Basic pane.

**Density**
: Type Information. The variable Density is used to define the desired average side of the elements of the grid or mesh when any of these is automatically

generated by the program, as described in the MeshMaker manual. It is associated with the Domain Outline layer. For example, if you create a square domain where each side is 10 cm and give the domain a density value of 1, when you tell the program to create the mesh it will generate elements with sides equal to 1 cm. For a quadrilateral mesh, it will create 100 elements. Changing the density value to 2 and remeshing, will give you 25 elements instead. (With a triangular mesh, you will get twice as many elements in each of the cases above.)

**Maps**

Type Maps. This layer is used to import digitized maps.

**Scrapbook**

Type Maps. This layer is used to create graphics from the PTC data for heads and/or concentrations.

**Rainl1**

Type Information. This layer is used to define the rain infiltration on the top layer (precipitation).

The number of each of the following layers created by the DLL varies depending on the number of formations specified in the Basic pane. For example, if the user defined two formations, the DLL will create two layers for the following variables: conductivities (condl1x and condl2x), dispersivities (displ1x, displ2x), storativities (Storl1, Storl2), initial conditions for flow (InitCL1, InitCL2) and concentration (Initc1, Initc2), boundary conditions for flow (BCTypL1, BCTypL2) and concentration (BCcTypL1, BCcTypL2), and boundary conditions for leakage (BCleakl1, BCleakl2). In addition, the DLL will create three layers for elevations (elevl1, elevl2, elevl3), required for specifying the bottom of the lower layer, the height of the same layer, and the height of the second (as explained in the *PTC Manual*).

For one geological formation, the layers are:

**condl1x**

Type Information. Conductivity, layer 1. This layer is used to define three variables: condl1x, condl1y, condl1z.

**displ1x**

Type Information. Dispersivity, layer 1. This layer is used to define three variables: displ1x, displ1y, displ1z.

**elevl1**

Type Information. This layer is used to define the elevation, interface 1 (bottom).

**elevl2**
Type Information. This layer is used to define the elevation, interface 2.

**InitCL1**
Type Information. Initial conditions for flow (heads), layer 1.

**Initc1**
Type Information. Initial conditions for transport (concentration), layer 1.

**Storl1**
Type Information. Storativity, layer 1.

**BCTypL1**
Type Information. Boundary Conditions for flow, layer 1. This layer is used to define 2 variables: BCTypL1 and BCValL1. Two variables are required because PTC stores two kinds of BCs for flow in the same array. BCs of type 1 are Dirichlet (specified value; BCTypL1 = 1); type 2 are flux BCs (BCTypL1 = 2).

**BCleakl1**
Type Information. Boundary Conditions for leakage. This layer is used to define 4 variables: BCleakl1, ref_head_leakl1, conductance_leakl1, ref_conc_leakl1. These values correspond to the four numbers required by PTC to define BCs of type 3 by node.

**BCcTypL1**
Type Information. Boundary Conditions for transport (concentration). This layer is used to define 2 variables: BCcTypL1 and BCcValL1. Same as with BCs for flow; BCcTypL1 = 1 is for Dirichlet (specified value at node). BCcTypL1 = 2 is for concentration of the incoming water at node.

**PTC data**
Type Data. This layer is used to import the numerical results of PTC into MeshMaker, for postprocessing (i.e., to make graphs).

# 5 How to define a model for PTC

To define a problem for PTC, you may proceed as follows.

## 5.1 Define the domain outline

This can be done in several ways:

1. the domain can be drawn directly into the **Domain Outline** layer from scratch, using the contour tool, or

2. it can be extracted from a map imported into the **Maps** layer in DXF format (or drawn from scratch but with the map on the background as reference), or

3. it can be specified in a text file and imported to MeshMaker. For example, to input a closed domain with the shape of a rectangle, create an ascii file with the following lines:

```
# This line is a comment
# Number of vertices, density
5,1
# x y for five points
10  10
10  100
100 100
100 10
10  10
```

   Lines beginning with #'s are comments, ignored by MeshMaker. The first line tells the program that five vertices will be listed, and assigns a density of 1 to the contour. Then follow the coordinates $x,y$ of the vertices, separated by blank spaces or tabs. The fifth one is the same as the first; this is the way to tell the program that the contour is closed.

To create the domain from scratch:
Define the scale of your drawing, as well as the drawing surface, as follows:

1. Make the **Domain Outline** layer the active layer.

2. On the Menu Bar, click on Special | Drawing size... and fill in the appropriate info on the window.

3. Then, again from the Menu Bar, click on Special | Scale and Units...

Now you can start drawing the Domain. Suppose you want to draw a closed domain. Pick up the contour tool from the palette to the left of the program's screen by clicking on it (it should turn black when selected). Bring the mouse pointer into the drawing area of the window (the white, big lower right portion of the program's window) and position the pointer on the coordinates of the first point of your drawing. (The coordinates of the mouse pointer are displayed above the drawing area, to the left, just below the menu bar.) Then click the mouse's left-button: the first point is set. Now move the pointer to the coordinates of the second

point. You will see that a black line appears on the screen, starting at your first point, and ending at the mouse cursor. When clicking on the second coordinate, the black line joins the two points defined, and by dragging the mouse again you will be setting the third point of your domain. You can add as many points as you wish to your contour. To close the contour, double-click on the last point you create. You will see a window with information related with the contour just created like the one shown in Fig. 4. In this new window, the important thing to set is the parameter Density appearing in the subwindow at the bottom. It controls the size of the elements on the mesh (its value is the length of the side of an element in the mesh), so the number of elements you want to have on the mesh is a function of the Domain's total area (shown in the upper subwindow) and of the Density you enter here. Its value is changed by clicking on the default value (equal to 0) and overwriting it.

With the contour tool you have just used, you can draw three different types of contours: closed, open and point. To select the type of contour to be drawn, click on the contour tool and hold the mouse button down; a menu will pop up with three different icons. The icons represent closed, open and point contours from top to bottom. Drag the pointer over the one you want to use, and release the button. Once selected, the icon on the contour tool will change correspondingly. Open contours are used to define special areas inside the domain, such as rivers. Point contours are used to define wells, observation points, sources, etc. Whenever you set any of these objects on the **Domain Outline** layer, you are forcing the program to put nodes (one node in the case of point contours) on the object when the mesh is generated.

## 5.2 Create the mesh

Make the **Mesh** layer the active layer. Click on the "magic wand" tool, bring the cursor into the domain and click on it. The mesh is created automatically.

## 5.3 Input the values for the different variables in your model

These variables are stored in Information layers, as explained above. If the value of a particular variable is constant over all the domain, then it is input through the "View | Layers" option in the Menu Bar. This will bring a window with a list of the layers already defined (upper subwindow) and with a list of the variables defined on each layer (lower subwindow). The window is shown in Fig. 5.

In the lower subwindow, the variables shown are the ones associated with the highlighted layer in the upper subwindow. Suppose you want to assign a constant value to the rain infiltration on top layer. Click on the **Rainl1** layer in the upper

subwindow. Move the cursor to the lower subwindow, where you will see that there is a variable Rainl1 already defined (it has the same name as the layer). In the Units column, the physical units of the value are written (just for information). In the third column, you see the type of variable (Real). Its value is shown in the last column, after a circled big i. To change the value, click on it and keep the mouse button down; a window with the legend "Expression" will pop up. Drag the pointer into it, and release the button. You will get a new window, which looks like a calculator on its upper-left side. To set a constant value now, click on the lower portion of the window, where there already appears a number, and overwrite it with the new value. When finished, press the OK button in the window. To learn more about how to use all the features in this window, see the MeskMaker manual.

The steps just described allow you to define a global value for a variable. In order to define a variable locally (on certain nodes or elements, or groups of them, for example), you have to use a different technique. For example, if you want to define a different value for the precipitation on some region in your domain (once you have input the global value) the procedure is as follows.

First, if you are still in the Layers window of the last example, close it and go back to the main window. Make the **Rainl1** layer the active layer. On the left panel you will see a button that looks like the contour tool you used to define the domain. It is in fact the same tool: in Information layers you can also create three types of contours (closed, open and point). In a Domain type layer, the tool is used to define the area of interest. In Information layers, it is used to define subregions of your domain in which the variables in the layer will take different values.

To create a closed contour, proceed as you did to create the domain. Pick up the contour tool, click the left mouse button where you want to put the first point, move to the second location, etc. When finished, double-click on the last point; you will get a window like the one shown in Fig. 4, but with the information of the new contour and with the values of the variables in the layer shown in the subwindow to the bottom. Enter the new values for the variables as you did for the Density parameter when you created the domain. The nodes you enclosed in the contour will have the values that you just entered for the variables in the layer.

There are three methods in which the program assigns values to nodes once you have created a contour on an Information layer. They are shown in the Layers panel described above (Fig. 5). There, below the second subwindow (where variables in the highlighted layer are listed), at the bottom of the window, there is a scrollable line with the legend "When probed for value, use:". The three methods to choose from are Nearest Contour method, Exact Contour method and Interpolation method. The first one, *Nearest*, sets the value of a variable at a node (or element) to that of the nearest contour. *Exact* sets the value to contours over the node or that contain it (in the case of closed contours), and the third one interpolates scattered

Table 1: Variables in MeshMaker and its corresponding PTC parameters.

| MeshMaker layer | MeshMaker variable | PTC distributed parameter |
|---|---|---|
| Rainl1 | Rainl1 | RAIN |
| condl1x | condl1x | CONDX |
| | condl1y | CONDY |
| | condl1z | CONDZ |
| displ1x | displ1x | ELONG |
| | displ1y | ETRAN |
| | displ1z | EVERT |
| elevl1 | elevl1 | ELEV |
| elevl2 | elevl2 | ELEV |
| InitCL1 | InitCL1 | INITH |
| Initc1 | Initc1 | INITC |
| Storl1 | Storl1 | STOR |
| BCTypL1 | BCTypL1 | BCFLO |
| | BCValL1 | BCFLO |
| BCleakl1 | BCleakl1 | LEAK |
| | ref_head_leakl1 | LEAK |
| | conductance_leakl1 | LEAK |
| | ref_conc_leakl1 | LEAK |
| BCcTypL1 | BCcTypL1 | BCCON |
| | BCcValL1 | BCCON |

values troughout the entire domain.

Following the steps just described, you can input the information for the entire model over the different layers: **Rainl1**, **condl1x**, **displ1x**, **elevl1**, **elevl2**, etc. Each Information layer keeps information related with PTC input data, as shown in Table 1. Notice that some MeshMaker variables are stored together in a single layer, like conductivities in $(x, y, z)$. (The PTC parameters are described in pp. 53–56 of the *PTC manual*.)

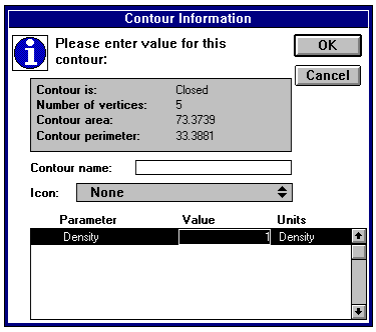


Figure 4: The contour information window.

# 6 Defining a simple flow-only problem

In the previous section, we described briefly how to enter information over the different layers. Here, we list the procedure to define a PTC problem involving only flow on a single geological formation.

The steps for this problem are:

1. Create the domain in the **Domain Outline** layer. Remember to enter the well locations of your problem as point contours here.

2. Create the mesh in the **Mesh** layer, using the magic wand tool.

3. Enter the global values for the following variables in the Layers panel (this panel is shown in Fig. 5:

   (a) Elevations. Layers **elevl1** and **elevl2**.

   (b) Conductivities. Layer **condl1x**, which contains three variables.

   (c) Storage coefficient. Layer **storl1**.

4. Enter boundary conditions for flow at particular locations or regions. Layer **BCTypL1**. Boundary conditions are defined using contours, not globally as in the previous layers. To specify them, make the **BCTypL1** layer active, pick up the contour tool, and create contours as necessary. Remember to copy the point contours you created in the **Domain Outline** layer to input sources (well locations) in this layer, and enter the correct numbers for the BC type and value. (Wells are BC type 2.)

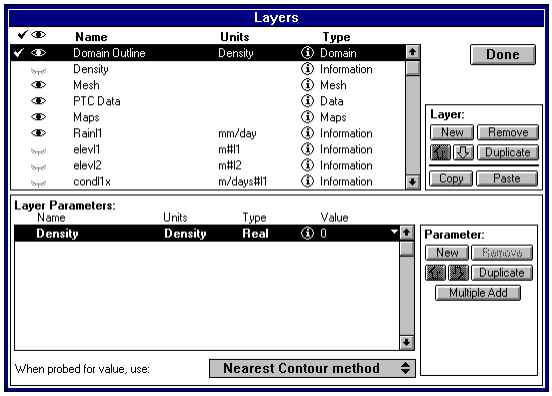


Figure 5: The layers window.

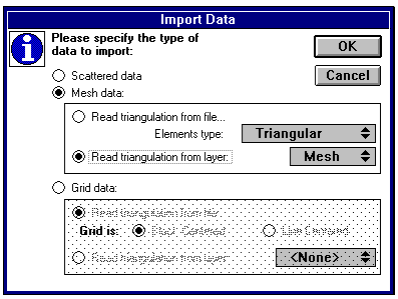


Figure 6: The Import Data window.

# 7 Running PTC

Once the problem is fully defined in MeshMaker, you can run PTC to obtain the solution. To launch PTC, make the **Mesh** layer active and select File | Export | Run PTC from the menu. This will bring back the "PTC configuration" tabbed dialog discussed before (see Sect. 3). You can change all of the settings you specified the first time you used this dialog, except for two entries: the type of mesh and the number of geological formations.

As mentioned before, this tabbed dialog lets you configure the PTC run. Table 2 lists the different **Control** panels and the PTC commands they define.

Clicking on the OK button this time starts the export process. First, you will get a new window, asking for information about General Export Arguments; nothing needs to be changed here, so simply click the OK button. In a second window, you are asked to enter the export file name. Type in the name, or use the default (`mesh.exp`), and press again the OK button.

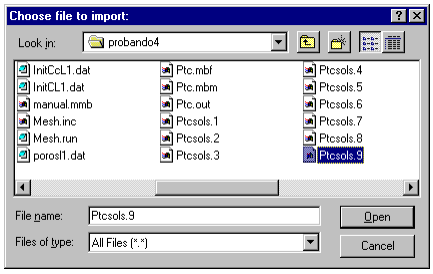


Figure 7: The Choose file to import: window.

Table 2: Tabs in the "PTC configuration" dialog and the PTC commands associated with them.

| MeshMaker dialog | PTC command |
|---|---|
| **Time** | gntime |
| **Graphs** | rdgrfkey |
| **Output** | rdprtkey |
| **General** | rdcontrl, |
| | rdsteady, |
| | rdwtable |

The export process starts, and the program reports it while in progress. The result is several files written in the subdirectory where you chose to save the export file. The files created all have the extension `.dat`, and its names are the names of the PTC variables in the Information layers: `Rainl1.dat`, `condl1x.dat`, `condl1y.dat`, `condl1z.dat`, etc. Two other files have the name you wrote for the export file, but with different extensions. If you used the default, `mesh.exp`, these files are `mesh.run` and `mesh.inc`; the first one is the main PTC run file while `mesh.inc` contains node coordinates and incidences.

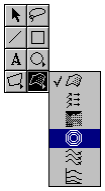

Figure 8: The **Maps** layer is active; the Drawing and Graphics Tools are shown.

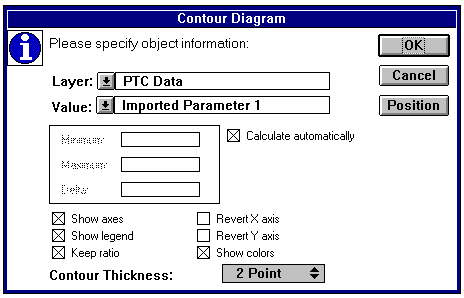


Figure 9: The Contour Diagram window.

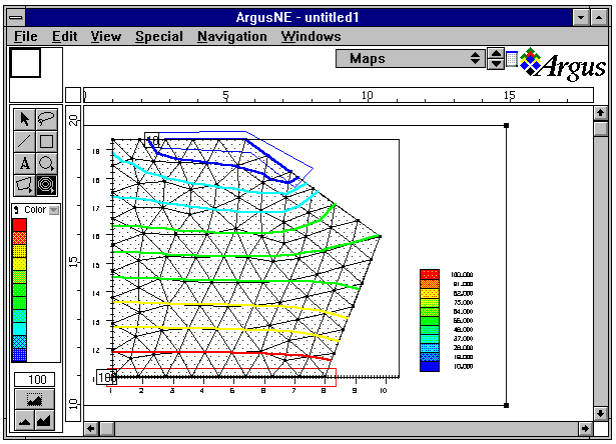


Figure 10: A contour diagram.

Right after the export process is finished, you will see a MS-DOS window open, and the command

```
ptc < mesh.run
```

(assuming you chose the default filename in the previous panel) being executed on it. This is the actual run of PTC on the data set just exported. When it is finished, the window will close by itself. The results of PTC will be saved in files with names `ptcsols.1`, `ptcsols.2`, etc.

# 8 How to import PTC results into MeshMaker

The output files from PTC can be brought back into MeshMaker for postprocessing. To create graphs, you will use two other predefined layers in your MeshMaker

project, namely, **PTC Data** and **scrapbook**. The import process has two steps, described in the following subsections.

## 8.1 Bringing PTC data into a MeshMaker Data layer

First, to bring PTC output into MeshMaker, make the **PTC Data** layer the active layer. Select File | Import PTC Data | Text File from the Menu bar. You will get a window like the one shown in Fig. 6. This window informs the program of the type of data you will be bringing in. The data are from a mesh, so select this entry, and also click on Read triangulation from layer (the **Mesh** layer, where the data come from). Click on the OK button. A second window will appear, like the one shown in Fig. 7. Enter the name of the file with PTC results you want to use (remember, these files have names `ptcsols.1`, etc.) and again press OK. If everything goes well, you should see a set of points in the layer, indicating that the values read were assigned to the nodes in the mesh.

The data just imported can be used for several purposes described in the MeshMaker manual. Here we will use them for making graphs.

## 8.2 Making graphs of imported data

The second step actually creates the graph. To proceed, make the **scrapbook** layer the active layer. As you can see in Fig. 8, in this layer there are several tools in the left panel. The tool in the lower right position is the Graphics tool, which is already opened in the window (to bring that menu of icons to the screen, click and hold the mouse left button on the tool). The six icons shown correspond to six different kinds of graphs: 3d surface, vector, color diagram, contour diagram, path-line and cross-section, from top to bottom.

To create a contour diagram with the data imported into **PTC data**, select the fourth icon. Move the mouse pointer to the drawing area. Click and hold the left mouse button and drag the cursor; a rectangle is drawn. When you release the mouse button, a window like the one shown in Fig. 9 will show up. There, you specifiy the Data layer from where the data will come, and several other options. To overlay the graph on top of your mesh, click on the "Position" button and select the option to "Overlay Source data" in the panel that shows up. When finished, click OK. A graph similar to the one shown in Fig. 10 will be created.