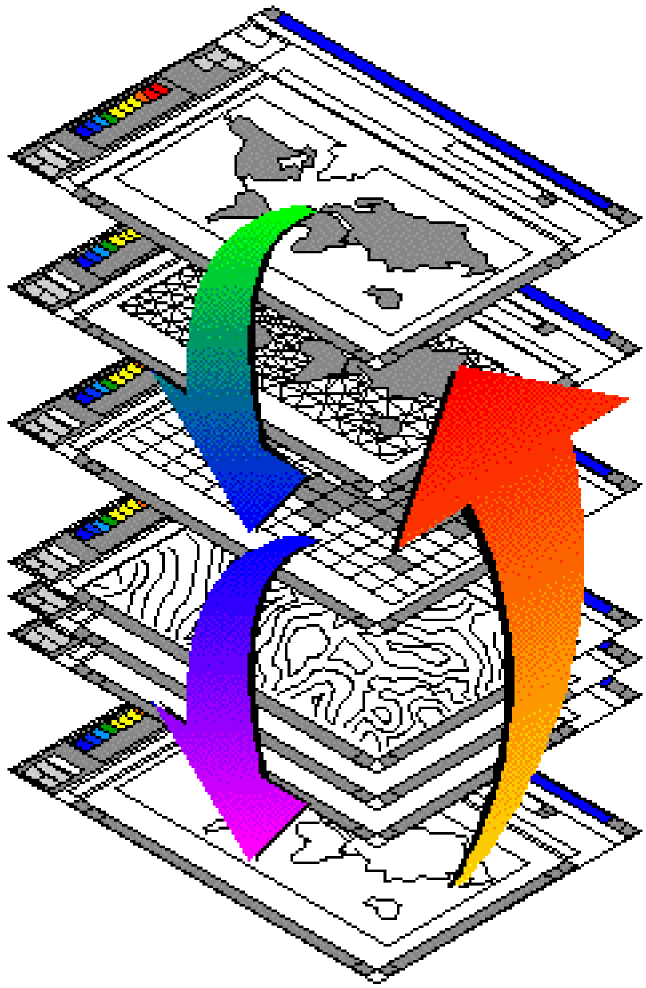
# User's Guide



# Argus ONE

*Argus Open Numerical Environments - A GIS Modeling System* *Version 4.0*

This product was developed using Neuron Data Open Interface Elements.

Sales and services offices at:

Argus Interware, Inc.

11 Merit Lane

Jericho, NY 11753

USA

Argus Documentation Serial Number 5220073-10

“Science ran too far ahead of us too quickly, and the people got lost in the mechanical wilderness, like children making over pretty things, gadgets, helicopters, rockets; emphasizing the wrong items, emphasizing machines instead of how to run machines.”

Ray Bradbury, The Martian Chronicles

# Introduction

Argus ONE is a *model-independent* Geographical Information System (*GIS*) for numerical *modeling*. Using a *conceptual-model* approach, combined with *export scripting* capabilities and *plug-in* support, Argus ONE is also an *application development environment* for developing and deploying g*raphical user interfaces* for numerical models.

**GIS -** Argus ONE provides a user environment where geospatial (map-type) information (or coverages) may be easily synthesized in preparation for use as input to numerical models. As in other GIS systems, the various types of geospatial information are stored and viewed in coverages or "layers" which the user may view and interact with on screen.

**Conceptual Model -** Mesh and grid layers, which are also available, are used to automatically create meshes and grids onto which a discrete realization of the "real" continuous world, described in the geospatial layers, is synthesized. Information is synthesized using mathematical, logical and spatial functions to define relations between layers. These relations form a *conceptual model* which represents the relations between geospatial entities as they are articulated by the underlying concepts of the model being used.

**On-Demand Synthesis and Analysis of Information (ODISA)-** ensures that your information is always model-ready. Regardless of changes you introduce in your information or meshes, Argus ONE automatically synthesizes the information according to the relations and rules set by the conceptual model to ensure your data is ready to be exported to your model.

**Model Independent -** Export scripting enables you to export the synthesized information to input files for your numerical model at the exact format it requires. Combining export scripting and the conceptual model approach, Argus ONE offers a *model independent* environment which enables you to use it as a pre-processor for any model. At the same time, it also enables you to interchangeably use geospatial coverages for any simulation model coupled with Argus ONE.

## Application Development Environment

Plug-in support allows one to automate the use of all of the above mentioned Argus ONE components. Argus' *Plug-in Extension (PIE)* technology enables a two-way communication between external programs and Argus ONE. Such external programs may be numerical models and any other algorithms or utilities that may *install menus*, *launch dialogs* and *execute external programs* in and from within the Argus ONE graphical user interface.

Argus ONE PIEs may be freely distributed to any user of Argus ONE.

# Contents

# Part 2 Basics for Using Argus ONE

# Before You Begin

Welcome to Argus ONE™, a general purpose, finite element and finite difference pre-and post-processor. Argus ONE is available on the Microsoft Windows, Apple Mac OS and X Windows operating systems.

Argus ONE provides the automated engineering tools you need for modeling. Creating and editing finite element meshes and finite difference grids; defining, manipulating and applying boundary conditions, initial conditions, and parameters to the meshes and grids; exporting that information to any numerical model (simulator) you use, and visualizing your model results are all integrated within the Argus ONE workplace. The various tools are organized in the following Argus ONE modules:

## GIS Module

*See Part 2 Chapters 2 and 3*

- **Information Layers** You can import, digitize and edit any type of spatial and nodal information such as boundary conditions, point sources and sinks, topography, and physical parameters, into information layers. Using this information you can define the domain's outline and the required mesh and grid densities and automatically assign physical information to mesh and grid objects. You can mathematically manipulate the information to calculate quantities such as formation's thickness, effective nodal replenishments, etc.

*See Supplement for version 2.5*

- **Data Layers** You can import and read scattered data, data on grids and data on meshes into Data layers to visualize your model results and to interpolate such data onto your meshes and grids.

*See Chapter 1, and all supplements*

- **Maps Layers** You can import text, DXF, GIS shape files and images into Maps layers to create background maps to help you navigate through your problem domain, annotate these maps with CAD drawing tools, and create post-processing visualization objects of your information distribution and model results.

## Mesh and Grid Modules

*See Part 3*

- **Finite Element Mesh Layers** You can automatically create complex finite element meshes and edit them manually. You can automatically assign data stored in information layers to elements and/or nodes and manipulate that information with respect to mesh parameters such as the element area. You can use complex search criteria to select groups of elements or nodes answering the specified criteria. Finally, you can export the mesh to the model of your choice.

*See Part 4*

- **Finite Difference Grid Layers** You can automatically create complex finite difference grids and edit them manually. You can automatically assign data stored in information layers to the blocks and manipulate that information with respect to grid parameters such as the block area. You can use complex search criteria to select groups of blocks answering the specified criteria. Finally, you can export the grid to the model of your choice.

### Programmable Export Modules

*See Chapter 5, Chapter 3 and supplements*

- **Export Templates** You can write export scripts which will export your data in the format required by your model.

## Argus ONE Grid and Mesh Modules

There are three different grid and mesh modules: Finite Difference Grid module, Triangular Finite Elements module and Quadrilateral Finite Elements module. The three modules are always present within the Argus ONE workplace but only those you have purchased and registered will allow you to save and export unlimited number of nodes, elements or blocks (cells). Unregistered modules are fully functional but do not allow you save information in their respective layers. You can however export a limited number of nodes or blocks to allow you to experiment with unregistered modules.

## Argus ONE Plug-In Extensions (PIEs)

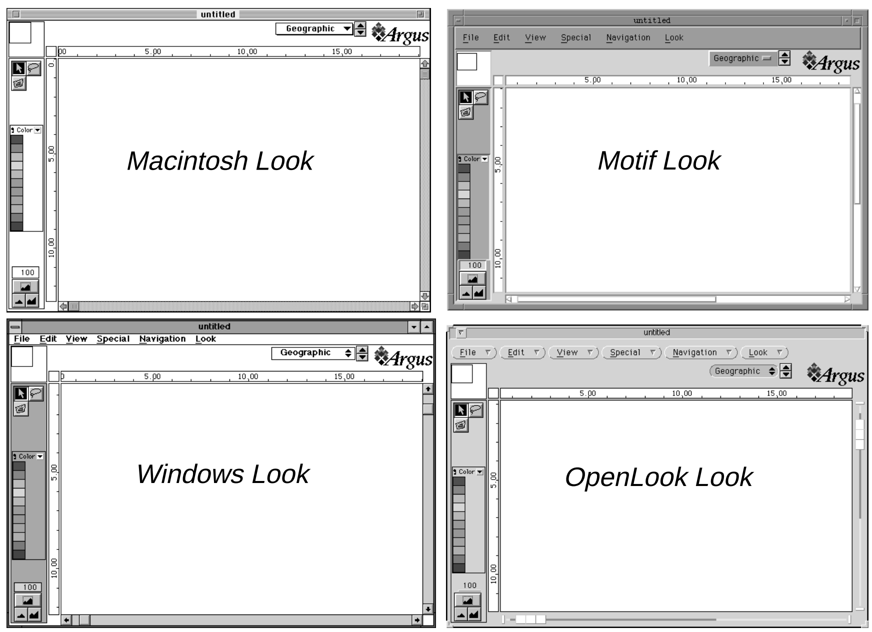
PIEs make Argus ONE a fully open and automated system, to which users can easily interface state of the art models and technologies. The Argus ONE PIE technology is fully documented and accompanied by many useful examples contributed by Argus and its users. The technology is an open architecture technology and is made available to you free of charge. For availability of documentation, technical notes and examples contact Argus. All PIE related materials are also posted on the Argus ftp site.

Many companies, organizations and universities are developing an abundance of scientific tools and fully interfaced numerical models using the PIE technology. PIEs for Ground Water flow and transport models such as MODFLOW, MOC3D, SUTRA, HST3D, and PTC are and will be offered to you as commercial products and some as public domain products. Other PIEs such as geostatistical PIEs, new auto mesh generators, import GIS information are also available.

# Argus ONE Documentation

*The supplements describe all features added since version 2. It is highly recommended that you read them since they contain information not mentioned elsewhere in the four parts of the User's Guide.*

*The Argus ONE Users's Guide* describes Argus ONE features and capabilities and gives procedures for using them. The book covers both Argus ONE and for all windowing systems they are available on. The actual windows, dialogs, menus and alerts looks are the same for all systems. Only the window frame, font styles, and colors might differ. Some windows and dialogs in this manual are of Microsoft Windows, some of Macintosh and some of Motif. The following pictures show Argus ONE and windows on four different windowing systems:




When instructions for the various windowing systems differ, you'll see a note such as "In Argus ONE for Windows" or "In Argus ONE for the Macintosh." Follow the procedures that apply to you.

The user's guide is intended for Argus ONE users of all experience levels. To help you learn and use Argus ONE efficiently, this manual is organized by tasks, beginning with the most common tasks and moving on to more advanced Argus ONE features.

Before you start using Argus ONE, it's important to understand the terms and notational conventions used in this documentation.

## General Conventions

- The world "choose" is used for carrying out a menu command or a command button in a dialog box.

- The phrase "Choose the OK button" means that you can either click the OK button with the mouse or press the ENTER key on the keyboard to carry out the default action.
- The word "select" is used for highlighting the part of the mesh, grid, contour or any selectable object that you want your next action to affect, and for selecting a specific dialog box option.
- Bulleted lists (•), such as this one, provide information, not procedural steps.
- A numbered list (**1**,**2**,**…**), indicates a procedure with one or more sequential steps.
- Whenever possible, general procedures are given that apply to the mouse. For equivalent keyboard procedures, see Appendix A, "Keyboard and Mouse Shortcuts."

## Mouse Conventions

You can either use a single-button mouse or multiple-button mouse with Argus ONE.

- If you have a multiple-button mouse, Argus ONE assumes that you have configured the left mouse button as the primary mouse button. Any procedure that requires you to click the secondary button will refer to it as the right mouse button.
- "Point" means to position the mouse pointer until the tip of the pointer rests on what you want to point to on the screen. For example, "Point to the node."
- "Click" means to press and immediately release the mouse button without moving the mouse. For example, "Click the element to select it."
- "Double-click" means to click the mouse button twice in rapid succession. For example, "Double-click the block to open the Block Information Dialog Box."
- "Drag" means to press the mouse button and hold it down while moving the mouse; then release the button. For example, "Drag the node."

## Keyboard Conventions

- Key names match the names shown on most keyboards and appear in small capital letters. For example the Shift key appears as SHIFT.
- The RETURN key and the ENTER key usually perform the same action in Argus ONE. In the User's Guide, "Press ENTER" means that you can press either ENTER or RETURN, unless specifically stated otherwise.
- The CTRL key and the COMMAND key perform the same action in Argus ONE. The COMMAND key appears on Macintosh keyboards, and the CTRL key appears on keyboards used on other windowing systems.

- The ALT key and the OPTION key perform the same action in Argus ONE. The OPTION key appears on Macintosh keyboards, and the ALT key appears on keyboards used on other windowing systems.
- A plus sign (+) used between two key names indicates that you must press both keys at the same time. For example, "Press COMMAND+OPTION" means that you must press the COMMAND key and hold it down while you press the OPTION key.

# Installing and Starting Argus ONE

Installing and launching Argus ONE differ from platform to platform. The following instructions tell you how to install Argus ONE on your computer. To find out how to start working with Argus ONE, refer to "Part One" in this manual.

## To Install Argus ONE for MS Windows 95 and NT

*To install and use Argus ONE, the Windows operating system must be installed on your local hard disk and not on a network disk.*

1. Insert the disk labeled "Installation Disk 1" in drive A.
2. From the File menu in either Program Manager or File Manager, choose Run.
3. Type **a:setup**
4. Press ENTER.
5. Follow the Setup instructions on the screen.
6. Before you run Argus ONE, make sure your MS Windows default font size is set to Normal and not to Large. If you try running with Large fonts some texts in Argus ONE's dialogs might become partially hidden.

## To Install Argus ONE for Power Macintosh

1. Copy the contents of your Argus ONE disk(s) into your hard disk.
2. Double-click the .sea file. You will be asked to choose the folder where you want Argus ONE installed. In versions later than 2.5 the application name is ArgusONE (Argus Open Numerical Environments)
3. Choose the folder.
After installing the application you can launch it from the finder.

## To Install Argus ONE for Unix Workstations

1. Create a directory for the application, for example,
/usr/bin/ArgusONE/
Command: mkdir /usr/bin/ArgusONE
Note: you can use any directory to upload the application, and move it to the final destination later on, thus super user privileges are not needed during this step.

2. Upload the application and the libraries from the tape.
   commands: cd /usr/bin/ArgusONE/
   tar x
3. Add the following command to the .shrc or .cshrc:
   setenv ND_PATH /usr/bin/ArgusONE
4. add /usr/bin/ArgusONE to the path
5. Typing ArgusONE will start Argus ONE.

The first time you run the application, a registration dialog appears. During the registration stage, you should have write access to the directory /usr/bin/ArgusONE/

After the registration process is complete, you can use the application with read access, and no write access is needed.

# Registering Argus ONE

On Macintoshs and PCs make sure the portability enabler is connected to the appropriate port. The first time you run the application, a registration dialog appears. Follow the registration instruction as described on the Application Registration Instructions card. These instructions can be also accessed from the registration dialog by clicking the Help button.

# Argus Product Support

Argus offers a variety of support options to help you get the most from your Argus product. This section summarizes these options.

If you have technical questions or problems operating your Argus product, please take a moment to check the following items. Many problems can be solved without our help.

1. Does your hardware meet the minimum requirements for the software? These requirements are listed on the next page. Check memory, disk storage, operating system type, and version.
2. Have you checked to see if the problem or question is explained in the documentation?
3. Are your hardware and peripherals set up according to the documentation? Are all cable connections secured?

Before calling please fill out the information below and have it available when you call Argus Product Support. This will enable the Support Specialist to answer your question more quickly.

| | |
|---|---|
| Software ________________________ | Version ________________________ |
| **Registration No**. ________________ | Computer Type & Model _________ |
| **Activation Code**__________________ | RAM (memory) ________________ |
| Hard Disk _______________________ | Operating System _______________ |
| System Version ___________________ | Network Type __________________ |

Argus ONE's version is specified in the greeting window that opens when you launch Argus ONE. On Macintosh versions you can see this window by choosing About Argus ONE from the Apple menu.

## Use Electronic Mail

Because of the very technical nature of Argus products, it is best that you eMail your problem description to the *Electronic Mail Technical Support Service Center.* Using the *Electronic Mail Technical Support Service* insures you the fastest and most professional response.

Internet Address: **support@argusint.com**

## Internet Home pages and Anonymous ftp site

Argus Internet Home pages and an anonymous ftp site are also available. New versions, updates, demos and other relevant materials are available through this site. You can also use certain directories to place your files for Argus Tech Support to examine. If you wish to use this service contact Argus Tech Support at support@argusint.com, or visit our home pages at:

**http://www.argusint.com**

## Fax Your Problem Description to Argus Product Support

You can fax Argus Product Support Services any time. Your fax will be transferred to a technical support person. Fax: dial **(516) 942-5040**

## Call Argus Product Support

You can reach Argus Product Support Services between 9:00 A.M. and 5:00 P.M. Eastern Standard Time (EST), Monday through Friday.

For assistance dial **(516) 931-4725**

# Hardware Requirements and Performance Considerations

*To improve redraw performance you can use Argus ONE's information hiding to decrease the number of graphic objects presented on the screen at the same time.*

Argus ONE is a graphic intensive application. Its computing demands stem mainly from the need to redraw graphic objects such as meshes, grids, polygons and maps. As you increase the complexity of your projects, that is, the number of graphic objects displayed, redraw time will become longer. To improve the redraw performance, before replacing your computer, you can gradually upgrade your hardware in the following order:

- Reduce Virtual Memory Paging by installing more RAM and/or upgrade to a faster hard disk. Increasing RAM size will also improve the general performance of Argus ONE in some memory intensive operations such as Auto Mesh Generation.
- Install a Graphic Accelerator Card.

## Platform Specific Hardware Requirements

The following table summarizes the minimum system and hardware requirements for the different computing platforms Argus ONE is supported on.

| Hardware & System Software | Minimum Requirements | | |
|---|---|---|---|
| | Power Macintosh | PC | UNIX Platforms |
| System Software | 7.6 or higher | MS Windows 95, MS Windows NT | UNIX with X Windows |
| CPU | 601 | Pentium, Pent. Pro recommended | |
| Mouse | Required | Required | Required |
| RAM | 32 MB | 32 MB or more recommended | |
| DISK | 5 /8 MB | 7 MB | 15 MB |
| Display | Color Display | VGA/SVGA Display, 65,000 colors | Color Display |
| Printer | Any Printer Supported by Operating System | | PostScript |

Part 1

# Basic Things to Know About Argus ONE

# Your First Argus ONE Project

Like most people, you probably want to get started on your work immediately. This section is designed to let you do just that. In the following pages, you can quickly learn the basic skills and concepts you will need the first time you use Argus ONE. You will learn how to:

- Create a project
- Edit your project
- Save your project
- Export your project
- Print your project

## Is This Section for You?

Unless you're experienced with Argus ONE, the answer is yes. You might want to skip material you're already familiar with, but you should pay particular attention to the tips and hints boxes. They provide information that can help you work more productively with Argus ONE right from the beginning.

### If you're new to Windowing Systems based Applications

Software applications developed for windowing systems are very easy to use. However, if you've been using mostly Mainframes and DOS based applications, you might find using Argus ONE very different. Argus ONE, as many other modern software packages is not menu driven, but object based. This section will help you familiarize yourself with some of the new concepts and techniques you will be using to operate Argus ONE.

### If you're new to Auto Mesh and Auto Grid Generation Techniques

If you have always created FEMs and FDGs manually, or by using batch oriented Automatic Mesh and Grid Generators (AMGs & AGGs), you may not know where to begin, or once started, what to do next with Argus ONE. This section guides you through a simple work scenario –from starting a new project to printing or exporting a final version. It also points out some important facts about meshing techniques.

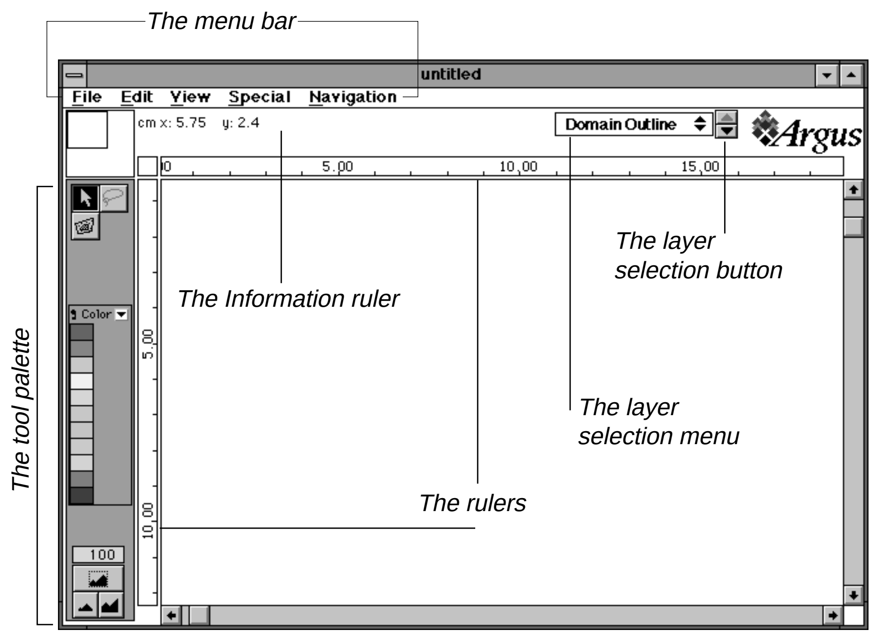
# Starting Argus ONE

Starting Argus ONE displays its main window. Menus, arranged at the top of the screen (Macintosh) or at the top of the window (other windowing systems), list the commands you'll use while working in Argus ONE. At the top of the window is the window's title. Below the window title (Macintosh), or below the menu bar (other windowing systems), is the information ruler. The information ruler provides you with information regarding mouse location, objects' information (nodes and elements), your current active layer, the coordinate system units and the current view with respect to the drawing size and the domain's outline. The inner most rulers are the drawing rulers which provide you with the location of objects on the screen in the units of your choice. On the left side of Argus ONE's window you will find the tool palette which includes the tools most often used while working in Argus ONE.

Once you launch the application, a new project window is displayed.

*A new project window*

*Use the layer selection button or menu to move between layers.*



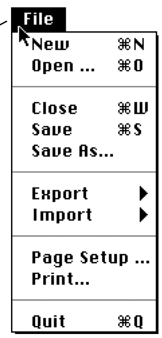

The project's coordinate system direction, scale and units, origin, drawing size, as many other parameters, are preset to some defaults. You can rely on these and other Argus ONE default settings until you're ready to change them.

## To resume work on an existing project

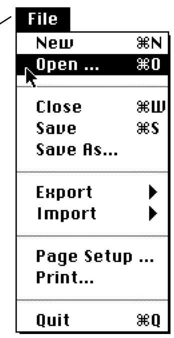
If you're resuming work on a project, you need to open the project to see it on your screen.

1. From the File menu, choose Open.

*Use this same technique to choose any other command from a menu.*




*Point to the File menu and press the mouse button...*




*...then drag the selection bar to Open and release the mouse button.*

2. Locate the file you want to open in the Open dialog box and open it by clicking the Open button.

On the Macintosh, the Open dialog box will show only valid Argus ONE files.

## To start a new project

Use the New command from the File menu to open a new untitled project window.

---

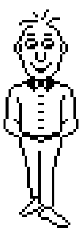
**Important note for Macintosh users:** If you close the last open window of the application, the application quits.

---

# Creating a Project

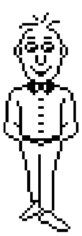
A typical project encompasses the range of tasks you would perform in order to describe in detail the problem at hand. Starting with defining the domain's outline from descriptive information, such as digitized maps in DXF format of boundary conditions, sources and sinks, adding initial conditions, material properties, and ending up with a mesh or grid of that domain.

Argus ONE workplace is based on CAD-like layers (transparency sheets or mylars) in which you create and maintain the different types of information you need using the same coordinate system origin, direction and scale. There are three main types of layers:



*Nature was not created on a grid or a mesh.*

- **Mesh and Grid** type layers, where the numerical type of information will be created, presented and edited.
- **Information and Data** type layers, where you describe the domain’s outline (shape), point sources/sinks or point loads, materials spatial distribution, and initial and boundary conditions. Argus ONE can automatically assign this information to the nodes, elements and blocks in the mesh and grid layers. Data type layers are described in the User’s Guide Supplement.
- **Maps type** layers, into which you can import digitized maps in DXF file format, Argus ONE generic contour format, and in which you create visualization objects (described in the User’s Guide Supplement) to investigate your model results and information distribution.



*Meshes and Grids are numerical entities, the data assigned to them comes from the real world.*

There are two different sub-types of information layers:

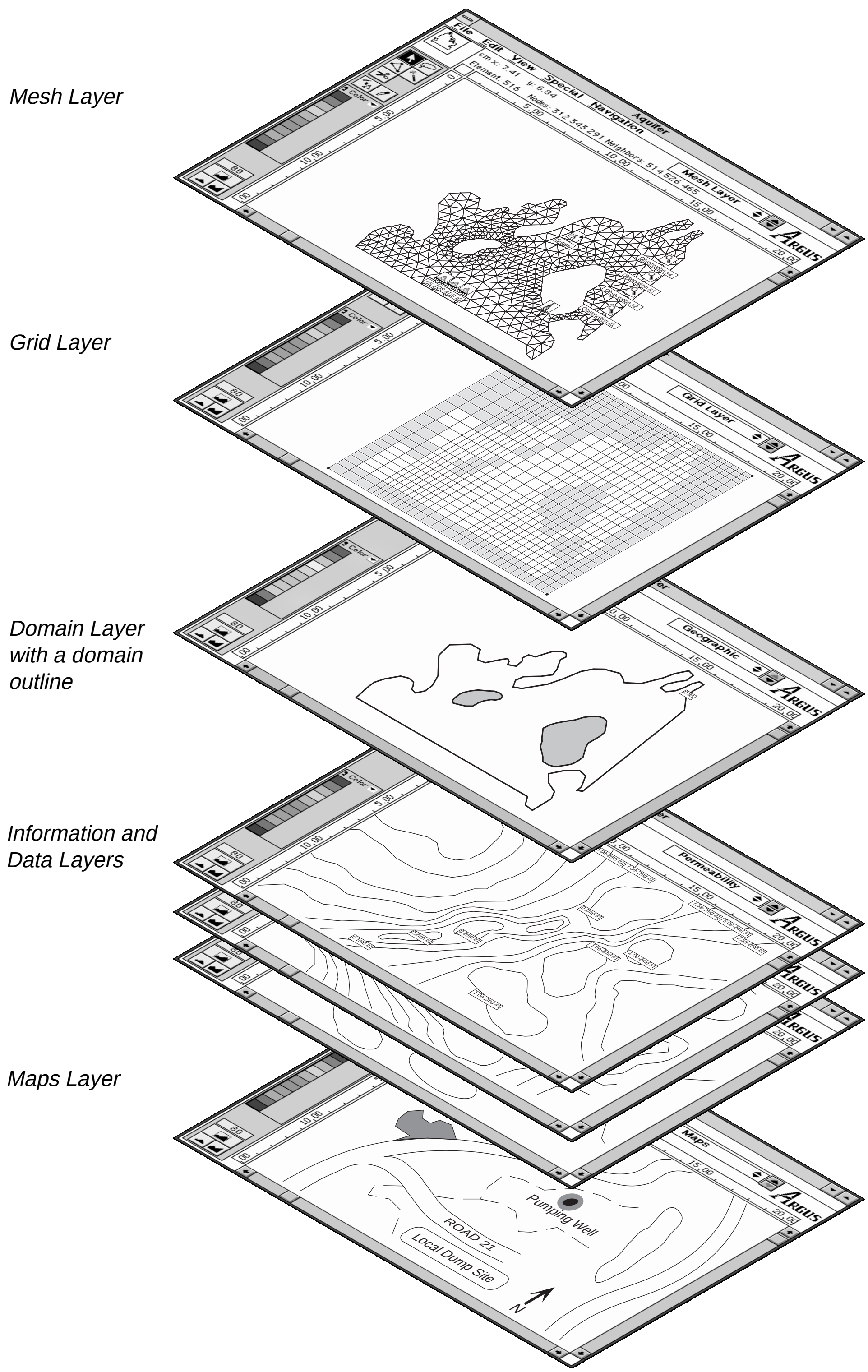
- **Domain type layers** Where you define the domain’s outline, related to as a “Domain Outline”, and its initial mesh and grid density.
- **General Information type layers** Where you store quantitative spatial information such as boundary conditions, physical parameters, initial conditions, topographies and also mesh and grid densities.

As you will learn later, you can manipulate information from different layers, assign layers’ values from other layers and assign and interpolate information from information and data layers to mesh and grid layers.

Layers of all types, except Maps type layers, can hold as many parameters as you need. Using these parameters allows you to describe additional information related to the layers’ objects. For instance, if you use a layer to describe the location and value of point sources or point loads, you can specify additional information such as the source value at different times, or a logical flag designating whether this source is continuous or a time-dependent step function.

***You can re-mesh and re-grid as many times as you need to create your preferred mesh or grid, and Argus ONE will automatically assign the correct data from the information and data layers to your elements, nodes and blocks.***

***The data you bring into the different information and data layers must be aligned to the same coordinates system. The origin, the scale and the coordinate system directions must be identical.***



You can create as many layers as you need from each of the different layer types.

When you create a new project document, Argus ONE creates for your convenience six layers, one of each of the different types. As you will see later, you may rename or even completely delete these pre-defined layers.

For more details about layers, refer to “The Argus ONE Workplace.”

## To Create a Finite Element Mesh of a Domain

To create a mesh, you must first define the domain you want to mesh. If you do not intend to use finite element meshes, or want to read now about finite difference grids, skip to the section “To Create a Finite Difference Grid of a Domain” on page 18.

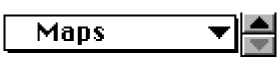
### Defining the domain

Defining the domain is performed in a Domain type layer by creating an outline of the domain including holes (islands), using the *Polygon tool*. A contour created in a Domain type layer is referred to as the *Domain Outline Contour*. A Domain type layer can also contain points for designating point sources and sinks or point loads, as well as open contours designating other boundaries such as rivers, faults, fractures, etc. These are explained in detail in the following parts of the manual.

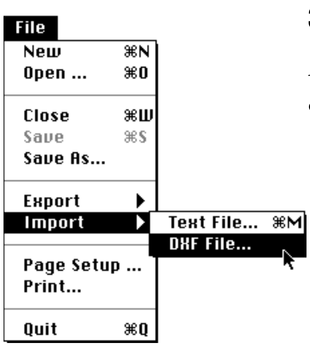
Associating a Domain type layer to a mesh layer tells Argus ONE what is the domain you want to mesh, and what will be the average element size in that domain.

Using the Polygon tool, you can define the outline of the domain by digitizing it directly from a background map of the area of interest.

### To import a background map stored as a DXF file



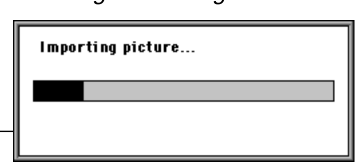

1. Click the bottom arrow in the *Layers Selection Button*  until the *Layers Popup Menu* reads Maps, the Maps layer is active.
2. From the File menu, choose Import DXF File...
3. In the Open dialog box, open the DXF file.



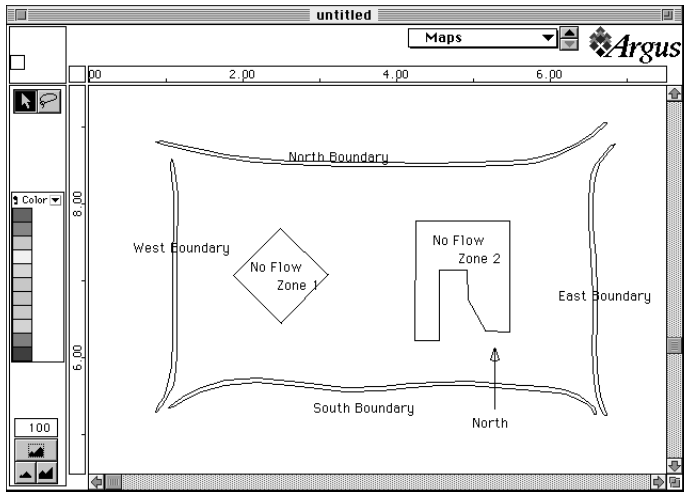

As Argus ONE imports the file, a ***progress*** dialog box appears to indicate that a time consuming operation is in progress.

*Progress Dialog box*

*Press the keyboard sequence to halt import. The keyboard combination on Windows and Unix platforms is* CTRL*+C.*



To learn more about the Maps type layers, about DXF files and DXF graphics objects manipulation refer to chapter 1, “The Argus ONE Workplace”.




*An Argus ONE window with a DXF file imported into a Maps type Layer.*

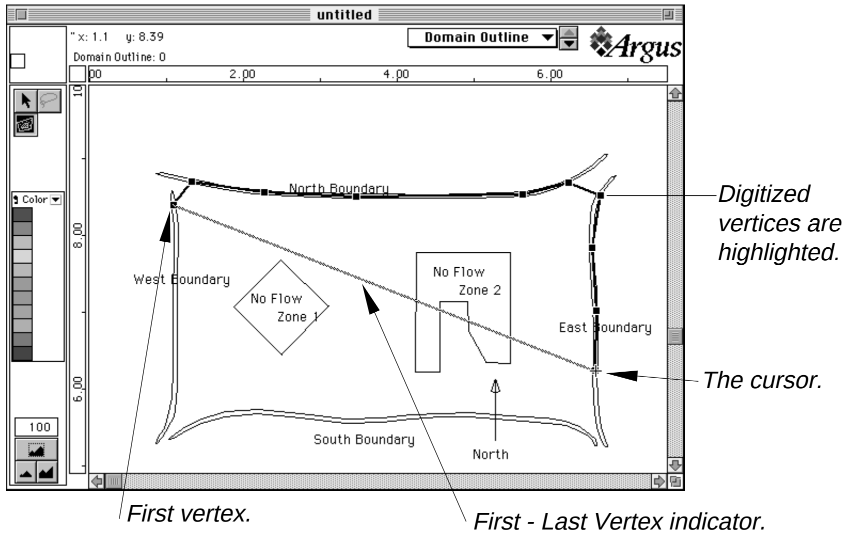
## To create a domain outline

1. Using the *Layer Selection Menu* or *Button*, activate the Domain Outline layer by bringing it to be the front most.
2. Click the *Polygon tool* in the tool palette.
3. Your cursor will change from an arrow to a cross +.
4. Click the cursor where you want to begin creating the outline contour (polygon).
5. Move the cursor to the point where you would like to have the second vertex of the polygon and click. Repeat this step until you're ready to close the outline contour.

   If you have mistakenly clicked a vertex you do not want to have, just hit the delete key on the keyboard and the last recorded vertex will be erased. You can DELETE back as many vertices as you wish, until you finally delete the whole polygon.
6. When you're ready to finish, double click the last vertex and close the outline contour, Argus ONE joins the last and first vertices.

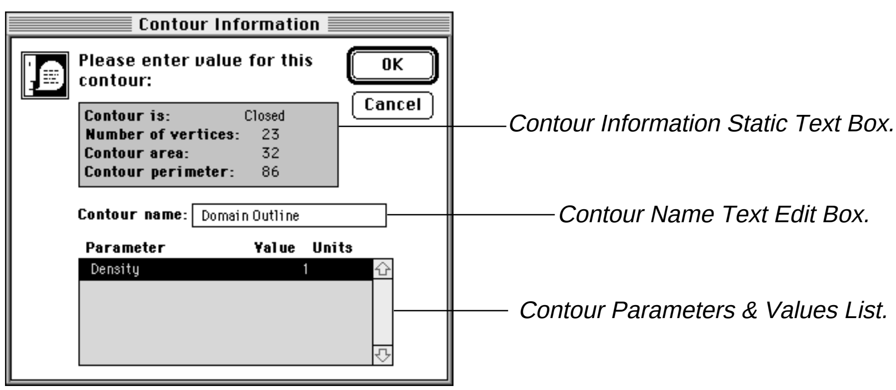
*A project window with a domain outline contour in creation.*

*It is irrelevant whether you record the contour clockwise or counter clockwise, Argus ONE takes care of such trivial chores for you. It also checks the validity of your contour as you record it. If you cross the contour itself, it will beep and will not record the illegal vertex.*




7. Once you have double clicked the last vertex, a dialog box appears asking you for the required average density of the mesh within the newly created domain outline contour. The density of the mesh is defined by Argus ONE as the average element size you would like to have.

*Contour Information dialog box, the first parameter is the Default Density value.*




To make a reasonable choice, look at the rulers and determine over how many units of ruler you want the average element to span.

8. Type this number in the text edit box and click the OK button.

If you wish to change your choice later on, you can do so by double-clicking the contour, which brings up this dialog again. You will see later how you can define different mesh densities for different zones of the mesh.

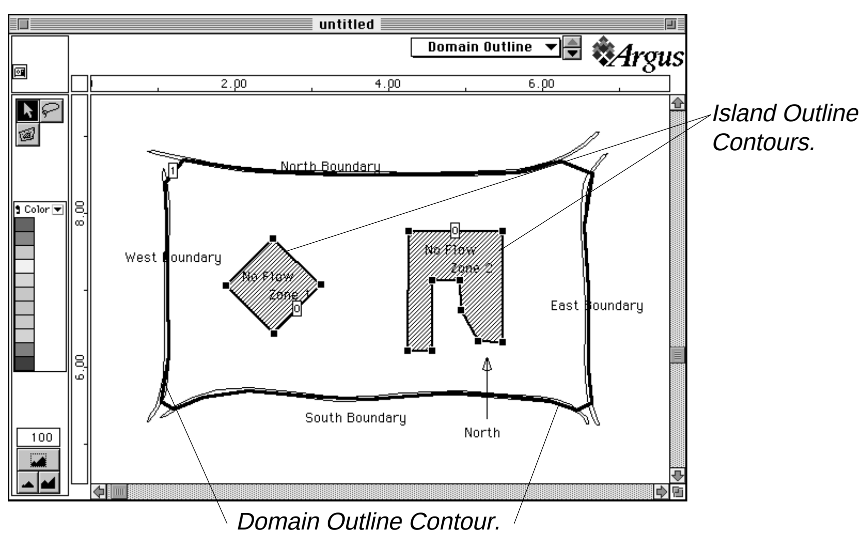
You can also name the contour. The name you give it will be drawn on it and will enable you to locate the contour by its name.

To create a hole, an opening or an island in the domain, you could continue using the Polygon tool, and using the same technique you've used for creating the outline contour, create an island inside the outline contour. However, to make good use of valuable information you brought into Argus ONE, the “No Flow Zone 1” and “No Flow Zone 2” close contours, Argus ONE enables you to copy these contours from the Maps layer and paste them in the Domain layer. Argus ONE will automatically transform them into islands.

### To create the Domain’s island contours using copy and paste

1. Using the *Layer Selection Menu* or *Button* activate the Maps layer by bringing it to be the front most.
2. Click the first contour you wish to copy, the contour vertices are highlighted to inform you it is selected.
3. SHIFT+Click the second contour you wish to copy to add it to the current selection.
4. From the Edit menu, choose Copy.
5. Using the *Layer Selection Menu* or *Button* activate the Domain layer by bringing it to be the front most.
6. From the Edit menu, choose Paste.

Argus ONE creates the two contours and designates them as islands (holes) by hatching them.




*A project window showing a Domain Outline contour with islands (holes).*

## Meshing the domain

To mesh (discretisize) the domain, make sure you're in a “mesh layer.” The layer selection popup menu should read: “Mesh Layer.”

*Tip: You can now hide the Maps Layer so that it will not disturb your work. From the View Menu, choose the Show/Hide menu item and hide the Maps Layer.*

If it doesn't, click on the appropriate arrow in the layer selection button until you enter that layer.

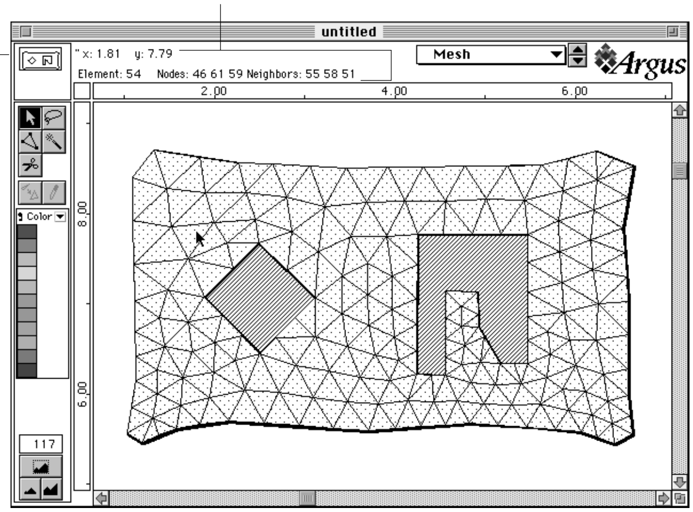
1. From the tool palette, select the *Magic Wand tool*.
2. Click the Magic Wand anywhere within the Domain Outline contour. The Magic Wand becomes active only when it is above a domain outline contour.

Sit back and watch the progress bar dialog box. When the black area completely fills the progress bar, the mesh appears.

*A project window after auto mesh generation.*

*The Information Ruler presents the current units, the Mouse Location and Information about the Element the mouse cursor lies above.*

*The Navigation Window facilitates orientation. It presents the Domain Outline objects of the Domain layer linked to the current Mesh Layer.*




## Using Information Contours to refine the mesh

The mesh you create has a tremendous impact on the solution of the numerical model, both with respect to its accuracy and to the computing resources it demands. Argus ONE Auto Mesh Generation is affected by many factors such as the segments' sizes of the domain outline contour, the distance of a contour to other contours or to itself and many other of which you have full control as you'll find out in the chapter "Finite Element Meshes."

However, you can also instruct the meshing module to refine the mesh at any point or area of need. You tell Argus ONE of your refinement needs by defining the areas to be refined using information contours in an information layer and assigning that layer as the Density layer for the specific mesh layer.

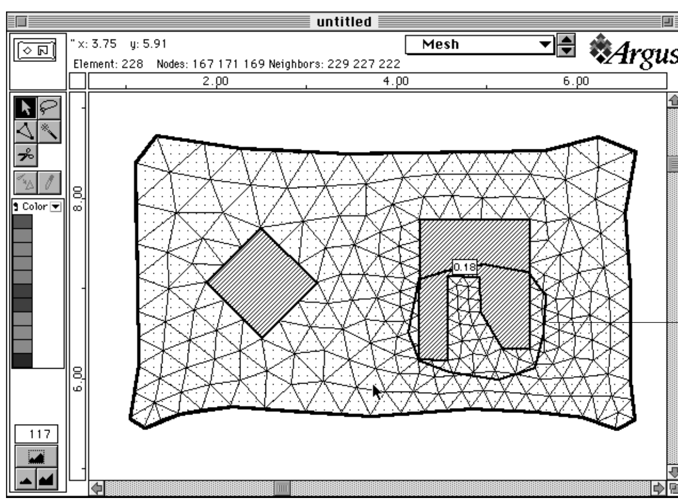
*If you decide to re-mesh your domain, the density information you store in a density layer is retained and will affect your new re-meshed mesh. In contrast, if you re-mesh a mesh that you have refined manually, this refinement will be lost.*

The mesh layer that is created for you when creating a new project document is already assigned a mesh density layer named “Density.” You can rename the layer or assign the mesh layer a different mesh density layer, as you will learn later in this manual.

### To define a Mesh Density Contour

1. Click the appropriate arrow in the Layer Selection button to move to the Density Layer.
2. Using the Polygon tool, create a contour around the area where you want to refine the mesh.
3. When you finish creating the contour, set its value to the size of elements you want to have in the area it encloses.
4. Click the appropriate arrow in the Layer Selection button to move back to the Mesh layer.
5. Click the Magic Wand tool anywhere within the domain. If you have already meshed the domain, Argus ONE asks you if you want to erase the existing elements.
6. Allow Argus ONE to erase the mesh and re-mesh it using the new density information.

To see the density contour through the elements, when the mesh layer is the active one, turn the elements transparent. From the View menu choose Opaque Elements.




*The mesh density value assigned to this mesh density contour is 0.18 units (inch).*

The final mesh density is a combination of the mesh density applied to the domain outline contour, the mesh density contours, and the size of the line segments constructing the domain outline contour. Other factors influencing the final mesh density such as sources and sinks, are explained in the chapter “Meshing A Domain.”

## Making Changes to the Mesh

*Manual changes to the mesh may be useful at times, however, remember that they will not be retained the next time you re-mesh the domain.*

The mesh you have just created is made up of objects. Each node and element is a dynamic object. You can select an object or a group of objects and edit them. There are many ways in which you can change the mesh. You can reshape the domain outline contour and re-mesh, change the density in specific zones and re-mesh, refine, smooth and edit the mesh manually.

In the following paragraphs we will outline some of these capabilities. To learn more about meshes refer to Part 3 “FInite Element Meshes.”

---

**Important Note**: All your actions are undoable. To undo an operation, choose the Undo command from the Edit menu.

---

### Selecting nodes and elements

To edit nodes and elements you must first select them. Selecting them tells Argus ONE that your actions are to affect the selected objects.

1. Choose the Arrow cursor from the tool palette.
2. Move the cursor above the mesh and watch as it changes from a solid arrow to a hollow one each time it is located over a node.
3. Click the mouse button when the cursor is over a node (it turns to a hollow cursor).

The node highlights itself to indicate it is selected.

Selecting an internal node changes its appearance to a hollow bullet, while selecting a boundary node, changes its appearance to a solid black bullet.

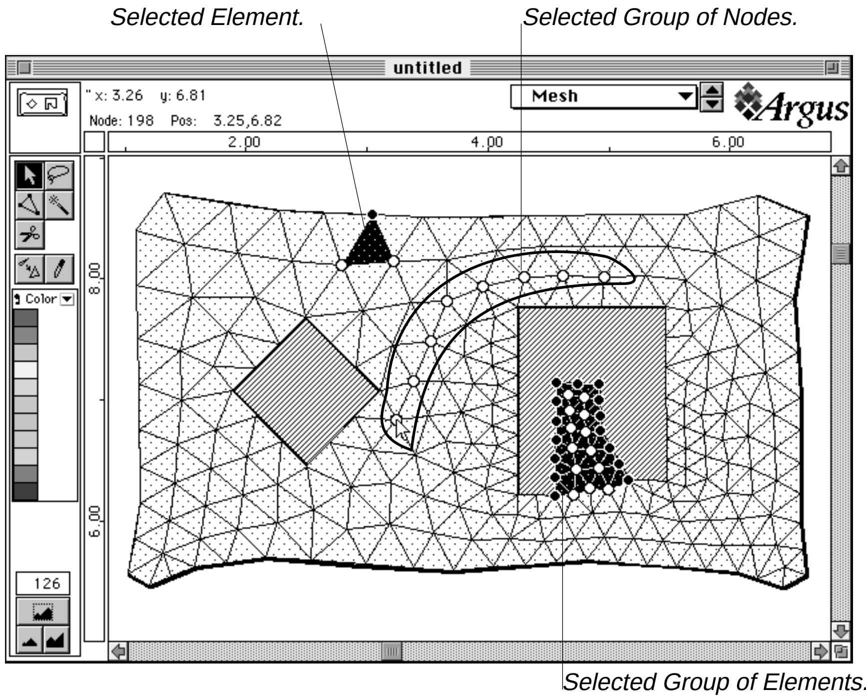
4. To select an element, move the cursor over an element and click the mouse button. The element highlights itself and its three nodes, to show that it is selected.

***Important note:***
*To start the Stretch-Band within a mesh, hold down the SHIFT key while clicking the mouse button.*

### Selecting groups of elements and nodes

- To select a group of elements and nodes, press the mouse button and move the mouse to create a stretch band. When you release the mouse button, all the elements and nodes enclosed by the stretch band are selected.
- You can also select any combination of nodes and elements by using the Lasso tool . Pick it from the tool palette, click and hold the mouse button as you move the cursor. After you release the mouse button, all elements and nodes enclosed by the lasso region are selected.
- To add or remove nodes and elements from the selection, simply hold down the SHIFT key while clicking the mouse on these objects.

*The selected, or highlighted area in a project is called the selection.*




## Editing nodes and elements

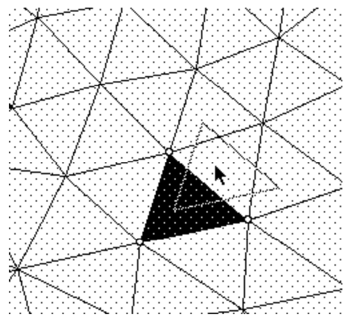
You can delete, reshape and detach elements. However, you can't delete nodes as such. A node is an integral part of an element. Hence, it is deleted only when all the elements it belongs to, are deleted.

### To delete and move elements

| User action and Argus ONE's reaction | Argus ONE's automatic validity testing |
|---|---|

- To delete an element or a group of elements, select it and hit the DELETE key on your keyboard.

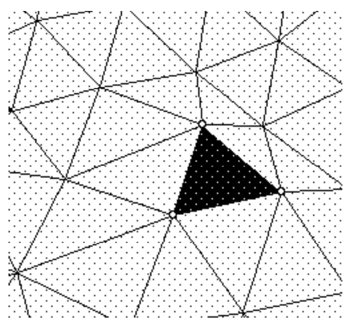
- To move an element or a group of elements, select it and move the mouse while pressing down the mouse button.



*If you release the elements in a position where they overlap other elements, Argus ONE presents a warning alert, and relocates the illegally moved elements to their original location.*

An outline of the elements you're moving follows your mouse movements.

- Release the mouse button and the elements redraw themselves in their new location.

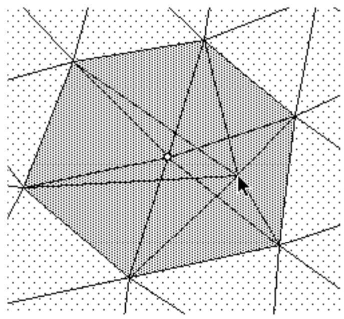


*If while moving the elements you stretch some elements so that their minimal angle becomes smaller than specified in the Preferences dialog, Argus ONE will present a warning alert, and will relocate the elements to their original position.*

Connected elements are distorted to adjust for the new location of the moved element.

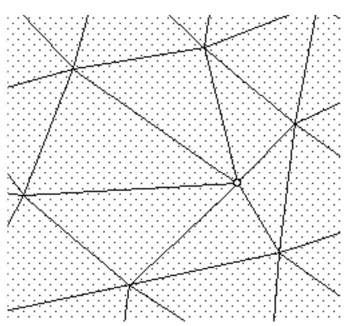
## Moving a node

| **User action and Argus ONE's reaction** | **Argus ONE's automatic validity testing** |
|---|---|
| • To move a node simply press the mouse button above the node and move the mouse while pressing down the mouse button. | |



An outline of the new shape of the connected elements follows your mouse movement, while also showing you the original node position.

| | |
|---|---|
| • All elements connected to the relocated node are reshaped to accommodate for the new node location. | *If you release the node in a position where it causes element overlapping, Argus ONE beeps, and relocates the illegally moved node to its original location.* |
| • Release the mouse button. The node relocates to its new position. | |



*If while moving a node you stretch some elements so that their minimal angle becomes smaller than specified in the Preferences dialog box, Argus ONE will present a warning alert, and will relocate the node to its original position.*

## To Create a Finite Difference Grid of a Domain

Argus ONE enables you to create “grid centered” and “block centered” finite difference grids. When you launch Argus ONE the default grid type is “block centered”, however, you can easily change this default. For additional information regarding finite difference grids, refer to Part 4 “Finite Difference Grids.”

To create a grid you do not have to first define a Domain Outline contour. However, if you do define a Domain Outline contour, Argus ONE will automatically generate a grid of the domain and deactivate blocks that are out of the domain.

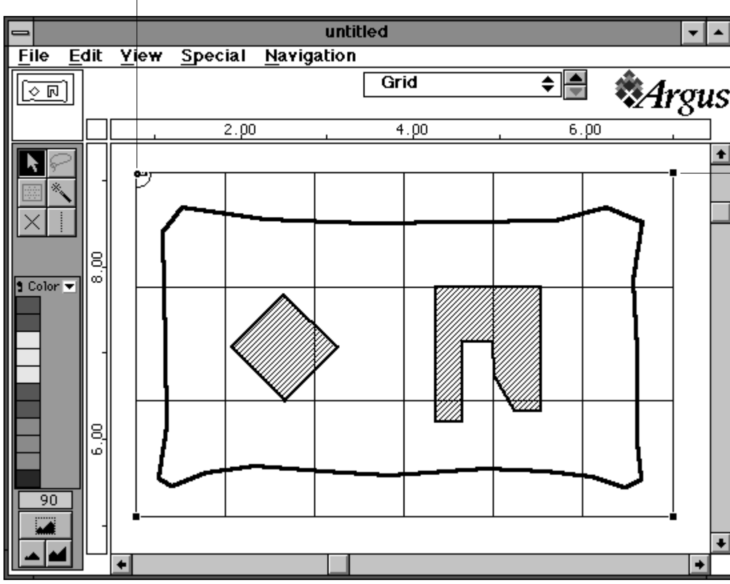
### To create a grid without defining a domain

To create a finite difference grid, make sure you're in a Grid type layer. The layer selection popup menu should read: “Grid”. If it doesn't, click the appropriate arrow in the layer selection button until you enter the Grid layer.

1. Select the *Grid Creation tool* from the tool palette.
2. Click and drag the Grid Creation tool from where you want to start creating the grid.
3. Release the mouse button where you want to end the grid creation.

The grid is automatically divided into evenly spaced rows and columns of one ruler unit. In the following paragraphs you will learn how to edit and refine the grid.

*Click the handle to unlock the grid and edit it.*




*The bullets mark a selected grid. Use the bullets to resize and move the grid.*

*A project window after manual grid generation.*

### To create a grid of a domain defined in a Domain Outline layer

If you have already read how to create a Domain Outline contour in the previous paragraphs, you can skip this section.

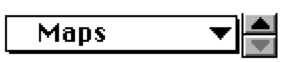
### Defining the domain

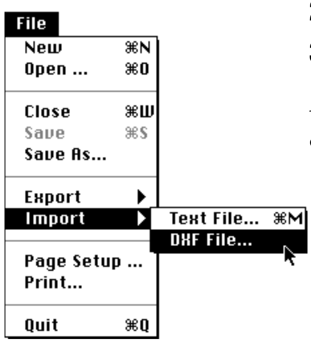
Defining the domain is performed in a Domain type layer by creating an outline of the domain including holes (islands), using the *Polygon too*l. A contour created in a Domain type layer is referred to as the *Domain Outline Contour.* A Domain type layer can also contain points for designating point sources and sinks or point loads, as well as open contours designating other boundaries such as rivers, faults, fractures, etc. These are explained in detail in the following parts of the manual.

Associating a Domain type layer to a Grid layer tells Argus ONE what is the domain you want to grid, and what will be the average block size in that domain.

Using the Polygon tool, you can define the outline of the domain by digitizing it directly from a background map of the area of interest.

### To import a background map stored as a DXF file



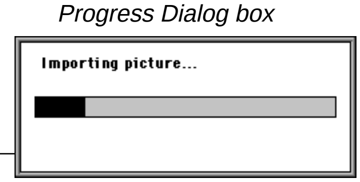



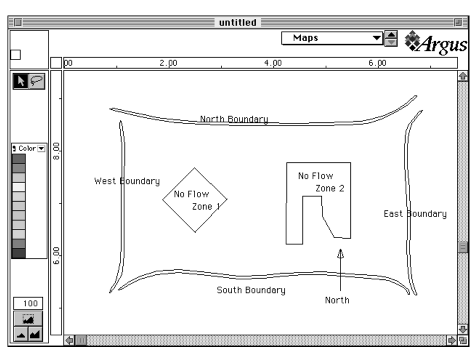

1. Click the bottom arrow in the *Layers Selection Button* until the *Layers Popup Menu* reads Maps, the Maps layer is active.
2. From the File menu, choose Import DXF File...
3. In the Open dialog box, open the DXF file.

As Argus ONE imports the file, a ***Progress*** dialog box appears to indicate that a time consuming operation is in progress.

Progress Dialog box

*Press the keyboard sequence to halt import. The keyboard combination on Windows and Unix platforms is* CTRL*+C.*




*An Argus ONE window with a DXF file imported into a Maps Layer.*

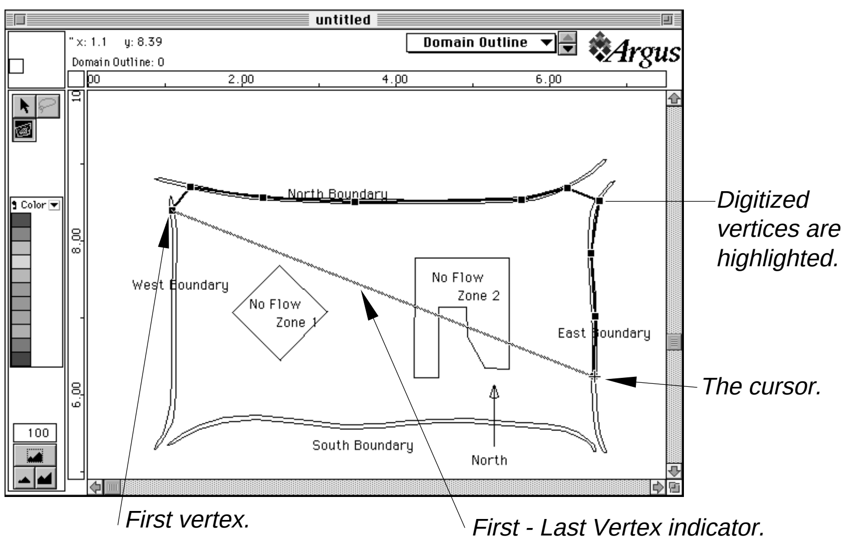
## To create a domain outline

1. Using the *Layer Selection Menu* or *Button*, activate the Domain Outline layer by bringing it to be the front most.
2. Click the *Polygon tool* in the tool palette.
3. Your cursor will change from an arrow to a cross +.
4. Click the cursor where you want to begin creating the outline contour (polygon).
5. Move the cursor to the point where you would like to have the second vertex of the polygon and click. Repeat this step until you're ready to close the outline contour.
6. When you're ready to finish, double click the last vertex and close the outline contour, Argus ONE joins the last and first vertices.

If you have mistakenly clicked a vertex you do not want to have, just hit the DELETE key on the keyboard and the last recorded vertex will be erased. You can delete back as many vertices as you wish, until you finally delete the whole polygon.

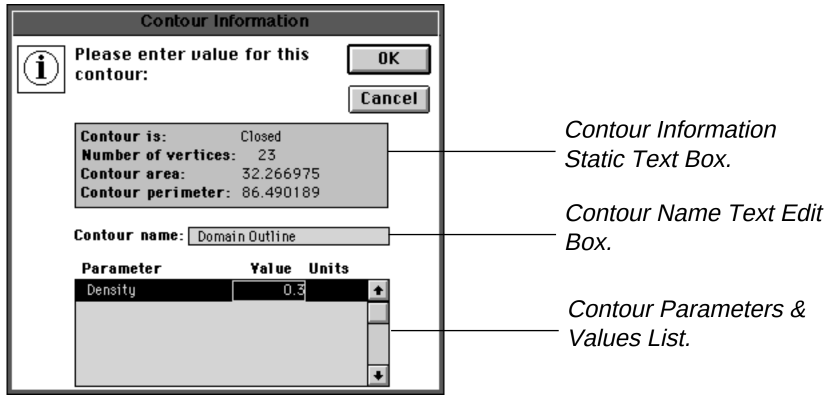
*A project window with a domain outline contour in creation.*

*It is irrelevant whether you record the contour clockwise or counter clockwise, Argus ONE takes care of such trivial chores for you. It also checks the validity of your contour as you record it. If you cross the contour itself, it will beep and will not record the illegal vertex.*




7. Once you have double clicked the last vertex, a dialog box appears asking you for the required average density of the grid within the newly created domain outline contour. The density of the grid is defined by Argus ONE as the default block size you would like to have.

Contour Information dialog box, the first parameter is the Default Density value.




To make a reasonable choice, look at the rulers and determine over how many units of ruler you want the average block to span.

**8.** Type this number in the text edit box and click the OK button.

If you wish to change your choice later on, you can do so by double-clicking the contour, which brings up this dialog again. You will see later how you can define different grid densities for different zones of the grid.

You can also name the contour, the name you give it will be drawn on it and will enable you to locate the contour by its name.

To create a hole, an opening or an island in the domain, you could continue using the Polygon tool, and using the same technique you've used for creating the outline contour, create an island inside the outline contour. However, to make good use of valuable information you brought into Argus ONE, the “No Flow Zone 1” and “No Flow Zone 2” close contours, Argus ONE enables you to copy these contours from the Maps layer and paste them in the Domain layer. Argus ONE will automatically transform them into islands.

### To create the Domain’s island contours using copy and paste

**1.** Using the *Layer Selection Menu* or *Button* activate the Maps layer by bringing it to be the front most.

**2.** Click the first contour you wish to copy, the contour vertices are highlighted to inform you it is selected.

**3.** SHIFT+Click the second contour you wish to copy to add it to the current selection.

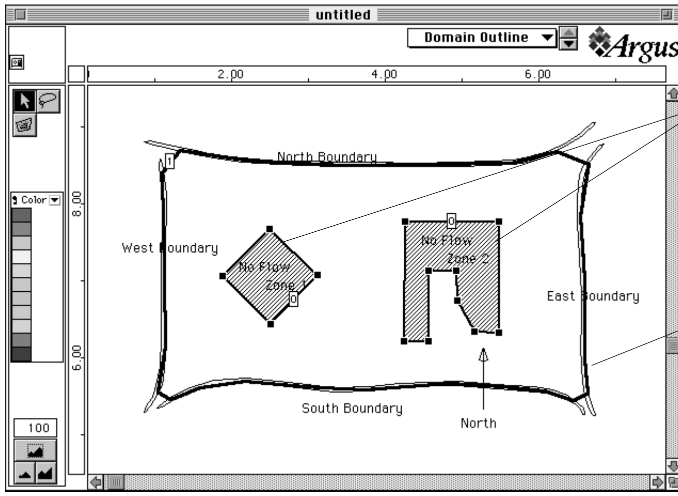
**4.** From the Edit menu, choose Copy.

**5.** Using the *Layer Selection Menu* or *Button* activate the Domain layer by bringing it to be the front most.

**6.** From the Edit menu, choose Paste.

Argus ONE creates the two contours and designates them as islands (holes) by hatching them.

*A project window showing a Domain Outline contour with islands (holes).*




### Automatically griding the domain

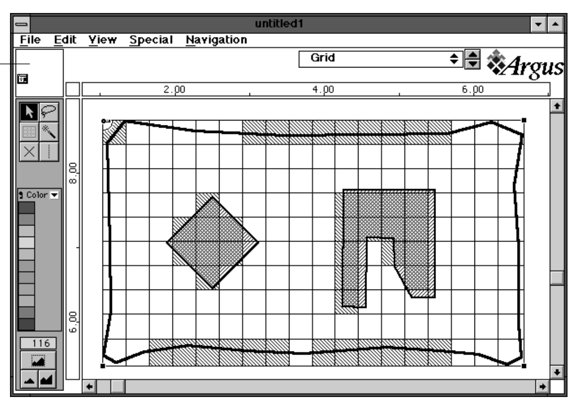
To automatically gird (create a finite difference grid) the domain, make sure you're in a Grid layer. The layer selection popup menu should read: "Grid ", unless you have changed the default layer name. If it doesn't, click on the upper arrow in the layer selection button until you enter that layer.

*Tip: You can now hide the Maps Layer so that it will not disturb your work. From the View Menu, choose the Show/Hide menu item to hide the Maps Layer.*

1. From the tool palette, select the Magic Wand tool .
2. Click the Magic Wand anywhere within the domain outline contour. The Magic Wand becomes active only when it is above a domain outline contour.

Sit back and watch the progress bar dialog box. When the gray area completely fills the progress bar, the grid appears. The automatic grid generation creates a grid of the size of the domain's outline contour enclosing rectangle.

*A project window after auto grid generation.*



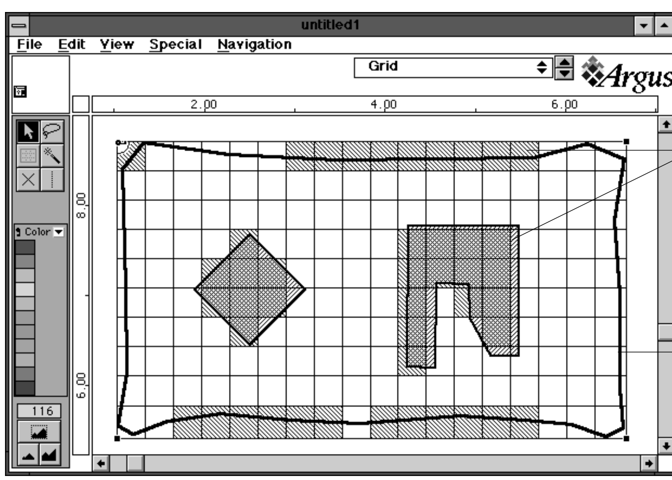
## Using Information Contours to refine the grid

The grid you create has a tremendous impact on the solution of the numerical model, both with respect to its accuracy and to the computing resources it demands. Argus ONE enables you to assign different grid densities to different areas in the domain. You can define the grid density in any information type layer and tell Argus ONE which is the information layer serving as the density layer for a specific grid layer. The grid layer that is created for you when creating a new project document is already assigned a grid density layer named “Density.” You can rename any layer and assign the grid layer a different grid density layer, as you will learn later in this manual.

### To define a Grid Density Contour

1. Click the lower arrow in the Layer Selection button to move to the Density Layer.
2. Using the Polygon tool, create a contour around the area where you want to refine the grid.
3. When you finish creating the contour, set its value to the size of blocks you want to have in the area it encloses.
4. Click the appropriate arrow in the Layer Selection button to move back to the Grid layer.
5. Click the Magic Wand tool anywhere within the domain. If you have already grided the domain, Argus ONE asks you if you want to erase the existing grid.
6. Allow Argus ONE to erase the grid and re-grid it using the new density information.

The final grid density is a combination of the grid density applied to the domain outline contour and the grid density contours defined in a Density layer.




*Blocks’ their center lies outside the domain contour or in islands are deactivated and marked by a hatch.*

*The grid density assigned to this domain outline contour is 0.15 units (inch).*

## Making Changes to the Grid

*Manual changes to the Grid may be useful at times, however, remember that they will not be retained the next time you re-grid the domain.*

The grid you have just created is made up of blocks (cells). Each block is a dynamic object. You can select a block or a group of blocks and edit them. There are many ways in which you can change the grid. You can change the density of the grid in specific zones, add rows and columns, delete rows and columns, and remove blocks from the domain.

In the following paragraphs we will outline some of these capabilities. To learn more about grids refer to Part 4 “Finite Difference Grids.”

To edit the grid and its blocks you must first select them. Selecting them tells Argus ONE that your actions are to affect the selected objects.

### Locking and unlocking the grid

To secure your work you can lock the grid. A locked grid does not allow you to move, resize or delete it. Locking the grid also does not enable you to move grid lines.

When you or Argus ONE create a grid, the grid is created in an unlocked mode. To read more about locking and unlocking the grid refer to chapter 9, “Creating and Editing a Grid”

### Selecting blocks

To edit blocks, you must first select them. Selecting them tells Argus ONE that your actions are to affect the selected objects.

1. If you locked the grid by mistake, click the handle at the top left corner of the grid .
   Or -
   from the Edit menu, choose Lock Grid.

The grid is unlocked, and you can work within it.

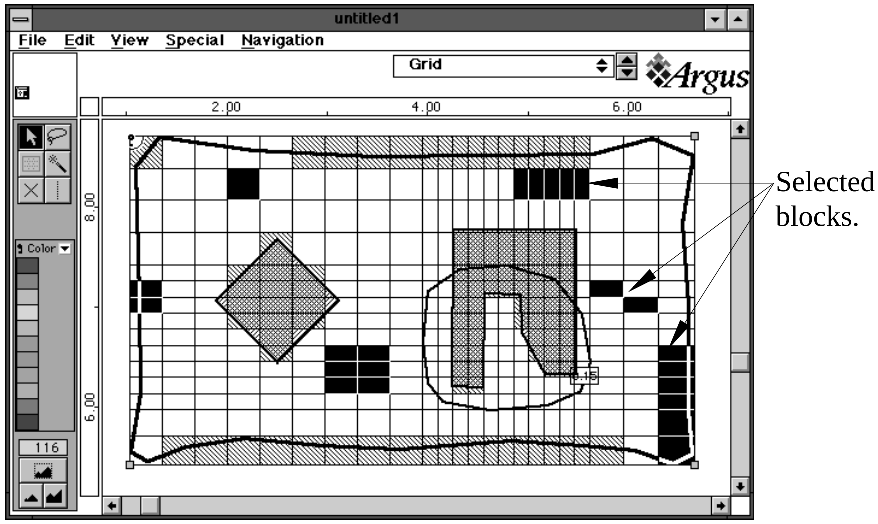
2. To select a block, move the cursor over it and click the mouse button. The block highlights itself to indicate that it is selected.

### Selecting groups of blocks

- To select a group of blocks, press the mouse button and move the mouse to create a stretch band. When you release the mouse button, all the blocks within the stretch band are selected.
- You can also select any combination of blocks by using the Lasso tool . Pick it from the tool palette, click and hold the mouse button as you move the cursor. After you release the mouse button, all blocks enclosed by the lasso region are selected.

- To add or remove blocks from the selection, simply hold down the SHIFT key while clicking the mouse on these objects.

*The selected, or highlighted area in a project is called the selection.*




## Editing the grid

You can change the grid density, that is, the blocks' height and width or the number of rows and columns, at any area you choose. There are three main capabilities to manually edit the grid: moving grid lines, adding grid lines and deleting grid lines. These capabilities are intended for final refinement of the grid. However, remember that it is much more sensible to refine the grid by using domain outline and density contours as they will remain even if you re-grid.

In the following paragraphs we will outline some of these capabilities. To learn more about grids refer to Part 4 "Finite Difference Grids."

## Adding rows and columns

When you first create the grid, Argus ONE divides it into rows and columns of one (1) unit. If one of the sides of the grid is too small, it divides it and creates rows or columns of 1/10, 1/100... of that side.

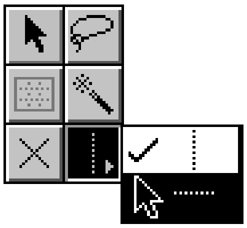
You can add grid lines one at a time or many at once.

## To add one Vertical grid line

1. From the tool palette, select the *Add Column Tool*, the tool highlights itself to designate that it is the active tool and the cursor changes its shape to a vertical line |.
2. Click the tool in the grid between any two grid lines.

A new vertical grid line is added and a new column is created.

### To add one Horizontal grid line



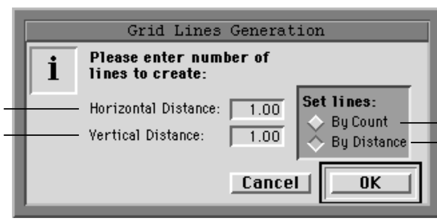
1. Click and hold the *Add Column Tool* to popup the menu and select the *Add Row Tool* ■, the cursor changes its shape to a horizontal line —.
2. Click the tool in the grid between any two grid lines.

A new horizontal grid line is added and a new row is created.

### To add a number of grid lines

1. Click the *Add Row* or the *Add Column Tool*.
2. Press the mouse button and move the cursor to create a stretch band spanning over the area you wish to add grid lines to. When you release the mouse button, the *Grid Lines Generation* dialog box appears.

*Enter the number of horizontal and vertical grid lines to add, or the distance between adjacent grid lines.*

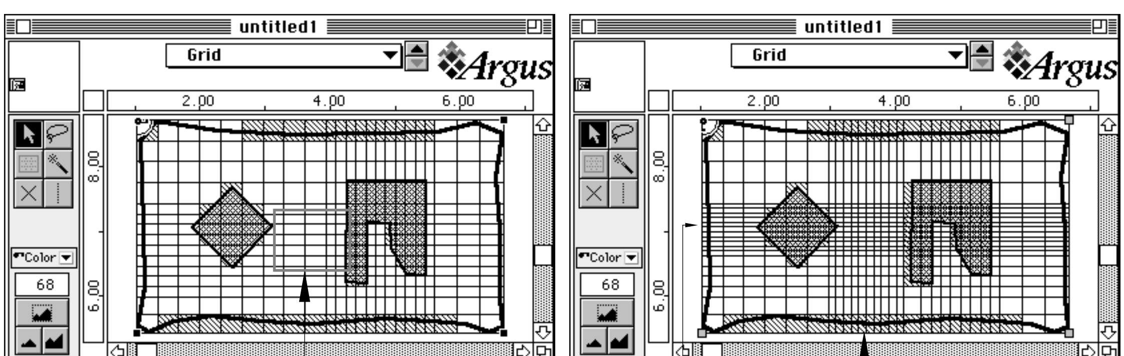



*Check to add desired number of grid lines.*

*Check to add grid lines at a specified distance.*

3. To add a specified number of grid lines, check the *By Count* radio button. Or - to add grid lines at a specified distance from each other, check the *By Distance* radio button.
4. Specify the number of grid lines you wish to add in both directions. Or - Specify the distance between adjacent grid lines.
5. Click the OK button. Grid lines are added as specified.

Specifying zero number of lines in the Horizontal or Vertical text edit boxes, will add grid lines only in the specified direction.




*Specifying add "by count" of ten by ten grid lines in the selected rectangle...*

*...adds them to the grid.*

## Deleting rows and columns

You can delete grid lines one at a time or many at once.

### To delete one grid line

1. From the tool palette, select the *Delete Grid Line Tool*, the tool highlights itself to designate that it is the active tool and the cursor changes its shape to an X shaped cross.
2. Click the tool in the grid on a grid line you wish to delete.

The grid line is deleted. If you click in the intersection of a vertical and a horizontal grid lines, both grid lines are deleted.

### To delete a group of grid lines

1. Click the *Delete Grid Line Tool*.
2. Click-drag the mouse and move the cursor to create a stretch band spanning over the area from which you wish to delete the grid lines. When you release the mouse button, all grid lines in the marked area, horizontal as well as vertical, are deleted.

## Moving a grid line

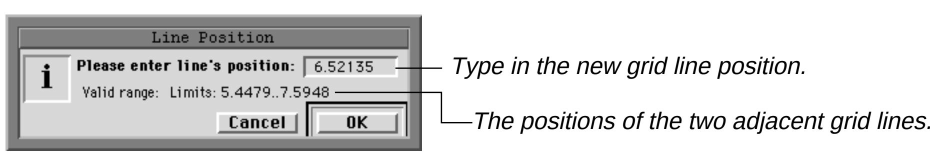
You can move a grid line by hand or re-position it at an exact location.

### To move a grid line

1. From the tool palette, select the *Arrow Tool*. When the cursor lies over a horizontal grid line its shape changes into and when it lies over a vertical grid line its shape changes into .
2. Click-drag the mouse button and move the line to its new desired location.

### To position a grid line at an exact location

1. From the tool palette, select the *Arrow Tool*.
2. Double-Click the grid line, the *Line Position* dialog box appears.




3. To re-position the grid line, type in the desired new location.

Argus ONE informs you of the valid range, that is, the locations of the two adjacent grid lines. If you try to type in a position out of the valid range, Argus ONE beeps and brings up the following alert.


## Deactivating blocks

When you use the Magic wand to grid a domain, the automatic grid generation also automatically marks blocks that their centers lie out of the domain outline contour, or within islands, as inactive blocks and hatches them to indicate they are deactivated. You can also deactivate blocks manually. If you create a grid manually or manually refine it, you can use the Magic Wand to automatically deactivate blocks without re-griding the domain.

When you export your grid, Argus ONE exports a matrix of the blocks in the grid in which deactivated blocks are assigned the integer zero (0) and others the integer one (1).

### To automatically deactivate blocks in a manually refined grid

1. Make sure the active layer is the Grid Layer.
2. From the tool palette, choose the Magic Wand tool  .
3. Click the Magic Wand anywhere within the grid.
   The following dialog box appears:


*Choosing the Re-Grid button erases the existing grid and automatically grids the domain.*

4. To only deactivate blocks, click the Deactivate button.

### To manually deactivate a selected group of blocks from the grid

1. Select the group of blocks.
2. From the Edit menu, choose Deactivate Blocks or use the keyboard shortcut.

### To reactivate an inactive block

1. Select the inactive block or blocks.
2. From the Edit menu, choose Deactivate Blocks or use the keyboard shortcut.

# Saving Your Work

When you're working in Argus ONE, you're actually working on a copy of your project temporarily stored in the memory of your computer. To save your work for future use, you must give the project a name and store it on a disk—either the hard disk of your computer or a “floppy” disk. It's a good idea to save your project approximately every 10 minutes. Save more often if you're doing complex work you don't want to redo. If a power failure or other problem occurs to interrupt your Argus ONE session, any work you haven't saved will be lost.

## To Save a Project

**1.** From the File menu, choose Save.

If you're saving the project for the first time, Argus ONE displays the *Save As* dialog box.

**2.** Type a name for the project in the Save as box.

Use a name that is not assigned to another project in the current folder. The name must comply with your computing platform file system naming conventions. For a detailed explanation refer to your operating system manual.

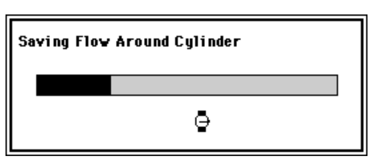
**3.** Make sure the currently open directory or folder is where you want to store the project in.

**4.** Choose the Save button. Your project remains open on your screen so you can continue working.

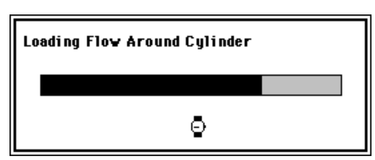
Argus ONE file format is multi-platform compatible. You can open any Argus ONE file regardless in what platform it was created. To transfer a file over the network between different platforms, just make sure the transfer type is set to BINARY.

### Save and Load Progress Bars

Saving and loading large Argus ONE project files may take some time. When loading and saving, Argus ONE presents you with a progress bar indicating the status of the save and load operations and specifies the file name.




*Save progress bar.*




*Load progress bar.*

# Adding Information to the Project

The design of a project usually begins by incorporating different types of information. That information is the basis for the definition of the problem domain and its characteristics. The information source is most often obtained from maps, either scanned or digitized and from nodal information listed in tables or files. The scale and units of the problem domain are also derived from such maps.

Incorporating maps or data from maps into your Argus ONE project, is supported by Maps type layers, Information type layers, and Data type layers described in the Supplement to this User's Guide. You can describe quantitative information in an Information type layer by using three information/location objects:

1. **The Point object**: Using the Point tool to create a Point object, allows you to define a point at some X and Y coordinates and assign information values at that location. The Point object enables you to describe point sources and sinks and point loads, as well as any discrete information.
2. **The Open Contour object**: Using the Open Polygon tool allows you to create an Open Contour object which is a line in the plane. You can assign this contour information values. The Open Contour object enables you to describe rivers and fractures as well as any equal value information along that line.
3. **The Close Contour object**: Using the Close Polygon tool allows you to create a Close Contour object which is a closed line in the plane. You can assign this line information values. The Close Contour object enables you to describe areas or zones having the same value. Using a number of Closed Contours, you can describe a surface in three dimensions such as the topography of a geological formation or the bathymetry of a lake or an ocean, initial and boundary conditions distribution on a plane, a physical parameter or material properties distribution.

An information layer containing a number of these objects is referred to as an *Information Contour Map*.

You may either digitize information objects from DXF files that were imported into a Maps layer, or you can import digitized data in the form of contour maps stored in Argus ONE's generic format.

The values of the contours can be automatically transferred to the nodes, elements or blocks in Mesh and Grid layers. Each node, element or block probes the information layers it is linked to and finds its value in them.

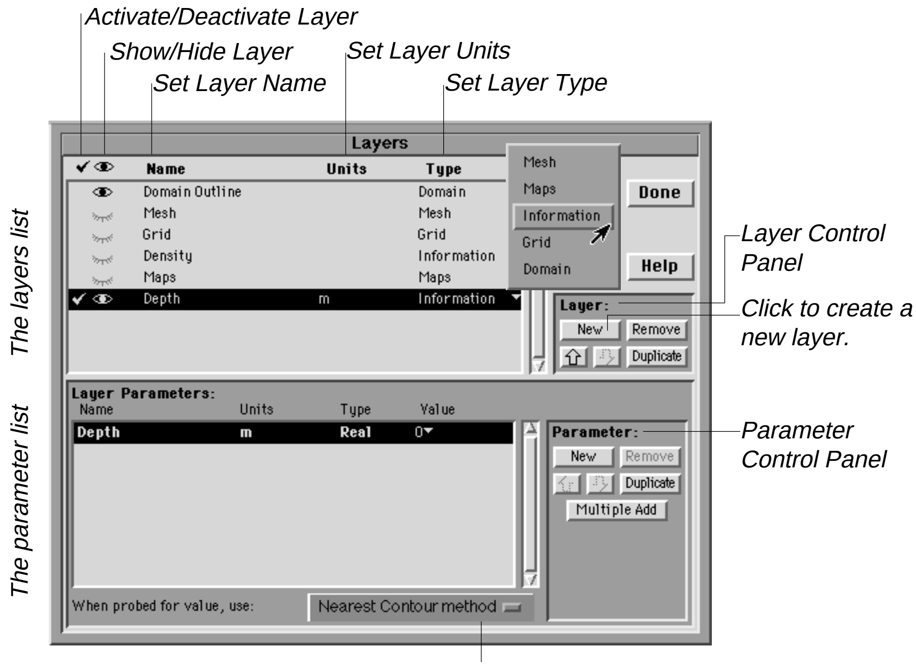
A node, an element or a block receives the value of the first contour that encloses it. You can also instruct Argus ONE to perform an interpolation of the values in an information type layer and to create complex mathematical relations involving information from any number and types of layers.

## Creating an Information Type Layer

**1.** From the *Layers Selection Menu* choose Layers...
Or -
From the View menu, choose Layers...

The *Layers* dialog appears.

*The Layers dialog box is where you define layers, set the active and visible layers, and define the link between the data in a layer and the nodes, elements or blocks in the Mesh and Grid layers.*




*Choose the contour interpretation method from the popup menu.*

The Layers Dialog is where you create and maintain the layers skeleton of your project. You can create, remove and duplicate layers, set their name, units and type, show, hide, activate or deactivate them, and create links between them.

### To create an information type layer

**1.** Click the New button in the Layer Control Panel.

A new layer is created. The default type is set to Information type layer, and the new layer's name is set to “New Layer.” The new layer is active (the front most) and visible. To change its name and units just click the appropriate field in the layer's line and start editing.

**2.** Click the Done button to close the dialog and return to your work.

To read more about the Layers Dialog refer to chapter 1 “The Argus ONE Workplace” and to chapter 2 “Working with Information Layers.”

You can now import digitized information in DXF format or Argus ONE's internal format. These files may contain information describing topography, bathymetry, materials or boundary and initial conditions, etc.

## Describing the Spatial Distribution of a Parameter

Information Contour Maps allow you to describe the spatial distribution of a parameter using the Point, the Open Contour and the Close Contour objects. A parameter can be the topography of a geological formation, the bathymetry of a lake, the spatial distribution of a material, initial and boundary conditions, the rainfall distribution, etc.

### To create contours describing a parameter distribution

For example, to create a contour describing the depth of the sea bottom:

**1.** Make sure you're in the new layer you've just created.

If you are not, click on the appropriate arrow in the layer selection button until you enter that layer.

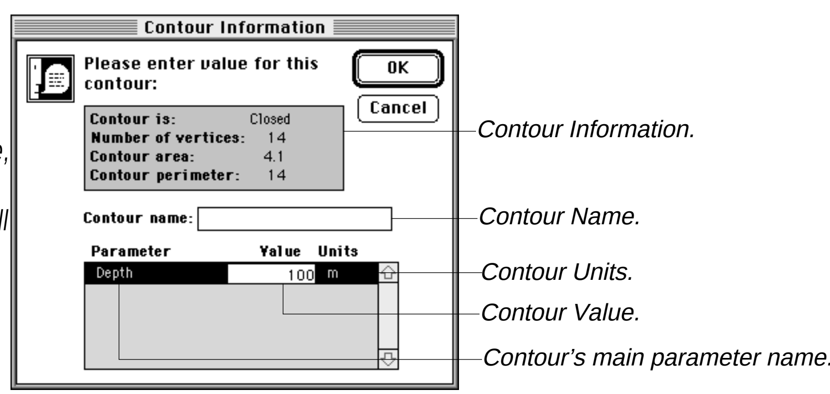
**2.** Click the Polygon tool in the tool palette. It looks a little different from the Polygon tool you've used before while creating a domain outline, and allows you to record contours within contours. You can also create open contours and points using the Open Polygon and the Point tools respectively.

**3.** Click the tool in the window to create the contours. As before, double-clicking closes the contour. Notice that Argus ONE does not let you record contour vertices such that contours intersect.

Once you have double clicked the last vertex of a contour, a dialog box appears, asking you for the value of new contour's main parameter.

**4.** Enter its value.

*You may change a contour's value at any time, by double clicking the contour. This dialog box will open.*



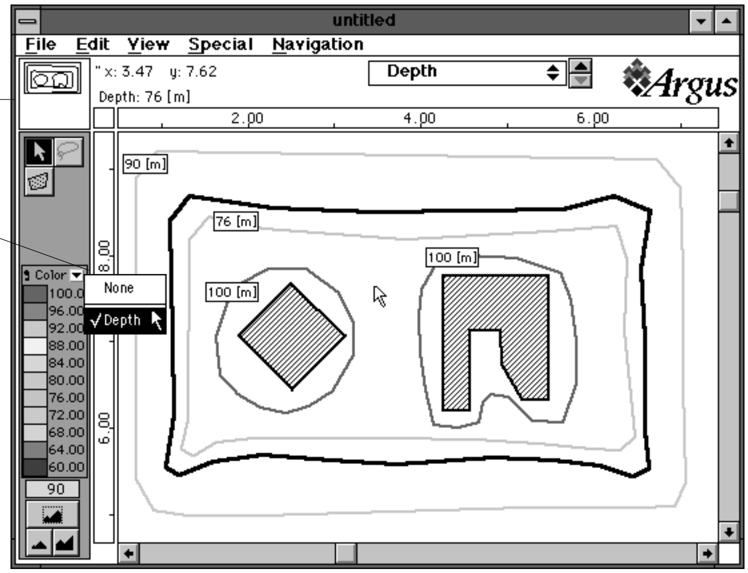
*A project window showing an Information type layer with contours describing depth variation.*

*A small text box indicating the contour's value is attached to each contour.*

*The Navigation window showing a thumbnail of the current layer contours.*

*Use the Color Popup to activate color interpreting of contours' values.*

*From the View menu choose the Calculate Colors command to automatically set the colors range.*




You may add additional Information type layers to describe quantitative or descriptive information.

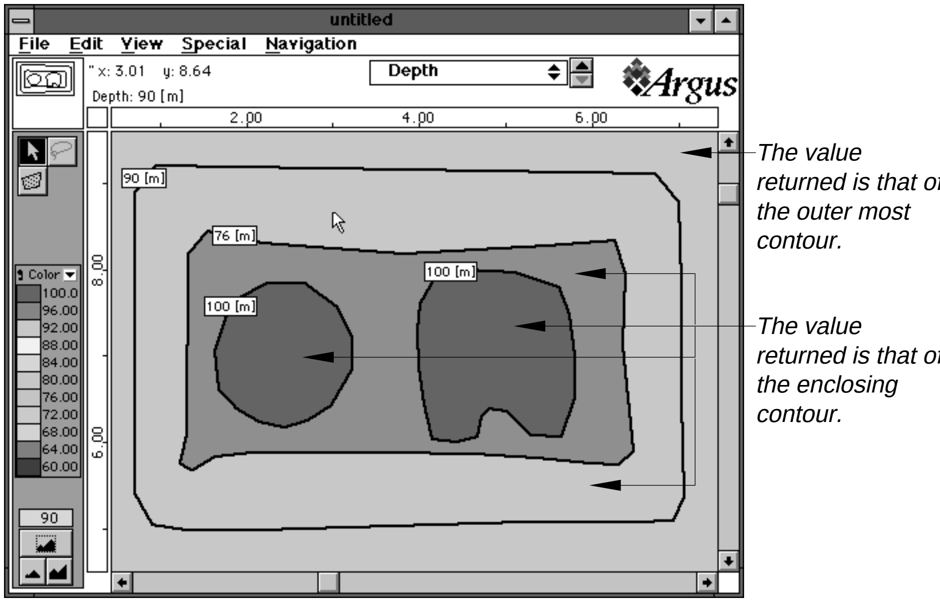
## Probing for Information

Argus ONE layers can be linked. You create a layer parameter and assign it a parameter from another layer. When an object or a parameter expression is evaluated, it probes the linked layer or layers for its value. The probing object, a node, an element, a block or the mouse location, sends its position (X and Y) to the layer being probed, which returns the appropriate value.

## Interpretation of Contour Data

Argus ONE supports a number of interpretation methods of contour maps data. The most basic and the default one, is the "Nearest contour" method. When a query is sent, the probing object's location is tested against all contours in the layer. The value returned is that of the first contour containing the tested object's coordinates. If the object's coordinates are not enclosed by a contour, the value returned is that of the nearest contour, either close, open or a point. If the layer contains no contours, the default value is returned. If there is only one contour in a layer, either close or open, the value returned is of that contour.

To read about the interpolation method and the "Exact contour" method, refer to part 2 chapter 2: "Working with Information Layers."




*A window showing contours data interpretation under the default "Nearest Contour" method.*

## Linking Mesh and Grid Layers to Information Type Layers

Linking layers allows you to assign values from one layer to another. Linking Information type layers to Mesh and Grid layers allows you to assign nodes, elements and blocks values from information layers. When you create such a link, each mesh or grid object probes the information layers it is linked to for its value in them. A node, for instance, is assigned its value at its X and Y position in the linked information layer.

### To link an information layer to a mesh and a grid layer

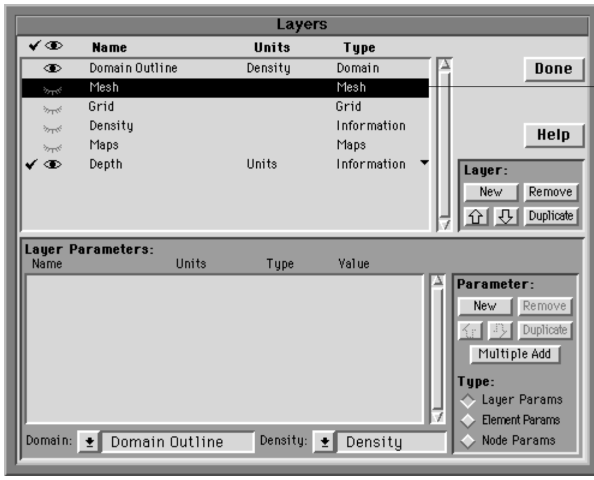
Linking layers is performed in the Layers Dialog. When you link one layer to another, a new layer parameter is created for that link. A parameter can be merely a constant, a linked layer, or a complex expression. To link the mesh and grid layers to the Depth layer you just created:

1. Open the Layers Dialog.
2. Click the Mesh layer line in the Layers list to select it. The line is highlighted to indicate it is selected.
3. In the Parameters control panel click the New button to create a new Mesh parameter.
4. Click and hold the mouse button in the Value field to open the popup menu.
5. Choose "Depth" from the popup menu and release the mouse button.
   The Mesh layer is linked to the Depth layer.

To learn more about information linking refer to chapter 2.

*Layer linking is a very important tool. It keeps your data linked to mesh and grid objects no matter how many times you change them.*

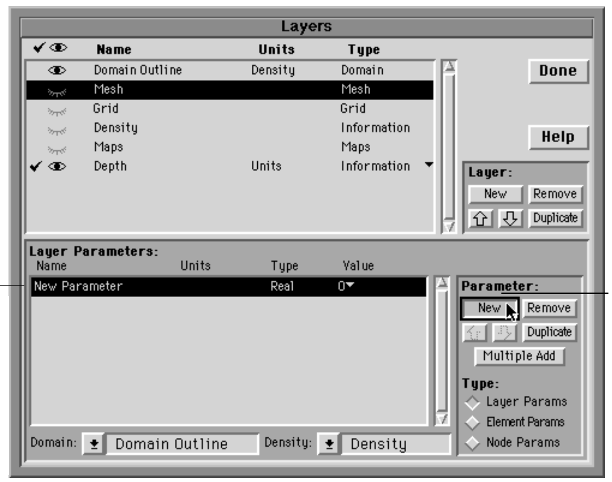
*Layer linking also enables you to create complex relations between different types of information.*




***Steps 1 & 2***

*Bring up the Layers Dialog and click the Mesh layer line to select it. The line is highlighted to indicate it is selected.*

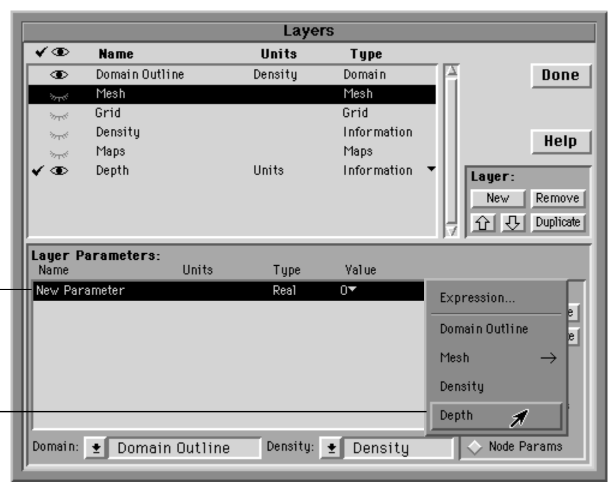
***Step 3***




*A new mesh parameter is created and assigned the name New Parameter.*

*Click the New button to create a new mesh layer parameter.*

***Steps 4 & 5***




*The parameter name is automatically changed to the name of the layer it is linked to.*

*Choosing the Depth layer from the parameter value popup links the new parameter to the Depth layer.*

## Editing and Adding Node, Element or Block Information

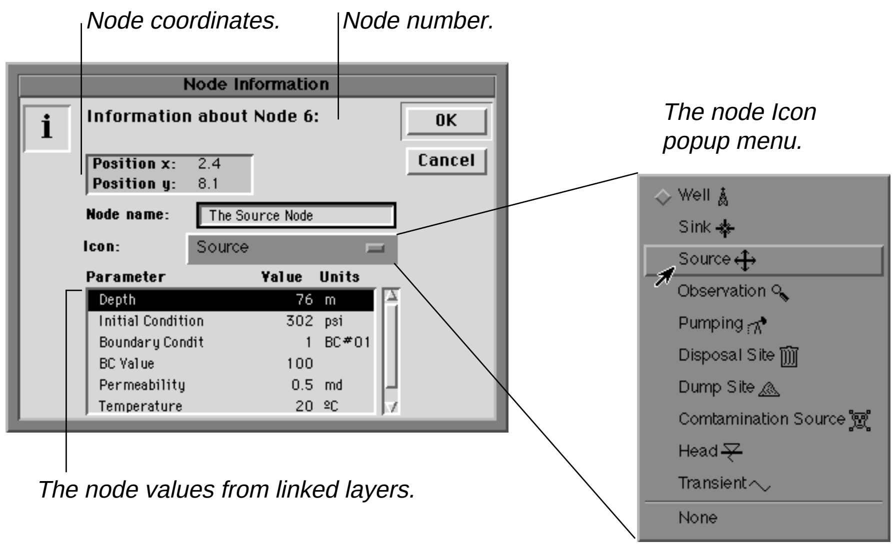
Since every object on the screen is a dynamic object, you may view and change its related information by double clicking on it. This is true for contours, nodes, elements and blocks.

You can view and edit the information linked to nodes and elements by selecting and double-clicking them. If you are using finite difference grid, the same applies to block information.

### To edit node information

- Double clicking a node or a selected group of nodes, brings up the Node Information dialog box.

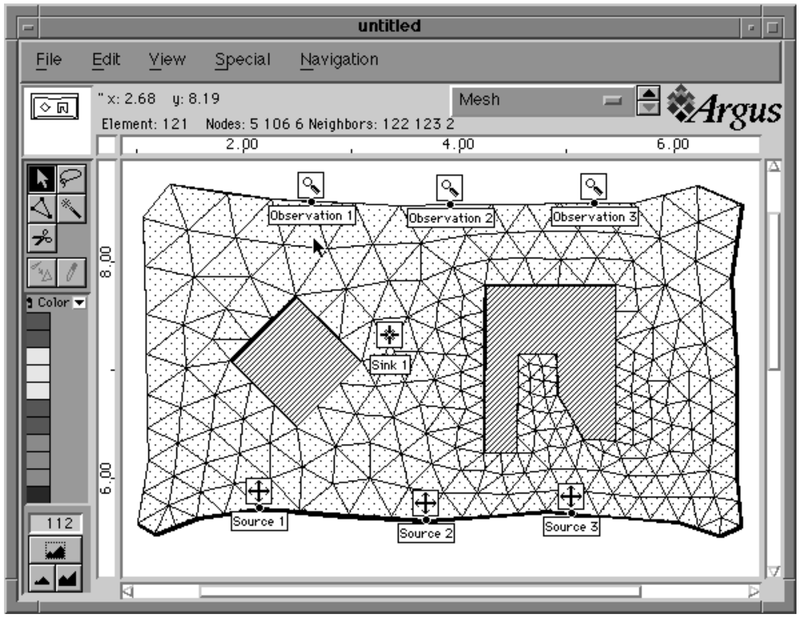
*Node Information dialog box.*




- To override the data automatically assigned to the node from an information layer just click the value field and enter the required data in the text edit box. To re-establish the link just select all the digits in the text edit box and delete them.
- To name the node, type its name in the Node name text edit box.
- To assign the node an icon, choose an icon from the Icon popup menu.

*Nodes Marked with names and icons.*

*If you move a node, its name and icon move with it.*

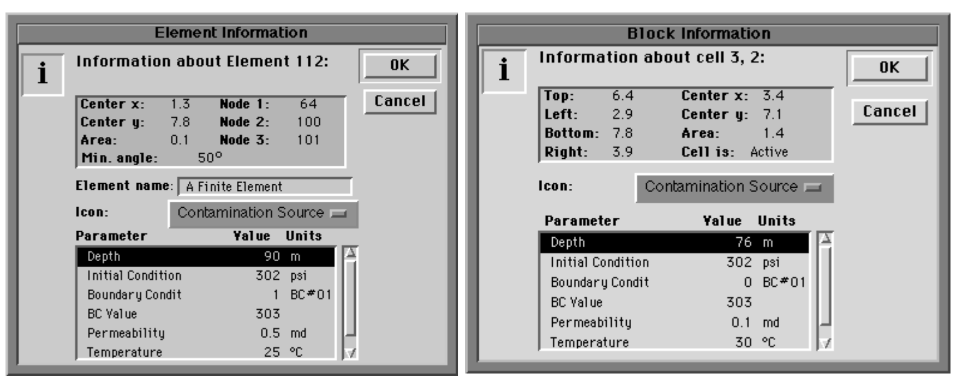



## To edit element and block information

Double clicking an element, or a selected group of elements, brings up the Element Information dialog box, where you can edit element information by following the same steps you have used to edit node information.

Double clicking a block, or a selected group of blocks, brings up the Block Information dialog box, where you can edit block information by following the same steps you have used to edit node information.

*Element Information and Block Information dialog boxes.*



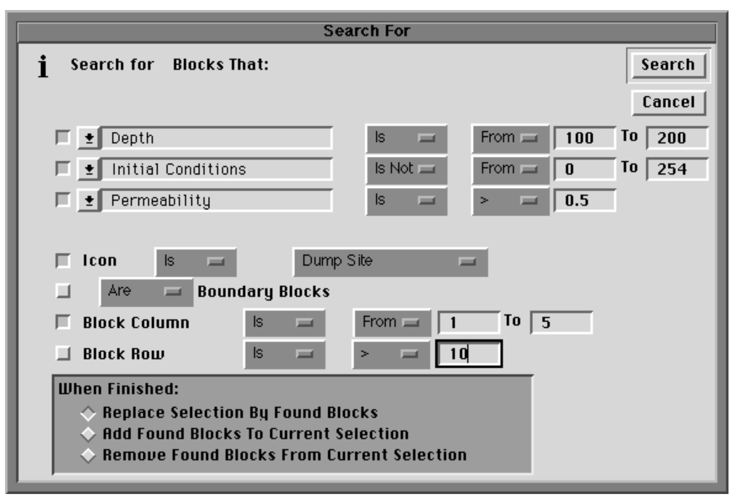
# Navigating Through Your Project


Argus ONE enables you to easily find your way through your project. You can use the zooming tools, the Goto Node, Goto Element, Goto Block and Goto Position commands to focus your view on a specific area or objects. The information ruler at the top of your window presents real-time information about your cursor position, the objects it lies above and the value in information layers.

The small Navigation window in the top left corner of the window presents a small copy of the layer's contours allowing you to easily orientate.

To obtain general information such as the number of elements, nodes, or blocks, you can use the Mesh Info... and Grid Info... commands.

To select an object or group of objects in a mesh or a grid type layer, based on the data associated with them, use the Search For... command. The Search For... dialog box enables you to set a complex search criteria that relates to all the types of information you have included in your project.




*The Search For dialog box for grid blocks.*

*Choosing the Search For... command from the Edit menu when the active layer is a mesh layer, brings up a similar dialog for nodes and elements.*

To learn more about the Search For... command refer to chapter 6 "Editing and Refining a Mesh" and to chapter 8 "Creating and Editing a Grid"

# Exporting Your Project

Exporting the data you have created with Argus ONE enables you to input it to processing and modeling programs. Exported data is saved to disk in ASCII format.

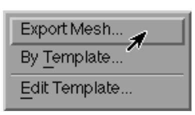
## Choosing the Output Files' Format

Argus ONE enables you to export your mesh and grid data in any format you choose. It allows you to set your own output format through the use of export templates you can define. It also allows you to quickly export your data using a generic file format, and a generic export template. Each layer type in Argus ONE has its own generic file format, and generic export template, discussed in parts 3 and 4 of this manual.

Exporting your data using the generic file format, is the fastest and simplest.

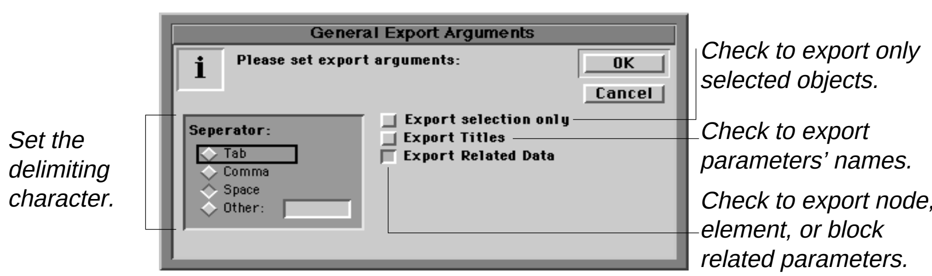
### To export your mesh or grid

1. If you have not yet defined the link between information layers and the mesh or grid objects, open the Layers dialog box and set the required links.
2. Make sure the active layer is the Mesh or Grid layer.
3. If you want to export only parts of the mesh or grid, select these parts.
4. From the File menu choose Export Mesh... or Export Grid..., the General Export Arguments dialog box opens.




The General Export Arguments dialog box allows you to set the delimiters between the different fields and to set some other variables.

*Selecting the Tab character as the delimiter will allow you to open the file in a spread sheet type application, where all data columns will be arranged in separate columns.*




- Click the OK button. The Save As dialog box opens.

5. Type the file name and set the directory or folder and click the OK button.

A progress dialog box indicating the progress of the export is presented.

# Previewing and Printing Your Project

To preview your whole project you can use the zooming tools located at the bottom left corner of the window.

- Click the Zoom Out tool  to zoom out until you have full view of your project.

## Printing Your Project

Printing your project is supported on all platforms Argus ONE is available on. Printing on the PC and Macintosh versions is supported to all printer types supported by the operating system. Argus ONE prints using the currently selected printer driver. Printing on Unix workstations versions is supported only on PostScript printers.

### To print the project

- From the file menu, choose Print.

Argus ONE displays the Print dialog box. Options in your Print dialog box differ, depending on your computer, operating system, and your printer driver.

Unless you change the options in the print dialog box, Argus ONE prints one copy of the entire project.

- Choose the Print button.

Argus ONE prints the project exactly as it looks on the screen. Layers and other information you choose to hide on the screen will be hidden on the hard copy too.

When printing, the zoom level is at 100 percent.

To reduce or enlarge the printed copy, use the Page Setup options from the Page Setup dialog.

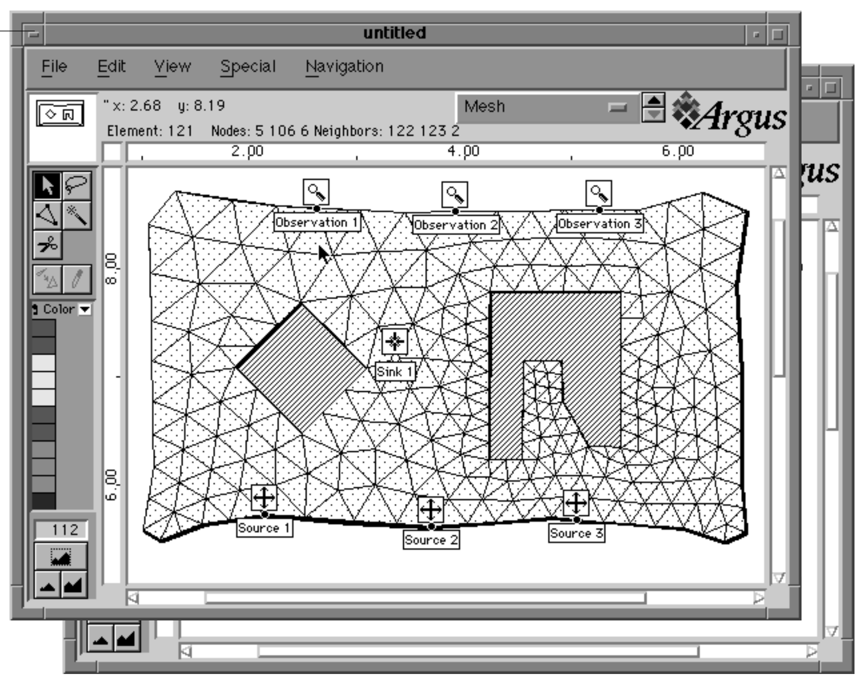
# Closing Your Project and Quitting Argus ONE

If you're ready to begin work on another project, save your current project and close its window to remove it from your screen. Make sure the window you close is not the last open window. If it is, Argus ONE quits. If you're finished working for now, you can quit Argus ONE.

## To Close the Project

- On the Macintosh, click the close box in the upper-left corner of the project window. On other platforms, double-click the close box in the upper-left corner of the project window,
  - Or,
  from the File menu, choose the Close command.


*Choose the Quit command to close all open windows and quit Argus ONE.*




*Use the close box to close this window.*

## To Quit Argus ONE

- From the file menu, choose Quit.

If you forget to save an open project before you close its window or quit Argus ONE, Argus ONE displays an alert message that asks you whether you want to save changes to the project. Click the Yes button in the alert box.

# Where to Go from Here

Once you've completed your first project with Argus ONE, you are ready to investigate other features and techniques. Or perhaps you'd like more help with a task described in this section. The following table directs you to more information.

| For information regarding | See |
|---|---|
| Working on your computer. | Your computer documentation. |
| Using the Argus ONE menus and commands and working in project windows. | Part 2, Chapter 1: “The Argus ONE Workplace” |
| Using Information layers and creating information contours. | Part 2, Chapter 2: “Working With Information Layers” |
| Using Data type layers and importing scattered, grided and meshed information. | Supplements |
| Visualizing your model results as well as mesh and grid information distribution. | Supplements |
| Importing DXF files to Maps type layers. | Part 2, Chapter 1: “The Argus ONE Workplace” |
| Importing and Exporting contours. | Part 2, Chapter 2: “Working With Information Layers” |
| Creating a Grid. | Part 4, Chapter 8: “Creating and Editing a Grid” |
| Creating a Mesh. | Part 3 Chapter 5: “Meshing a Domain” |
| Exporting a Mesh. | Part 3, Chapter 7: “Exporting and Importing a Mesh” |
| Exporting a Grid. | Part 4, Chapter 9: “Exporting a Grid” |
| Creating and Using Export Templates | Part 2, Chapter 4: “Export Templates” |
| Printing only some of the layers or some of the pages. | Part 2, Chapter 1: “The Argus ONE Workplace” |

# The Mathematics Beneath Argus ONE

Argus ONE is a powerful pre-processor application, rich with features that make it easy to describe complex physical problems.

To support the discretization and other numerical needs for such complex problems, Argus ONE relies on sophisticated mathematical algorithms.

Argus ONE also includes a set of validity checking algorithms that are performed on-line to check the validity of user actions, such as creating and moving contours, nodes and elements.

Since we realize you will be using Argus ONE for engineering and scientific work requiring high level of accuracy, this chapter will give you a quick overview of Argus ONE's underlying mathematical algorithms and concepts.

# Numerical Precision and Coordinate Systems

Argus ONE enables you to set the coordinate system you wish to work with. This includes setting the coordinate system orientation, its scale and its units.

Your work is carried out in real coordinates. All of Argus ONE internal data representations for real type variables are of double precision depending on your platform representation of double precision.

# Mesh Density

The mesh density is a new proprietary concept. It was developed and implemented to allow you to easily define the mesh density anywhere within the domain to be meshed, without forcing you to define the number of divisions along the domain boundaries and to subdivide the domain into sub-regions.

Mesh density is a real type variable set by you to define the average size of an element at a given area of the mesh. Mesh Density is measured in the current scale and units.

The final mesh density is affected by the following factors:

- The mesh density you apply to the domain outline contour.
- Additional mesh densities you set in the mesh density layer, through density contours.
- Size of segments constructing the domain outline contour. Segments, smaller than the user defined densities, either by the domain outline contour's density or by the mesh density contours, override the original densities if they are 2.5 times smaller than the latter.
- The internal distance between domain outline vertices.
- Internal distance between domain outline holes (islands) and their distance from external outline vertices.
- The internal distance between Point objects (sources/sinks) and their distance from domain outline contour vertices.
- The internal distance between Open Contour objects and their distance from domain outline contour vertices.

Argus ONE’s MeshMaker modules weigh the user defined densities, such that the smallest must apply and the larger are recommendations. It gradually increases the element size from the smallest size required by you up to your largest required size, while keeping the elements as close to equal edges triangles as possible.

# Auto Mesh Generation

Argus ONE's meshing modules use a proprietary improved Delaunay triangulation algorithm.

The auto mesh generation is based on three main algorithms; the mesh density, the tessellation and the smoothing.

Together with the mesh density tool, Argus ONE allows you to easily mesh complex geometries, including "islands" and "half islands", resulting in meshes with no acute elements. The meshing module default criteria for acute elements is set to 22.5º (degrees) as recommended by Strang and Fix in their book "An Analysis of the Finite Element Method", however, you may override this criteria.

Our tests show that finite element meshes of thousands of elements created without using the mesh density layer, contain usually no more than 2 pro-mil acute elements. When using the mesh density layer to apply mesh densities in areas where you expect acute elements, your mesh will contain no acute elements.

After auto mesh generation, Argus ONE enables you to select and view the acute elements, if any exist.

# Mesh Smoothing

Mesh smoothing is carried out by the meshing module as part of the auto mesh generation procedure. However, you can select the whole mesh or parts of it, and perform additional smoothing.

The smoothing algorithm is an iterative procedure that moves the nodes to the center of gravity of their respective polygons. The meshing module default number of iterations is set to three. Although we have found that after three iterations sufficient convergence is achieved, you can change this default to any number you wish.

# Mesh Refinement

You can refine any set of selected elements. Refinement is performed by creating three new nodes on the mid-point of each of the element's edges, thus dividing the refined element into four similar elements. Adjacent elements are split into two elements to avoid transition elements.

Refinement tends to degrade the adjacent elements. Refining a region by assigning it a lower density value and re-meshing produces much better elements.

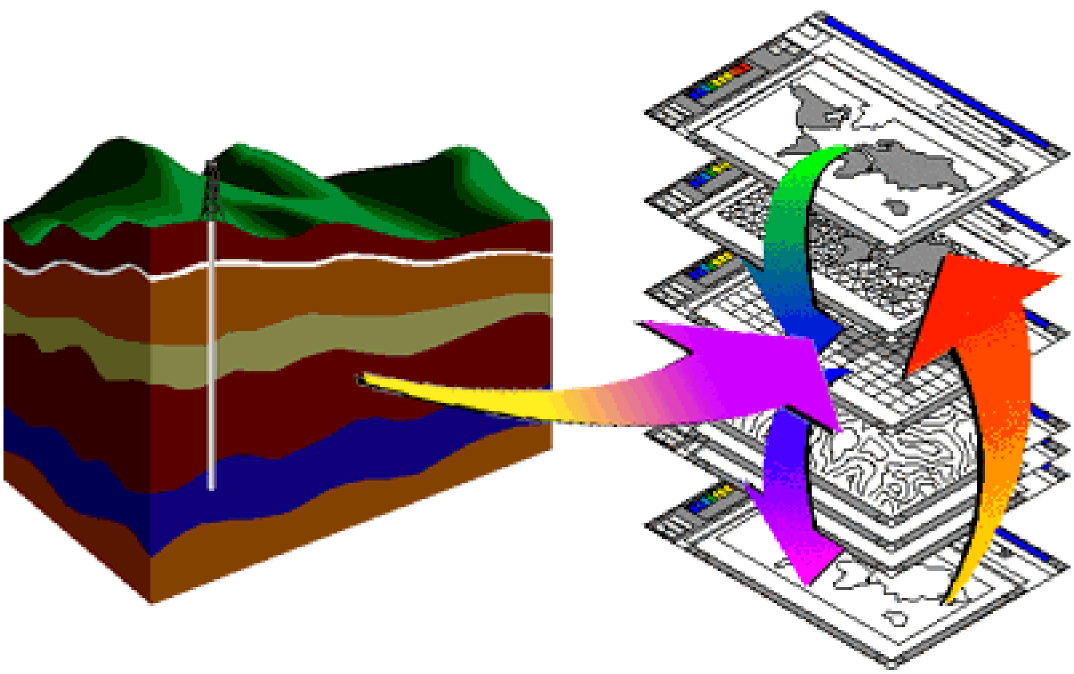
# Renumbering (BandWidth Optimization)

If the solver you use needs to receive an optimized BandWidth matrix, you may use Argus ONE bandwidth optimization (renumbering) algorithm. The renumbering algorithm is a fast “minimum degree” heuristic algorithm.

# Interpolation of Information Layers’ Data

You can direct Argus ONE to interpolate data you have entered in an Information layer using the contour mapping tools. The interpolation is performed to give interpolated values at an X and Y position. Argus ONE interpolation is based on a proprietary “Inverse Distance weighted Interpolation” algorithm. Other interpolators are available through third-party developers using the Argus PIE technology.

Part 2



# Basics for Using Argus ONE

# The Argus ONE Workplace

# Overview

Argus ONE is a very productive and enjoyable place to work in. Everything you need to define, mesh, grid, add information to, and navigate through, is on the screen or at your fingertips. The Argus ONE menus list commands that automate many tasks. The tool palette puts commonly used commands in easy reach. The rulers and zoom palette help you orientate through the mesh and grid, and information layers allow you to combine all types of information into a comprehensive project.

# The Window

Argus ONE windows contain the same elements on all windowing systems it is available on. The basic elements of the window differ between the various windowing systems. However, all of the basic elements, except for the iconize button not present on the Macintosh window, have a similar function.

## Basic Elements of the Window

### The Title Bar

The title bar shows the project's name.

### The Scroll Bars

Use the scroll bars to scroll to different areas in the project.

### The Close Box

Double-Click (Macintosh click) the close box to close the window. You can also choose the Close command from the File menu.

### The Size Box or Window Frame

On the Macintosh, use the Size box to resize your window.
On other widowing systems use the Window Frame to resize the window.

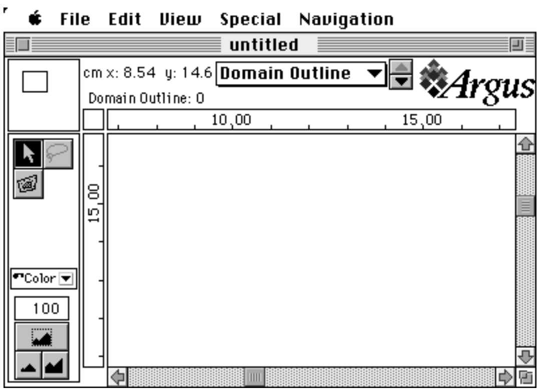
This enables you to view windows from different applications while still having a view of your Argus ONE project.

### The Maximize Box or the Zoom Box

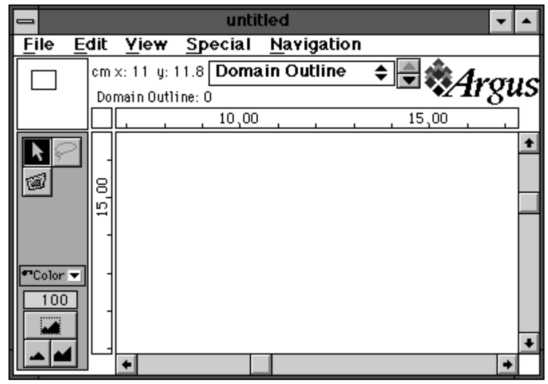
On the Macintosh, click the Zoom box. On other windowing systems, click the Maximize box. The window zooms to your screen size. Click it again and the window zooms back to its previous size.

For further platform specific matters refer to your computer user manual.

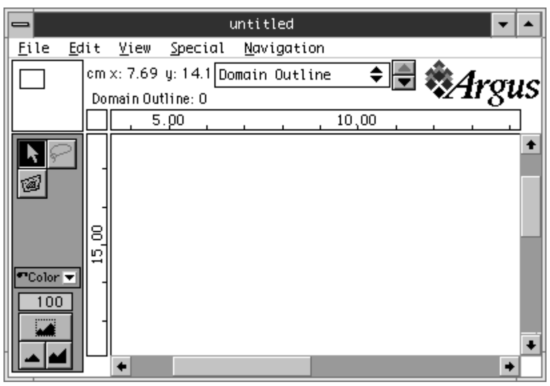
*Argus ONE window on the Apple Macintosh*



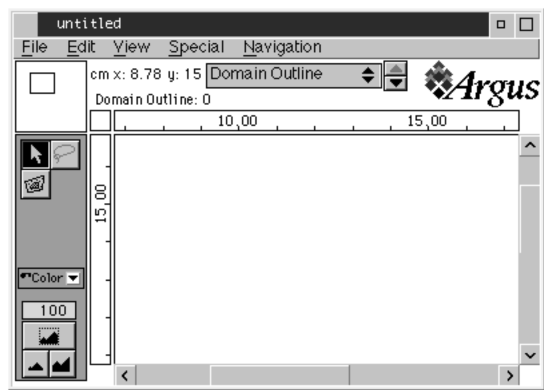

*Argus ONE window on Microsoft Windows*



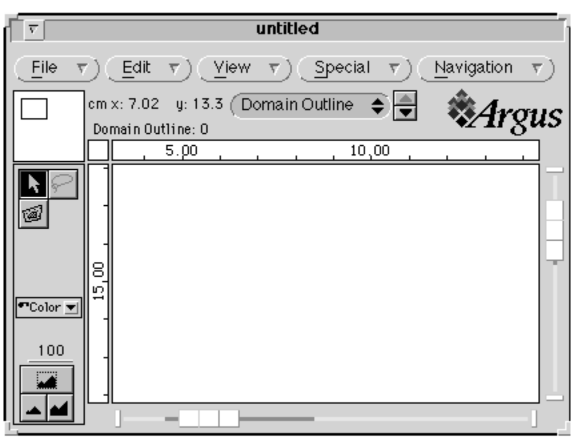

*Argus ONE window on IBM OS2 1.2*



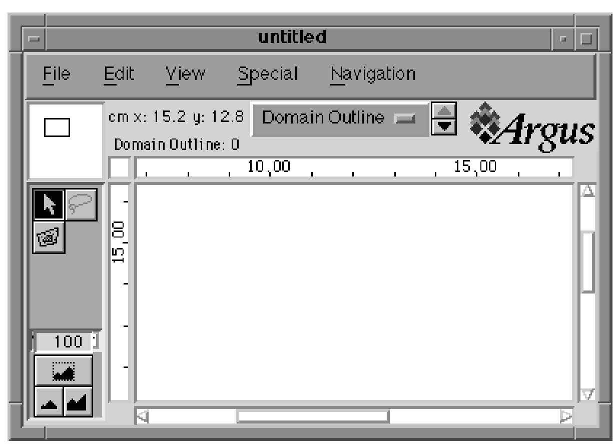

*Argus ONE window on IBM OS2 2*




*Argus ONE window on Sun's OpenLook*




*Argus ONE window on Motif*



## The Coordinate Rulers

The vertical and horizontal rulers reflect the current coordinate system, its scale and its units. Changing the coordinate system is described in detail later in this chapter.

## The Information Ruler

The information ruler keeps you informed with regard to your cursor's position, the current units, and the active layer. The cursor's position describes location and information about objects the mouse lies above.

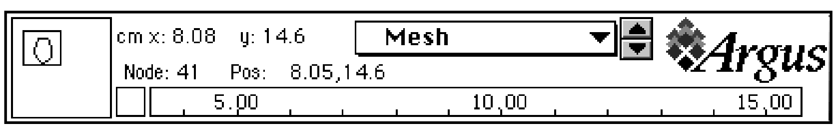
### Cursor's position

Cursor's position is given in the current units and coordinate system direction by the “x:” and “y:” values.

While working in a mesh layer, the information ruler presents information about the node or element the cursor lies above. While working in a grid layer, the information ruler presents information about the row and column the cursor lies above. While in an information layer, the information ruler presents the layer value at the cursor’s location according to the active interpretation mechanism.

### Node information

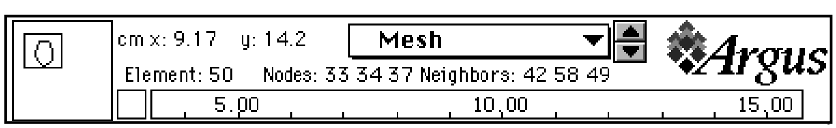
Whenever the cursor lies over a node, the node's number and position are presented in the information ruler.




The “Pos:” coordinates indicate the exact node coordinates. When the cursor enters the vicinity of the node (a rectangle of 4 pixels surrounding its actual location), the ruler shows the node's number and its coordinates.

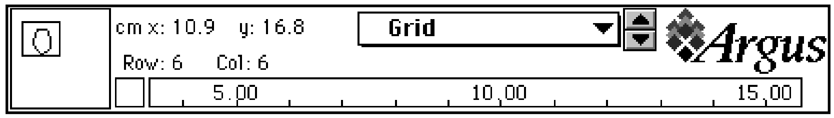
### Element information

Whenever the cursor lies over an element, the element's number, its three nodes' numbers and the numbers of its neighboring elements, are shown.



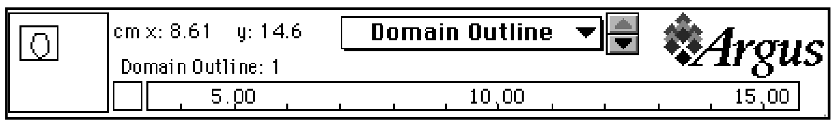
### Block information

Whenever the cursor lies over a grid block, the block's row and column are shown.

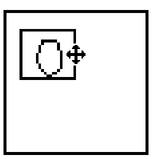



### Information Layers value

When the currently active layer is an Information type layer (including a Domain type layer), the layer's name and the value of the layer's main parameter at the cursors location are shown. The information presented reflects the interpretation mechanism applied to the layer.

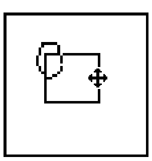



### The Navigation window



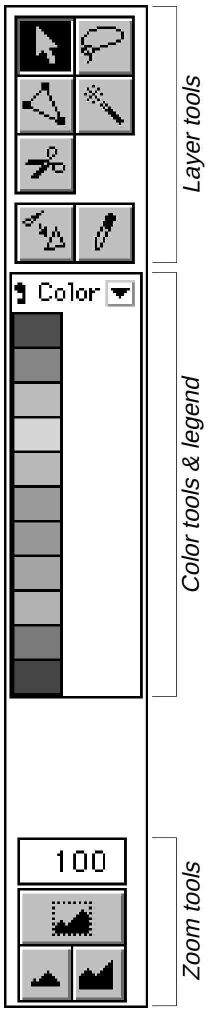
The square frame on the left corner of the information ruler is the navigation window. The outer square frame is a reduced size representation of the drawing size. The internal rectangle is a reduced size representation of the current window size. The arbitrary shape or shapes presented in the navigation window are reduced size outlines of the contours in the active layer. If the current layer is a mesh or a grid layer, the contours in the domain layer associated with them are presented.

#### To focus your view on a specific area using the navigation window



1. Place the cursor above the internal rectangle.
   Your cursor shape changes into a cross ✥.
2. Click–drag the cursor to move the internal rectangle.
3. Release the mouse button to place the internal rectangle above the area of interest.

The window is centered on the area of interest.




# The Tool Palette

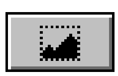
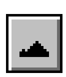
The tool palette contains the tools you'll use most often during your work. Tools help you perform tasks that demand your direct access to the working area, such as selecting, creating, and editing objects. Argus ONE tools are context sensitive, that is, they will operate when appropriate.

Some of the icons in the tool palette represent buttons rather than tools. Buttons are activated (highlighted) only when a group of objects they operate on is selected. The buttons are grouped at the bottom of the layer specific tools.

There are three tool groups in the tool palette. The Color buttons and legend, the Zoom tools and the Layer specific tools. The layer specific tools in the palette are layer dependent. That is, depending on the layer you are currently working in (the active layer), some of them may be inactive (dimmed) or they may have a different shape.

## The zoom tools

At the bottom end of the tool palette you will find the zooming tools. These allow you to zoom the view to the required size, thus enabling you to view a specific zone of your work in more detail, or to view the whole project at one glance.

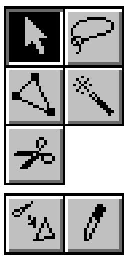
| Tool | Tool's name and function | Tool activated when |
|---|---|---|
| 100 | The Zoom by Percent tool. Type in the zoom level and hit the RETURN key. Your view is zoomed to the required zoom level. | Always |
|  | The Magnifying Glass tool. Allows you to zoom in and out in various ways using the mouse. | Always |
|  | The Zoom In button. Click the button to zoom in to twice the current zoom level. | Always |
|  | The Zoom Out button. Click the button to zoom in to half the current zoom level. | Always |

To read more about zooming techniques refer to the paragraph "Zooming Techniques" in this chapter.

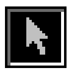
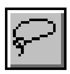
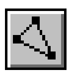
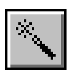
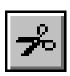
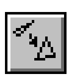
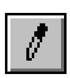
## Mesh layers tool palette

The mesh layer tool palette contain all the tools you need to easily create and edit a mesh and its objects, the elements and nodes. To operate on the mesh objects, select a tool and use it on the appropriate object.

A tool is active only when you can use it. When a tool is not available, it is dimmed to indicate you can not use it. When you select a tool it highlights itself to indicate it is the active tool, and the cursor changes its shape to suit the selected tool. The different cursors are presented in the paragraph “Mouse Cursors and Techniques” later in this chapter.



*Mesh layer tools*

| Tool | Tool’s name and function | Tool can be activated when |
|---|---|---|
|  | The Arrow tool. The default tool in all layers. Allows you to select, edit and move objects. | Always. |
|  | The Lasso tool. Allows to select all objects within an arbitrary shape. | There are one or more objects in the current layer. |
|  | Element creation tool. Allows you to manually create elements. | A mesh layer is the active layer. |
|  | The Magic Wand. Click the magic wand anywhere within a domain outline contour to automatically mesh that domain. | In a mesh layer, when the domain layer associated with it contains a domain outline contour. |
|  | The Detach tool. Detaches the selected elements from the mesh. | In a mesh layer containing two or more connected elements. |
|  | The Smooth button. Smooth the selected elements. | In a mesh layer, when one or more groups of elements are selected. |
|  | The Refine button. Splits the selected elements into four similar elements. It also splits the surrounding elements into two. | In a mesh layer, when an element or some groups of elements are selected. |

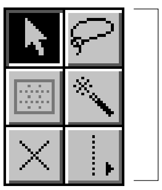
To learn more about using the mesh tools refer to chapter 5 “Meshing a Domain” and chapter 6 “Editing and Refining a Mesh.”

## Grid layers tool palette

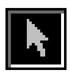
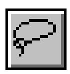
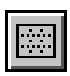
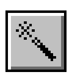
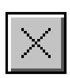
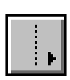
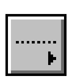
The grid layer tool palette contain all the tools you need to easily create and edit a grid and its object, the blocks. To operate on the grid objects, select a tool and use it on the appropriate object.

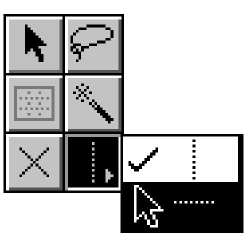
To resize, move or delete the grid, you must first open the handle on its top left corner.

A tool is active only when you can use it. When a tool is not available it is dimmed to indicate you can not use it. When you select a tool it highlights itself to indicate it is the active tool, and the cursor changes its shape to suit the selected tool. The different cursors are presented in the paragraph “Mouse Cursors and Techniques” later in this chapter.



*Grid layer tools*

| Tool | Tool’s name and function | Tool can be activated when |
|---|---|---|
|  | The Arrow tool. The default tool in all layers. Allows you to select, edit and move objects. | Always. |
|  | The Lasso tool. Allows to select all objects within an arbitrary shape. | There are one or more objects in the current layer. |
|  | Grid creation tool. Allows you to manually create a grid. | A grid layer is the active layer, and there is no other grid in that layer. |
|  | The Magic Wand. Click the magic wand anywhere within a domain outline contour to automatically grid that domain. | In a grid layer, when the domain layer associated with it contains a domain outline contour. |
|  | The Delete rows and columns tool. Deletes the selected rows and columns. | In a grid layer containing a grid. |
|   | The Add Row or Column tools. Add a column or a row to the grid.<br>To select a tool, click and hold the mouse button until the menu pops up.<br>Click-drag the tool and it will add multiple rows or columns. | In a grid layer, when within the grid. |



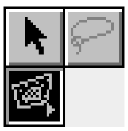
To learn more about using the grid tools refer to chapter 8“Creating and Editing a Grid.”

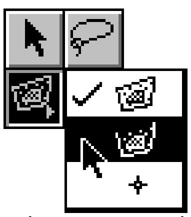
## Domain layers tool palette

The domain layer tool palette contain all the tools you need to easily create and edit domain outline objects, the Point, Open Contour and Close Contour objects. To operate on the domain layer objects, select a tool and use it on the appropriate object.

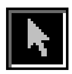
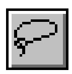
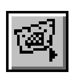
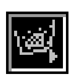
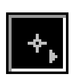
When you select a tool it highlights itself to indicate it is the active tool, and the cursor changes its shape to suit the selected tool. The different cursors are presented in the paragraph "Mouse Cursors and Techniques" later in this chapter.

*Domain layer tools*



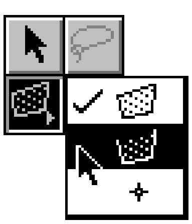


*The contour tool popup menu for domain type layers.*

| Tool | Tool's name and function | Tool can be activated when |
|---|---|---|
|  | The Arrow tool. The default tool in all layers. Allows you to select, edit and move objects. | Always. |
|  | The Lasso tool. Allows to select all objects within an arbitrary shape. | There are one or more objects in the current layer. |
| <br><br> | **1.** Click and hold the mouse button until the popup menu appears.<br>**2.** Move the cursor to the item you wish to select.<br>**3.** Release the mouse button.<br>The Close Polygon tool. Allows you to create a Close Contour.<br>The Open Polygon tool that allows you to create an Open Contour.<br>The Point tool that allows you to create a Point Object. | A domain layer is the active layer. |

## Information layers tool palette

*Information layer tools*



The information layer tool palette is almost identical to that of the domain layer. The only difference is that the Close and Open Polygon tools icons do not include the "island" shape.

To learn more about using the information layer tools refer to chapter 2 "Working with Information Layers."

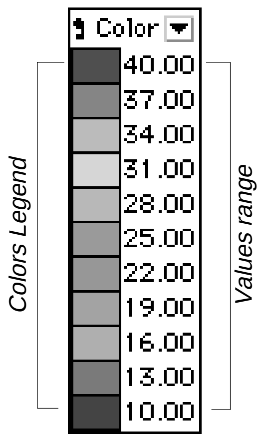
## Maps layers tool palette

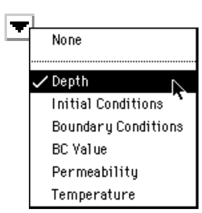
The maps layers tool palette is explained in detail in the Supplement chapter.

## The color tools and legend

Colors are off


The color palette.
Colors are on




Layer's parameter popup menu.

The color tools enable you to turn colors on and off, set a layer's parameter to be evaluated by color, and present you with the range of values and their respective colors.

### To turn colors off

- Click the Color handle.
  The handle moves into a horizontal position. Colors are turned off in all your project layers, and the color legend roles up.

### To turn colors on

- Click the Color handle.
  The handle moves into a vertical position. Colors are turned on in all your project layers, and the color legend roles down.

### To set the evaluated parameter

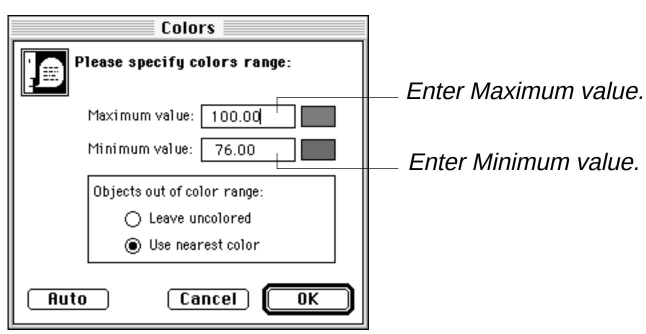
- Click and hold the mouse button in the Color popup arrow to open the parameter popup menu and choose one of the layer's parameters.
  The objects in the layer are evaluated for the selected parameter and assigned a color.

### To automatically set the color range

- From the View menu, choose Calculate colors.
  Each object in the layer is evaluated for the selected parameter, the minimum and maximum values are calculated and the color scale is adjusted to the calculated range.

### To fully control the color range

- From the View menu, choose the Colors... command.
  The Colors dialog box opens.




- To set the range manually type in the required minimum & maximum values.
- To automatically calculate the range, click the Auto button.
  This is equivalent to choosing the Calculate colors menu item.
- To leave objects that are out of the range uncolored, choose the Leave uncolored radio button.

## Zooming Techniques

Zooming allows you to easily navigate within your project canvas. It also enables you to work in fine detail. If for instance, you need to digitize a background map in a high resolution, you can zoom in before you start digitizing a contour and using the auto-scroll capability, which is available during object creation and editing, digitize the whole contour. There are few ways you could zoom a view.

- Click the Zoom by percent text box, type the zoom level in percent. For example, to zoom in four times type 400 and hit the ENTER key on your keyboard.
- To zoom in to twice the current zoom level, click the Zoom In button.
- To zoom out to half the current zoom level click the Zoom Out button.

To magnify a specific area of the view:

1. Click the Magnifying Glass tool to select it, the cursor changes into a magnifying glass .
2. Click and drag the tool around the area you want to magnify.
   A stretch band follows your cursor movements.
3. Release the mouse button.
   Your view is zoomed in so that the area contained in the stretch band occupies the whole window.

To zoom in around a specific point to twice the current zoom level while maintaining it at the center of the screen:

1. Hold down the OPTION key (Macintosh) or the ALT key (other platforms), the cursor changes into the magnifying glass .
2. Click the point.
   Your view is zoomed in to twice the current zoom level and is centered around the point you clicked.

To zoom out around a specific point to half the current zoom level while maintaining it at the center of the screen:

1. Hold down the OPTION+SHIFT keys (Macintosh) or the ALT+SHIFT key (other platforms), the cursor changes its shape into a .
2. Click the point.
   Your view is zoomed out to half the current zoom level and is centered around the point you clicked.

The Zoom by percent text edit box presents the current zoom level. To return to 100 percent zoom level, hold down the CONTROL key and click the Zoom by percent text box.

---

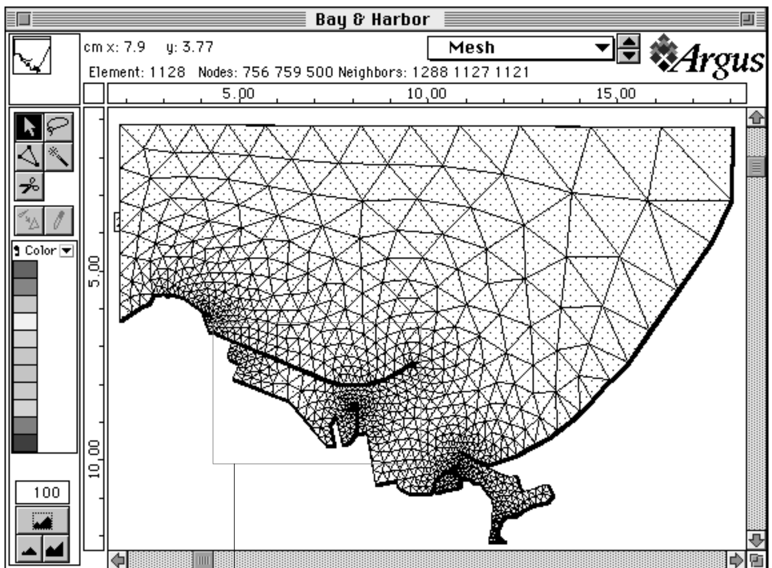
**Important note**: Your maximum zoom level is limited to “maxint” pixels.

---

## A Zooming example

To magnify a rectangle around the head of the large breakwater in the following Argus ONE window, select the Magnifying Glass and stretch that rectangle.

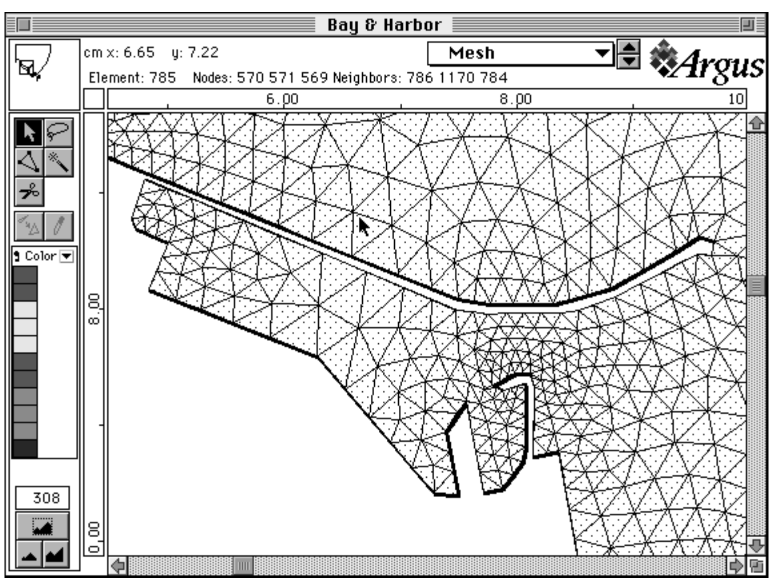
*A mesh of a bay and harbor at full view.*




*The zoom stretch-band rectangle.*

*Mesh after zoom in.*

*Detailed view of the mesh in the inner harbor.*



## The Coordinate System

When you start a new project in Argus ONE, you should set the coordinate system to fit the coordinate system of the data you're about to import into the project and the one you intend to work with. Argus ONE allows you to change some of the coordinate system's parameters after you've started to work.

### Controlling the coordinate system

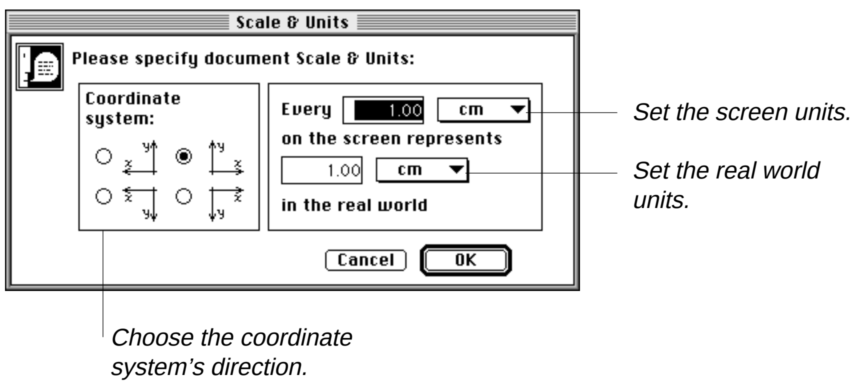
The Scale & Units dialog allows you to control the coordinate system using four parameters.

- Screen Units (scale nominator)
- Real World Units (scale denominator)
- Scale, the Ratio of Screen Units to Real World Units
- The Coordinate system directions

#### To open the Scale & Units dialog

- From the Special menu, choose the Scale & Units... command.
  The Scale & Units dialog box is opened.

*The Scale & Units dialog box.*




When you create a new Argus ONE document, its default coordinate system settings is:

- Screen Units — centimeters
- Real World Units — centimeters
- Scale — 1:1
- Coordinates direction — right handed (mathematical, positive X to the right, positive Y upwards)

The rulers reflect your scale and screen units only in 100 percent zoom level. Changing the zoom level is equivalent to changing the scale and vise a versa. If you are not comfortable working in your current 100 percent zoom level you can change the scale to enlarge or reduce your objects' sizes on the screen.

### Screen units

There are three available screen units:

- Centimeters
- Millimeters
- Inches

Choosing the setting 1.00 Inch, means that, at 100 percent zoom level, one tick mark on the ruler will be of length one (1) over "number of real world units" inches. Changing the screen units between the three available options reduces or enlarges the screen object size accordingly. If for instance, you change the screen units from centimeters to millimeters, all objects are scaled by the ratio <centimeters/millimeters>=1:10.

#### To choose the screen units

- Select the desired units form the screen units popup menu.

### Real World units

Real World units are nothing but a name presented in the information ruler. Changing the Real World units from meters to kilometers for instance, has no effect on your project window, or on the size of your object on the screen.

#### To choose the Real World units

- Select the desired units form the Real World units popup menu.

Argus ONE assumes that you use a 72 dpi (dots per inch) screen with an aspect ratio of 1. This insures the WYSIWYG (What You See Is What You Get) looks and that circles for instance look like circles and not like ellipses. If your screen specifications are different, read the installation guide to change your basic Argus ONE settings to compensate for these differences.

### The Scale

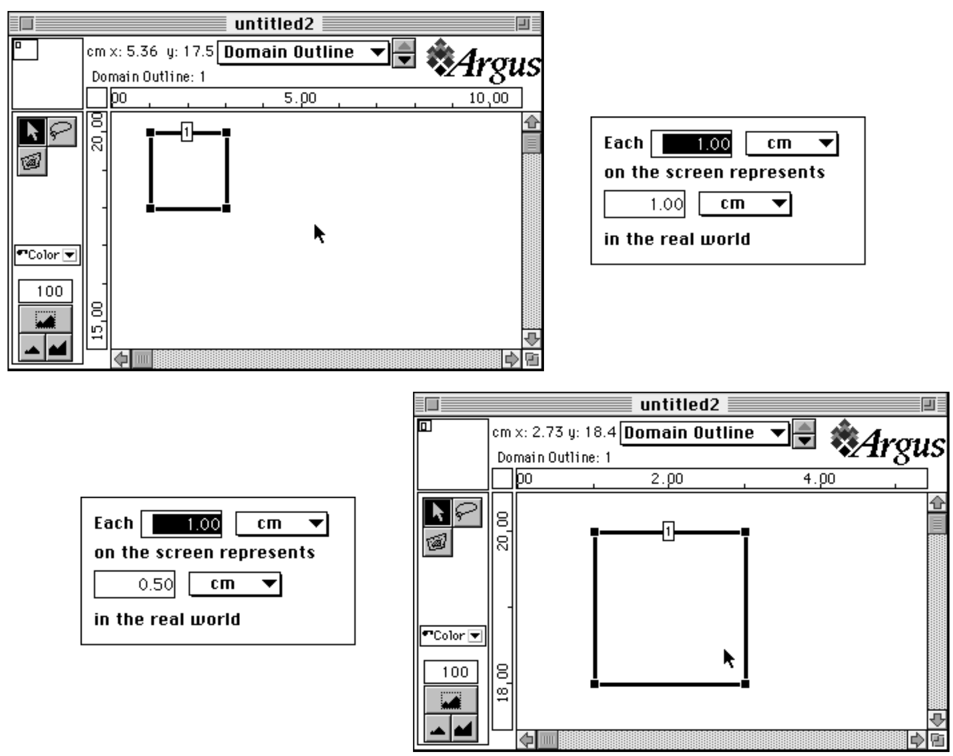
If information related to your project is supplied to you in real world units, it might happen that your domain outline is, for instance, given in millions of meters. Setting the scale and origin enables you to work comfortably in Argus ONE’s workplace, while maintaining the real coordinates in which your data must be exported to your numerical model.

For a given screen units, the scale is defined as the ratio of the number of screen units over the number of real units.

$$Scale = \frac{number\ of\ screen\ units}{number\ of\ real\ world\ units}$$

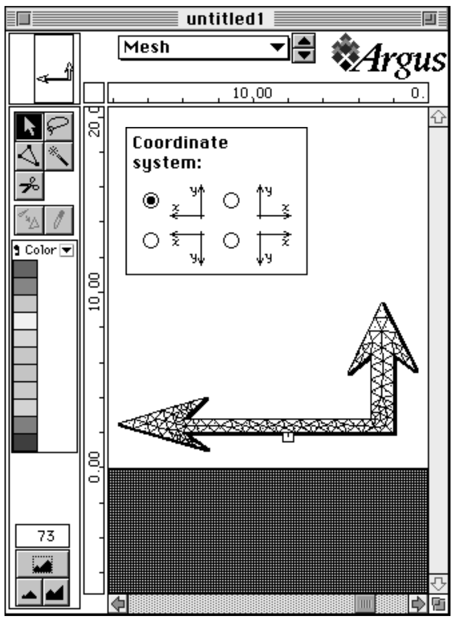
Changing the scale allows you to reduce or enlarge objects' appearance on the screen. All of Argus ONE's objects will be scaled, including contours, meshes, grids and picts.

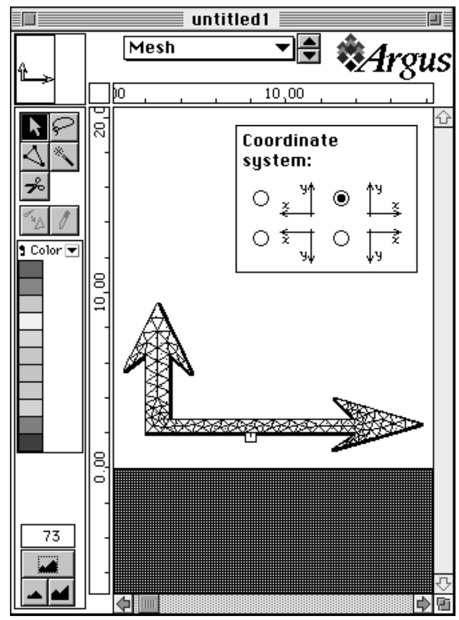
For instance, to enlarge the screen appearance of the contour in the following example, change the number of real world units from 1 to 0.5 or change the number of screen units from 1 to 2.

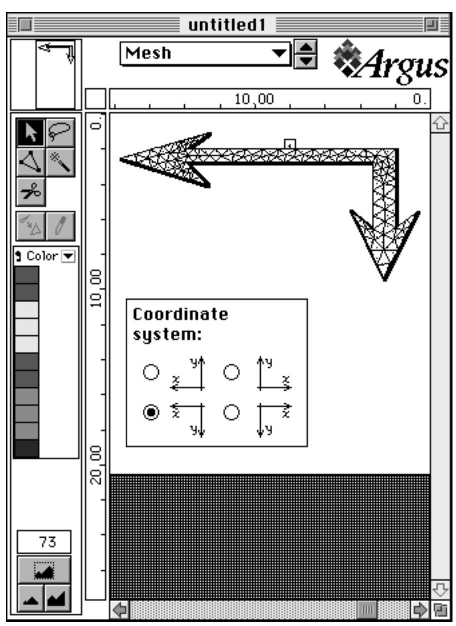


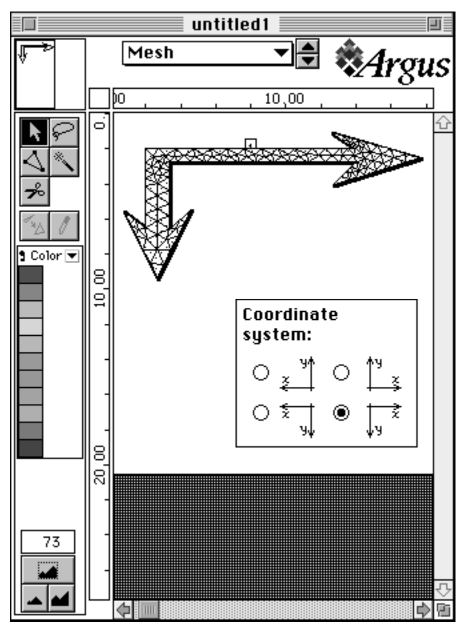
## The coordinates direction

You can change the coordinate system's directions by selecting the appropriate radio button. The origin of the coordinate system and the rulers change to comply with the new directions. The following four screen-shots demonstrate the four available coordinate system directions and how Argus ONE changes the workplace to reflect the different directions.



## The Drawing Size

When you open a new project, Argus ONE sets the drawing size to some default. Argus ONE does not allow you to create, move or stretch objects out of the current drawing size. However, you can move objects that were imported to an area out of the drawing size, into it. You may also enlarge and reduce the drawing size.

### Zooming and Drawing Size

*Enlarging drawing size decreases maximum zoom level.*

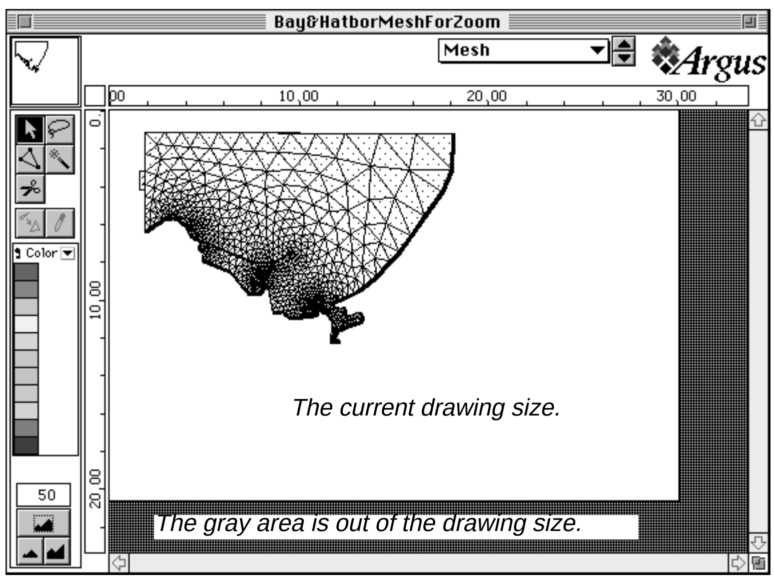
When you enlarge or reduce the drawing size, Argus ONE respectively reduces or enlarges the maximum available zoom size.

### Drawing Size and Scale & Units

If you have completely used up your drawing size you can enlarge it again by enlarging the current scale. This way each object on the screen is resized by the ratio of the current and the new scale while maintaining your maximum available drawing size.

### To view the entire drawing size

- Zoom down until you see the drawing size completely surrounded by a gray area.

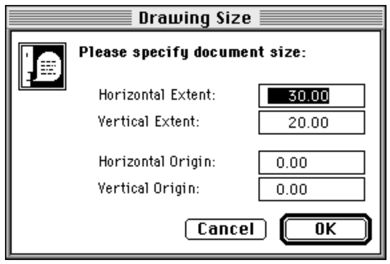



You can redefine the document's size to be up to 11.5x11.5 meters measured in any units. As explained before, the maximum drawing size is affected by the current zoom level and scale, and is limited by “maxint” number of pixels.

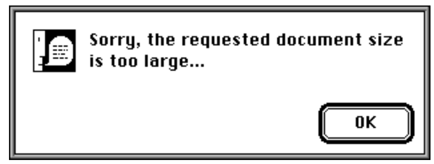
### To set the Drawing Size

1. From the Special menu, choose the Drawing Size... command. Argus ONE presents you with the following dialog box:

*The Drawing Size dialog box.*




*If the horizontal or vertical extent you choose are too large this alert is presented.*




2. Type the required horizontal and vertical extents in the current units.

If you type in numbers which exceed the currently allowable drawing size, Argus ONE presents an alert and automatically presents again the Drawing Size dialog box.

---

**Important note:** If you reduce the drawing size, some objects might be left partially or fully out of the new drawing size. Also, enlarging or reducing the document size is always performed towards the origin and not from it.

---

### To set the coordinate system's origin

1. From the Special menu, choose the Drawing Size... command.
   You are presented with the Drawing Size dialog box.
2. Type in the required Horizontal and vertical origins.

If, for instance, your project spans in the horizontal from 1,100,000 meters to 1,600,000 meters, you can set the horizontal origin to 1,000,000. Every object you create, a contour, an element, a node or a grid block will have the correct coordinates, that is, it will include the 1,000,000 meters deduced from the origin. Using the Scale & Units command in the above example, you could also set the scale to 1:1000 and your project will span on the screen only 6 screen units.

# Layers

Layers are one of the very basic concepts underneath Argus ONE's workplace. Your work is always carried out in one of the layers. Working with layers is actually like working with transparency sheets that you overlay. However, layers are much more convenient and powerful. You can hide any number of layers from view at any time. Hiding layers is very important when your workplace becomes cluttered by too much information. Argus ONE's layers are much more powerful then standard CAD layers since they transfer information between themselves and affect other layers.

There are six types of layers in Argus ONE's workplace:

1. **Domain type layers** Where you define the domain's outline, related to as a "Domain Outline", and set the default mesh and grid densities for mesh and grid layers that might be associated (linked) with it.
2. **General Information type layers** Serving as containers of quantitative spatial information such as property boundaries, boundary conditions, physical parameters, initial conditions, topographies and also mesh and grid densities. Working with these layers is described in detail in chapter 2 "Working with Information Layers" and chapter 3"Layer Parameters."
3. **Data type layers**, into which you read and import scattered, grided and meshed data. This layer type enables you to read large number of points with related information, created by a numerical model or sampled during a survey. Using data type layers you can import your model results or any other point-wise information into Argus ONE. To read more about data layers refer to Supplement for version 2.5.
4. **Mesh type layers**, where you create and manipulate your finite element meshes.
5. **Grid type layers**, where you create and manipulate finite difference grids.
6. **Maps type layers**, into which you can import digitized maps in DXF and Shape file formats, annotate your project and create post-processing objects.

## Mesh Related Layers

A mesh layer must be assigned a domain type layer, and an information type layer serving as its density layer. Such three layers enable you to define and mesh a domain. Working with these layers is described in detail in chapter 2 "Working with Information Layers" and chapter 6 "Meshing a Domain."

## Grid Related Layers

A grid layer can be assigned a domain type layer, and an information type layer serving as its density layer. Such three layers enable you to define and grid a domain. Working with these layers is described in detail in chapter 2 "Working with Information Layers" and chapter 8 "Creating and Editing a Grid."

## Maps Type Layers

Maps type layers enable you to perform two main activities in them:

- You can use maps layers to create and import background maps for your project.
- You visualize and investigate your model results and the distribution of parameters on your meshes and grids. To read about these post-processing and visualization capabilities refer to the **Supplement** chapter.

You can import DXF maps and Argus ONE contour maps into your project. Digitized maps are very often saved in DXF format and you can also instruct most of your CAD applications to export your drawings in DXF format and import them into Argus ONE.

You can use maps layers containing such DXF imported files to digitize information contours in information layers within Argus ONE's workplace without the need for a stand alone digitizing table. You might use maps to digitize domain boundaries, material distributions, topographic, legal, geological, boundary and initial condition information. Maps layers also help you orientate within complex domains, where the outline of the domain is not sufficient for orientation.

If the DXF file you import contains polyline type objects, you can import them directly into an information layer, providing the contours are valid Argus ONE contours (see chapter 2).

You can also import Argus ONE exported contour files, copy contours from other layers and paste them into Maps type layers, and copy graphic objects from a maps layer to any other layer.

Maps type layers can serve as scratch-pad layers since they do not check the graphics objects brought into them for contour validity as the other layers do.

### Supported graphic objects

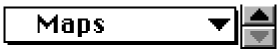
A DXF file might contains vast amounts of information. Argus ONE import command currently supports the following DXF graphic objects:

- **Line** — defined by its two end points
- **Polyline** — one or more consecutive line segments
- **Circle** — defined by its center and radius
- **Arc** — defined by its center
- **Text** — the characters and a position

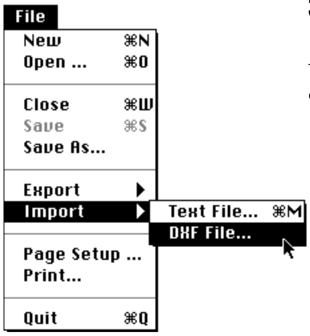
The “Z” attribute and color of these graphic objects are imported as well. The text imported is scalable and rotatable.

Argus ONE graphics objects, the **point object**, the **open contour** and **close contour** objects can also be imported and pasted into a Maps layer.

## To import a DXF file



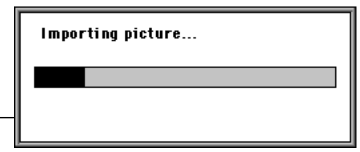

1. Click the bottom arrow in the *Layers Selection Button* until the *Layers Popup Menu* reads Maps. The Maps layer is active.
2. From the File menu, choose Import DXF File...
3. In the Open dialog box, choose the file and click the Open button.

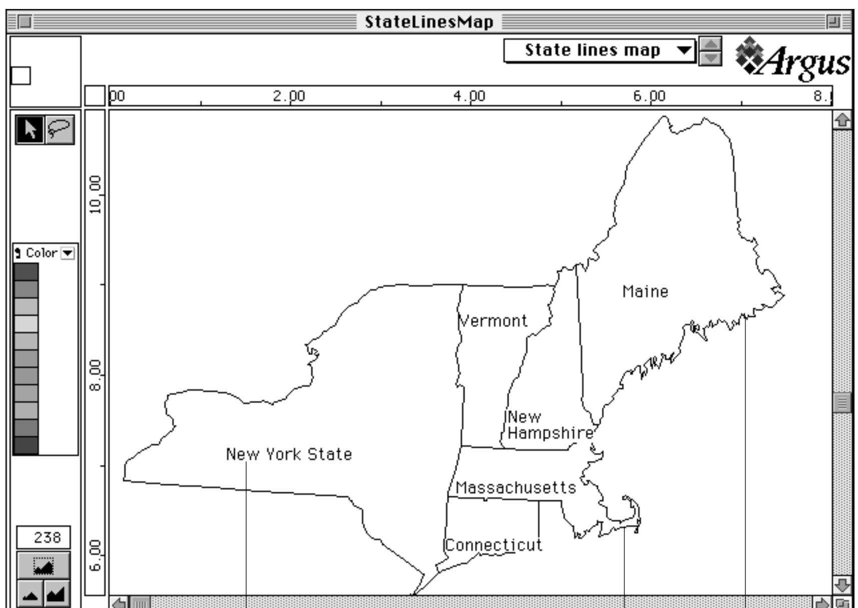



As Argus ONE imports the file, a ***progress*** dialog box appears to indicate that a time consuming operation is in progress.

*Progress dialog box*

*Press the keyboard sequence to halt import. The keyboard combination on Windows and Unix platforms is* CTRL*+C*




*A Argus ONE window with a maps layer containing an imported DXF file of some state lines.*

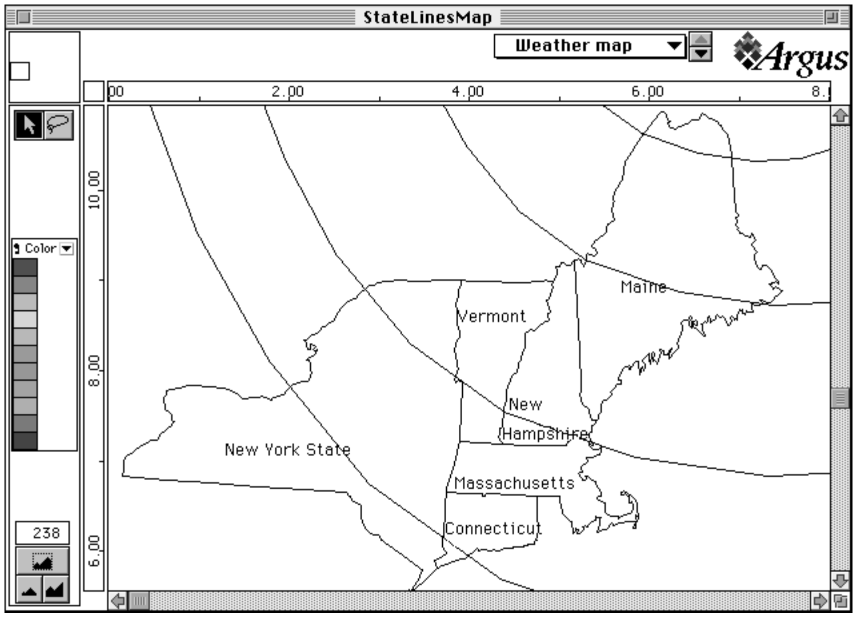
A text graphic object

Polyline graphic objects

## To import additional DXF files

If you have more than one map you need to import, you can repeat steps 2 and 3 for as many DXF files as you have, or you can create additional Maps layers and import the different DXF files into different Maps layers. Using a number of maps layers will enable you to control the visibility of the different maps.

For instance, if you need to bring in a state line map of the area and a weather fronts map, importing them into separate maps layers will enable you to present both of them, or just one of them.




*A Argus ONE window with a weather front map overlaid on the state lines map.*

### To paste contours in a Maps layer

You can copy any number and types of Information contours from information layers and paste them into a Maps type layer.

1. In an information layer, select the contours.
2. From the Edit menu, choose Copy.
3. Activate the maps layer by clicking the appropriate arrow in the *Layers Selection Button* or choose it from the Layers popup menu.
4. From the File menu choose Paste.

You can also copy contours from other project windows.

### To import a Argus ONE generic contour file

1. While in a Maps type layer, from the File menu, choose Import Text File...
2. In the Open dialog box, choose the file and click the Open button.
   The file is imported into the active Maps layer.

To learn about Argus ONE contour file generic format, how to import, copy, and paste contours, refer to chapter 2 “Working with Information Layers.”

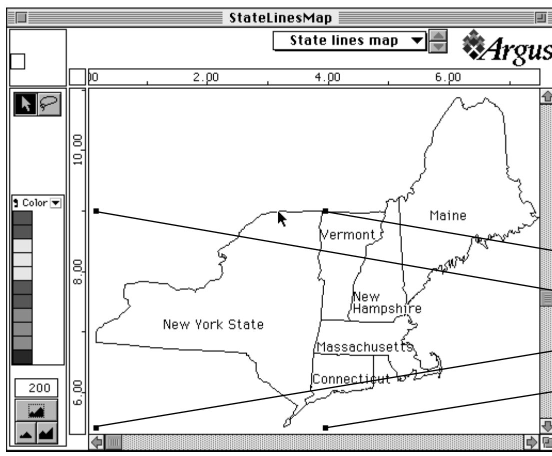
### Coordinate system considerations

When you import and paste graphic objects from sources external to the current project, you must take into account the possibility that the source and target coordinate systems might differ. They might differ in scale, units, coordinate system direction, drawing size, and origin. If they do, the objects you bring in might look different from what you expect, be completely out of the current drawing size, or too small to be noticed in the current zoom level.

### To select a graphic object

1. Make sure the layer in which you want to select the object is the active layer.
2. Place the cursor at some point along the object's outline and click the mouse button.
   Handles at the four corners of the object's enclosing rectangle are highlighted to indicate it is selected. You can move, delete and copy selected objects.

***Note:*** *If you are in a very high zoom level, the object's highlighted handles might be outside of your current view. To make sure the object is selected, zoom out to reveal at least one of its handles.*




*These four handles mark the four corners of the selected object's enclosing rectangle.*

### To select a number of graphic objects

1. Make sure the layer in which you want to select the objects is the active layer.
2. Place the cursor at some point along an object's outline and click the mouse button.
3. Hold down the SHIFT key while selecting the next object to add it to the current selection.
   Or Click-drag the mouse across the area you want to select objects in.

### To select all graphic objects in a layer

- From the Edit menu, choose the Select All command.

### To delete a selection of graphic objects

- From the Edit menu, choose the Clear command or hit the DELETE key.

### To copy graphic objects from a Maps layer

You can copy graphic objects from a maps layer and paste them into other maps or information layers, either in the same project or in another project.

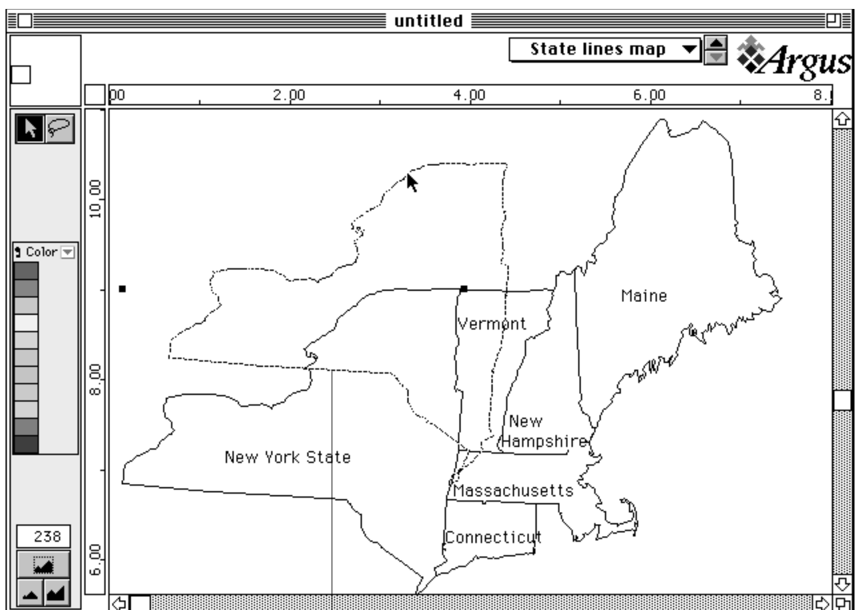
1. Select the graphic object or objects you want to copy.
2. From the File menu, choose Copy.
   The objects are stored in Argus ONE's internal clipboard.

### Moving graphic objects in a Maps layer

You can move graphic objects anywhere within the drawing size.

1. Select the object or objects to be moved by clicking on its borders.
2. Click-drag the mouse to the new location.
   An outline of the picture's rectangle follows your mouse movements.
3. Release the mouse button to relocate the objects.

*A graphic object on the move.*




*The object's enclosing rectangle outline.*

---

**Important note:** To undo moving an object, select the Undo Drag command from the Edit menu, to redo it after undoing it, select the Redo Drag command from the Edit menu. You can undo and redo the move, paste, import and delete commands.

---

## Setting Your Layers

Until now you have learned how to control the planar view of the Argus ONE workplace. The layers, stacked one on top of the other, create the third dimension of the workplace. The layers are containers of spatially distributed information, and at the same time are information records which can hold many information fields, which are referred to as layers' parameters.

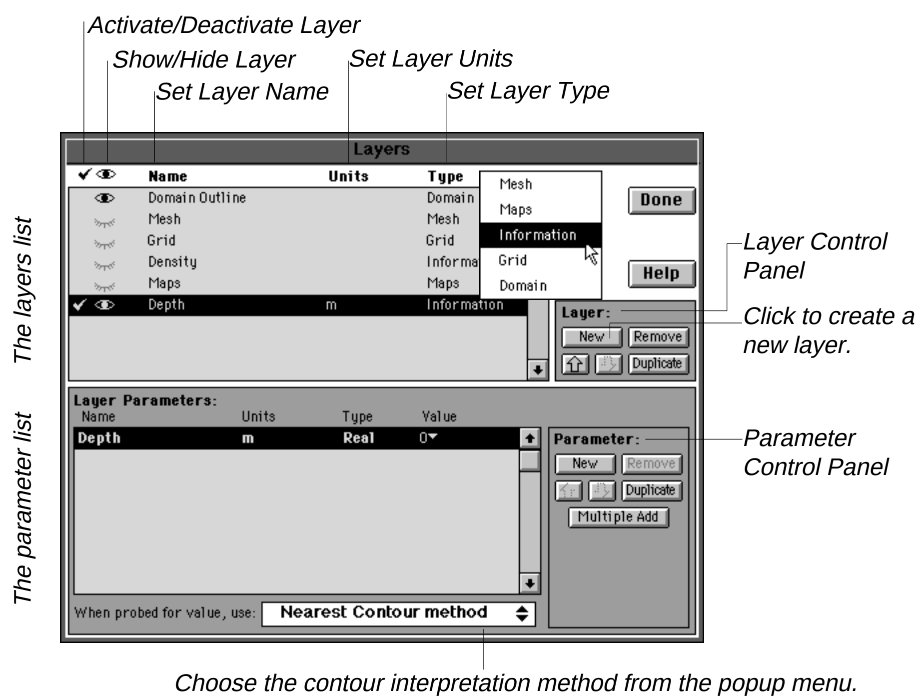
Setting and controlling your layers is performed in the Layers dialog box. You can create, edit, delete and order your layers in this dialog.

### To open the Layers dialog box

- From the View menu, choose the Layers...command
  Or -
  In the Information ruler, click the Layers Selection popup menu and choose Layers...

The Layers dialog appears.

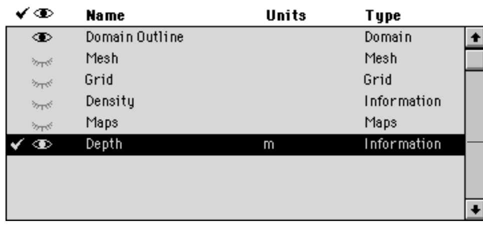
*The Layers dialog box is where you define layers, set the active and visible layers, and define the link between layers.*




The Layers dialog box is constructed of two main parts. The upper part, containing the layers list and the layer control panel, and the lower part, containing the parameters list and the parameter control panel. Like in any other Argus ONE window or dialog, to operate on a layer or its parameters, you must first select it.

### To select a layer

- In the layers list, click anywhere within the layer line.
  The line is highlighted to indicate it is the active layer.

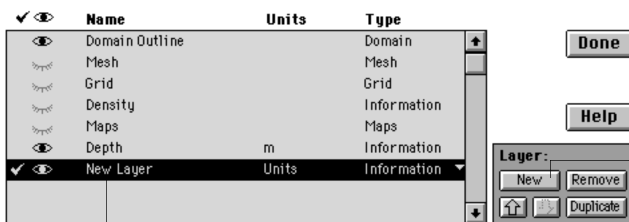



*A selected layer's line.*

### To create a new layer

When you create a new layer, its line is entered after the line that is selected when you create it.

1. Click the mouse anywhere in a layer's line.
2. Click the New button in the layers control panel.
   A new layer is created and highlighted.




*Click the New button to create a new layer.*

*A newly created layer.*

The newly created layer appears below the layer that was highlighted when you clicked the New button.

### To change the layer's name

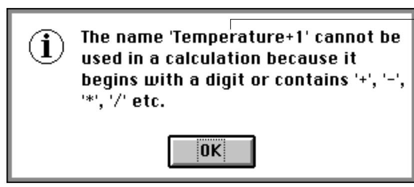
**Important note**: *You can not create or rename a layer to a name that is already assigned to another layer.*

When you create a new layer it is assigned an initial name. The initial name is always the string "New layer" unless there already exists a layer in that name. In this case the string "New Layer" is suffixed by the next available digit, starting from 1. For instance, if there exists a layer by the name "New Layer", a newly created layer will automatically be named "New layer 1". This insures that each layer has a unique name.


- Start typing the layer's name.
  Or -
  Click the line within the name field and type the layer's name. The cursor changes into an I Beam cursor to indicate you are in text edit mode.

A layer's name is used by Argus ONE to show you the current layer's name in the information ruler. When you export your mesh and grid with the related data from information layers, you can instruct Argus ONE to add the layer name at the top of the corresponding data column. Although Argus ONE does not limit the name length, do not use very long names as they might not fit in the window title.

Although a layer's name can contain any character, if you use a digit as the name's first character, use punctuation marks, or use the numeric keys such as the plus sign, Argus ONE will not be able to use this name in an expression. If you create such a name the application brings up an alert to warn you, but allows you to keep the invalid name.




*This layer name is invalid for use in expressions.*

## To set the layer's units

The layer's units can be any string you choose.


1. If your cursor is in the name field, hit the TAB key to move to the Units field.
   Or -
   If in another layer's line, double-click the Units field in the layer's line you wish to change its units.
2. Type in the units name.

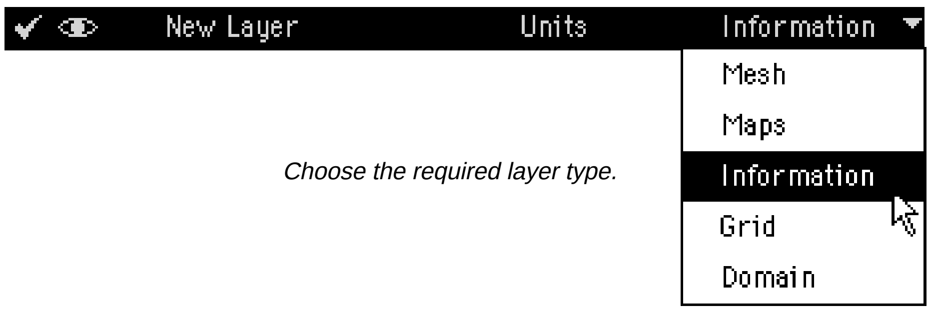
Since the units you choose appear in a text box attached to every contour you create, concatenated to the value you specify for the contour, keep the units' name short so they do not clutter your workplace. The units' description appears also in the Contour Information dialog box.

Some times you will need to describe the same parameter in few user layers. For example, you might be describing the permeability of two formations. In such cases, although you might name both layers' units as "md", you need a way to differentiate between the two on the screen. To do so, add a suffix such as the letters 1 and 2 to the "md" string.

### To set the layer's type

After creating a new layer, and until you click the Done button, you can change a newly created layer's type. When you first create a layer its type is set to an Information type layer. The small triangle at the right corner of the type field indicates that you can still change its type.

1. Click and hold the mouse button in the Type field.
   A menu with the available types pops up.
2. Choose a type and release the mouse button.
   The layer's type is changed.




*Choose the required layer type.*

### To show and hide a layer

When you create a layer, it is visible.

- To Hide a layer, click the Eye icon.
  The eye is shut, to indicate the layer is hidden.
- To Show a layer, click the Eye Lashes icon.
  The eye is open to indicate the layer is visible.


### To turn a layer active

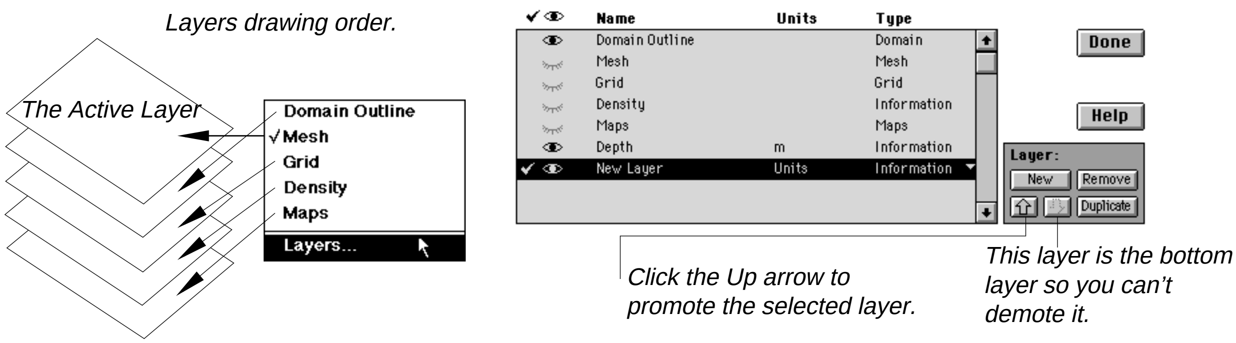
An active layer is the front most layer in which you can work.

- Click the left most field column.
  A √ mark is turned on to indicate the layer is active.


Only one layer at a time can be active in a project window. You can not turn a layer active if it is hidden, you have to make it visible first.

## To change layers' order

You can set the order in which layers are listed in the layers list. The order you set in the layers list is also the order in which the layers appear in the Layers Selection menu and in the Show/Hide layers sub-menu. Layer ordering also defines the order in which they are drawn on the screen. Layers are drawn after the active layer starting from the first in the list.




## To remove a layer

Removing a layer deletes the layer and its contents from the project.

1. Select the layer to be removed.
2. In the layers control panel, click the Remove button.
   The layer is deleted.

If the layer you try to remove serves as a Domain or Density layer to one of your mesh or grid layers Argus ONE alerts you and does not allow you to remove the layer.


## To duplicate a layer

To shorten layers' definition Argus ONE enables you to duplicate a layer.

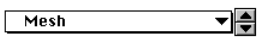
1. Select the layer you wish to duplicate.
2. In the layers control panel, click the Duplicate button.
   The layer and all its parameters are duplicated.

To keep layer naming valid, Argus ONE adds a suffix digit to the name of the duplicated layer. If the originating layer's name has a suffix digit, the digit is incremented to the next available digit.

## Moving Between Layers

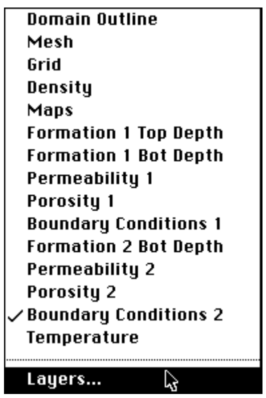
Moving between layers enables you to set the active (front most) layer. The active layer is the layer that is currently above all other layers and in which you can operate on objects.

There are three techniques you can use to set a layer to be the active layer:




1. Click the up or down arrow of the Layer selection button.

   Using the Layer selection button enables you to scroll between the visible layers. If no visible layer is available, The arrow is dimmed.




2. Choose the layer's name from the Layers selection popup menu in the information layer.

   Hidden layers' names are dimmed. The active layer is marked with a check mark to its left. The active layer's name appears in the Layers selection popup menu.

3. You could also open the Layers dialog box and set the layer to active by checking its active check mark.

## Showing and Hiding Layers

Since Argus ONE enables you to view and manipulate large amounts of information, your workplace may become cluttered. To allow you to have a clear view of your work and to concentrate on different sets of data at different times, Argus ONE supports many tools for information hiding.

Argus ONE enables you to fully control the types of data you want to see. You can control layers' hiding as well as hiding of the various objects in a layer. The View menu is where you determine what is shown and what is hidden. The following paragraphs describe layers' hiding. To learn other information-hiding techniques, refer to the chapters "Working with Information Layers", "Editing and Refining a Mesh" and "Creating and Editing a Grid."

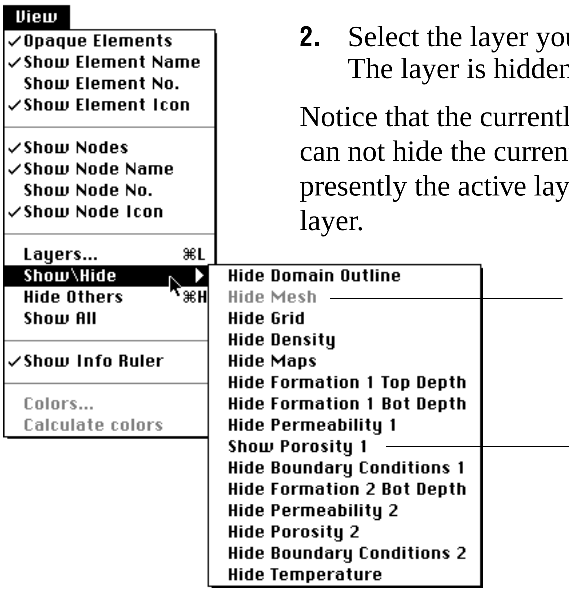
As described before, you can set the active layer and the visible layers in the Layers dialog box. However, to quickly hide and show layers, you should use the following techniques:

### To hide a layer

1. From the View menu, Choose the Show/Hide popup menu. The visible layers are marked as Hide “Layer name”, and the invisible ones are marked as Show “Layer name”.

2. Select the layer you wish to hide.
   The layer is hidden.

Notice that the currently active layer (Mesh) is dimmed, that is because you can not hide the currently active layer. If you want to hide a layer that is presently the active layer, you must first set another layer to be the active layer.




*The active layer, Mesh layer, is dimmed because you can not hide the active layer.*

*The hidden layer (Porosity 1) says: “Show Porosity 1”. The Show/Hide prefix is changed to reflect the visibility status of the layer.*

The layers you hide are temporarily removed from the workplace and their names appear dimmed in the View menu. When you “scroll” between layers, using the layer selection button (in the information ruler), the hidden layers are skipped.

### To show a layer

From the View menu choose the Show/Hide popup menu and select “Show Layer” for the layer you wish to be visible.

## Show All and Hide Others

### To view only one layer

- From the View menu, choose Hide Others.
  All layers except the active layer are hidden.

### To show all layers

- From the View menu, choose the Show All command.
  All layers are shown.

## Using the Layer Selection Button to Show and Hide Layers

Since you will mostly be using the Layer selection button in the Information ruler to change the active layer, Argus ONE provides you with some shortcuts that enable you to combine layer selection and hiding.

The following table describes the four available techniques for moving, hiding and showing layers using the Layer selection button.

| Use this keyboard combination while clicking the Layer selection arrows | To move to |
|---|---|
| Click | The next visible layer. |
| SHIFT+Click | The next layer, either hidden or visible. If it's hidden, it becomes visible. |
| OPTION + Click (Macintosh)<br>ALT + Click (Other platforms) | The next visible layer and hide all others. |
| OPTION + SHIFT + Click<br>ALT + SHIFT +Click (Other platforms) | The next layer, either hidden or visible and hide all the others. |

## Keyboard Shortcuts for Moving, Showing and Hiding layers

*Use Command-Down or Up Arrow to move between layers.*

The speed up your work, Argus ONE enables you to use keyboard shortcuts of the above described methods for moving, hiding and showing layers.

Simply replace every Click in the above description, by holding the COMMAND + Up or Down-arrow keys (Macintosh) or the CONTROL + Up or Down-arrow keys (Other Platforms). For example, instead of clicking in the layer selection button to move between layers, hold down the COMMAND key and press the down or up-arrows on your keyboard. For a detailed description of these shortcuts, refer to Appendix A.

## Inactive Layers and Selected Objects

Object selection, and hence object manipulation, is possible only in the active layer. When you change the active layer, the selected objects' selection highlighting in the layer you leave is turned off, and the selected objects in the activated layer are highlighted. Argus ONE does not forget the selected objects in a layer you deactivate, when you re-activate a layer, all selected objects in it are highlighted.

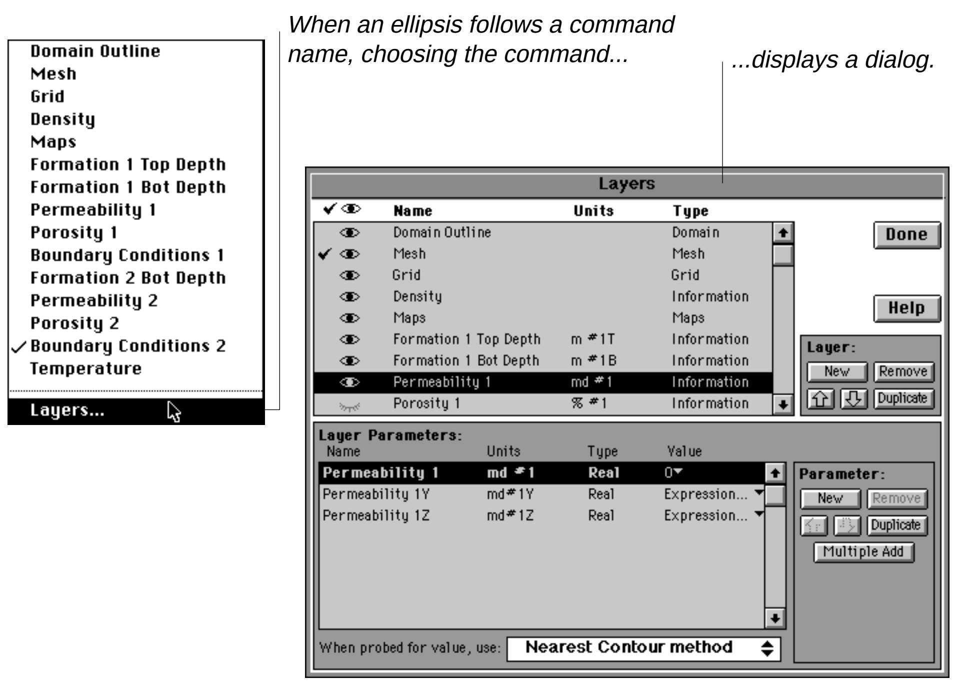
# Menus and Commands

Choosing a command tells Argus ONE what to do next – print a project, mesh a domain, export the mesh, and so on. Commands that carry out similar actions are grouped on a menu. For example, the File menu has commands that you use to open, print, and save your projects. The menus are listed on the menu bar across the top of the Argus ONE screen.

To see the commands on a menu, point to the menu name and press the mouse button. The menu drops down, displaying the commands. To scroll through open menus, drag across the menu bar with the mouse.

To use some commands, you must first select the objects you want the command to act on. These commands, dimmed on a menu, are unavailable until a selection is made. For example, you can't choose the Goto Node... command on the Navigation menu until you have created at least one element.

Argus ONE carries out some commands right away. If more information is needed to complete a command, Argus ONE displays a dialog box. You select options in the dialog box to control how the command is carried out.



## Choosing Commands

You can choose commands from menus or press the shortcut keys assigned to commonly used commands. On the menus, the shortcut key is shown to the right of the command.

You can choose some commands by clicking or double-clicking certain areas of the project window. For example, clicking the Zoom In button is a quick way to choose the Zoom In command. For a complete list of key and screen shortcuts, see Appendix A, “Keyboard and Mouse Shortcuts.”

## Undoing Commands

Undoing commands is a very powerful tool. It enables you to carry your work without worrying too much about simple mistakes you make. If you regret moving an object, resizing or deleting it, you can issue an undo command. The undo command is always available for your last action.

### To undo a command

- From the Edit menu, choose the Undo command.
  Or -
  On the Macintosh, use the COMMAND + Z keyboard shortcut.
  On all other platforms, use the CONTROL + Z keyboard shortcut.

### To redo an undone command

You can also redo the command you have just undone.

After you used the Undo command the Undo menu item changes into the Redo menu item.

# Mouse Cursors and Techniques

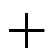
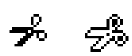
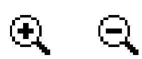
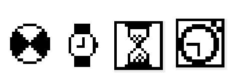
In Argus ONE’s workplace the mouse cursor changes to reflect the different tools you are using, to indicate that a time consuming process is in progress and to give you feedback about objects it lies above.

## Cursor Shapes

When you point the mouse to different parts of your screen, the cursor's shape changes, signifying the different tasks you can perform and reflecting its position above different objects. Some of the tools from the tool palette also change the cursor shape; these cursors are noted in the discussion about related commands.

If the cursor assumes a shape you don't want to use – for example, you accidentally clicked the Magic Wand, changing the cursor to a Magic Wand – click the mouse in the Arrow cursor tool in the tool palette to restore the cursor to its usual shape.

The following table lists the common cursor shapes.

| Cursor shape | Significance |
|---|---|
|  | This is the default cursor. The pointer is in the menu bar, window title, tool palette, scroll bars or the working area. Use this cursor to click or drag items or choose commands. When above nodes, it becomes hollow to make it easier for you to select a node. |
|  | When you select the Lasso tool the cursor changes its shape to reflect the current tool. |
|  | When selecting the Manual Element Creation tool, the cursor changes to this shape. If it lies above an existing node it changes to a hollow cross. |
|  | Selecting the Contour Creation tool, changes the cursor's shape to the following shape. |
|  | Selecting the Detach tool, changes the cursor's shape to scissors . If it lies above a node it turns to hollow scissors. |
|  | Selecting the Magic Wand, changes the cursor's shape to a Magic Wand. If the wand does not lie above a domain outline contour, it looses its magic and dims out. |
|  | Macintosh: Holding down the OPTION key, changes the arrow cursor to a magnifying glass. Holding the OPTION and SHIFT keys turns it to a Zoom Out tool.<br>Other Platforms: use the ALT key instead the OPTION key. |
|  | Argus ONE is performing a timely task. |

## Basic Mouse Techniques

The following table provides a review of basic mouse techniques you'll need to know to work in Argus ONE. For more information and practice using these skills, see “Your First Argus ONE Project” earlier in this manual.

| To | Do this |
|---|---|
| Point | Position the mouse cursor on or next to something. |
| Click | Position the cursor, and then quickly press and release the mouse button. |
| Double-click | Position the cursor, and then quickly press and release the mouse button twice. |
| Drag | Position the cursor. Press and hold down the mouse button as you move the cursor to the desired position. Then release the button. You often drag something to a new location or drag through a list to select an item. |

## Constraining Cursor Movements

You can constrain cursor movements during the creation and editing of objects.

### Angle constraints

When you create an element or a contour, you can constrain the angle of the current line segment in steps of 15° starting at 0°. During element creation you can constrain the first and second sides.

- To constrain a line segment to 0°, 15°, 30°, etc., press SHIFT and drag at the approximate angle.

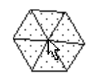
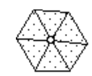
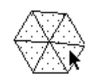
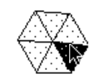
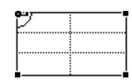
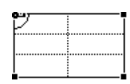
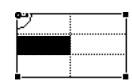
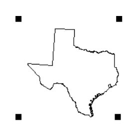
## Selecting and Opening Objects

Argus ONE's workplace contains the objects you create. These are nodes, elements, grids, blocks, contours and maps layer objects. You can reshape, delete and move each of these objects.

To operate on objects you must first select them. When you select objects, Argus ONE changes their status and indicates they are selected by highlighting them. Argus ONE highlights different objects in different ways.

Since Argus ONE's objects carry not only shape information, you can open objects to view and set the object's related information. The following paragraphs describe selecting and opening techniques.

## Selecting objects

| To select: | Method | Visual Feedback | |
|---|---|---|---|
| | | deselected | selected |
| Node | Bring the cursor over the node and click. The node is highlighted to indicate it is selected. |  |  |
| Element | Bring the cursor over the element and click. The element is highlighted to indicate it is selected. |  |  |
| Grid | Bring the cursor over the grid and click. Handles at the four grid corners are highlighted to indicate it is selected. |  |  |
| Block | Bring the cursor over the block and click. The block is highlighted to indicate it is selected. |  |  |
| Contour | Bring the cursor over the contour and click. The contour's vertices are highlighted to indicate it is selected. |  |  |
| Maps object | Bring the cursor over the graphic object and click. Handles at the four object corners are highlighted to indicate it is selected. |  |  |

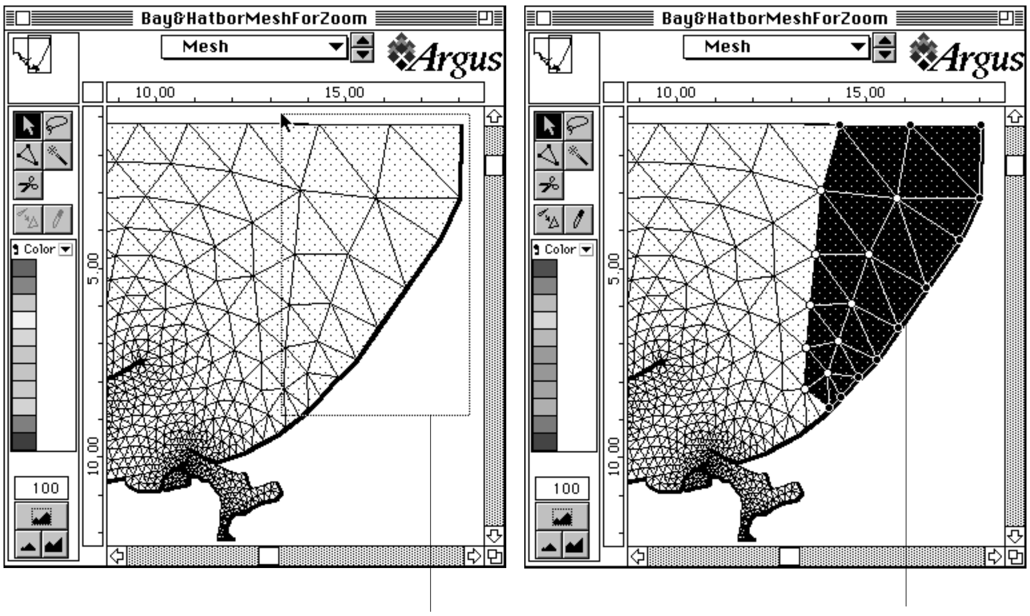
When you need to select a group of objects you can use the Stretch-Band tool or the Lasso tool. The former is a quick selection tool, the latter enables you to create complex selection regions.

### Selecting multiple objects using the Stretch-Band

- Select the Arrow tool from the tool palette.
- Click-drag the cursor to create the Stretch-Band.
  An outline of the selection rectangle follows the cursor.

***Important note***

*To start the Stretch-Band within the mesh, hold down the SHIFT key while clicking the mouse button.*




*The Stretch-Band rectangle.* *Multiple elements and nodes are selected.*

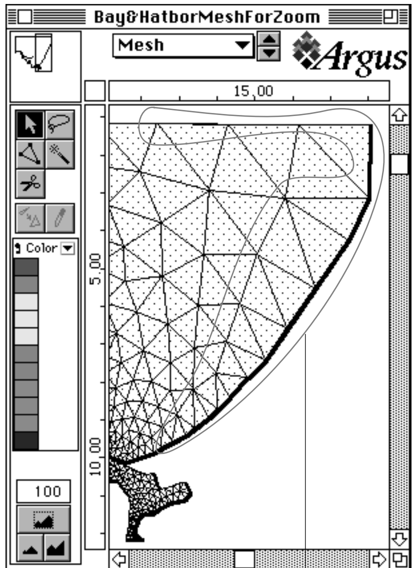
- Release the mouse button.

All objects enclosed by the Stretch-Band rectangle are selected.

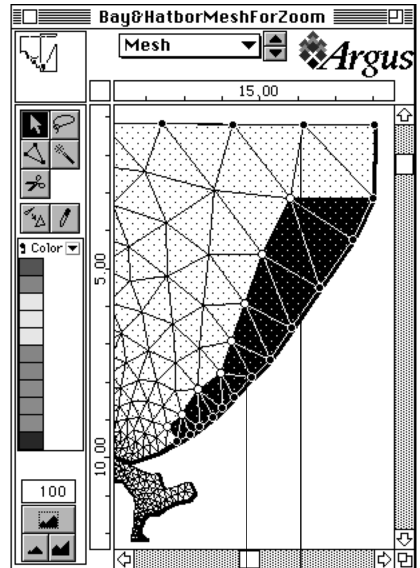
To start creating a stretch-band within a mesh, hold down the SHIFT key when clicking the mouse. After the stretch-band appears you can release the SHIFT key.

### Selecting multiple objects using the Lasso tool

- Select the Lasso tool from the tool palette.
- Click-drag the cursor to create the lasso. An outline of the selection region follows the cursor.



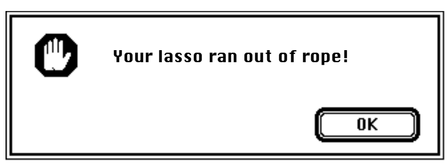

*Outline of a lasso selection region.*




*Multiple elements and nodes selection.*

- Release the mouse button.
  All objects enclosed by the lasso region are selected.

Argus ONE enables you to create very long lassos. However, they are not infinite. If you exceed the lasso limits, Argus ONE will present you with the following message:




It will then close the lasso and select the objects contained in it.

## Adding to, and removing from, the selection

To add or remove objects from the current selection, you can use the click, stretch-band or lasso tools while holding the SHIFT key.

## Clearing the selection

- To clear the selection, just select another object or click the cursor somewhere else within the window.

### Selecting all objects in a layer

- From the Edit menu, choose Select All to select all the objects in the active layer.

### Opening objects

All Argus ONE's objects carry additional information that you can access. For instance, a contour carries its value information, an element, a node and a grid block contain specific information and values from Information layers, etc. To access the object's related information:

- Double–click the object.

A dialog box containing the object's information is presented.

# Orienting in the Workplace

A typical project encompasses large amounts of data of different types and sources, as well as complex meshes or grids with thousands of nodes, elements or blocks. When constructing the project, as well as when iterating through the model calibration and realizations (predictions), one needs all the help he/she can get to orientate through the project.

Argus ONE furnishes you with many tools and capabilities to allow you to easily find your way within the project.

## Using the Rulers

Present at all times, are the rulers:

The coordinate rulers - including tick marks.

The Information ruler - showing the current mouse location, nodes, element and neighbors, contour's value, the current units, the active layer and the Navigation window.

## Using the Maps Layer

Usually the outline of the domain is not sufficient for orientation. Importing DXF digitized maps of cities, roads, rivers, etc. and drawing using the CAD tools available in maps layers, will help you construct a meaningful picture of the region under investigation.

## Using Node, Element and Block Names and Icons

To relate numerical data to the physical data, use the node, element and block names and icons. For instance, naming and adding a well icon to a node representing a well, will help you to visually find it.

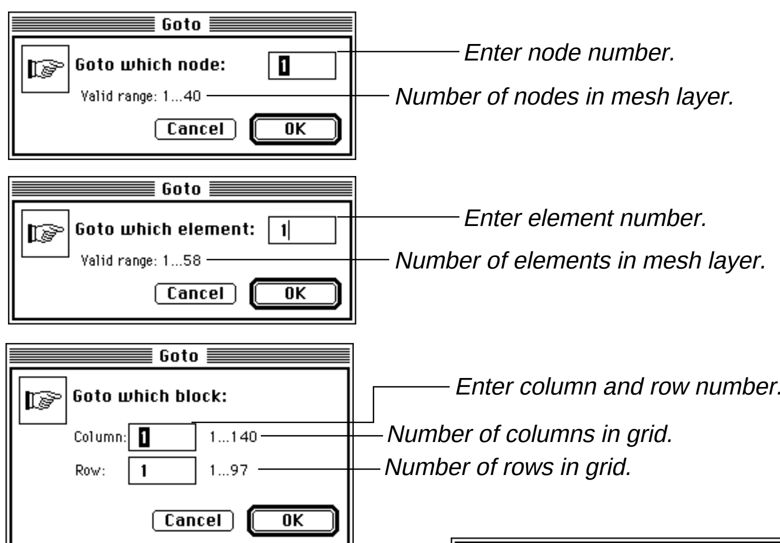
## Zooming

Argus ONE furnishes you with many zooming shortcuts. To focus on a specific area, or to find out where a specific area is located within the problem region, use these tools.

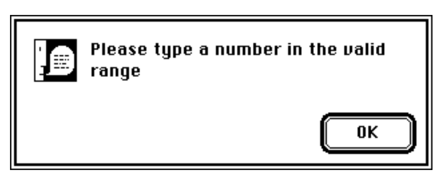
## Goto Node, Goto Element, Goto Block

To find a node, an element or a block in a grid, use the Goto commands. Each of these commands is available only in its respective layer.

1. From the Navigation menu, choose Goto Node... Goto Element..., or Goto Block....
   Argus ONE presents you with one of the following dialogs:




*If the number you enter is out of range, an alert explaining the problem is presented.*



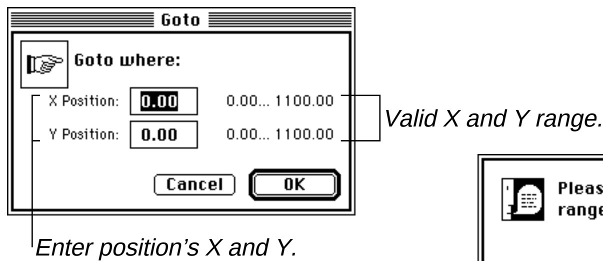

2. Click the OK button.

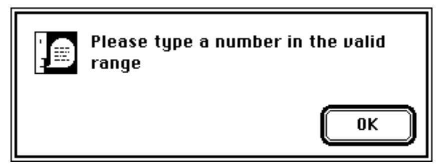
Argus ONE scrolls to the selected object, centers it in the window, and selects it.

## Goto Position

If you need to place an object at a position that is out of the current window you can use the Goto Position... command, instead of scrolling.

1. From the Information menu, choose Goto Position...
   Argus ONE presents you with the following dialog:




*Out of range alert.*

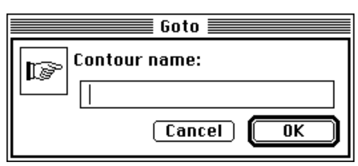
2. Enter the X-coordinate and hit the TAB key to move to the Y field.
3. Enter the Y-coordinate.
4. Click the OK button.

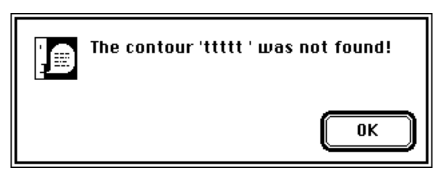
Argus ONE scrolls the position entered to the center of the window.

## Goto Contour

When in an Information or Domain type layer you can use the Goto Contour name command to find a contour. You need type only the first number of layers differentiating it from other contours.

1. From the Information menu, choose Goto Contour...
   Argus ONE presents you with the following dialog:




2. Enter the contour's full name or just few of its first characters.
3. Click the OK button.

Argus ONE scrolls the to the first created vertex of that contour and selects the contour. If the name you entered was not found, an alert is presented.

## Using the Search For Command

The Search For command can be used to navigate within a project. The objects answering the search criteria are selected. To read more about using the Search For... command refer to chapters 3, 7 and 9.

# Working with Information Layers

# Overview

Argus ONE enables you to incorporate quantitative information in three major methods; in Information type layers, Data type layers and on numerical objects such as mesh nodes and elements or grid blocks. Using the first two and linking numerical objects to the information stored in Information and Data type layers is the most recommended method. When using the latter, your data is lost every time you re-mesh/grid your domain.

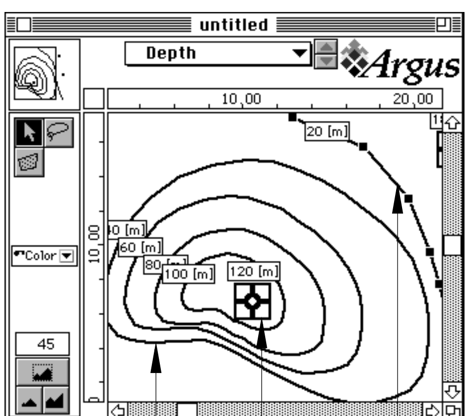
Storing your data in Information type layers is accomplished through the use of the Point, Open and Close Contour objects. In this chapter you will learn how to create, edit, copy, paste, import and export these objects. Using Data type layers is described in detail in the **Supplement** chapter.

In the next chapter you will learn how to link and manipulate information from different layers and create complex expressions. As you'll find out, these manipulations can save you considerable coding in your simulator.

Although contours are used in Argus ONE in two different contexts; the Information type layers, and the Domain type layers, they are created using the same techniques.

While in an Information type layer, contours are used to describe the spatial distribution of a parameter or a function, in Domain type layers, they are also used to define the outline of a domain to be descretisized. The outline of a domain must be a close contour of any shape and might include islands, open contours and point objects. The domain outline contour also carries information about the initial mesh and grid densities.

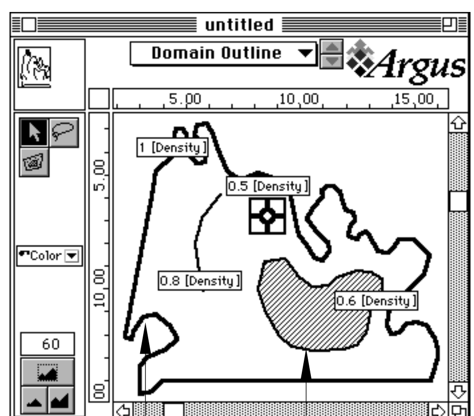
*Information type layer. Contours show their main parameter value.*




*Close contour* *Open contour*

*Point*

*Domain type layer. Contours show their main parameter value, the density.*




*The domain outline contour is marked by a heavier line.*

*Islands are marked by a hash*

## Contour Maps Objects

The three objects used in information layers' contour maps are the Point, the Open Contour and the Close Contour. Together they enable you to describe spatial information in three dimensions regardless of the method your data is made available to you.

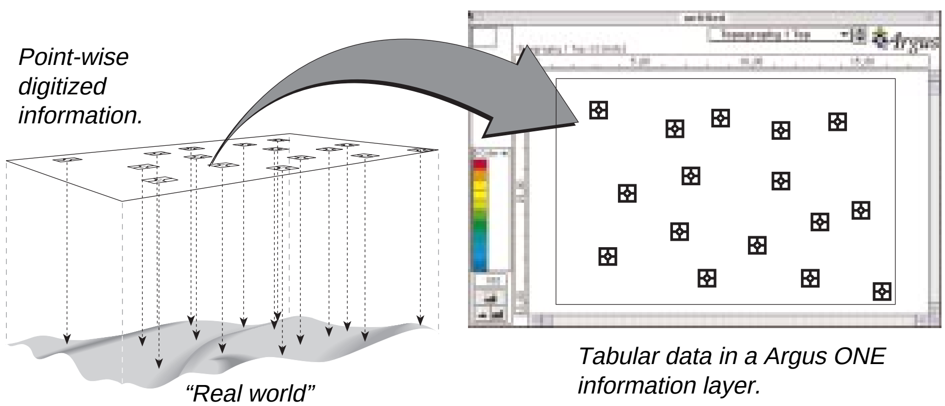
### Tabular information source

***Important note:***

*If your data is comprised of hundreds or thousands of observation points, import that information into a Data type layer and not into an Information type layer.*

Often, your data is made available to you in the form of a table. For instance, coordinates of observation or measurement points with their values. When you import tabulated data, it is stored in an information layer using the point objects. Your contour map will be a collection of points with values. If you wish to interpolate the data, Argus ONE enables you to do so.

Using point objects for tabular data should be used if you have a reasonable number of observation points. If however, your data was sampled on hundreds or thousands of points, or was created on a grid or a mesh you should use Data type layers to import that information into Argus ONE.

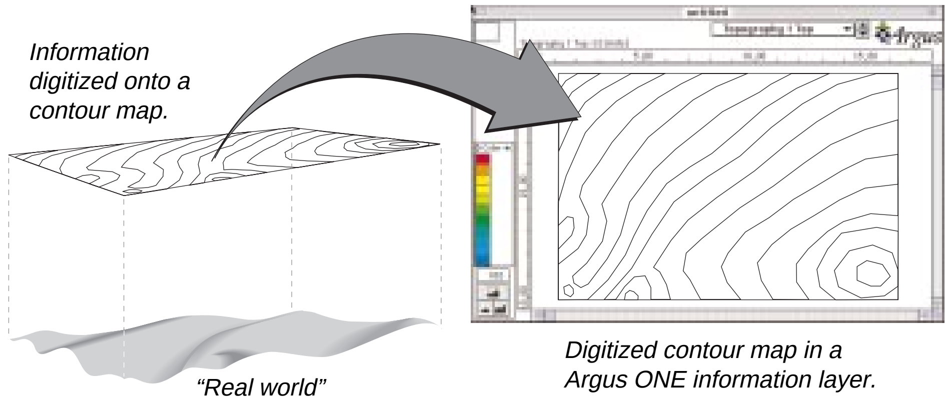



*Tabular data in a Argus ONE information layer.*

### Contoured data

Many times you do not get the original nodal data as recorded in the field, but after it has gone some manipulation, such as creating contours through equal value points. Such are topography maps, geological maps, parameters' distribution, etc. Such contour maps may contain point, close and open contours. This information is usually available in digitized format, or you can easily digitize it from a map using a digitizing table.

When you import digitized maps, your data is stored in an information layer using the point, close and open contours. Your contour map will be a collection of points and contours with the data associated to them.



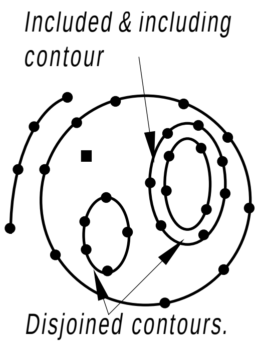

### Contours in a domain type layer

Close contours are used to describe the domain’s outline or border. Open contours are used to describe line entities such as rivers, faults, etc. Point objects can be used to describe point sources or point loads. Open contours will force elements to be created along them (element sides will not cross the open contour), and point objects will force a node through them.

## Contour Parameters and Values

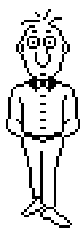
When you create information type or domain type layers, they are created with one parameter. This automatically-created parameter is where you assign a contour’s value. You can create as many parameters as you need for a layer, and they will act the same as the one automatically created. You assign each contour you create a value for each of its parameters. The automatically-created parameter, called the main layer parameter, has a special meaning in a domain type layer. It is named Density, and is interpreted during auto mesh generation and auto grid generation as the required average element and block size. On the meaning and use of mesh and grid density, refer to the chapter “Meshing a Domain” and the chapter “Creating and Editing a Grid.”

# Contours and Contour Maps Definitions

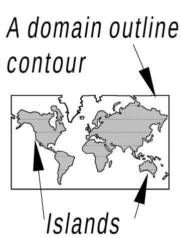



## The Rules

1. An Argus ONE contour is a point, a close or an open polygon made up of one or more vertices connected by straight line segments.
2. A layer with contours is a contour map.
3. A contour can not intersect itself.
4. A contour can not intersect other contours.
5. Close contours can contain other points and close contours, or be contained by other contours. They can also have disjoined contours.
6. Open contours must be disjoined from other contours.

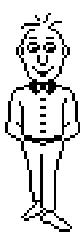


Domain outline contours are slightly different from all other contours. Their definition complies with the above rules except for the two last rules. A domain outline contour is further defined by the following rules:




1. A domain outline contour can not include or be included by another domain outline contour. Domain outline contours can only be disjoined.
2. A domain outline contour can include any number of open contours. They are used to mark domain lines such as faults, rivers and other internal boundaries.
3. A domain outline contour can have “islands” which are part of it.
4. Islands are contours contained by, and belonging to their domain outline contour. They are used to mark regions within the domain outline contour that you do not want to mesh or grid.

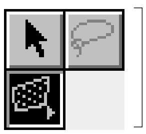
## Argus ONE Remembers these Rules for You



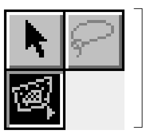
You don't have to memorize these rules. While you create, paste, import, and edit contours, Argus ONE performs validity tests of your actions to make sure that contours you create comply with the above described basic rules of contour maps.

Each time you violate one of these rules, Argus ONE beeps or opens an alert and does not allow you to complete the illegal operation.

# Creating Contours




*Close Polygon tool*




*Close Polygon tool*

## To Start Creating a Close Contour

1. Make sure the active layer is the layer you wish to create a contour in.
2. If you are digitizing the contour from a background picture in the maps layer, zoom-in so that you can easily digitize in the resolution you need.
3. Select the Close Polygon tool from the tool palette, it has a slightly different icon in domain type layers and information layers.
4. Click the cursor where you want to begin creating the contour.
5. Move the cursor to the point where you want to have the next vertex and click. Continue this step until you're ready to close the contour.

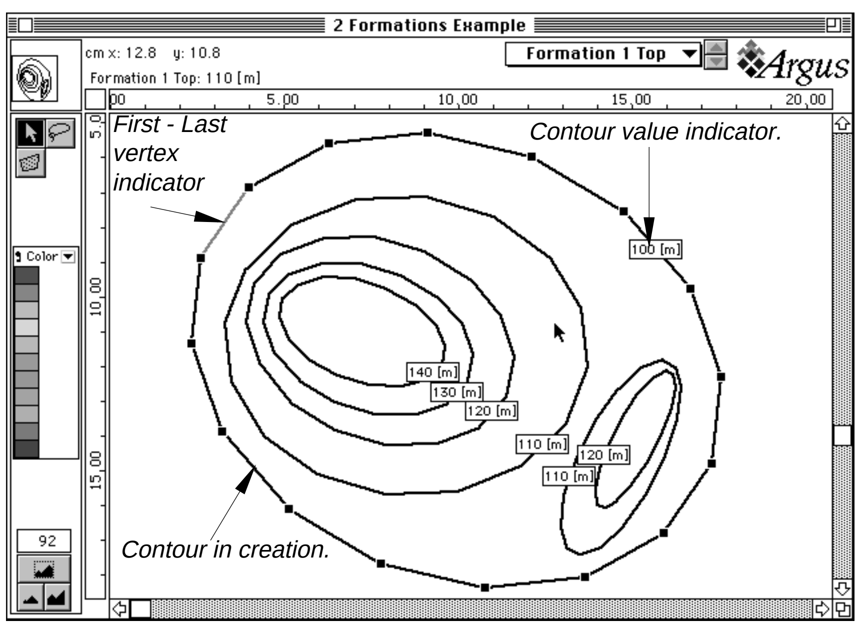
   A gray line connecting the first vertex with the last recorded vertex follows your cursor movements. This helps you remember where is the first vertex.
6. When you are ready to close the contour, double-click the last vertex.

---

**Important note:** It is irrelevant whether you record the contour clockwise or counter clockwise.

---

*Argus ONE checks the validity of your contour as you record it. If you cross the contour itself or another contour, Argus ONE beeps and does not record the offending vertex.*



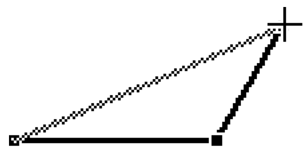

If you reach the end of the window, just move the cursor outwards and the window will scroll with you to reveal hidden areas of the drawing size.

## Constraining Cursor Movements

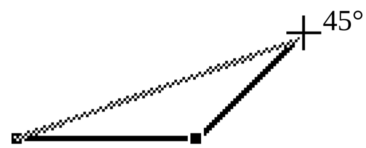
Constraining the cursor enables you to create contour segments at specified angles.

### To constrain cursor movements

To constrain a line segment to 0°, 15°, 30°, etc., hold down the SHIFT key and drag at the approximate angle.



*Without SHIFT*




*With SHIFT*

## Deleting Vertices while Creating a Contour

While you record a contour, you might need to change a vertex location, or delete it all together. You can delete vertices while recording a contour.

### To delete vertices during contour creation

- If you wish to backtrack, just hit the DELETE key on the keyboard and the last recorded vertex will be erased.

You can delete back as many vertices as you wish, until you finally delete the whole contour. If you do delete the whole contour, Argus ONE beeps to alert you.

## Closing the Contour

- When you're ready to finish, double–click the last vertex to close the contour, Argus ONE joins the last and first vertices.
  Or -
  Click the first vertex to close the contour.

## Assigning the Contour a Value

Once you have double clicked the last vertex, Argus ONE presents you with a dialog box asking you for the contour's value.

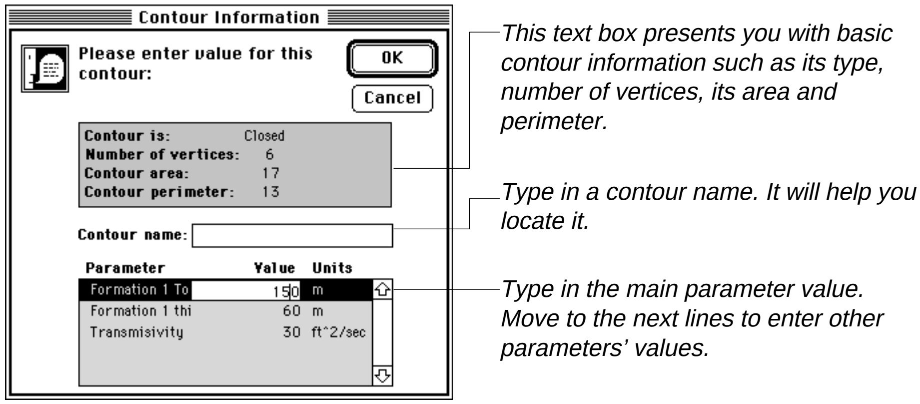
The initial value assigned to the contour by Argus ONE is the default value you assigned to the layer's main and other parameters when creating the layer.

### To assign contour values

**1.** Type the value in the text edit box.

***Important note:***

*If you click the Cancel button, the contour is deleted.*




**2.** Click the OK button. Clicking the Cancel button erases the contour.

You can also set the contour's value at a later time.

## Naming the Contour

It is good practice to name your contours. Naming them will help you find them later on, using the Goto Contour... command. You do not have to worry about cluttering your view, since you can hide and show contours' names and values at any time.

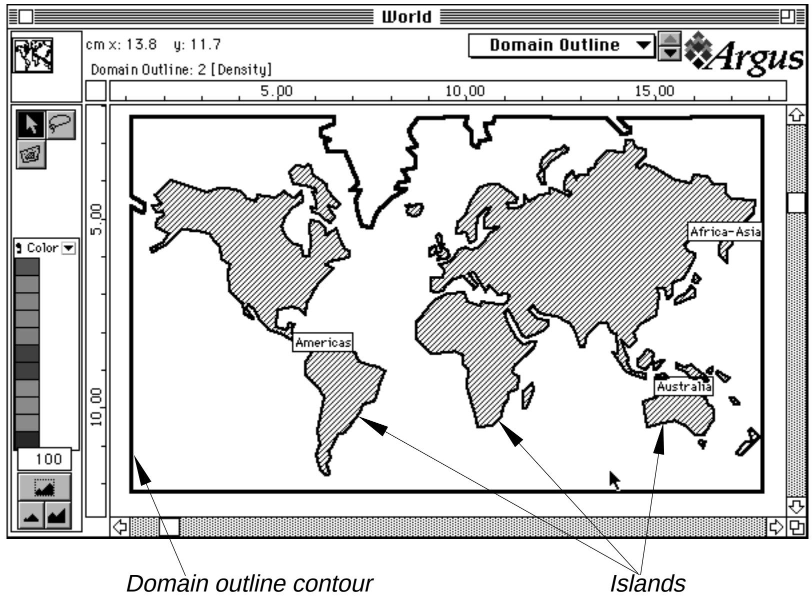
### To name a contour

- After you finished creating the contour, enter its name in the Contour name text edit box.

You can always change the contour's name or delete it all together as you will learn later in this chapter.

## Adding Islands to a Domain Outline Contour

Islands are parts of a Domain Outline contour. To create islands within the Domain Outline contour use the same techniques you use for creating contours.




When you finish creating an island, Argus ONE adds a hash pattern to the island to distinguish it from the domain.

## Creating Open Contours

### What are open contours?

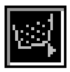
Open contours are contours that their first and last vertices are not connected.

### When to use open contours?

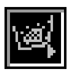
In many cases where you import or digitize contour maps, some of the contours might end at the map boundary. Argus ONE enables you to create open contours. Like in real maps, open contours can not be included within other contours.

However, in domain type layers, open contours can be created within a domain outline contour. You can use open contours within a domain outline contour to represent a fault, a river or any other internal boundary. When a mesh is created using a domain layer containing open contours, auto mesh generation creates the elements such that they do not cross the open contour. In a grid layer, you can sum up the number, value and length of open contours crossing a grid block.

### To create an open contour



*Open Polygon tool in Information type layers.*



*Open Polygon tool in Domain type layers.*

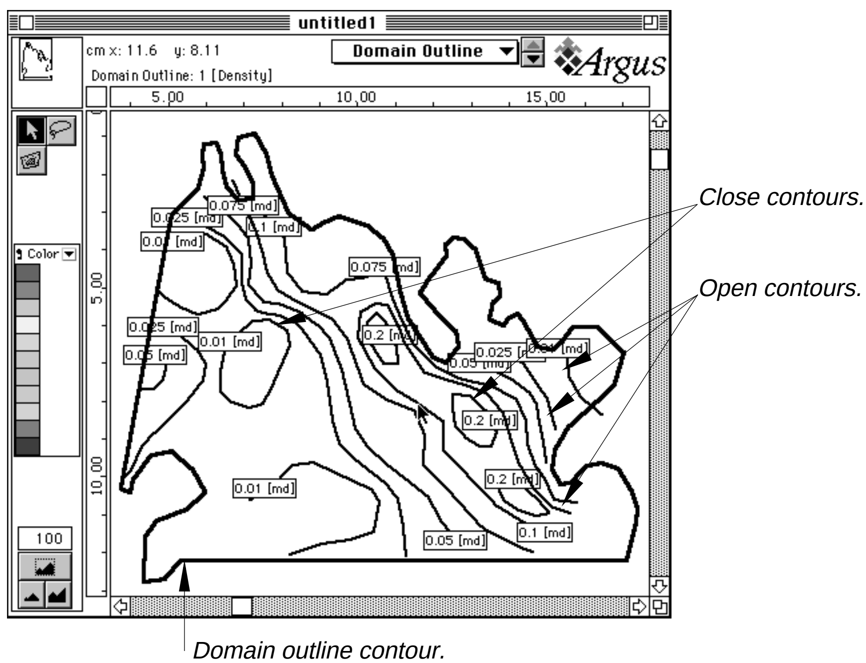
1. Click and hold the currently active polygon tool until the popup menu opens. Select the Close Polygon tool from the popup menu.
2. Use the same techniques you use to create a close contour.
3. To end the contour creation double-click the last vertex. The contour is left open.
   The Contour Information dialog box appears.

---

**Tip:** To create a close contour using the open polygon tool, click the first vertex again.

---

4. Enter the contour’s value.




*An aquifer domain outline overlaying permeability contours.*

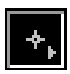
## Creating Point Objects

### When to use point objects?

As explained before, you will most often use point objects when your source of information is in a tabulated form. You may also have a mixed information source, combining a contour map and nodal information. For instance, some physical parameter distribution has been contoured and you also have additional highly accurate information from some observation points, such as wells, that you want to take into account.

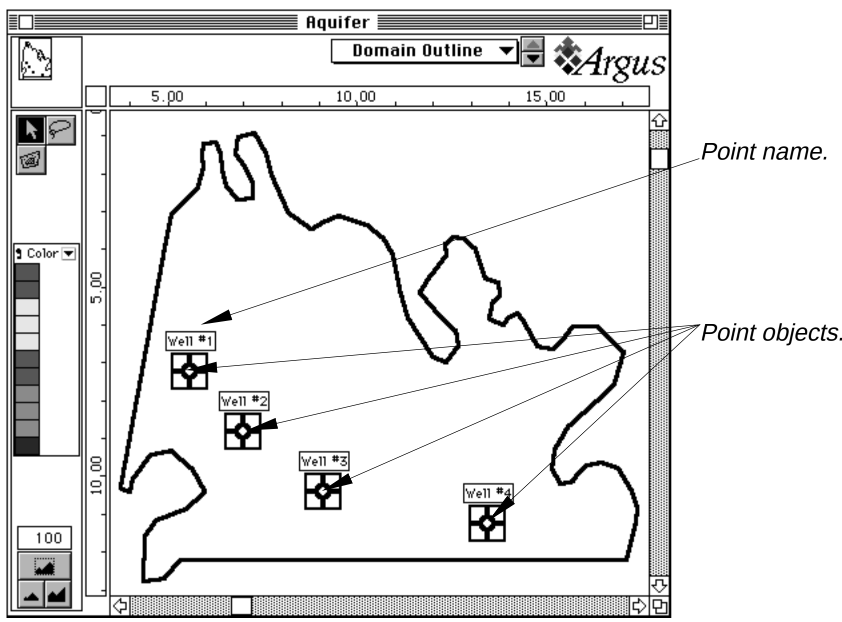
In domain type layers, point objects are used to describe the location of sources/sinks or point loads introducing mass, contaminant or load into the system. Point objects in a domain outline, will force the auto mesh generation to create a node above them. As you will learn later, this enables you to incorporate point boundary conditions types and values. In a grid layer, you can sum up the number and value of point objects contained a grid block.

### To create a point object



*The Point tool.*

1. Click and hold the currently active polygon tool until the popup menu opens. Select the Point object tool from the popup menu
2. Click the mouse where you want the point object to be. The Contour Information dialog box appears.
3. Enter the point's value.




## Editing Contours

Argus ONE enables you to edit contours. While editing the contours, Argus ONE performs validity tests of your actions to ensure that the contour map remains valid. Some additional editing options are described in the **Supplement** chapter.

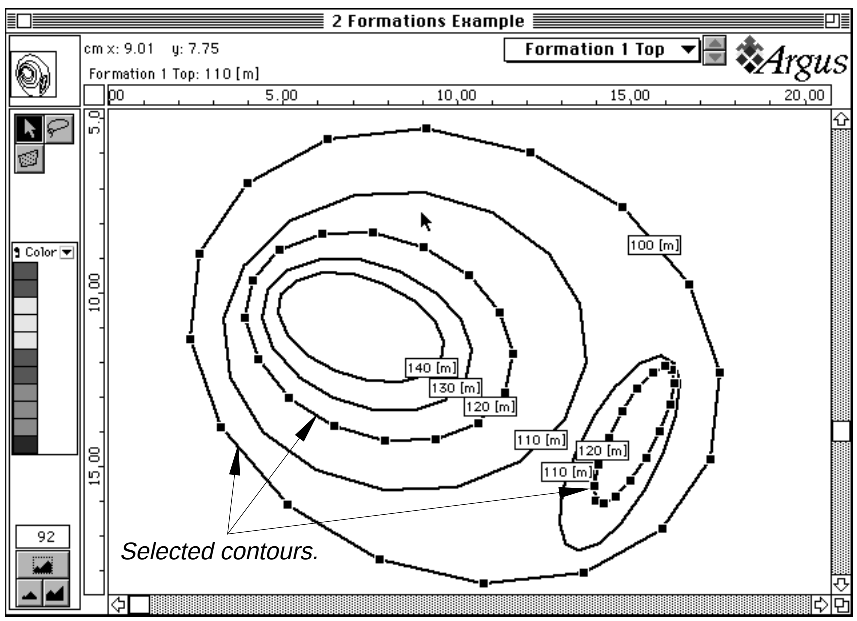
## Selecting Contours

If you wish to change a contour's shape, location or value, select it first.

### To select a contour

1. Make sure the active layer is the layer its contours you wish to select.
2. Using the Arrow cursor click within the contour. If the contour includes another contour, click in the area between the two contours.
   To select a point object or an open contour, click the object itself.

The vertices of the contour are highlighted as bullets to mark it's selected. A point object icon is highlighted.




*An information layer with selected and deselected contours.*

### To add a contour to the selection

- SHIFT-Click within or on the contour to be added to the selection.

### To remove a contour from the selection

- SHIFT-Click within or on the contour to be removed from the selection.

### To select all contours

- From the Edit menu, choose Select All.

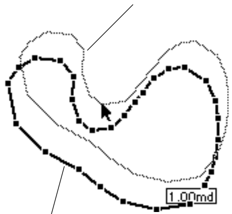
## Deleting Contours

- Select the contour or group of contours you want to delete and press the DELETE key on your keyboard.

## Moving Contours

### To drag contours

Contour's outline.




Contour's original location.

1. Select the contour or group of contours you want to move.
2. Drag the selected contours to their new locations.
   An outline of the contour follows your movements to indicate its future location.
3. Release the mouse button.

---

**Note:** You can not drag contours outside the drawing rectangle. However, if a contour was left outside the current drawing size you can drag it back into the drawing rectangle.

---

### Validity tests

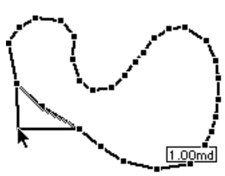
If one or more moved contours intersect another contour, Argus ONE beeps and relocates the violating contour to its original position.

---

**Tip:** If you want to move a contour outside of the current screen, just drag it outside the window borders, Argus ONE scrolls the window.

---

## Reshaping a Contour




*Shape-up.*

1. Select the contour.
2. Drag one of the bullets to move the vertex.

You can move the vertices in any direction. The vertex always remains between its two adjacent vertices.

### Validity tests

If one or both of the moved segments intersect another contour or the contour they belong to, Argus ONE beeps and relocates the violating vertex to its original location.

## Changing the Contour's Values and Name

Changing contours' values is a powerful tool. If you run your model on different realizations, defined by different initial conditions or parameters that you have described using information layers, you can easily change the contours' values to take into account the new realization.

### To change a contour's values

1. Double-click within the contour.
   The Contour Information dialog box appears.
2. Type the new values and name in the appropriate fields.
3. Press the OK button.

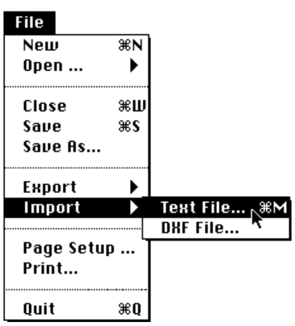
# Importing Contours

Argus ONE enables you to import Point, Close and Open contours. If you have digitized data of contour maps such as domain boundaries, initial conditions distribution, or material distributions, you can import them directly into the relevant layer. You can import contours to all the layers with the exception of the mesh and grid layers. You can also import DXF files into information and domain type layers.

The contours you import must comply with all the above described rules regarding the creation of contours at each of the different layer types. You should also make sure that the imported contours are in the correct scale, units and coordinate system orientation.

## To Import Contours

You can import contours from files saved in Argus ONE generic contour file format (Text File...), or directly from DXF files (DXF File...).




### To import contours from a generic file

1. Activate the layer you wish to import contours to.
2. From the File menu, choose Import Text File...
3. Use the standard Open file dialog box to choose the file containing the contours.
4. Click the OK button.

Argus ONE draws the contours on the screen.

### To import contours from a DXF file

Importing DXF files is supported only for the polyline and line DXF objects, for further reading refer to chapter 1 “The Argus ONE Workplace.”

## Contours' Data Format

Argus ONE supports the following data formats for importing contours. The data can be only in ASCII format.

### General description of a contour in an import/export file

A contour is described by: its name, the number of its vertices, its parameters' values and a list of the X and Y coordinates of the vertices. Data fields are delimited by one character — "The delimiter". The delimiter can be any character but: a digit, period, "e" or "E".

### The first line of a contour's description

- Number of vertices Delimiter Contour's main parameter value Carriage return

The contour's value field is optional. You can have additional parameters' value fields, they should appear after the main parameter value field. If the layer you import the contours into, has less parameters assigned to it than the number of parameter value fields in the file, the additional ones will be skipped.

### A line describing a vertex

- X Vertex coordinate Delimiter Y Vertex coordinate Carriage return

These lines are repeated for every vertex in the contour. They are ordered in their drawing order such that the previous line holds the current vertex predecessor vertex and the following line its successor.

### The last line of a contour description

- Carriage return (an empty line)

### Multi contours file

A file may contain any number of contours, mixing open contours, close contours and points. After the last line of a contour's description (the empty line) a new contour may be added.

### Close, Open and Point contours representation

An open contour with n vertices is described by n lines of pairs of vertices' coordinates. In contrast, a close contour of n vertices is described by n+1 lines of pairs of vertices' coordinates where the last line repeats the first vertex. This is how Argus ONE distinguishes a close contour from an open one. An open contour with one vertex only, describes a point object.

### An example of a file ready for import

The following table presents an example file that can be interpreted by Argus ONE as a valid import contours file. This example describes three contours, the first being a close contour, the second an open contour and the third a point.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 13 | tab | 100.5 | tab | 22.4 | tab | 201.5 | ↵ |
| | 7.6552777778 | tab | 3.0691666667 | ↵ | | | | |
| | 4.5155555556 | tab | 1.5169444444 | ↵ | | | | |
| | 3.81000 | tab | 2.5047222222 | ↵ | | | | |
| | 3.8805555556 | tab | 3.3161111111 | ↵ | | | | |
| | 4.2333333333 | tab | 3.9158333333 | ↵ | | | | |
| *Close Contour* | 5.0447222222 | tab | 4.5508333333 | ↵ | | | | |
| | 5.8208333333 | tab | 5.9266666667 | ↵ | | | | |
| | 5.5738888889 | tab | 6.66750 | ↵ | | | | |
| | 4.9741666667 | tab | 7.4788888889 | ↵ | | | | |
| | 4.7272222222 | tab | 8.8194444444 | ↵ | | | | |
| | 5.2916666667 | tab | 9.4191666667 | ↵ | | | | |
| | 8.6077777778 | tab | 6.35000 | ↵ | | | | |
| | 7.6552777778 | tab | 3.0691666667 | ↵ | | | | |
| | | | | ↵ | | | | |
| | 6 | tab | 85.5 | ↵ | | | | |
| | 9.4191666667 | tab | 1.2347222222 | ↵ | | | | |
| | 10.26583 | tab | 2.5752777778 | ↵ | | | | |
| *Open Contour* | 10.01888 | tab | 4.12750 | ↵ | | | | |
| | 10.37166 | tab | 5.5738888889 | ↵ | | | | |
| | 10.93611 | tab | 6.66750 | ↵ | | | | |
| | 10.12472 | tab | 8.6077777778 | ↵ | | | | |
| | | | | ↵ | | | | |
| | 1 | tab | 44.432 | ↵ | | | | |
| *A Point* | 12.4191666667 | tab | 11.2347222222 | ↵ | | | | |
| | | | | ↵ | | | | |

## A comment line

The file may contain any number of comment lines. A comment line starts with the # character. Argus ONE skips such lines. A comment line in which the # character is followed by a space and any number of characters is reserved for future use by Argus ONE.

---

**Important note:** In commenting your contour files, try to refrain from using the sequence #space. It might be used by future versions of Argus ONE.

---

The contour's name line for instance starts with the sequence #space.

### The contour's name line

The first line in a block of contours might be the contour's name line.

- # Name: Contour's name

The contour's name line is optional.

### Comment lines

When you export or copy contours, Argus ONE adds comment lines to the file. The following lines are an example of a file containing one close contour named "A Close Contour", having four vertices and four parameters including the main parameter.

```
# Name:A Close Contour↵
# Points Count,Value↵
4,1,22.33,44.55,55.66↵
# X pos,Y pos↵
5.50333,5.43639↵
8.60778,7.90583↵
10.8303,6.8475↵
5.50333,5.43639↵
↵
```

### Ordering domain outline contours

A file that is to be imported as domain outlines into the domain type layer may contain multiple domains. Argus ONE imports the contours one after the other, and tests them against all existing contours. If the current contour being read is disjoined from all other domain outline contours, Argus ONE imports it as a new domain outline contour. If the contour is contained within a domain outline contour it is being imported as an island of that contour.

Although you do not have to order the contours in the imported file, it might be easier for you to order the contours such that each domain outline contour is followed by the contours of its islands, that is the way Argus ONE orders the contours when you export or copy contours.

## Adjusting the Coordinate Systems

In case you need to import a set of contours that were digitized relative to a coordinate system with units other than those currently set, you can change the units of the project, import, and change back to the previous units. However, if the file being imported is relative to a different scale or directions, you must externally translate it before importing it.

If you do not adjust the coordinate systems prior to importing, and the imported data was digitized with respect to a different coordinate system than the project's current coordinate system, the imported contours might look very different from their original form, or be completely out of sight.

## Validity Tests

Argus ONE performs validity tests on imported files. These include data integrity tests and legality tests of the contours being imported, depending on the layer they are being imported to. These tests are based on the set of contouring rules explained earlier in this chapter.

Argus ONE tests the validity of each new contour read from the imported file against all the contours that are already present in the layer. If it finds a violating contour, it beeps, does not register the contour and continues reading the following contours in the file.

# Exporting Contours

Exporting contours allows you to document your contours and their values in a general format. For a description of the Export file format refer to the paragraph “Contours' Data Format” earlier in this chapter.

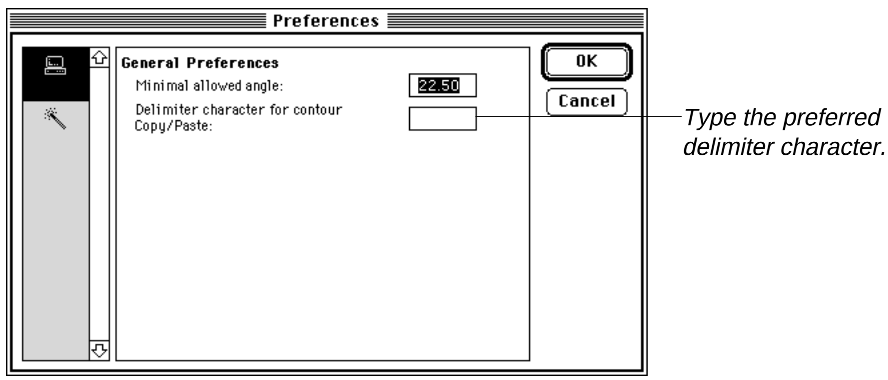
## Choosing the Delimiter

Argus ONE enables you to choose the delimiter character that will be inserted between data fields when copying or exporting contours.

### To choose the delimiter character

From the Special menu, choose Preferences...
The Preferences dialog box is presented.

*The preferences dialog box.*




You can enter any character or set of characters as delimiters. However, if you use more than one character, a digit, period, the letter “e” or “E”, you will not be able to paste contours you copy. In case you changed the delimiter to one that is not accepted by Argus ONE as a valid one for pasting, remember to change it back before attempting to paste contours.

## To Export Contours from a Layer to a File

1. Activate the layer from which you want to export contours.
2. From the File menu, choose Export “Layer’s Name”...
3. Enter the file name.
   Argus ONE suggests a default name that is a string containing the layer’s name suffixed by “.exp”.
4. Click the OK button.

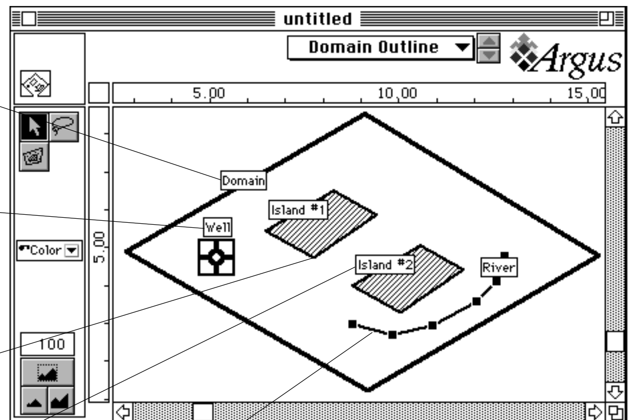
### An Example of an Exported Domain Outline Contour

A Domain Outline contour having two islands, one open contour, and one point is presented on the screen shot to the right. Its exported values are presented in a spreadsheet application window.

*Exported domain layer*

| | A | B |
|---|---|---|
| 1 | # Name :Domain | |
| 2 | # Points Count | Value |
| 3 | 5 | 1 |
| 4 | # X pos | Y pos |
| 5 | 9.10167 | 8.54083 |
| 6 | 2.78694 | 4.90722 |
| 7 | 9.17222 | 1.23833 |
| 8 | 15.2753 | 4.80139 |
| 9 | 9.10167 | 8.54083 |
| 10 | | |
| 11 | # Name :Well | |
| 12 | # Points Count | Value |
| 13 | 1 | 0 |
| 14 | # X pos | Y pos |
| 15 | 5.22111 | 4.69556 |
| 16 | | |
| 17 | # Name :Island #1 | |
| 18 | # Points Count | Value |
| 19 | 5 | 0 |
| 20 | # X pos | Y pos |
| 21 | 6.49111 | 5.43639 |
| 22 | 8.29028 | 6.49472 |
| 23 | 9.41917 | 5.85972 |
| 24 | 7.76111 | 4.73083 |
| 25 | 6.49111 | 5.43639 |
| 26 | | |
| 27 | # Name :Island #2 | |
| 28 | # Points Count | Value |
| 29 | 5 | 0 |
| 30 | # X pos | Y pos |
| 31 | 8.78417 | 3.99 |
| 32 | 10.5833 | 5.04833 |
| 33 | 11.7122 | 4.41333 |
| 34 | 10.0542 | 3.28444 |
| 35 | 8.78417 | 3.99 |
| 36 | | |
| 37 | # Name :River | |
| 38 | # Points Count | Value |
| 39 | 6 | 0 |
| 40 | # X pos | Y pos |
| 41 | 12.8058 | 4.76611 |
| 42 | 12.5942 | 4.09583 |
| 43 | 12.065 | 3.56667 |
| 44 | 10.9008 | 2.93167 |
| 45 | 9.8425 | 2.68472 |
| 46 | 8.78417 | 2.96694 |

*The domain layer.*




*The table on the left contains the exported data from this domain outline layer.*

# Copying and Pasting Contours

Argus ONE supports copying and pasting of contours by automatically implementing the above described Export and Import commands. When you copy a contour, the number of its vertices, its parameters' values, its name and the coordinates of the vertices are copied to the clipboard. They are copied using the delimited character you set in the preferences dialog, in the same format they are written into an export file. You can either paste the copied contours into an application, paste it directly into another layer in your project, or into another project layer.

If you copy a domain outline contour with its islands and internal open contours, and paste it into an Information layer, both contour and islands will be drawn in the Information layer as regular contours, while the open contours will be skipped.

## To Copy Contours

1. Activate the layer you want to copy contours from.
2. Select the contour or group of contours to be copied.
3. From the Edit menu, choose Copy.

The contour is now in the clipboard.

You can now paste the contour into another layer, another layer in another Argus ONE project, or another application all together. As long as you do not copy or cut other contours, the clipboard retains your copied information.

When you copy a contour or a group of contours, the information copied into the clipboard includes the contour vertices, its name and its parameters' values. Pasting the contour, pastes the parameters' values as override values. Override values are explained later in this chapter.

## Taking Advantage of Copying and Pasting Contours

Combining the contours' copy and paste capabilities with a text editor, a word processor or a spread sheet application enables you to enhance your working environment.

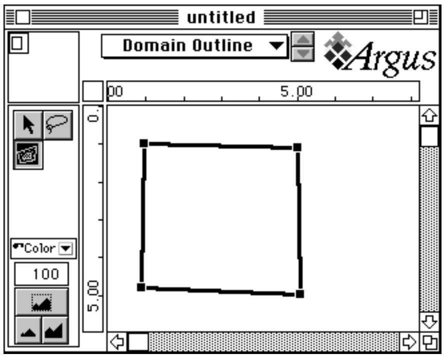
### Editing contours using other applications

If you need a quick way to numerically edit the contours' vertices, you can copy them into a spreadsheet application and edit them there. After you edit them, you can copy and paste them back into the layer. When you paste the edited contours back into a layer, Argus ONE checks their validity, and will not paste offending contours.

For an example, let's look at the case in which you want to create an exact square contour at a specific location.

You could use Argus ONE's facilities of zooming-in and creating contours with constraints, to record the vertices at the exact points.

However, if you have already created the contour, but not at the exact coordinates you wished, you can copy the contour to an editor or spreadsheet and correct the vertices' values there.




*Copy this contour and paste it into a spreadsheet to reshape it.*

| | A | B |
|---|---|---|
| 1 | # Name: | |
| 2 | # Points Count | Value |
| 3 | 5 | 1 |
| 4 | # X pos | Y pos |
| 5 | 0.9525 | 0.987778 |
| 6 | 5.00944 | 1.09361 |
| 7 | 5.08 | 4.97417 |
| 8 | 0.881944 | 4.79778 |
| 9 | 0.9525 | 0.987778 |

*Vertex X's Coordinate* *Vertex Y's Coordinate*

*Contour's vertices pasted in a spreadsheet.*

Now copy the contour from the spreadsheet or editor and paste it back into the layer. Be sure to delete the previous contour first, otherwise Argus ONE will not allow you to paste the corrected contour as it will overlap the original one.

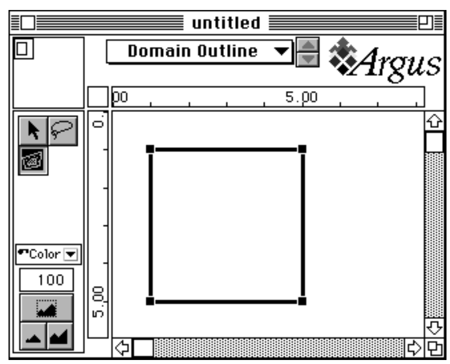
*Contour's value*

*Contour's name*

*Number of vertices + 1*

| | A | B |
|---|---|---|
| 1 | # Name: | |
| 2 | # Points Count | Value |
| 3 | 5 | 1 |
| 4 | # X pos | Y pos |
| 5 | 1 | 1 |
| 6 | 5 | 1 |
| 7 | 5 | 5 |
| 8 | 1 | 5 |
| 9 | 1 | 1 |

*The contour's fixed coordinates.*




*Paste the fixed contour.*

## Duplicating a Contour

To duplicate a contour use the above explained contours coping and pasting techniques. However, if you want to paste the duplicated contour in the same layer, you must first move the original copy to the side such that when you paste the duplicated contour it will not overlap it.

# Setting Your Views

When you edit contours in a layer you can take advantage of Argus ONE's information hiding capabilities to concentrate on the objects you need. In the following paragraphs we will outline some of these capabilities. For a detailed explanation of showing and hiding layers refer to part 1, chapter 1 in this manual.

All the commands you will use to set your views are presented in the View menu.

## Showing and Hiding Layers

Showing and hiding layers enables you to view the layer you work in with respect to only some of the layers. Hiding and showing layers and moving between layers is explained in detail in the chapter “The Argus ONE Workplace.”

### To hide or show a layer

- From the View menu, choose Hide or Show “Layer name.”

## Zooming

When you want to edit a part of the contour that is too small to edit in the current zoom level, Zoom In, or, to select large areas of the layer, Zoom Out to refrain from tedious autoscroll. Zooming also enables you to digitize and edit contours in high resolution. A detailed discussion about zooming is presented in the chapter “The Argus ONE Workplace.”

## Showing and Hiding Contour Information


Some of the information you can assign to contours can be graphically presented on the screen. You can show and hide this information so that your workplace does not become cluttered.

### To show and hide the contour’s value

The contour’s value, concatenated with the parameter’s units is displayed in a small text box annotated to the contour’s first vertex. The show/hide status of the contour’s value is layer dependent. Showing or hiding contours’ values takes place only in the active layer.

### To hide contours’ value

1. Make sure the layer in which you want to hide the contour’s value is the active one.
2. From the view menu, choose Show Contour Value.
   All contours’ value in the layer are hidden.

The check mark, √, next to the menu item is removed to indicate that this option is turned off.

#### To show contours' value

1. Make sure the layer in which you want to show the contour's value is the active one.
2. From the View menu, choose Show Contour Value.
   All contours' value are shown.

A check mark, √, next to the menu item indicates that this option is turned on.

### To show and hide the contour's name


A contour to which you assign a name, display its name in a text box above the contour's value. The show/hide status of the contour's name is layer dependent. Showing or hiding contours' names takes place only in the active layer.

#### To hide contours' name

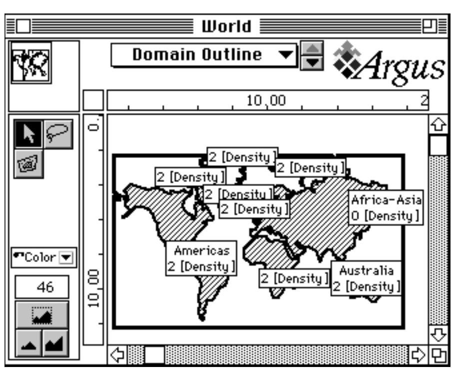
1. Make sure the layer in which you want to hide the contour's name is the active one.
2. From the View menu, choose Show Contour Name.
   All contours' names in the layer are hidden.

The check mark, √, next to the menu item is removed to indicate that this option is turned off.

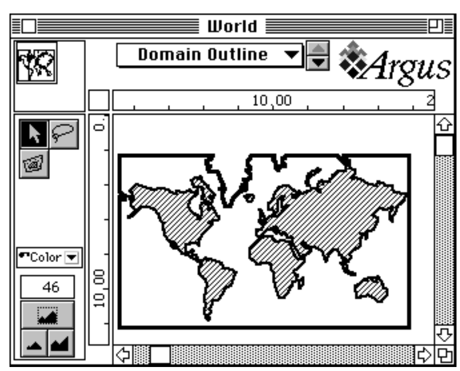
#### To show contours' name

1. Make sure the layer in which you want to show the contour's name is the active one.
2. From the View menu, choose Show Contour Name.
   All contours' names are shown.

A check mark, √, next to the menu item indicates that this option is turned on.

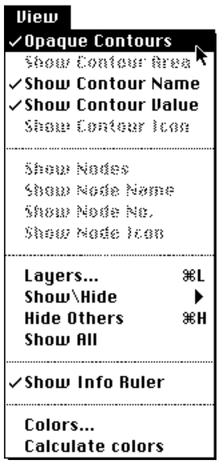



*Contours' names and values are visible.*




*Contours' names and values are hidden.*

## Seeing Through Information Contours




As explained in chapter 1, you can instruct Argus ONE to evaluate contours in an Information layer using colors. When you turn colors on, and set an evaluation parameter, contours are colored to reflect their values. When colors is on, contours can be set transparent or opaque. Again, if you need to see other layers with respect to the current layer, you can turn the contours transparent, so that only the contours' outlines are colored.

### To turn contours transparent

1. Make sure the layer in which you want to turn contours to transparent is the active one.
2. From the View menu, choose Opaque Contours.
   All contours are made transparent.

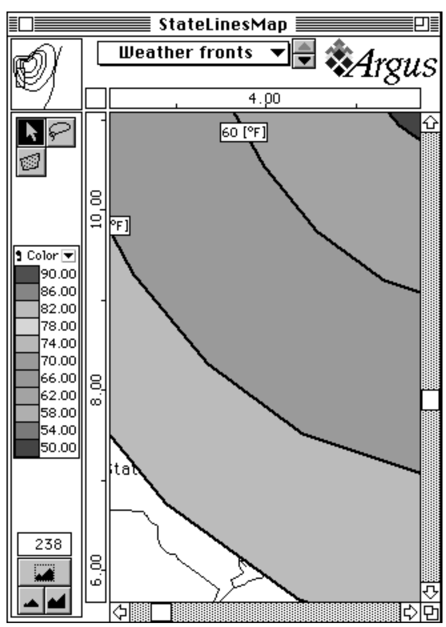
The check mark, √, next to the menu item is removed to indicate that this option is turned off.

### To turn contours opaque

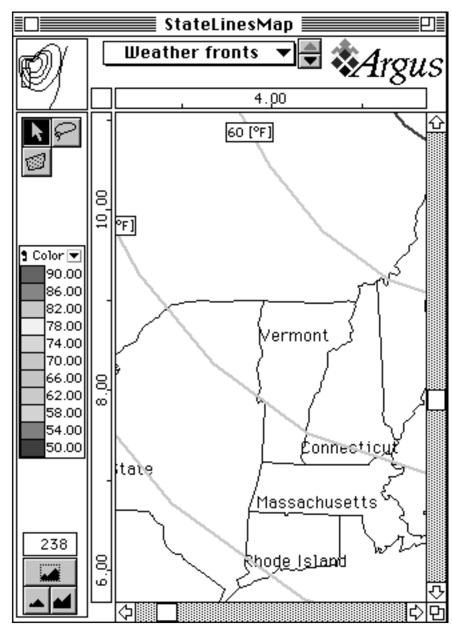
1. Make sure the layer in which you want to turn contours opaque is the active one.
2. From the View menu, choose Opaque Contours.
   All contours are made opaque.

A check mark, √, next to the menu item indicates that this option is turned on.

*Contours can be made opaque only when Colors is on.*




*To reveal hidden information...*




*...turn the contours transparent.*

# Contours Interpretation Methods

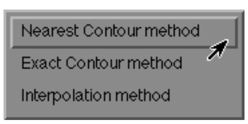
As you will learn in the next section of this chapter, Argus ONE layers can be linked. When you create a layer parameter and assign it a parameter from another layer you link them. When an object or a parameter expression is evaluated, it probes the linked layer or layers for its value. The probing object, a node, an element, a block or the mouse location, sends its position (X and Y) to the layer being probed, which returns the appropriate value. The value a layer returns on a probe query, is a function of the layer type and in information type layers, it is a function of the contour interpretation method you choose for the linked layer.

Argus ONE supports three contour interpretation methods. These three methods are designed to support three different types of data.

## The "Nearest Contour" Method

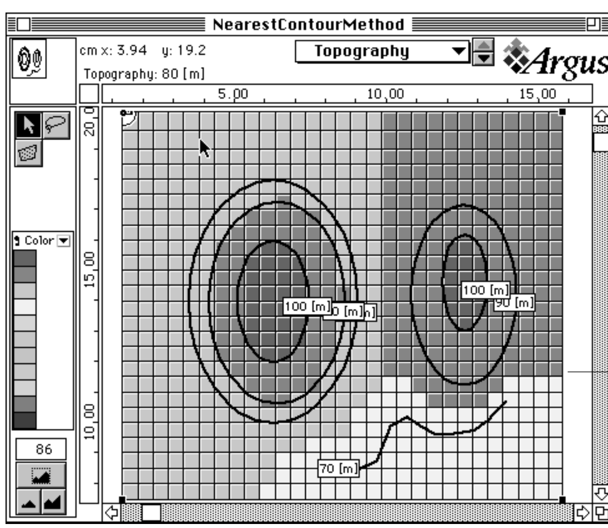
This method is designed for contours describing information such as topography and bathymetry.

*From the Layers Dialog.*




The value at any point in a layer using the "Nearest Contour" method, is that of the first enclosing contour. Values in areas which are not enclosed by a contour, are that of the nearest contour, either close or open, but not of a point object. If the layer contains no contours, the value at any point is set to the default value. If there is only one contour in a layer, either close or open, the value at any point is that of that contour.

In the following example, a grid is linked to an information layer named "Topography". The Topography layer was set to use the "Nearest Contour" interpretation method. Notice that blocks' values are changing along lines which are of equal distance from the outer most contours.



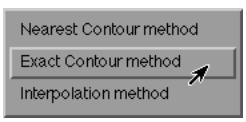

*A finite difference grid linked to an information layer describing the topography of a formation or the bathymetry of a lake. Note that since the "Nearest Contour" interpretation method is applied to the "Topography" layer, the grid blocks values change along lines lying on equal distance from disjoined contours.*

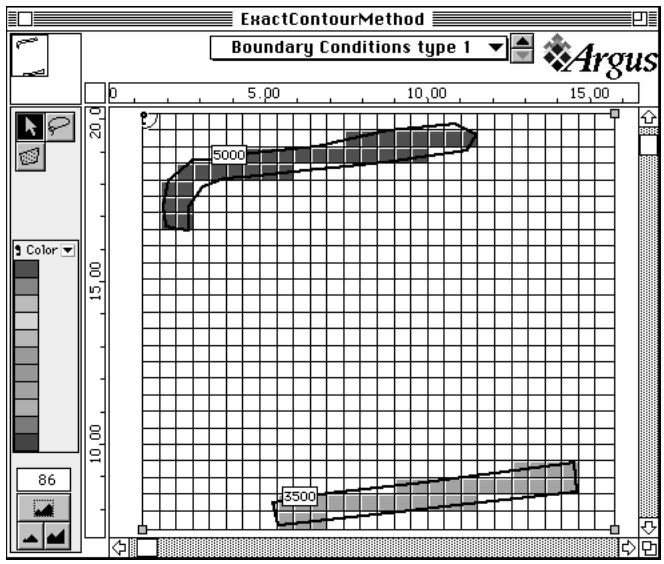
## The “Exact Contour” Method

This method is designed for contours describing zone-wise type of information, such as the distribution of a physical parameter in a geological formation or boundary conditions in and around the investigated domain. This type of information is characterized by the presence of a default value describing most of the domain’s area with some zones having different values. Boundary conditions, for instance, are null over most of the domain but have some specific value at points, lines and some areas in the domain and along its boundaries.

In the following example, a grid is linked to an information layer named “Boundary Conditions Type 1”. The Boundary Conditions Type 1 layer was set to use the “Exact Contour” method.

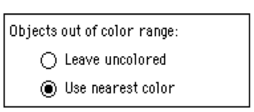
*From the Layers Dialog.*




*A finite difference grid linked to an information layer describing the Boundary Conditions of Type 1 (such as constant head). Note that since the “Exact Contour” interpretation method is applied to the this layer, only grid blocks assigned a boundary condition different then the layer’s default value (0=none) are colored.*

*Tip: To instruct Argus ONE not to color blocks having the layer’s default value, use the Colors dialog to set the colors range and choose the Leave uncolored option.*




Using special functions together with the “Exact Contour” method, enables you to assign grid blocks and mesh elements, boundary conditions values and types for boundaries lying along lines such as rivers, faults, slurry walls, etc.

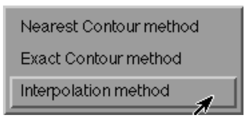
To read more about using layer parameters and functions refer to the chapters “Layer Parameters”, “Editing and Refining a Mesh” and “Creating and Editing a Grid.”

## The "Interpolate" Method

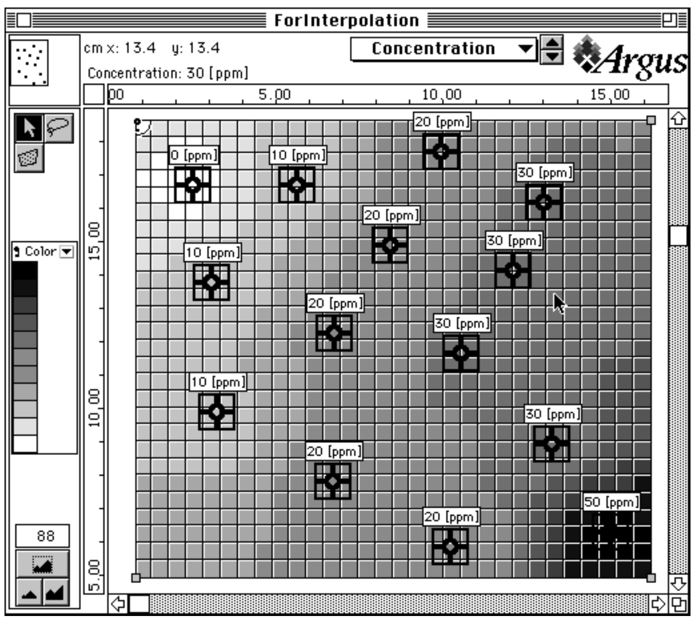
This method is specifically designed for information that is available to you point-wise. Tabulated information from observation points, sources and sinks is usually the type of information you need to interpolate, to obtain values for points you do not have values for such as element centers, mesh nodes and grid blocks' centers.

Argus ONE interpolation is based on a proprietary "Inverse Distance weighted Interpolation" algorithm.

*From the Layers Dialog.*




In the following example, a grid is linked to the information layer Concentration. The Concentration layer was set to use the "Interpolate contour" method. Grid blocks are assigned interpolated values.




*A finite difference grid linked to an information layer describing the initial concentration of a contaminant. Note that since the "Interpolate Contour" interpretation method is applied to the "Concentration" layer, the grid blocks values are the interpolated values of the point-wise information.*

---

**Important Note:** Choosing the "Interpolate Contours" method for an information layer, interpolates contour values as well. Argus ONE disassembles the contours to their vertices and treats them as discrete points. All vertices on a contour are considered as equal value points, the value of the contour they form.

---

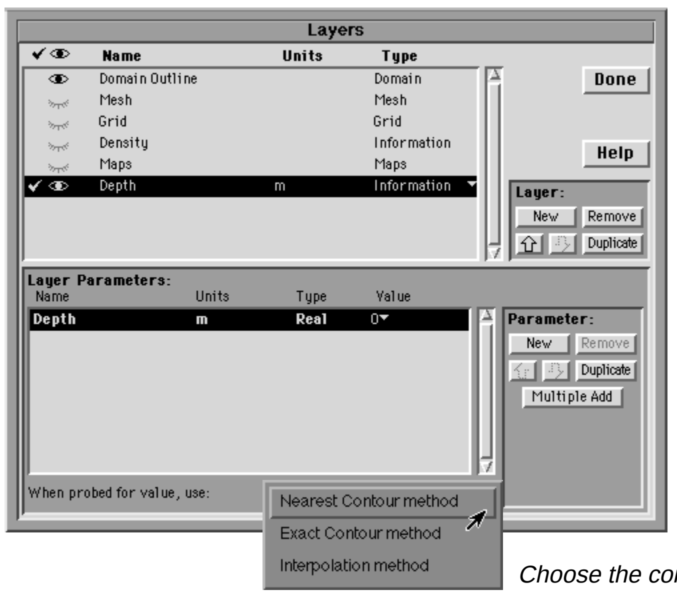
You can also use the Interpolate function available through the Expression Dialog Funcs menu to instruct Argus ONE to interpolate any expression you define.

### Setting the Contours Interpretation Method

You can change the interpretation method of an information layer at any time. When you first create a new layer, Argus ONE sets it to the default interpretation method which is the “Nearest Contour” method.

#### To set or change a layer’s interpretation method

1. Open the Layers dialog.
2. Choose the layer its interpretation method you wish to change.
3. From the menu at the bottom of the dialog, choose the preferred method.




*Choose the contour interpretation method from the popup menu.*

## Searching for Contours in an Information Layer

As you add information to your project, using the Information layers, you're actually building a database. Argus ONE enables you to perform extensive and complex searches of the contours based on their data. Searching for contours is especially useful when you import detailed tabular data into an information layer.

The Search For... command is the graphical interface to the query language you use to build your queries. It allows you to perform searches based on all the types of data assigned to a contour.

- Contour name.

- Contour type - Close, Open and Point object.
- Contour parameters data.
- Contour Icon - Refer to the **Supplement** chapter.

You can use these searching capabilities to orientate within the project and search your data visually as well as quantitatively.

The Search For dialog enables you to search for contours, define a search criteria and to decide how to treat the current selection.

After you have selected a group of contours based on some search criteria, you can extend or reduce the found group by adding or removing from it another group of nodes or elements satisfying a different search criteria.

You can also negate each search criteria.

## Using Contours Parameters to Define Search Criteria

Since you can assign contours as many parameters as needed, you can use these parameters to create intricate search criteria.

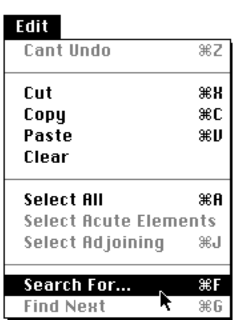
## To Search an Information Layer

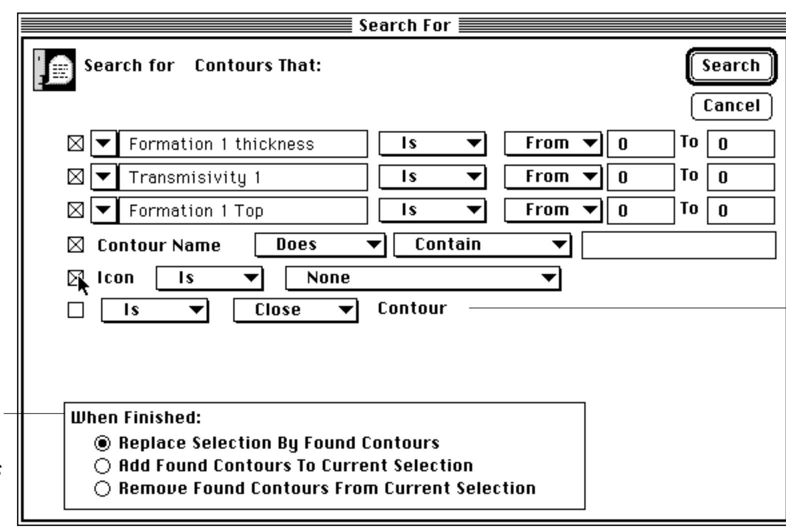
1. Make sure the active layer is an Information Layer.
2. From the Edit menu, choose the Search For... Command.

The following dialog appears.

3. Set all the fields you need in order to define the search criteria.
4. Press the Search button.

While searching, a progress dialog appears to indicate the search progress.




*Set the search range*

*Using several search processes together with different settings of the current found set enables you to expand or reduce the search scope.*

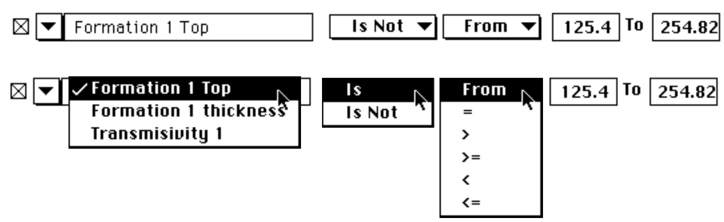
## Defining a search criteria

In the following paragraphs each of the fields and options in the search dialog are explained.

### Choosing the contours parameters to be searched

You can create a search criteria containing up to three of the layers parameters. If you need to define a search criteria based on more parameters, execute one search for the first three, and then define and execute more searches using the Add Found To Current Selection option.

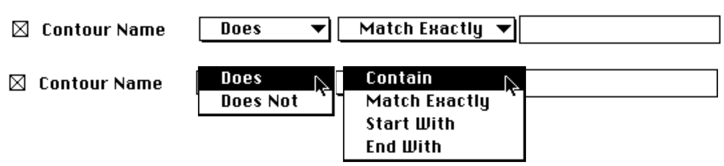
- From each of the three Parameters popup menus, choose the parameter to be searched for, and assign its range. Check the check box.




### To search for contours’ names

Assigning names to very special contours can help you to orientate within an Information layer.

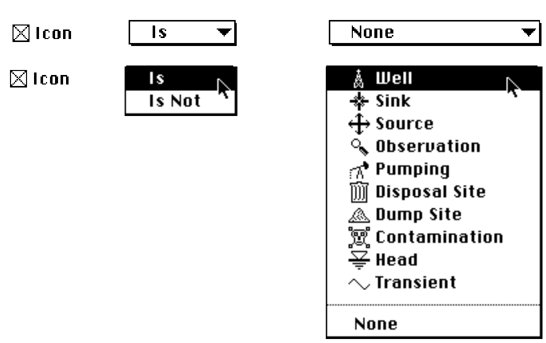
- Enter the icon and specify the search criteria. Check the check box.




### To search for contours’ icon

Assigning icons to very special contours can help you to orientate within an Information layer. To read about assigning icons refer to the **Supplement**.

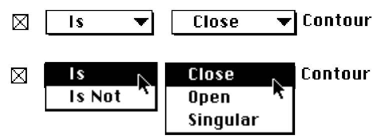
- Enter the name and specify the search criteria. Check the check box.



### To search by contours' types

You can instruct Argus ONE to search only for contours of a specific type, that is, for close, open and point contours.

- Select the contour type. Check the check box.




The contours matching the search criteria are selected. If no contours matching the search criteria are found, Argus ONE reports it with the following alert:


## Expanding the Search Scope

By default the dialog is set to "Replace Selection By." If when invoking the dialog some contours are already selected, they are de-selected and replaced by the newly found set.

To expand or reduce the search scope over an already found set, change this field to "Add Found Contours To Current Selection" or "Remove Found contours From Current Selection" and define the additional search.

# Layer Parameters

# Overview

*While a Argus ONE layer is like a data base, layer parameters are like data base fields. Linking layer parameters allows you to create a relational data base for your data.*

You might ask yourself why should you have more than one parameter per layer. The first and most important reason to add parameters to layers, is for linking layers. A layer parameter can be assigned any other layer, or layer parameter, and by that, link the two layers. For instance, if you wish to link depth variation stored in an information layer, to mesh elements or nodes or to grid blocks, you create a mesh or a grid parameter and assign it the layer describing the depth variation. Each parameter can hold such a link.

*Layer parameters allow you store different information describing the same object.*

The second reason to have multiple layer parameters is that the phenomenon you need to describe might have more than one value defining it. For instance, a material property in heterogenous media might need three parameters to describe its value in each of the three coordinate system directions. Another case would be describing boundary conditions. A boundary condition is defined by its type and value, so to describe boundary conditions you would create two parameters.

*Layer parameters enable you to create meaningful and elaborate expressions to define relations between different data describing your problem.*

The third reason is the ability to create and assign complex and elaborate expressions based on information parameters to other parameters. For example, you need to compute the intensity of a physical phenomenon defined as the difference between its ceiling and floor values, at each element, node or block. However, you have received two maps describing the ceiling and floor distributions. To assign the intensity to an element, node, or a block you would only have to create a parameter expression subtracting the two and link it to the mesh or grid layers.

As you will learn in the following sections, a layer parameter can be a constant, a link to another layer and a complex expression including constants, links, mathematical and layer specific functions. A layer parameter is context sensitive, that is, it is evaluated at the current layer context. For instance, a mesh layer parameter which points to an information layer will calculate its value at elements’ centers and node positions for any mesh you create.

*You can create and use as many layer parameters as you need.*

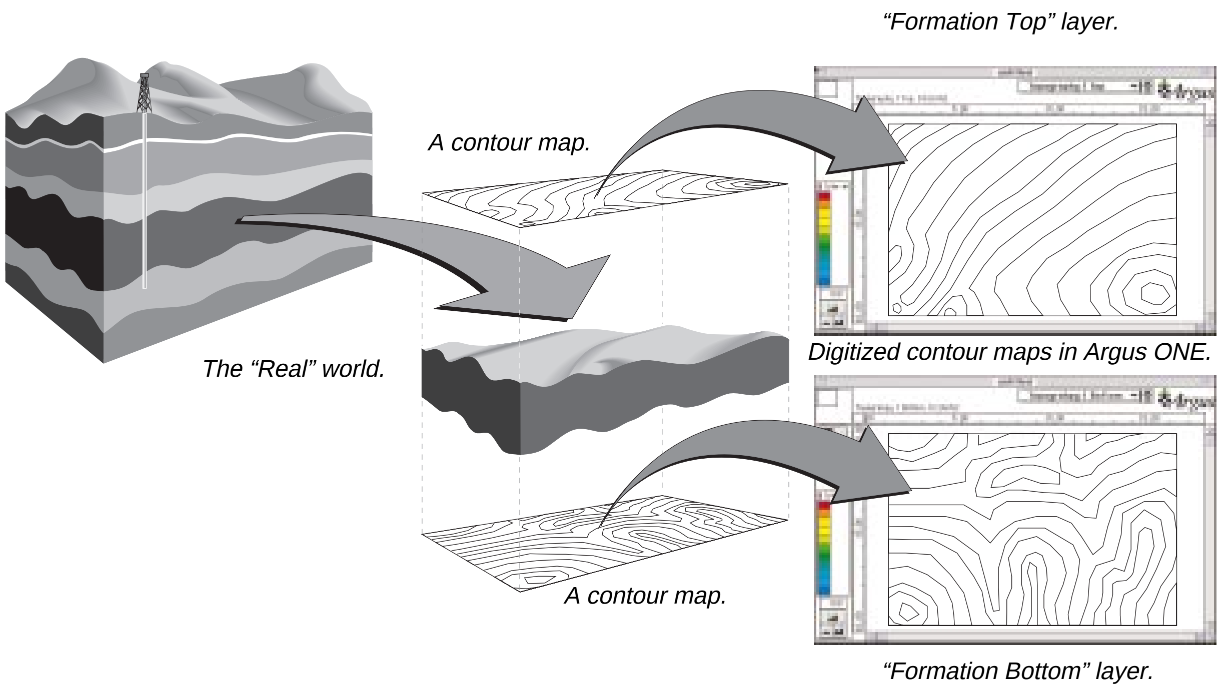
*Some of Argus ONE layers are automatically assigned parameters while to others you assign them yourself.*

Argus ONE layers can be assigned as many parameters as you need. When you create an Information type layer it is automatically assigned a layer parameter, bearing the layer’s name. Mesh and Grid type layers can also be assigned as many parameters as you need, but when you create them they are not assigned a parameter. Map type layers are the only layers you can not assign parameters to, since they are designed to hold descriptive graphical information and not quantitative numerical information. Data type layers are automatically assigned parameters as you read or import information into them. To read more about Data type layers refer to the **Supplement** chapter.

To read about layer creation and manipulation, refer to chapter 1 earlier in this manual.

# An Example

You need to assign your mesh or grid objects the intensity (thickness) of the geological formation presented in the following picture. You have received (or digitized yourself) two digitized contour maps describing the surfaces of the top and bottom of the formation.




To calculate the formation's thickness:

1. Create two information layers and import the digitized maps into them. Name the two formations "Formation Top" and "Formation Bottom" or any other names you choose.
2. Create a second parameter in one of the layers and name it for instance "Formation Thickness".
3. Assign the "Formation Thickness" parameter an expression subtracting the two layers:
   `Formation Top - Formation Bottom`
4. Link the parameter "Formation Thickness" to a mesh or a grid layer.

If you export this mesh or grid, each element, node or block will be automatically assigned the correct intensity value. No matter how many times you edit, change and refine your mesh or grid, every object in them will always be assigned the correct thickness (intensity) value.

# Creating and Manipulating Layer Parameters

You can create, remove, rename, reorder and change layer parameters, at any time. You operate on layer parameters in the Layers dialog box.

### To create a layer parameter

When you create a new layer parameter, its line is entered after the line that is selected when you click the New or Duplicate buttons.

**1.** From Layers Selection menu in the information ruler, choose Layers... The Layers dialog box opens.

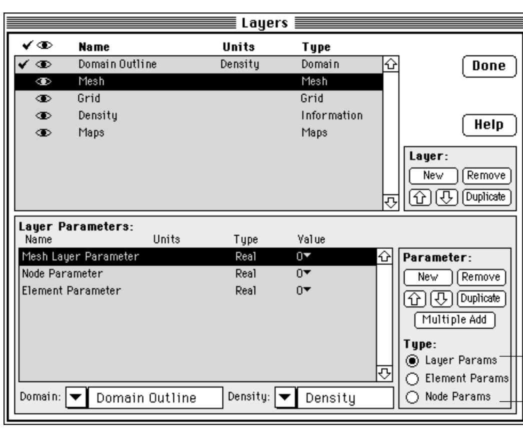
**2.** In the Layers (upper) list, select the layer's line.

**3.** In the Parameters control panel click the New button. A new parameter is created.

When you create a new layer it is assigned an initial name. The initial name is always the string "New Parameter" unless this layer already has a parameter in that name. In this case the string "New Parameter" is suffixed by the next available digit, starting from 1. For instance, if there exists a layer parameter by the name "New Parameter 1", a newly created layer parameter will automatically be named "New Parameter 2". This insures that each layer parameter has a unique name.

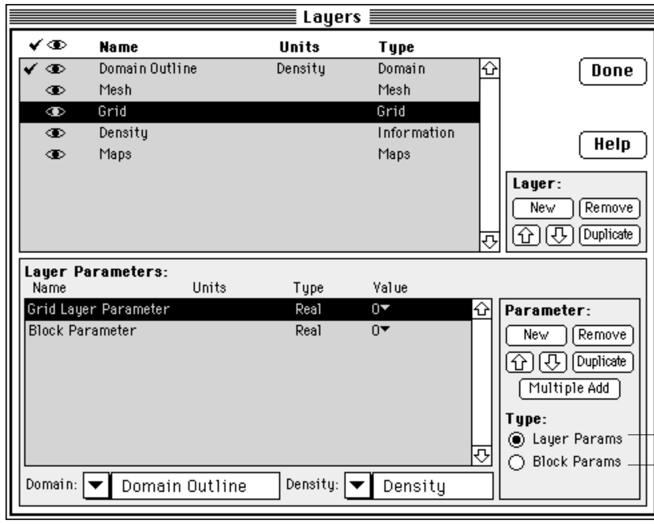
## Creating object specific parameters

When you create parameters in any Argus ONE layer, these parameters are assigned to all objects in the layer. For instance if you create a parameter in a mesh layer, Argus ONE sets it by default to be a mesh parameter. Since it is a layer parameter, both nodes and elements will be assigned this parameter. However, Argus ONE allows you to create object specific parameters in the mesh and grid layers. Such parameters are assigned only to objects of the parameter's type. Object specific parameters enable you to access object specific functions in the expression dialog.




***Creating mesh, node, and element type parameters.***

*Click the radio button to set the parameter type.*




***Creating grid, and block type parameters.***

*Click the radio button to set the parameter type.*

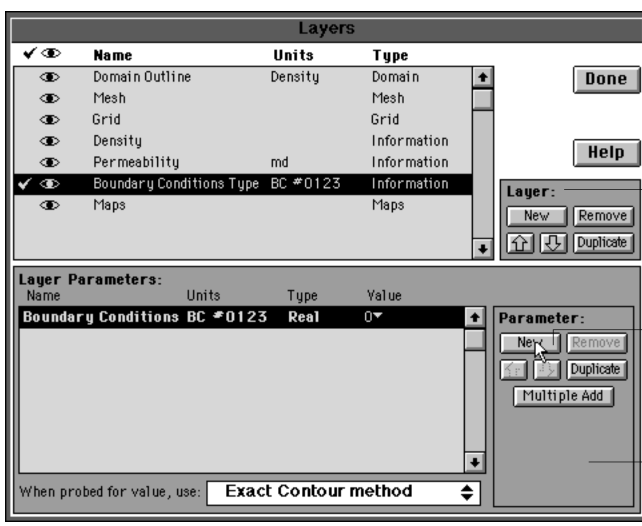
For further reading about object specific parameters and functions refer to the sections "Layer Specific Functions" and "Creating and Editing Expressions" later in this chapter.

## Creating a layer parameter – an example

For instance, to describe boundary conditions types and values, create an information type layer, set it to "Exact Contour" method, and name it. The main parameter might hold the boundary condition type, where 0 (zero) denotes no boundary condition, 1 denotes Dirichlet type, 2 denotes Neumann type and 3 denotes mixed boundary condition. (These are mere suggestions, you can use any naming and type assignment numbering).

*The layers list*

*The parameters list*



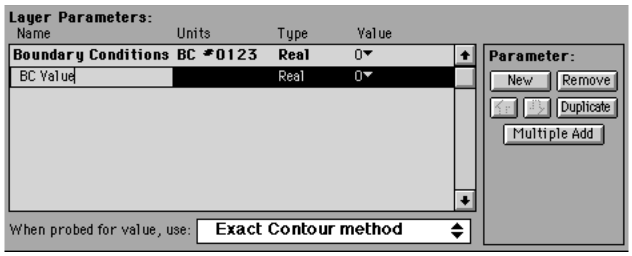

*Layer Control Panel*

*Click to create a new layer parameter.*

*Parameter Control Panel*

To create a second layer parameter to hold the boundary condition's value:

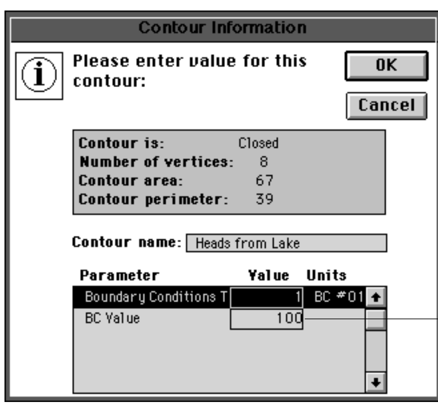
1. Click the New button in the "Parameter Control Panel.
2. Name the newly created parameter "Boundary Condition Value" or "BC Value" or any other name you find suitable.




The newly created parameter line is inserted below the parameter line that was selected while you clicked the New button.

Notice that when you create a new layer parameter, Argus ONE automatically assigns the parameter a default value of 0 (zero). You can change the default value at any time. The default value can be a constant, a link to another layer parameter or an expression. When you create a contour, it is automatically assigned the default value. To read more about parameters' values refer to the following sections in this chapter.

If you create a contour in this layer, the contour can be assigned the two required values to define a boundary condition.




*The Contour Information dialog showing the two contour values required to define a boundary condition.*

*A manually overridden value is indicated by a black frame surrounding the parameter's value text edit box.*

## Default vs. Manually Overridden Parameter Values

As briefly explained in the previous paragraph, a parameter always has a default value. The default value is the value you assign the parameter when creating it, or 0 (zero) automatically assigned to it by Argus ONE. The default value can be a constant, a link to another parameter, or an expression.

Every Argus ONE object is actually a detailed data structure. An object's data structure holds the object location, type, any additional data it can be assigned, such as name and icon, and the values of the parameters, it or the layer containing it, is assigned.

When you create or change an object, the object's information dialog box presents you with all layer's and object's parameters and their values. An object's parameter value is the parameter's default value, unless you manually override it for that object. If the parameter is an expression or a link to another parameter, its value is evaluated at the object's evaluation context to produce the object's default value for that parameter.

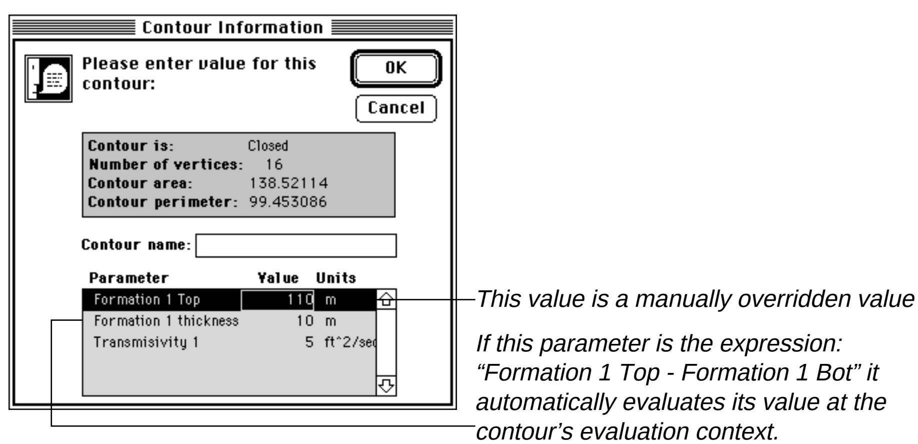
If you manually enter a value for an object parameter value, you override the default value. In this case, if the parameter is a link to another parameter, or an expression, these are not resolved, and the object's value for that parameter will be the overridden value.

Manually entered values are most common when working with information type layers. The contours or points you import or digitize, always have at least one "z" value assigned to them. These values are assigned to the object and override the parameters's default value. Other parameters containing expressions or links are usually automatically calculated.

---

**Important note:** When it comes to mesh or grid objects such as nodes, elements, and blocks you should try to avoid using overridden values since they are lost each time you delete the mesh or grid.

---




## To manually override a parameter value

- Enter a value.
  The parameter's value text box is enclosed by a black frame to indicate the value is not the default value, but a manually overridden value.

### To remove the manual override

To remove the manually overridden value:

1. Select the text edit box and delete the manual entry.
2. Hit the RETURN key, or click the mouse outside that text edit box.
   The black frame enclosing the text edit box is removed to indicate the parameter's value is now evaluated to the default value.

## Manipulating Parameters

### To change a layer parameter's name

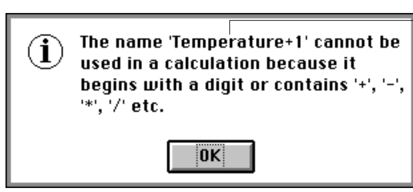
**Important note**: *Argus ONE prevents you from creating or renaming a layer parameter to a name that is already assigned to another parameter in that layer.*

As you will learn later in this chapter, linking a parameter to another is performed by using the parameter's name. When you change a layer parameter's name, Argus ONE automatically renames it in all its occurrences.


- Start typing the parameter's name.
  Or
  Click the line within the name field and type the parameter name.

The cursor changes into an I Beam cursor to indicate you are in text edit mode.

Although a parameter name can contain any character, if you use a digit as the name's first character, use punctuation marks, or the numeric keys such as the plus sign, Argus ONE will not be able to use this name in an expression. If you create such a name the application brings up an alert to warn you, but allows you to keep the invalid name.




*An invalid parameter name.*

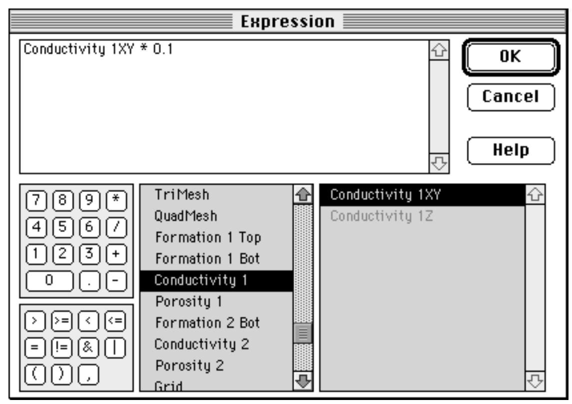
### Parameters' full names

A parameter's name is defined by the name of the layer it belongs to, suffixed by the actual parameter name. Information type layer are automatically assigned a first parameter by Argus ONE. The first parameter's name in an information type layer is the layer's name. To change it you have to change the layer's name. The names of any additional parameters you create within an information type layer, or for that matter, any mesh or grid parameters, are made of the layer's name suffixed by the parameter's name.

For example, if you create a layer named “Conductivity 1” with two parameters named “Conductivity 1XY”, and “Conductivity 1Z”, their full names when referenced will be:
“Conductivity 1.Conductivity 1XY” and “Conductivity 1.Conductivity 1Z”.

To keep things simple, Argus ONE does not show the parent layer’s name if you refer to the layer’s main parameter. For example, to create an expression defining the “Conductivity 1Z” parameter as 10 percent of the “Conductivity 1XY” parameter:

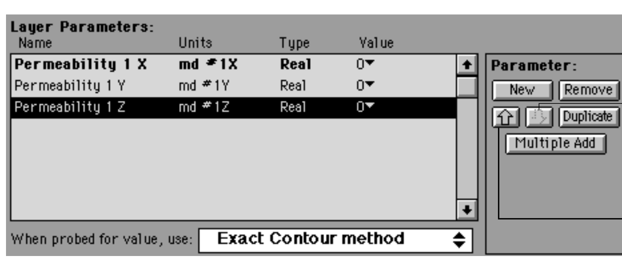
1. In the parameters list click the line of the Coductivity1Z parameter to select it.
2. Click the Value field and from the popup menu choose Expression.
   The Expression dialog box opens.
3. Select the layer “Conductivity 1” from the layers list. The layer’s parameters will be listed in the list to the right.
4. Double-click the “Conductivity 1XY” parameter to enter it into the expression text edit box.
5. Enter the multiplication sign followed by 0.1.
   The expression created is: Conductivity 1XY * 0.1.




To read more about expressions refer to the following section in this chapter.

## To change parameter ordering

You can set the order in which parameters are listed in the parameter list. The order you set in the parameter list is also the order in which these parameters appear in the Expression dialog parameters list.




*The parameter is the bottom parameter so you can’t demote it.*

*Click the Up arrow to promote the selected parameter.*

### To remove a parameter

Removing a layer parameter deletes the parameter from the layer.

1. In the Layers dialog box, select the line its parameter you wish to remove.
2. In the parameters list, select the parameter to be removed.
3. In the parameter control panel, click the Remove button.
   The parameter is deleted.

If the parameter you try to remove is referenced by an expression, Argus ONE alerts you and enables you to either go on and delete it, or cancel.


If you choose to go ahead and delete a referenced parameter, on evaluation of the expression this parameter will return -0 (minus integer zero), denoting it can not be resolved.

### To duplicate a layer parameter

To shorten layer parameter definition, Argus ONE enables you to duplicate a parameter.

1. Select the parameter you wish to duplicate.
2. In the parameter control panel, click the Duplicate button.
   The parameter is duplicated.

To keep parameter naming valid, Argus ONE adds a suffix digit to the duplicated parameter. If the originating parameter name has a suffix digit, the digit is incremented to the next available digit.

# Linking Parameters

Linking layers, as explained in part 1, is actually performed by means of linking parameters. Linking parameters allows you to reference parameters from any layer, including from the same layer. Parameter linking is actually a basic form of Argus ONE expressions, explained in the next section. You will use parameter linking for many purposes such as linking information layers to mesh and grid layers, creating expressions, exporting your data, etc. However, since you will be using parameter linking very often, Argus ONE supports few shortcuts to simplify and shorten your work.

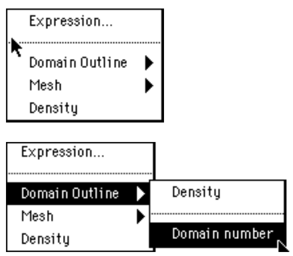
## To Link Parameters

A link is always created between two parameters, the linked parameter and the referenced parameter. By linking two parameters you make the linked parameter point to the referenced parameter.

**1.** Open the Layers dialog.

**2.** Select the layer in which you want to create a linked parameter.

**3.** In the parameter control panel, click the New button to create the linked parameter and assign it a name.




**4.** Click and hold the mouse in the Value field of the newly created parameter line and select the Expression popup menu.
The Expression dialog opens.

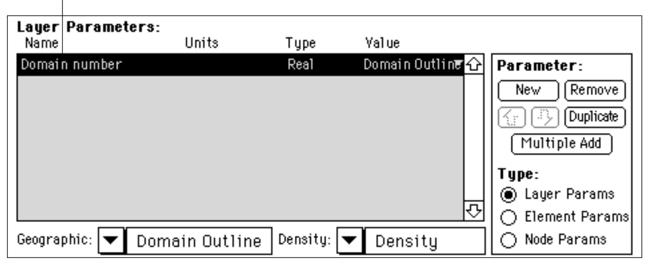
**5.** In the Functions and Layers list (the left one) click to select the layer the referenced parameter belongs to.
The layer's parameters are listed in the list to the right.

**6.** Double-click the referenced parameter to enter it to the text edit box on top.

**7.** Click the OK button.

*You can change the name and the linked parameter at any time.*

*This parameter is linked to a domain outline layer parameter named Domain number.*

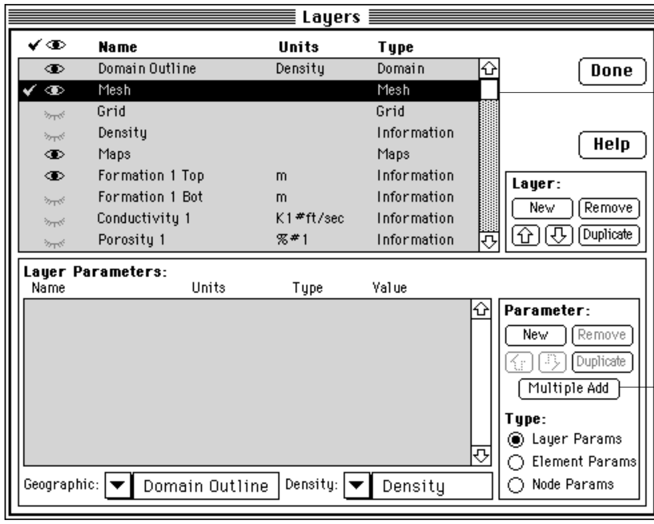



## To Link Many Parameters At Once

If you need to create many linked parameters at once, Argus ONE enables you to do so using the Multiple Add command. The Multiple Add command creates new parameters and links them to the referenced parameters in a single step.

**1.** Open the Layers dialog.

**2.** Select the layer to which you want to add new linked parameters.

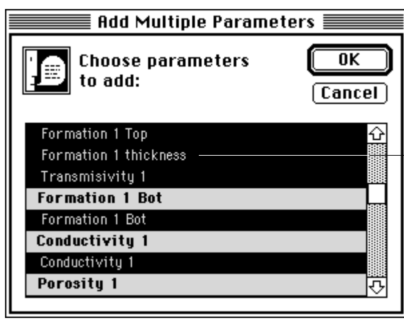



*To link many parameters to a layer, select the layer line to which you want to add the new linked parameters.*

*Click the Multiple Add button to link many parameters at once.*

3. In the parameter control panel, click the Multiple Add button to create the linked parameters.
   The Add Multiple Parameters dialog box opens and presents you with the list of available parameters.

*The Add Multiple Parameters dialog box.*



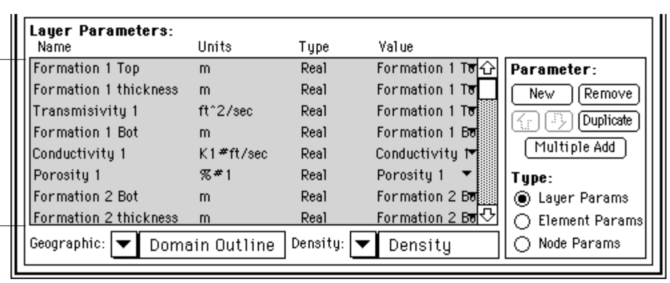

*Layers' names appear in bold, layers parameters are indented below the layer name.*

*Click a parameter to add it to the selection.*

*Click a selected parameter to remove it from the selection.*

4. Click the mouse in every parameter's line you wish to link to add it to the selection. Selected parameters' lines are highlighted.
5. To remove a parameter from the selection, just click it again to deselect it.
6. When you finish adding parameters to the selection, click the OK button.
   For each parameter you selected, Argus ONE automatically creates a linked parameter named after the referenced parameter name.

*A list of parameters added using the Multiple Add button.*



### Linked Parameters' Naming

When you link parameters using the Multiple Add command, a linked parameter is assigned the name of the referenced parameter. The full name of a parameter is made of the parameter name, prefixed by the name of the layer it is defined in.

For example, you intend to create a number of disjoined meshes in a mesh layer, and you need to mark all elements belonging to a certain mesh with a unique domain number. Create a second parameter in the Domain Outline layer and name it "Domain number". When you open the expression dialog you will see the full name of the referenced parameter which in this example will be:
"Domain Outline.Domain number"

While creating parameters yourself, or when Argus ONE creates them for you when you link parameters, Argus ONE adds a suffix digit to parameters having identical names. This is done to avoid duplicate names thus keeping parameter names valid. If the originating parameter name has a suffix digit, the digit is incremented to the next available digit.

---

**Important note:** A circular reference is not allowed. If you use Argus ONE expression dialog popup menus to reference layer parameters, it does not enable you to create circular references.

---

## Expressions

An Argus ONE expression is like a formula. Entering an expression is the basic technique you use to analyze data in your project. With an expression you can perform operations such as addition, multiplication and comparison, on layers' and objects' values.

***Important note:*** *When you assign an element, a node or a block an expression, the expression is evaluated for every element, node or block in the mesh or grid.*

An expression can include any of the following elements: operators, reference to other layer parameters (links), values and functions. To enter an expression in a layer parameter, you enter a combination of these elements in the expression dialog box.

An expression is evaluated in its evaluation context. The evaluation context for elements' parameters, is the element's center coordinates, for nodes, it is the node's coordinates, and for blocks it is the block's center coordinates. Some functions' context is the object's area or perimeter.

When you assign an element, a node or a block an expression, the expression is evaluated for every element, node or block in the mesh or grid.

## Using Expressions

An *expression* combines values with operators, such as a plus or minus sign, in a layer parameter, to produce a new value from existing values. An expression can include layer parameters, constant values, and can also use general and layer specific functions.

You can think of an expression as one side of an equation whose result is evaluated and shown in the layer.

## Using Operators

You can use *operators* to specify the operation, such as addition, subtraction or multiplication, to perform on the operands (values on either side of the operator) in the expression. Argus ONE uses three types of operators as shown in the following table.

| Type of operator | Example | Operator | Description |
|---|---|---|---|
| **Arithmetic**<br>Performs basic mathematical operations; combines numeric values and produces numeric results. | The expression 20*2/4 multiplies 20 by 2 and divides the result by 4. | + | Addition |
| | | - | Subtraction (negation if used with one operand only) |
| | | / | Division |
| | | * | Multiplication |
| **Comparison**<br>Compares two values and produces the logical value `True` `(1)` or `False` `(0)`. | The expression Depth>50 produces the value True if the value of Depth is greater than 50. If the value is less than or equal to 50, this expression produces the logical value False. | = | Equal |
| | | > | Greater than |
| | | < | Less than |
| | | >= | Greater than or equal to |
| | | <= | Less than or equal to |
| | | != | Not equal to |
| **Logical**<br>Performs basic boolean operations; combines logical values and produces logical results. | The expression A\|B produces the value True if either A or B are True. If both are False, this expression produces the logical value False. | & | And |
| | | \| | Or |

## Order of Evaluation of Operators, Functions and Parameters

If you combine several operators, functions and parameters in a single expression, Argus ONE performs the operations in the order shown in the following table.

| Operator/Function/Parameter | Description |
|---|---|
| Functions and Parameters | Argus ONE functions and layer parameters. |
| () | Parentheses |
| - | Negation (of a single operand or in parentheses) |
| * and / | Multiplication and division |
| + and - | Addition and Subtraction |
| =, <, >, <=, >=, != | Comparison |
| & and \| | Logical |

If the expression contains more than one operator with the same priority, for example, several comparison operators, Argus ONE evaluates the operators from left to right. If you want to alter the order of evaluation, use parentheses to group expressions. Argus ONE first calculates the expressions in parentheses, and then uses those rules to calculate the full expression as shown in the following table.

| This expression | Produces this value |
|---|---|
| 3+5*4 | 23 |
| (3+5)*4 | 32 |

## Using Linked Parameters

A linked parameter points to a referenced parameter. Links tell Argus ONE which parameter to look in to find the values you want used in an expression. With linked parameters, you can use data stored in different layers of a project in one expression and use one or more referenced parameter's value in several other expressions in parameters.

Referencing parameters is based on the layer name (layer first parameter) and the secondary parameters if any. When you use the Expression dialog to view a parameter, the referenced parameters are displayed using their full name.

# About Functions

In Argus ONE, functions are calculation tools that you can use to perform value-returning, decision-making and action-taking operations automatically. Argus ONE provides a wide variety of functions that perform many different types of calculations. Most of these functions are described in this chapter, while newly added ones, are described in **Supplement** chapters.

Argus ONE has two types of functions: general functions and layer specific functions. You can use general functions in every layer type. For example, the general function Cos() returns the cosine of an angle; you can use this function in any layer type.

Layer specific functions are available only in the appropriate layer. For example, the mesh layer function NumNodes(), returning the number of nodes in a mesh layer, has no meaning in non mesh type layers.

**All of Argus ONE functions can be also used in export templates.**

## Function Categories

Argus ONE functions can be divided into five main categories: mathematical, trigonometric, logical, geometric and layer specific functions. These are described in the following table.

| Category | Action |
|---|---|
| **Mathematical** | Performs mathematical actions on values |
| **Trigonometric** | Performs trigonometric actions on values |
| **Logical** | Returns logical values |
| **Geometric** | Returns location |
| **Mesh type layers** | Returns mesh layer related information |
| Element | Returns element related information |
| Node | Returns node related information |
| **Grid type layers** | Returns grid layer related information |
| Block | Returns block related information |
| **Information type layers** | Returns Information layer related values |
| Contour | Returns contour related information |

## How Function Descriptions Are Organized

To help you find information, function descriptions follow a template consisting of some or all of the following parts:

- **Name** The name of the function as you type it into Argus ONE.
- **Return value** What the function returns, or what actions the function performs, and suggestion for using the function.
- **Syntax** The name of the function, including all required and optional arguments in the correct order.
- **Arguments** A description of each argument in the order of the function's syntax. If the argument is mandatory it appears in bold italic lettering otherwise it appears italic lettering.

## Using Arguments

Arguments are the information that make up an expression and that a function uses to produce a new value or to perform an action. Arguments are always located to the right of the function name and are enclosed in parentheses. Most arguments are expected to be of a certain data type. The argument you give should be either the appropriate type or another type that Argus ONE can convert to the appropriate type.

An argument can be anything that produces a desired data type. For example, the `Power` function, raises a given number to the power of the second given number. You can give the `Power` function any of the following three kinds of arguments, which produces a number:

- A value that is a number (constant), such as:
  `Power(3,2)`
- An expression that results in a number, such as:
  `Power(3,If(ElemOnBoundary(),2,0))`

  Using a function as an argument to a function, as in the preceding example, is called nesting functions. In this example, the `If` function is an argument to the `Power` function. You can nest as many levels of functions as needed.
- A reference to a layer parameter that contains a number or an expression that results in a number, such as:
  ```
  Power(Rainfall*ElementArea(),If(ElemOnBoundary,2,0))
  Power(ElementReplenishment,If(ElemOnBoundary,2,0))
  ```

  The second preceding example is equivalent to the first if the layer parameter ElementReplenishment is defined as:
  `Rainfall*ElementArea()`

# Mathematical Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Abs*(number)*** | ***number*** | Real number<br>Returns the absolute value of ***number.*** |

Example:
`Abs(-5.0)` returns `5.0`

`Abs(layer 1 - layer 2)` returns the absolute of the difference in values between layer 1 and 2 at the point of evaluation.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Div*(number,divisor)*** | ***number***<br>***divisor*** | Real number<br>Real number<br>Returns the integer division INT***(number/divisor)*** rounded towards zero. |

Example:
`Div(7,2)` returns `3.00`
`Div(7,-2)` returns `-3.00`
`Div(-7,2)` returns `-3.00`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Exp*(number)*** | ***number*** | Real number<br>Returns the natural exponent of ***number.*** |

Example:
`Exp(4)` returns `54.5982`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Log*(number)*** | ***number*** | Real number<br>Returns the natural logarithm (ln) of ***number.*** |

Example:
`Ln(54.5982)` returns `4.00`

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **Log10*(number)*** | ***number*** | Real number<br>Returns the decimal logarithm (log) of ***number.*** |

Example:
`Log10(1000.00)` returns `3.00`

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **Mod*(number,divisor)*** | ***number***<br>***divisor*** | Real number<br>Real number<br>Returns the remainder (modulus) after ***number*** is divided by ***divisor.*** The result has the same sign as ***number.*** |

Example:
`Mod(7,2)` returns `1.00`
`Mod(7,-2)` returns `1.00`
`Mod(-7,2)` returns `-1.00`

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **Power*(number,exponent)*** | ***number***<br>***exponent*** | Real number<br>Real number<br>Returns the value of ***number*** raised to ***exponent.*** |

Example: `Power(7,2)` returns `49.00`

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **Sqrt*(number)*** | ***number*** | Real positive number<br>Returns the square root of ***number.*** |

Example: `Sqrt(9)` returns `3.00`

# Trigonometric and Hyperbolic Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ACos*(number)*** | ***number*** | Real, $-1 \le number \le +1$<br>Returns $Cos^{-1}$(***number***) in radians. |

Example: `ACos(-1)` returns `3.14159`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ASin*(number)*** | ***number*** | Real, $-1 \le number \le +1$<br>Returns $Sin^{-1}$(***number***) in radians. |

Example: `ASin(-1)` returns `-1.5708`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ATan*(number)*** | ***number*** | Real number<br>Returns $Tan^{-1}$(***number***) in radians. |

Example: `ATan(412)` returns `1.56837`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ATan2*(number,divisor)*** | ***number***<br>***divisor*** | Real number<br>Real number including zero<br>Returns $Tan^{-1}$(***number/divisor***) in radians. |

Example: `ATan2(2,0)` returns `1.5708`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Cos*(number)*** | ***number*** | Real number in radians, if the angle is in degrees, multiply it by π/180 to convert it to radians.<br>Returns the cosine of ***number.***<br>Cycle: 2π. |

Example: `Cos(3.14)` returns `-0.99999`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **CosH*(number)*** | ***number*** | Real number<br>Returns the hyperbolic cosine of ***number.*** |

Example: `CosH(4)` returns `27.3082`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Sin*(number)*** | ***number*** | Real number in radians, if the angle is in degrees, multiply it by π/180 to convert it to radians.<br>Returns the sine of ***number.***<br>Cycle: 2π. |

Example: `Sin(1)` returns `0.841471`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **SinH*(number)*** | ***number*** | Real number<br>Returns the hyperbolic sine of ***number.*** |

Example: `SinH(3)` returns `10.0179`

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **Tan*(number)*** | ***number*** | Real number in radians, if the angle is in degrees, multiply it by $\pi/180$ to convert it to radians.<br>Returns the tangent of ***number.***<br>***number*** $\neq (2k+1)\pi$, Cycle: $\pi$. |

Example: `Tan(3.141592654)` returns `4.10207e-10`

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **TanH*(number)*** | ***number*** | Real number, or any expression that can be evaluated to a real number.<br>Returns the hyperbolic tangent of ***number.*** |

Example: `Tan(12)` returns `1.00`

# Geometric Functions

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **X*()*** | ***no arguments*** | Returns the X coordinate of the evaluation context. |

Example:
`X()` in an element context, returns the X coordinate of the element center.
In information type layers, returns the X coordinate of a contour's first vertex.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **Y*()*** | ***no arguments*** | Returns the Y coordinate of the evaluation context. |

Example:
`Y()` in an element context, returns the Y coordinate of the element center.
In information type layers, returns the Y coordinate of a contour's first vertex.

# Logical Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **If*(condition,true_val, false_val)*** | ***condition*** | Any value or expression that can be evaluated to True or False. |
| | ***true_val*** | The value returned if ***condition*** is True. |
| | ***false_val*** | The value returned if ***condition*** is False. |
| | | Returns ***true_val*** if ***condition*** evaluates to True and ***false_val*** if it evaluates to False. |

Any number of If functions can be nested as *true_val* and *false_val* arguments to construct more elaborate tests.

Example:
`If(Formation1.Thickness<0,-1,Formation.Thickness)`
This function tests the thickness of a formation at each evaluation context, such as a node, and will return the value -1 for nodes at which the thickness is less than zero. At nodes the thickness is larger than or equal to zero, it will return the formation thickness.

# Search Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Index*(selector,number_1, number_2,numbe_3,..... number_n)*** | ***selector*** | Real number |
| | ***number_1... number_n*** | A series of real numbers. |
| | | Returns the value of an element in a series, selected by ***selector***. |

Example:
`Index(5,42.5,48.9,112.4,5.3,8.9,4.0,44.8)` returns `8.9`

# Action Taking Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Interpolate*(expression (Information Layer parameter))*** | ***expression*** | An expression containing any number of Information type layer parameters.<br>Interpolates all Information type layer parameters contained in the expression. |

Example: `Interpolate(Rainfall*ElementArea)` returns the interpolated value of the information layer “Rainfall” multiplied by the element area, at the evaluation context. In the above example the evaluation context is at the element center’s coordinates. `Interpolate(Conductivity*Thickness)` return the interpolated value of the “Conductivity” layer parameter multiplied by the interpolated value of the “Thickness” layer parameter at the evaluation context.

An expression can hold as many information layer parameters as needed, all parameters’ values will be interpolated before the expression is evaluated.

To read more about Argus ONE’s interpolate function refer to the chapter “Working with Information Layers.”

# Layer Specific Functions

Layer specific functions are layer and object dependent. Layer specific functions are available for mesh and grid type layers. Both are categorized to layer and layer’s objects functions.

## Mesh Layer Functions

Mesh type layers have three types of functions: Mesh layer functions, returning general mesh layer information, element functions, returning element information, and node functions, returning node information.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **MeshArea*()*** | ***no arguments*** | Returns the total area of the elements in a mesh layer. |

If the mesh layer contains more than one mesh, or some disjoined elements, this function returns the total area of all the elements.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NumElements*()*** | ***no arguments*** | Returns the total number of elements in a mesh layer. |

If the mesh layer contains more than one mesh, or some disjoined elements, this function returns the total number of elements in the layer.

Note: Used mainly in Export templates to loop over the number of elements.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NumElemParameters*()*** | ***no arguments*** | Returns the number of element type parameters of a mesh layer. |

Note: Used mainly in Export templates to loop over the number of element related parameters.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NumNodeParameters*()*** | ***no arguments*** | Returns the number of node type parameters of a mesh layer. |

Note: Used mainly in Export templates to loop over the number of node related parameters.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NumNodes*()*** | ***no arguments*** | Returns the total number of nodes in a mesh layer. |

If the mesh contains more than one mesh, or some disjoined elements, this function returns the total number of nodes in the layer.

Note: Used mainly in Export templates to loop over the number of nodes.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NumParameters*()*** | ***no arguments*** | Returns the number of mesh layer related parameters. |

Note: Used mainly in Export templates to loop over the number of mesh layer related parameters.

## Mesh Layer, Element Functions

These functions can be invoked only for element related parameters. To read more about setting mesh layer parameter types, refer to the chapter "Editing and Refining a Mesh."

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ElementArea*()*** | ***no arguments*** | Returns the area of the element. |

Example: `(Rainfall*ElementArea())` if rainfall is an information layer describing the rainfall distribution, returns the replenishment rate (mass) contributed by the element.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ElementNumber*()*** | ***no arguments*** | Returns the number of the element. |

Note: Used mainly in Export templates to export the element number.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ElementOnBoundary*()*** | ***no arguments*** | Returns True (1) if one or more of the element's nodes lie on the perimeter of the domain occupied by elements, otherwise returns False (0). |

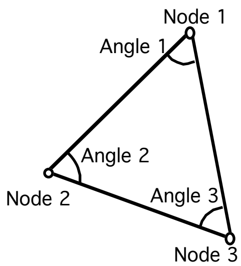
Example: `ElementOnBoundary()` returns 1 if the element is a boundary element, otherwise returns 0. Can be used in an Export template using the export template If statement to export only boundary elements.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **GCenterX*()*** | ***no arguments*** | Returns the X coordinate of the element's center of gravity. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **GCenterY*()*** | ***no arguments*** | Returns the Y coordinate of the element's center of gravity. |

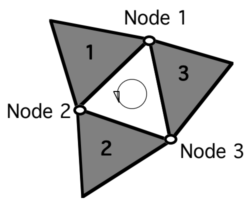
| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **MinAngle*()*** | ***no arguments*** | Returns the element's minimal angle in degrees. |

Example: Create an element parameter and assign it the MinAngle() function. Use the color palette to view the distribution of elements by minimum angle.




| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthAngle*(angle_number)*** | ***1, 2, or 3*** | Returns the element's ***angle_number*** angle in degrees. |

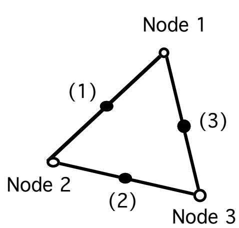
Example: `NthAngle(1)` returns the value of the elements' first angle corresponding with the element's first node.




| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthNeighbourNum*( neighbour_num)*** | ***1, 2, or 3*** | Returns the number of the element's ***neighbour_num*** neighboring element. Returns 0 if there is no such element. Returns zero for all three arguments if the element is disjoined. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthNodeNum*(node_num)*** | ***1, 2, or 3*** | Returns the node number of the element's ***node_num.*** |

Note: Used mainly in Export templates to export the element's node numbers.




| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthSideCenterX*(side_num)*** | ***1, 2, or 3*** | Returns the X coordinate of the element's ***side_num*** side midpoint. |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NthSideCenterY*(side_num)*** | ***1, 2, or 3*** | Returns the Y coordinate of the element's ***side_num*** side midpoint. |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NthSideLength*(side_num)*** | ***1, 2, or 3*** | Returns the length of the element's ***side_num*** side. |

## Mesh Layer, Node Functions

These functions can be invoked only for node related parameters. To read more about setting mesh layer parameter types, refer to the chapter "Editing and Refining a Mesh."

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **EffectiveArea*()*** | ***no arguments*** | Returns the node's effective area. Node effective area is defined as the sum of third of the area of each of the elements connected to the node. |

Example: `(Rainfall*EffectiveArea())` if rainfall is an information layer describing the rainfall distribution, returns the replenishment rate (mass) contributed by the node.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NodeNumber*()*** | ***no arguments*** | Returns the number of the node. |

Note: Used mainly in Export templates to export the node's number.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NodeOnBoundary*()*** | ***no arguments*** | Returns True (1) if the node lies on the perimeter of the domain occupied by elements the node belongs to, otherwise returns False (0). |

Note: Can be used in an export template using the export template script If statement to export only boundary nodes.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **PositionX*()*** | ***no arguments*** | Returns the node's X coordinate. |

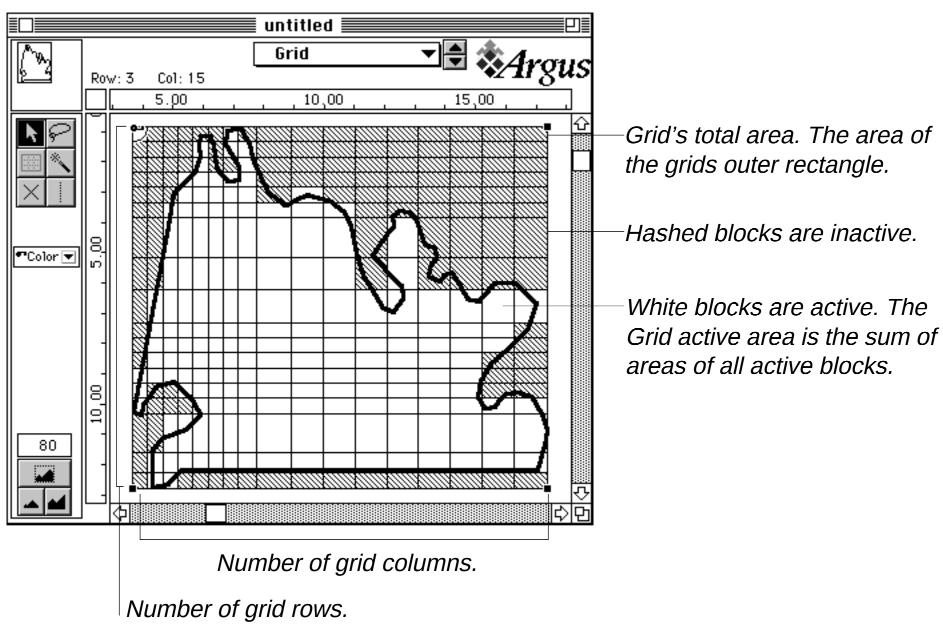
Note: Used mainly in Export templates to export the node's X coordinate.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **PositionY*()*** | ***no arguments*** | Returns the node's Y coordinate. |

Note: Used mainly in Export templates to export the node's Y coordinate.

## Grid Layer Functions

Grid type layers has two types of functions: Grid layer functions, returning general grid layer information and block functions, returning block information.



| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ActiveArea*()*** | ***no arguments*** | Returns the total area of active blocks in the current grid layer. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NumColumns*()*** | ***no arguments*** | Returns the number of grid columns. |

Note: Used mainly in Export templates to loop over the number of grid columns.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NumParameters*()*** | ***no arguments*** | Returns the number of grid related layer parameters. |

Note: Used mainly in Export templates to loop over the number of grid layer related parameters.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NumRows*()*** | ***no arguments*** | Returns the number of grid rows. |

Note: Used mainly in Export templates to loop over the number of grid rows.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **TotalArea*()*** | ***no arguments*** | Returns the total grid area. |

Example: The expression `(TotalArea()-ActiveArea())` returns the area of all inactive blocks in a grid.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthColumnPos** ***(col_num)*** | ***col_num*** | ***col_num***'s grid line coordinate.<br>To get ***col_num*** use the function **Column()**.<br>In a grid export template ***column*** can be obtained by calling the macro variable $Column$ within a loop over columns.<br><br>Returns ***col_num***'s grid line coordinate.<br><br>When in "grid centered" grid, this coordinate is identical to the block center's X coordinate.<br><br>When in "block centered" grid, this coordinate is the minimum X coordinate of ***col_num*** (or block) depending on the coordinate system direction. |

Example: Used mainly in grid Export templates to export grid lines coordinates.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthRowPos** ***(row_num)*** | ***row_num*** | ***row_num***'s grid line coordinate.<br>To get ***row_num*** use the function **Row()**.<br>In a grid export template ***row_num*** can be obtained by calling the macro variable $Row$ within a loop over columns.<br><br>Returns ***row_num***'s grid line coordinate.<br><br>When in "grid centered" grid, this coordinate is identical to the block center's Y coordinate.<br><br>When in "block centered" grid, this coordinate is the minimum Y coordinate of ***row_num*** (or block) depending on the coordinate system direction. |

Example: Used mainly in grid Export templates to export grid lines coordinates.

## Grid Layer, Block Functions

These functions can be invoked only for block related parameters. To read more about setting grid layer parameter types, refer to the chapter “Creating and Editing a Grid.”

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockArea*()*** | ***no arguments*** | Returns the block’s area. |

Example: `(Concentration*BlockArea())` if Concentration is an information layer describing the contaminant concentration distribution, the expression returns the contaminant mass contributed by the block.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockCenterX*()*** | ***no arguments*** | Returns the block’s center X coordinate. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockCenterY*()*** | ***no arguments*** | Returns the block’s center Y coordinate. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockIsActive*()*** | ***no arguments*** | Returns True (1) if block is active, otherwise returns False (0). |

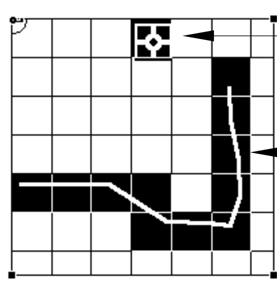
| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockOnBoundary*()*** | ***no arguments*** | Returns True (1) if block has at least one side on the grid edge, or block is inactive. Otherwise returns False (0). |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Column*()*** | ***no arguments*** | Returns the column number the block belongs to. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **CountObjectsInBlock*( layer_name,*** *contour_type**)*** | ***layer_name*** | The name of the layer for which you need to count the number of objects of type *contour_type* intersecting the block or contained in it. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the number of objects of type *contour_type* in layer ***layer_name***, intersecting or contained in the block. If *contour_type* is omitted, returns the total number of all objects intersecting or contained in the block. |

Example: The function `CountObjectsInBlock(Domain Layer)` counts the number of objects in the layer "Domain Layer" intersecting or contained by the block and returns:
0 — if the block is not intersected by, or contains no objects.
1 — if the block is intersected by, or contains one object of any type.
N — if the block is intersected by, or contains N objects of any type.

Example: The function `CountObjectsInBlock(Domain Layer,1)` counts the number of open contours in the layer "Domain Layer" intersecting or contained by the block and returns:
0 — if the block is not intersected by, or contains no open contours.
1 — if the block is intersected by, or contains one open contour.
N — if the block is intersected by, or contains N open contours.



*This highlighted block on top of the point object will return the value 1 for the function*
`CountObjectsInBlock(Domain Layer,0)`

*These highlighted blocks along the open contour will return the value 1 for the function*
`CountObjectsInBlock(Domain Layer,1)`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthBlockSideLength*( h_v_side)*** | ***h_v_side*** | Block's horizontal or vertical side<br>1 for vertical<br>2 for horizontal<br><br>Returns the ***h_v_side*** length of the block. |

Example: `NthBlockSideLength(1)` returns the block's vertical length.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NumBlockParameters*()*** | ***no arguments*** | Returns the number of block and grid related layer parameters. |

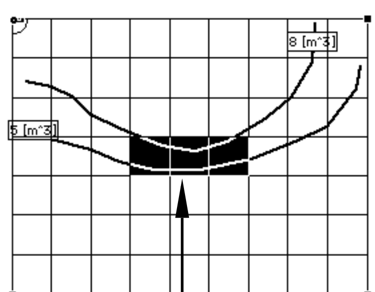
Example: Used mainly in Export templates to loop over the number of block layer related parameters.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Row*()*** | ***no arguments*** | Returns the row number the block belongs to. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **SumObjectsInBlock*( parameter_name,*** *contour_type****)*** | ***parameter_name*** | The name of an information type layer parameter for which you need to sum the values of objects of type *contour_type* intersecting the block or contained in it. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the sum of the objects' values of type *contour_type* in parameter ***parameter_name***, intersecting or contained in the block. If *contour_type* is omitted, returns the sum of values of all objects intersecting or contained in the block. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **WSumObjectsInBlock*( parameter_name,*** *contour_type* **)** | ***parameter_name*** | The name of an information type layer parameter in which you need to calculate the weighted sum of objects of type *contour_type* intersecting the block or contained in it.<br>The weighted sum is defined as the sum of the products of each intersecting segment length, multiplied by its value. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the weighted sum of objects' values of type *contour_type* in parameter ***parameter_name***, intersecting or contained in the block. If *contour_type* is omitted, returns the weighted sum of values of all objects intersecting or contained in the block. |

Example: The function `WSumObjectsInBlock(Fractures Layer,1)` computes the product of each intersecting or contained segment length by its value and sums them.




*If contour_type is 0 or if it is omitted and a point object is encountered,* `WSumObjectsInBlock` *returns the value of the point object.*

*In these highlighted blocks the function* `WSumObjectsInBlock(Fractures Layer,1)` *returns the sum of the two products of the length of intersecting segments multiplied by their contour value.*

# Creating and Editing Expressions

You create and edit expressions in the *expression dialog*. The expression dialog can be invoked from the layers dialog and from the export template dialog. In both it allows you to access all operators, functions and layer parameters.

Using the expression dialog you can create complex expressions without typing, thus reducing typing errors in your expressions to minimum. However, if you prefer, you can still enter a whole expression by hand, or paste them from another text editor.

## The Expression Dialog

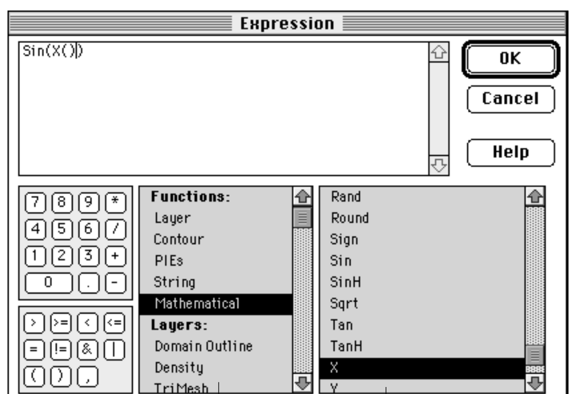
The expression dialog is like a small calculator. It is made up of three panels and two lists allowing you to access all the ingredients necessary to create an expression. When you finish creating an expression and click the OK button, the dialog tests the expression you've created to see if it is valid, that is, if it is free of syntax errors and that functions and parameters you used do exist.

The three panels are: the calculator panel, the logical operators panel, the and the expression editor panel.

*The expression editor panel.*

*The calculator panel.*

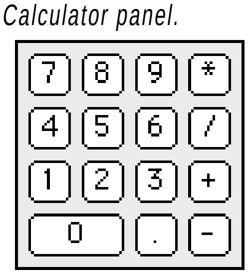
*The logical operators panel.*




*The Functions Groups and Layers list.* *The Functions and parameters list.*

### The calculator panel

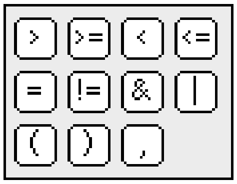
*Calculator panel.*




This panel contains the digits, the period, and the four basic arithmetic operators. You use this panel by clicking the mouse in the appropriate buttons. Each button you click produces the appropriate digit or operator in the expression editor at the insertion point. You can use the keyboard instead of clicking the buttons in the calculator panel.

To enter the expression: 7 + 3 click the 7 button, than the + button and than the 7 button.

## The logical operators panel




*Logical operators panel.*

This panel contains the logical operators. You use this panel by clicking the mouse in the appropriate buttons. Each button you click produces the appropriate operator in the expression editor at the insertion point. You can use the keyboard instead of clicking the buttons in the panel.

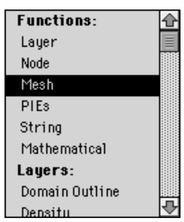
To enter the expression: 7 >= 3 click the 7 button, than the >= button and than the 7 button.

## The Functions, Layers and Parameters Lists

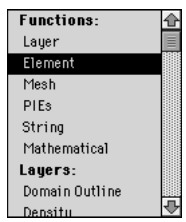
Two lists enable you to access all your layer parameters and Argus ONE functions. You access parameters and functions by first selecting the layer or function group they belong to, from the Function-Groups and Layers list, and then selecting them from the Functions and Parameters list.

### The Functions-Groups and Layers list

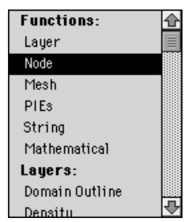
This scrolling list presents all the available groups of functions and layers. The available Function-Groups change according to the parameter type you are creating an expression for. To Select a function or a layer parameter click the group or layer they belong to.



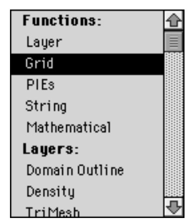

*Mesh layer. A mesh parameter expression is edited.*



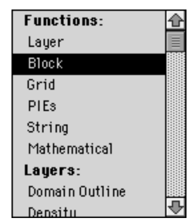

*Mesh layer. An element parameter expression is edited.*




*Mesh layer. A node parameter expression is edited.*



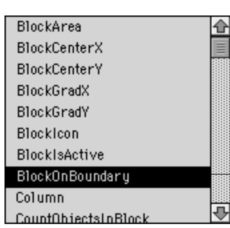

*Grid layer. A grid parameter expression is edited.*




*Grid layer. A block parameter expression is edited.*

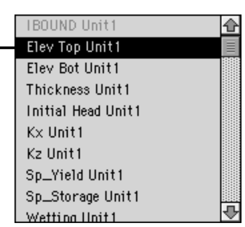
### The Functions and Parameters list

This scrolling list presents all the functions or parameters belonging to the functions-group or layer selected in the Functions-Groups and Layers list. To insert a function or a parameter into the Expression Editor Panel double-click the function or parameter. It is inserted at the insertion point.



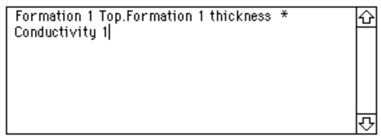

*Double-click a parameter to insert it in the expression editor.*

*Double-click a function to insert it in the expression editor.*



### The expression editor panel

The expression editor is where the expression you create is presented. It is a text editor in a list dialog allowing you to scroll up and down to see the full length of the expression. You can use all the keyboard characters, cursor keys, DELETE key and copy and paste keyboard shortcuts. This enables you to copy and paste a whole expression, or part of an expression from an external source or from another parameter expression.



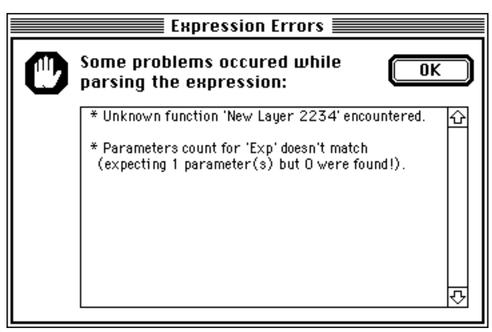

## Expression Validity Testing

When you OK the expression dialog Argus ONE interprets the expression you entered and searches for expression errors. If it detects errors it puts up a dialog explaining the errors. Each error description starts with an * followed by the function, parameter or operator Argus ONE can not resolve, followed by a description of the problem.

For example, in the following expression the layer parameter “Layer 2234” does not exist, and the arguments of the function Exp() are missing.


*Erroneous expression.*

The error messages are presented in the following dialog.




*Each error description starts with an asterisk and can span over several lines.*

# Infinities and Non Numbers

An expression, might, when evaluated, yield infinity or a meaningless result. For example, an expression which is defined as the division of two parameters, could be evaluated to infinity, if the denominator parameter becomes 0 (zero), or to a meaningless result, if both parameters are 0 (zero). If the evaluation result is infinity or meaningless, Argus ONE assigns the parameter the $NaN value, where NaN stands for Not-a-Number.

## $NaN Resulting Operations

When the result of a parameter's evaluation is meaningless or infinity, Argus ONE returns the string $NaN. The following operations cause Argus ONE to return $NaN:

- A mathematical operation on NaNs
- Divide by zero
- Mod() function, where the second argument is 0 (zero)
- Sqrt(), Log() and Log10() functions, where the argument is a negative number
- Tan() function, where the argument is an integer multiple of π/2
- ASin() and ACos() functions, where the argument is not in the range -1 to +1
- ATan2() function, where both arguments equal zero

# Not Available Values

You may assign any parameter and object's parameter the $N/A (Not Available) value.

## Assigning $N/A to the Default Parameter Value

Assigning a parameter $N/A as its default value, is especially useful when the parameter's layer uses the "Exact Contour" interpretation method. For example, to have a layer describing boundary conditions, set the layer to the "Exact Contour" method and assign it $N/A as its default value. Then, describe the different boundaries and their values using contours. An object probing this layer will return the value of a boundary condition, if it lies above a boundary, and $N/A if it does not.

## Assigning $N/A to an Object's Parameter

You may also assign $N/A to an object's parameter. This is useful when some objects in a layer do not have a value for some of the parameters assigned to them. For example, consider a layer describing temperature and humidity measurements read by a number of gauges. If some gauges did not measure the humidity, you can assign their humidity parameter the $N/A value.

# Opening and Saving Your Projects

# Overview

To work on a project, you open the project to display it on your screen. You can edit the project, edit the mesh and grid, and print it. To preserve your work, you will need to save the project. Saving a project stores the version of the project you see on the screen in a file on a disk. Be sure to save your projects regularly. Also note that since Argus ONE files are cross platform binary compatible, you can easily transfer your files to your colleagues, or work on the same project on the different platforms you have Argus ONE installed on.

# Opening a Project

You can open a project in a few ways:

### From within Argus ONE

- From the File menu, choose the Open command.

### From the desktop or the shell

All the operating systems Argus ONE is supported on, allow you to open a file or a number of files from their desktop or the shell. If you are not familiar with opening a file from your desktop or shell, refer to your operating system documentation.

When you open a project stored on the disk, Argus ONE loads it from the file on the disk into memory. As long as you work on an open project, Argus ONE maintains the file in memory. The copy on the disk is not changed until you save the project.

## Memory Considerations

Argus ONE's memory requirements are relative to the size of your project. Each object you add to the project, such as a layer, a contour, an element, and a grid, increases the memory Argus ONE needs.

To open a project, Argus ONE must have sufficient memory to read the entire project into memory. If Argus ONE does not have enough memory, it will alert you of the problem.

Memory shortage can happen while you open a large file, or while you work on the project and create objects in it. You can increase the memory allocated to Argus ONE. Increasing the memory available for an application differs among the various operating systems and computing environment.

### To increase the memory allocated to Argus ONE

On a Macintosh you can set the memory allocated to the application in the Finder. On MS Windows and Unix Workstations you will have to increase the page file size. For a detailed discussion of memory usage refer to your computing platform documentation.

## Opening Multiple Projects

Argus ONE allows you to open as many project files as you need. Each project is presented in a different Argus ONE window. You can copy graphic objects and contours from one project window to another.

### To open another project

1. From within any of Argus ONE current open windows, choose the Open command in the File menu.
2. In the Open dialog box, choose the file and click the OK button.

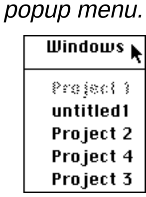
### To move between open project windows

If you have a number of open Argus ONE windows at the same time, you can bring any of them to be the front active window by clicking in it (Macintosh) or clicking on the window’s frame (other platforms). If some of the windows are out of sight, beneath others, you can use the Window List popup menu, to bring any of the windows to the front.

#### To activate a window using the Window List popup menu

1. **Macintosh** - While holding the COMMAND key click the mouse anywhere within an Argus ONE window.
   **PC MS Windows** - Click the right mouse button while holding the CTRL key anywhere within an Argus ONE window.
   **Unix** - Click the middle mouse button anywhere within the window.
   The window list menu pops up.

*The window list popup menu.*




*The currently active window name is dimmed.*

2. From the popup menu, choose the required window name.
   The window is sent to the front.

# Saving a Project

When you open a project, Argus ONE copies it from the disk and displays the copy in a project window. As you work, you're actually making changes to the copy of the project temporarily stored in memory on your computer. To keep your latest work safely on the disk, you should periodically save your project. You should also save backup copies of important projects.

A good rule of thumb is to save every 10 to 15 minutes, or after you've completed any work you wouldn't want to redo. If you save frequently, you won't lose much work if a power failure or other problem occurs. It is also a good habit to save a project before you make a major change to it, such as re-meshing or refining the mesh.

When you save a project, the project remains open on your screen so you can continue working. If you're finished with a project, close it to prevent accidental change to it. When you quit Argus ONE, if any changes to the project have not been saved, an alert is brought up to ask you if you want to save these changes.

## To Save a Project

The first time you save a project, you name the project and tell Argus ONE where you want the project stored. Each time you save the project thereafter, the current version of the project on your screen replaces the version on the disk. The project remains open so you can continue working.

**1.** From the File menu, choose Save.

Argus ONE displays the Save As dialog box.

**2.** Type the project's name in the Save as box.

File naming conventions on the different computing platforms vary. On MS Windows running under DOS, file naming is very limited. If you intend to send your files to a colleague working on MS Windows under DOS, keep in mind that your file names will be truncated to eight characters followed by a period, followed by three additional characters.

**3.** Open the directory/folder you want to save the current project in.

**4.** Click the Save button.

If you assign a name of a project already on the disk or in the current folder, Argus ONE asks you if you want to replace the existing project. Choose No to preserve the existing project, and then type a different name. Choose Yes to replace the existing project with the one on the screen.

## To Rename a Project or Copy It to Another Location

When you choose the Save command, Argus ONE saves the open project under the name and location you last gave it using the Save As dialog box. Use the Save As command to create more than one version of the project, or to save copies on another disk for safekeeping. You can save each version under a different name, or save them under the same name in different folders or on different disks.

1. From the File menu, choose Save As.
2. If you want to rename the project, type a new name in the Save as box.
3. Open the folder/directory or select the disk where you want to store the project.
4. Choose the Save button.

The original version of the project is closed, and the new project is opened. The new project contains any changes you made since saving the original project. The original version is closed without saving the changes you made since last saving it. To keep these changes save the original project before saving it under a different name and/or location.

## To Close a Project

- From the File menu, choose Close.
  Or
- Macintosh: Click the close box in the upper-left corner of the project window.
  Other Platforms: Double-click the close box in the upper-left corner of the project window.

If you have made changes to the project since saving it, Argus ONE displays a message asking you if you want to save the changes.

- Choose Yes to save changes.
- Choose No to discard changes.
- Choose Cancel to leave the project open without saving it.

## To Save an Open Project as You Quit Argus ONE

- From the File menu, choose Quit.

If you've made changes to the project since you last saved it, Argus ONE displays a message asking you if you want to save these changes.

- Choose Yes to save the changes or No to discard them.
  Or
- Choose Cancel to continue working on this project.

# Saving a Condensed Version of a Project

## What Is a Condensed Project?

A condensed version of a project is a version of the project saved without the meshes and/or grids. A typical project might constitute of several mesh layers and/or grid layers you created to describe different realization of the problem under investigation. Each mesh or grid layer might contain thousands of elements and/or grid blocks. The disk space needed to store meshes and grids is very large compared to the disk space required for storing information and domain type layers and their objects (contours). For instance, a project containing a mesh of 1,000 elements and 14 layers might occupy four times the disk space as the same project saved without the mesh.

## When to Save a Condensed Project?

1. If you need to transfer a project to a colleague who also owns a copy of Argus ONE through an electronic mail system.
2. If you are required, by a court of law for instance, to archive your data.
3. If you are required to be able to reproduce your results at any time in the future.
4. If you finished working on a project and are short of disk space.
5. If you want to carry with you many projects on a portable.

To recreate your data at any time, you can save only information, domain and maps type layers without saving the meshes and grids. When you or your colleague wants to work with the mesh or grid again, all you or he/she has to do is click the magic wand within the domain outline contour to re-mesh or re-grid. The mesh or grid will then be recreated, and elements and nodes will automatically be assigned their values from the user layers contours.

---

**Important note:** If however, you have extensively edited the mesh and added node, element and block information, deleting the mesh or grid will erase all this information. This is another very good reason not to use specific element, node or block information.

---

### To save without the mesh

1. Save a copy of the project under a new name.
2. Delete meshes and grids.
3. Save the project.

You should now have a much smaller version of the project.

# Exporting a Project

# Export Templates - Overview

The development and use of export templates is only available through the use of the **Programmable Export Module**. All Plug-In Extensions (PIEs) make extensive use of the **Programmable Export Module** but you can use them without having to purchase this module.

Export templates give you control over your export file format. This enables you to directly export the information you’ve created in Argus ONE to any numerical model you use in the exact file format the model requires.

In an export template you can specify into which files you want to redirect the output, which parameters you need to export, set the delimiters, set the format of a parameter, loop over nodes, elements and blocks, calculate new expressions during export, set the line length, etc.

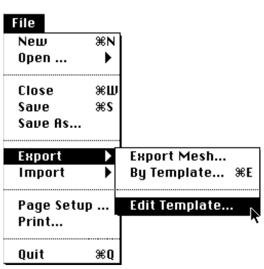
You can name each export template you create, save it to disk, and load it when needed. This enables you to maintain as many export templates as you need for the various models or versions of a model you use.

The export template is created using the *Template Editor* dialog. The template is written in a script like language. You can either use the dialog buttons to construct an export script, or write an export script in you favorite text editor and load it later into Argus ONE.

You can create dynamic templates like those used by Argus ONE. Dynamically defined templates are constructed of script commands and script commands arguments that are resolved during the export process. For instance, if you use the command “Loop over Nodes” in a script, Argus ONE will loop over the number of nodes in your mesh layer while exporting it. This way you do not have to worry about the validity of your export template, it will always be valid, no matter how many nodes the mesh is made of.

# Creating and Editing an Export Template

An export template is layer type dependent. You can create export templates for mesh and grid type layers. An Argus ONE project contains an export template for each of these two layer types which is created when you create a new project,.




### To open the template editor dialog

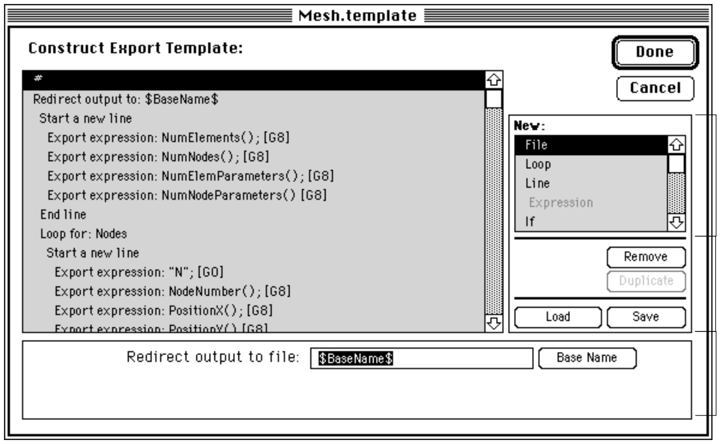
**1.** Make sure a mesh or a grid layer is the active layer.

If it is not, click on the appropriate arrow in the layer selection button until you enter a mesh or a grid layer.

**2.** From the File menu, choose the Export menu item, and from the submenu choose Edit Template...

The Template Editor dialog box opens.

*Template's Name*




*Script Commands control panel.*

*Script Editing control panel.*

*Argument Editing control panel.*

*The Template Editor dialog box showing a default mesh export template.*

## To edit a template

You can either edit an existing template or create your own template. Using a pre-defined template as the basis for your template is an easy way to create your first templates. Editing templates in the Template Editor dialog is executed by clicking the appropriate buttons in the various control panels.

### To select a command

Selecting a command tells Argus ONE that you wish to edit that command or to add another command after the selected command.

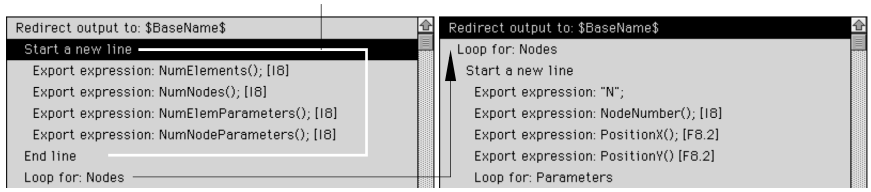
- Click the command's line, or if it's a block command click its first or last line.

You can edit the command arguments, change the command's location within the script, remove the command or duplicate it.

### To remove a command

- Select the command line. If the command is a block command, select the command's first or last line.
- Click the Remove button in the Script Editing control panel.

*Click the Remove button to remove this command.*




*The command as well as all the commands it contains are removed.*

The command and all the commands its contains are removed.

### To create a new mesh or grid export template

1. Click the first line in the template to select it.
2. Click the Remove button in the *Script Editing control panel*.
   If the template contains more than one “Redirect output to:” commands, repeat the last two steps for all file commands.
   The default template is cleared, so that you can start creating your own template.
3. Using the buttons in the *Script Commands control panel* create your script.

### To save your export template


1. Click the Save button in the Script Editing control panel.
   The Save Template As dialog opens.
2. Type the name under which you wish to save the template.
3. Click the OK button.

The template is saved in an ASCII file. You can load this file using the Load button.

### To end template editing


- Click the Done button in the Template Editor dialog box.

If you click the Done button and did not save the template first, the template is saved within the project until you quit the application or close the project window. If you do save the project the template is saved with it.

### To load a template from a disk file


1. Click the Load button in the Script Editing control panel.
   The Open File dialog box opens.
2. Locate the template file on your disk, and choose Open.
   The new template is read and added after the selected line.

To replace the current template by the one read from the disk, remove the existing template.

### Script validity checking

When you create and edit an export template script using the Template Editor you can not create syntax errors since the commands are automatically inserted in the right syntax and order. The buttons for invalid commands are dimmed to keep you from introducing mistakes. For instance, while editing the block command “Start a new line”, the Line button is dimmed to prevent you from writing a line within a line.

When you load an export template script from disk, its syntax and structure are checked for you. If a syntax error or a structure problem are detected, template loading is continued. All script commands, but the offending ones are loaded and presented in the dialog.

No validity checking is performed over commands arguments or expressions. These are resolved and evaluated only during the actual export process. If an error is detected at this stage, the export will be terminated and the export file will include all data that was successfully interpreted and evaluated.

## Creating Export Template Scripts

An export template script is made of script commands. These commands allow you to specify the file you redirect your output to, the parameters to be exported, etc. A script command may have command arguments. It can be either a block script command or a one line command. A block script command can contain one or more other commands. For instance, the command to create an output file and redirect the output to that file is:

```
Redirect output to: $Basename$
     .
     Other Command
     .
     Other Command
     .
End File
```

The commands within a block command are indented. Arguments can be explicit or dynamic. A dynamic argument is resolved when the script is interpreted, that is, during the actual export process.

Newly added script commands and export macros are listed in the **Supplements** chapters.

## Script Commands

| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **File** | **Redirect output to:**<br>**End File** | ***Filename*** **or** ***$Basename$*** |

- This command can be nested in another File command.
- This command is a block command.

This command redirects the output of all commands contained within it, to the file specified as the command's argument. This command's argument can be a dynamic argument or a specific name.

The command "`Redirect output to:Myfile.dat`" always redirects the output to the file "`Myfile.dat`".

The command "`Redirect output to:$Basename$`" redirects the output to a file its name you will specify while exporting. If for instance you need to export node related information to a file with an extension `.nod` and element related information to a file with an extension `.elm`, you could write the following script:

```
Redirect output to: $Basename$.nod
     Node Command
     Node Command
End File
Redirect output to: $Basename$.elm
     Element Command
     Element Command
End File
```

If during export you set the filename to be Myproj1, node related information will be written to the file `Myproj1.nod` and element related information will be written to the file `Myproj1.elm`.

| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **Line** | **Start a new line:**<br>**End Line** | |

- This command can not include another Line command.
- This command is a block command.

This command redirects the output of all commands contained within it, to the a new line.

```
Start a new line
     Expression
End line
```

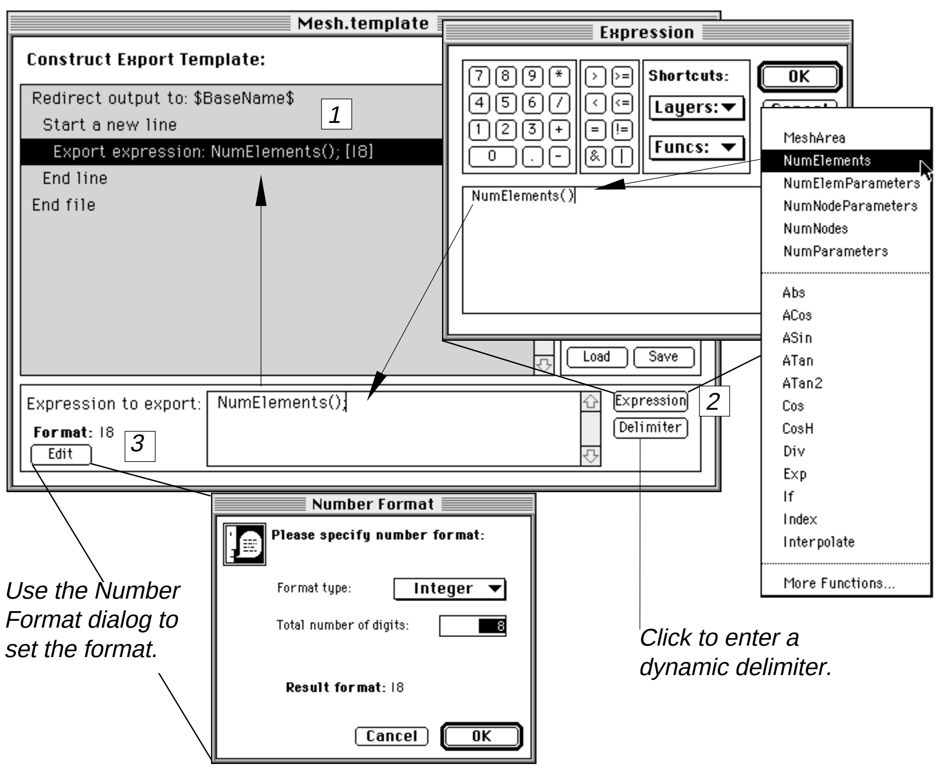
| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **Expression** | **Export expression** | ***Delimiter*** **or** ***Argument [Format]*** |

This command exports an expression in the specified format. For instance, to export the number of nodes, click the Expression button and type the argument NumNodes(), or choose it from the Funcs popup menu.
`Export expression: NumNodes()[i5]`

An expression can be either a literal, that is, any string, or a mathematical expression. A mathematical expression can include any valid Argus ONE function or parameter. Argus ONE functions and parameters include mesh, node, element, and grid functions, mathematical and logical operators, mathematical functions, layer parameters and constants.

### To insert an expression using the expression dialog

1. Click the Expression button in the Script Commands control panel.
   A new script line is created.
2. Click the Expression button in the Argument Editing control panel.
   The Expression dialog box opens.




3. Click the Edit button in the Arguments Editing control panel to set the expression format.

### To edit an expression

You can either enter an expression manually by typing it in the Argument Editing control panel, or use the Expression dialog to create it. The latter is safer since you avoid syntax mistakes. Creating an error free export template is very important since the template is interpreted only during the actual export is in process, and if erroneous, it might result in corrupted export files.

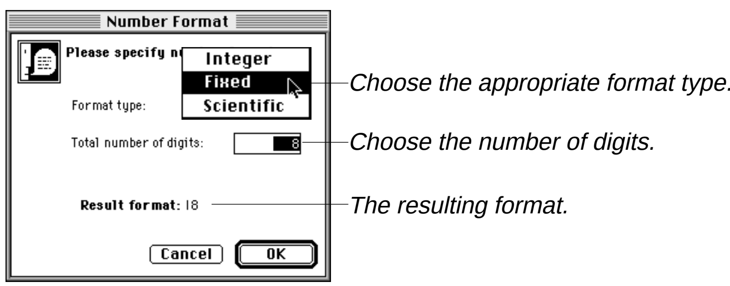
### Using the Expression dialog

The expression dialog is the same expression dialog you use to create parameter expressions in the Layers dialog. For a detailed discussion of the Expression dialog refer to chapter 2 "Working with Information Layers."

### Using the Number Format dialog

Each expression must be specified a format. When you create an expression, Argus ONE defaults to [F8.2]. To change the format you can either type it yourself or use the Number Format dialog. It is recommended that you use the dialog unless you are certain you are familiar with the format syntax.

1. Click the Edit format button in the bottom left corner of the Arguments Editing control panel.
   The Number Format dialog appears.




2. From the format popup menu, choose the required format type.
3. Enter the number of digits.
4. Click the OK button.

### Number Formats

The three available format types, are the Fixed, Integer and Scientific formats.

The total number of digits field defines the total number of places an expression can occupy, starting at the beginning of the line, or after an expression or delimiter before it, and up to the first character of the next field or the next delimiter. This include minus and plus signs, periods and the "E" character if using a scientific format.

If the total number of digits is smaller than available, the numbers are padded by spaces from left.

If the order of magnitude of the number is larger than the specified total number of digits, Argus ONE adds the necessary number of places to present the number. If your simulator does not use free format reading, you will have to design your format to match the simulators input format such that Argus ONE will not add digits.

The number of fraction digits field defines the total number of digits after the period.

| With this format | This Number | Is displayed this way |
|---|---|---|
| I5 | 12345 | `12345` |
| | -123 | `␣-123` |
| | 1234567 | `1234567` |
| F8.2 | 10.42 | `␣␣␣10.42` |
| | 123.4567 | `␣␣123.45` |
| E10.2 | 123.4567 | `␣␣1.23E+02` |
| E10.6 | 123.4567 | `1.234567E+02` |

### Setting delimiters between arguments in a line

The input format to your numerical model contains lines describing more than one variable. Some of the input formats require that multiple fields in a line be delimited. A delimiter may be any character or group of characters inserted between fields in a line. Argus ONE gives you full control of the delimiter character. You can set any character or group of characters to be the delimiter, you can have different delimiters at different places in a line and in a file, or you can decide not to use delimiter between certain fields.

To create an explicit delimiter just insert an Export Expression script line and specify the delimiter as literal text, that is, text between quotes.

To use a dynamic delimiter, just insert the “;” character after the expression, or click the Delimiter button in the Argument Editing control panel. The dynamic delimiter is replaced by the delimiter you set in the General Export Parameters dialog box.

| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **Loop** | **Loop for:**<br>**End loop** | ***Elements, Element Parameters, Nodes, Node Parameters, Parameters, Blocks, Block Parameters, Columns or Rows, Variable*** |

- This command is a block command.

The Loop command enables you to loop over a group of objects and export their values. For instance, to create a simple script looping over elements and exporting their numbers, their area and their node numbers:

### Looping over elements

1. Click the Loop button.
2. In the Arguments Editing control panel choose Elements from the popup menu.
3. Click the Line button to create a new line.
4. Click the Expression button in the Script Commands control panel to create a new script line.
5. Click the Expression button in the Arguments Editing control panel to invoke the Expression dialog.
6. From the Functions popup menu choose “ElementNumber()” function.
7. Click OK to close the Expression dialog.
8. Click the Edit button in the Arguments Editing control panel to edit the format, choose Integer from the popup menu and set the desired format and click OK to confirm your choice.
9. Click the Delimiter button in the Arguments Editing control panel to insert a dynamic delimiter.
10. Repeat steps 4-9 for each additional expression.
    Your script should look as follows:

```
Loop for: elements
    Start a new line
        Export expression: ElementNumber();[I8]
        Export expression: ElementArea();[F10.4]
        Export expression: NthNodeNum(1);[I5]
        Export expression: NthNodeNum(2);[I5]
        Export expression: NthNodeNum(3)[I5]
    End line
End Loop
```

This loop will loop over the mesh elements and export “elements” number of lines containing the element number, element area and the element’s first, second and third node numbers.

#### Looping over nodes

```
Loop for: nodes
    Start a new line
        Export expression: NodeNumber()[I8]
    End line
End Loop
Loop for: nodes
    Start a new line
        Export expression: PositionX()[F10.5]
    End line
End Loop
Loop for: nodes
    Start a new line
        Export expression: PositionY()[F10.5]
    End line
End Loop
```

Each of these three loops will loop over the mesh nodes and export “nodes” number of lines. The first “nodes” number of lines containing the node number, the second “nodes” number of lines containing the node’s X coordinate, and the third “nodes” number of lines containing the node’s Y coordinate.

#### Looping over parameters

Looping over parameters enables you to export all the parameters linked to nodes or elements by using a single script command.

1. Click the Loop button.
2. In the Arguments Editing control panel choose Elements from the popup menu.
3. Click the Line button to create a new line.
4. Click the Expression button in the Script Commands control panel.
5. Click the Expression button in the Arguments Editing control panel to invoke the Expression dialog.
6. From the Functions popup menu choose “ElementsNumber()” function.
7. Click OK to close the Expression dialog.
8. Click the Edit button in the Arguments Editing control panel to edit the format, choose Integer from the popup menu and set the desired format and click OK to confirm your choice.
9. Click the Delimiter button in the Arguments Editing control panel to insert a dynamic delimiter.
10. Click the Loop button.

11. In the Arguments Editing control panel choose Element Parameters from the popup menu.
12. Click the Expression button in the Script Commands control panel.
13. In the Arguments Editing control panel click the Delimiter button.
14. In the Arguments Editing control panel type $Parameter$.

```
Loop for: Elements
     Start a new line
         Export expression: ElementNumber();[I8]
         Loop for: Element Parameters
            Export expression:;$Parameter$[F10.4]
         End loop
    End line
End loop
```

---

**Tip:** Inserting a dynamic delimiter in the loop over parameters, before the $Parameter$ will insert a delimiter before each new entry in the line, and will not insert a delimiter at the end of the line, where it is not needed.

---

These two nested loops will loop over the mesh elements and for each element export a line containing the element number followed by the number of parameters associated with each element.

| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **Matrix** | **Export matrix:**<br>**End matrix** | ***any valid expression*** |

- This command can not be included in a Line command.
- This command is a block command.

The Matrix command enables you to export an expression for each cell in a grid layer using one command. You can create the same export file using a loop over columns nested within a loop over rows. However, the matrix command is much faster, and it also automatically places a dynamic delimiter between fields.

For instance, the following example will export a matrix for each grid related parameter.

```
Loop for: Block Parameters
    Export matrix: $Parameter$ [F8.2]
End loop
```

| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **If** | **If**<br>**End if** | ***0, 1 or a logical expression*** |

- This command is a block command.
- The script command within the If block will be executed only when the If argument is true.
- True = 1, False = 0, anything different from zero returns True.

The If command enables you to nest script commands within it, that will be executed only if the logical expression holds. For instance, to export the node number and coordinates of boundary nodes only:

```
Redirect output to: $BaseName$
     Loop for: Nodes
          If: NodeOnBoundary()
               Start a new line
                  Export expression: NodeNumber(); [I8]
                  Export expression: PositionX(); [E15.5]
                  Export expression: PositionY() [E15.5]
               End line
          End if
     End loop
End file
```

| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **Comment** | # | ***Any text*** |

The comment command enables you to comment your export template scripts to make them more readable by you. For instance, to clarify the last example you can add comments at important points.

```
Redirect output to: $BaseName$
     Loop for: Nodes
     # Loop to export Boundary Nodes Only!!!
          If: NodeOnBoundary()
          # This if statement is true for boundary nodes!
               Start a new line
                  Export expression: NodeNumber(); [I8]
                  Export expression: PositionX(); [E15.5]
                  Export expression: PositionY() [E15.5]
               End line
          End if
     End loop
End file
```

The comments you insert in a script are not exported.

### To comment your export file and to add literals

To add remarks to your export file use the Export expression command with a literal. For instance, to make your template, write the string “My first export file” at the top of the file:

```
Redirect output to: $BaseName$
    Start a new line
        Export expression: “My first export file”
    End line
End file
```

Using the literal in an expression allows you to specify different delimiters at different places in a line or different delimiters in different lines. For instance, the following example uses two different delimiters:

*In this line the delimiter is a comma.*

*In this line the delimiter is a TAB.*

```
Start a new line
    Export expression: NodeNumber() [I8]
    Export expression: “,”
    Export expression: PositionX() [E15.5]
End line
Start a new line
    Export expression: NodeNumber() [I8]
    Export expression: “^t”
    Export expression: PositionX() [E15.5]
End line
```

The first line block will use the comma character to delimit the NodeNumber from the PositionX. The second will use the TAB character to do the same.

To insert any text (literal) within a file use the Export expression as explained above.

Another example:

```
Start a new line
Export expression: “EL”;
Export expression: ElementNumber(); [I8]
Export expression: “   RG”;
Export expression: Mesh.Domain Number ; [I5]
Export expression: “   3";
Export expression: NthNodeNum(1); [I8]
Export expression: NthNodeNum(2); [I8]
Export expression: NthNodeNum(3) [I8]
End line
```

Using the above example with the comma character as the delimiter will produce a line looking like this:

```
EL,      83,   RG,    2,    3,      44,      55,   11
```

## Dynamic Arguments Interpretation and Resolving

Dynamic arguments are interpreted and resolved during export. Layer parameters and expressions are also evaluated during the export process. Dynamic arguments include general arguments such as the $Basename$ as well as loop iterators such as the $Row$ and $Column$.

### Resolving $Basename$

While exporting a mesh layer using the following template, the `$BaseName$` dynamic argument will be replaced by the file name supplied in the Save As... dialog.

```
Redirect output to: $Basename$.nod
    Node Command
    Node Command
End File
Redirect output to: $Basename$.elm
    Element Command
    Element Command
End File
```

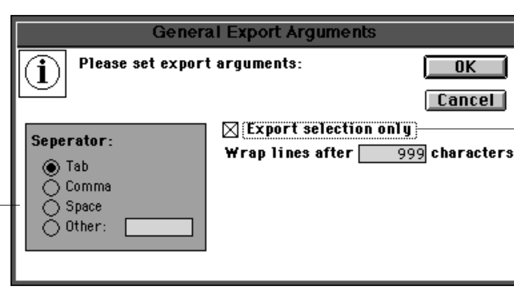
For instance, if you supply the name MESHEXP, the two occurrences of the argument `$Basename$` will be replaced by MESHEXP, to create two files named MESHEXP.nod and MESHEXP.elm.

### Resolving the dynamic delimiter “;”

While exporting a mesh layer using the following template, the dynamic delimiter “;” will be replaced by the delimiter supplied in the Dynamic Export Arguments dialog.

```
Start a new line
    Export expression: NodeNumber(); [I8]
    Export expression: PositionX(); [E15.5]
    Export expression: PositionY() [E15.5]
End line
```

*The delimiter you choose here will replace all occurrences of the Dynamic delimiter “;” you specified in the Export Template.*




*Check to export selection only.*

*Set the line length for all export files.*

## Loop iterators

Each export template script loop command has its loop iterator. To use a loop you do not always have to call its iterator. However, some functions might need the loop iterator to be resolved during export. Such functions might be the grid functions NthColumnPos() and NthRowPos(), and the export template script command Matrix.

### The $Parameter$ loop iterator

When using the $Parameter$ loop iterator within a loop over parameters, it will be resolved for every parameter assigned to the layer yielding the parameter's name.

### The $Element$ loop iterator

When using the $Element$ loop iterator within a loop over elements, it will be resolved for every element resulting in the element's number. This is equivalent to using the element function ElementNumber().

### The $Node$ loop iterator

When using the $Node$ loop iterator within a loop over nodes, it will be resolved for every node resulting in the node's number. This is equivalent to using the node function NodeNumber().

### The $Block$ loop iterator

When using the $Block$ loop iterator within a loop over Blocks, it will be resolved for every block yielding the block's number designated by row,column.

### The $Column$ loop iterator

When using the $Column$ loop iterator within a loop over grid columns, it will be resolved for every column resulting the current column number. If used within the function NthColumnPos($Column$), it results in the current column position.

### The $Row$ loop iterator

When using the $Row$ loop iterator within a loop over grid rows, it will be resolved for every row resulting in the current row number. If used within the function NthRowPos($Row$), it results in the current row position.

## Resolving loops

Loops are resolved during export to their value in the exported layer.

- Loop for nodes will loop for “number of nodes” in the exported mesh layer.
- Loop for elements will loop for “number of elements” in the exported mesh layer.
- Loop for blocks will loop for “number of blocks” in the exported grid layer and will export the blocks’ expression row by column, looping over the columns first.

.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1,1 | 1,2 | 1,3 | . | . | . | . | . | 1,columns |
| 2,1 | 2,2 | . | . | . | . | . | . | . |
| 3,1 | 3,2 | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| rows,1 | . | . | . | . | . | . | . | rows,columns |

- Loop for rows will loop for “number of rows” in the exported grid layer and will export rows’ expressions by row.
- Loop for columns will loop for “number of columns” in the exported grid layer and will export column’ expressions by column.
- Loop for parameters will loop for the number of parameters that are assigned to the elements, nodes or layer, depending on the loop it is nested in.
  For instance, to loop over all element parameters:

```
Loop for: elements
    Start a new line
        Export expression: ElementNumber();[I8]
        Loop for: Element Parameters
            Export expression:;$Parameter$[F10.4]
        End loop
   End line
End loop
```

In the above example, if the elements are assigned three parameters, the `$Parameter$` argument will be evaluated three times in each line within the loop to export the three element related parameters.

# Export Template Examples

## Mesh Template Example

The following listing is the default mesh export template. The output format this template creates is identical to Argus ONE's generic mesh export format explained in chapter 8 "Exporting ans Importing a Mesh."

You can change this template and save it under a different name. If you altered this template and saved the project, but need to recover it, just open a new project window and save the default template from that project. You can now load it back into the other project.

*Export general mesh information.*

*Export node related information.*

*Export element related information.*

```
Redirect output to: $BaseName$
  Start a new line
    Export expression: NumElements(); [I8]
    Export expression: NumNodes(); [I8]
    Export expression: NumElemParameters(); [I8]
    Export expression: NumNodeParameters() [I8]
  End line
  Loop for: Nodes
    Start a new line
      Export expression: "N";
      Export expression: NodeNumber(); [I8]
      Export expression: PositionX(); [F8.2]
      Export expression: PositionY() [F8.2]
      Loop for: Node Parameters
        Export expression: ;$Parameter$ [F8.2]
      End loop
    End line
  End loop
  Loop for: Elements
    Start a new line
      Export expression: "E";
      Export expression: ElementNumber(); [I8]
      Export expression: NthNodeNum(1); [I8]
      Export expression: NthNodeNum(2); [I8]
      Export expression: NthNodeNum(3) [I8]
      Loop for: Element Parameters
        Export expression: ;$Parameter$ [F8.2]
      End loop
    End line
  End loop
End file
```

## Grid Template Example

The following listing is the default grid export template. The output format this template creates is identical to Argus ONE's generic grid export format explained in chapter 10 "Exporting a Grid."

You can change this template and save it under a different name. If you altered this template and saved the project, but need to recover it, just open a new project window and save the default template from that project. You can now load it back into the other project.

*Export general grid information.*

*Export columns and rows coordinates.*

*Export block activity matrix.*

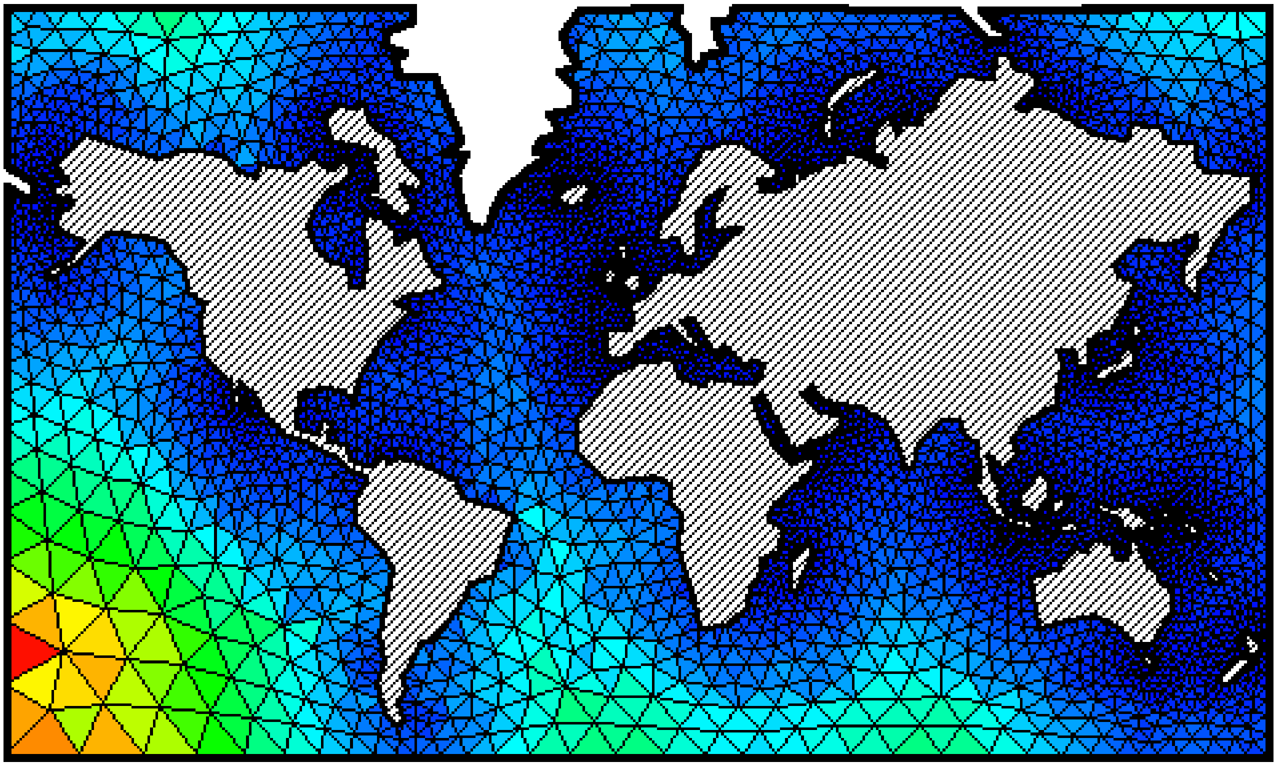
*Export block and grid related parameters.*

```
Redirect output to: $BaseName$
  Start a new line
    Export expression: NumRows(); [I8]
    Export expression: NumColumns(); [I8]
    Export expression: NumBlockParameters()+1 [I8]
  End line
  Loop for: Rows
    Start a new line
      Export expression: NthRowPos($Row$) [F8.2]
    End line
  End loop
  Loop for: Columns
    Start a new line
      Export expression: NthColumnPos($Column$) [F8.2]
    End line
  End loop
  Export matrix: BlockIsActive() [I1]
  Loop for: Block Parameters
    Export matrix: $Parameter$ [F8.2]
  End loop
End file
```

# Script Commands Reference

| Command Syntax | Command Arguments | Command Name |
|---|---|---|
| `Redirect output to:`<br>`End File` | Explicit: *<FileName>*<br>Dynamic: *$BaseName$* | File |
| `Start a new line`<br>`End line` | | Line |
| `Export Expression:` | *any valid expression* | Expression |
| `Loop for:`<br>`End loop` | *Mesh Layers*<br>*Parameters - Layer*<br>*Nodes*<br>*Node Parameters*<br>*Elements*<br>*Element Parameters*<br>*Objects - Nested within loop for Elements*<br><br>*Grid Layer*<br>*Parameters - Layer*<br>*Blocks*<br>*Block Parameters*<br>*Rows*<br>*Columns*<br>*Objects - Nested within loop for Blocks*<br><br>*Variables* | Loop |
| `Script Comment` | *# any text* | Comment |
| `Delimiter` | Explicit: *"any character"*<br>Dynamic: *;* | Delimiter |
| `If:`<br>`End if` | Explicit: *0* for False<br>anything else True<br>Dynamic: *any valid expression* | If |
| `Export matrix` | *any valid expression* | Matrix |

**For additional script commands see Supplement for version 3 and 4.**



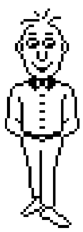
# Finite Element Meshes

# Meshing a Domain

# Overview

The two Argus ONE mesh modules support two types of elements; the Triangular element and Qquadrilateral element. Except for their different shapes,, the two types are handled by Argus ONE in an identical manner. Thus, although all the examples and illustrations are of triangular elements, the methods described are identical for Quadrilateral element meshes.

The mesh you create has a tremendous impact on the solution of your numerical model, both with respect to it's accuracy and to the computing resources it demands. To get the best possible solution while maintaining computing needs to the minimum required, one usually tries to comply with the following three discretization rules:



*The three meshing rules.*

- Keep the number of nodes to the minimum to save computing resources.
- Have sufficient number of nodes at zones where high gradients of the function solved are encountered, or where the domain boundaries are needed to be described in detail.
- Create a mesh free of acute and obtuse elements.

As you might have read before, the only thing you have to do to mesh a domain, is to define it by creating a domain outline contour, and then, click the Magic Wand in it. However, to create a mesh that complies with your numerical and physical needs as described above, you must tell Argus ONE of these needs. You do so by defining the domain, setting different mesh densities to different zones in it, and finally letting Argus ONE meshing module mesh it.

The three main stages in meshing a domain are:

1. Defining the domain by creating a domain outline contour.
2. Setting mesh densities.
3. Clicking the Magic Wand in the domain.

You define the domain in a domain outline type layer by creating a outline contour. Creating contours is explained in detail in chapter 2 "Working with Information Layers."

In this chapter you will learn the techniques that will enable you to create a mesh that will answer your numerical needs. This chapter explains the effects of the mesh density layer and of the shape of the domain outline contour on the mesh.

Argus ONE's automatic mesh generation (AMG) engine performs the meshing in two major steps: creating the elements and smoothing them.

# Mesh Related Layers

To allow Argus ONE to automatically mesh a domain, a mesh layer must be associated with two other layers: a domain outline layer and a density layer.

The domain outline contours in a domain layer tell Argus ONE's AMG what is the domain it needs to mesh when you click the magic wand. As you will find out soon, it also contains additional information such as the default mesh density you assign to domain outline contours in the density parameter, etc.

The density layer allows you to specify different element sizes in different areas of the domain, overriding the default density defined by the domain outline contour.

## Assigning the Mesh Layer Domain and Density Layers

When you create a new Argus ONE project, it is created with one mesh layer, a domain layer named Domain Outline, and an information layer called Density. This predefined mesh layer is automatically associated with the domain and density layers. You can change the domain and density layers a mesh layer is associated with, at any time.

As you might have read before, any information type layer or a domain type layer may serve as the density layer you assign to a mesh layer. However, only a domain type layer can serve as a mesh layer's domain layer.

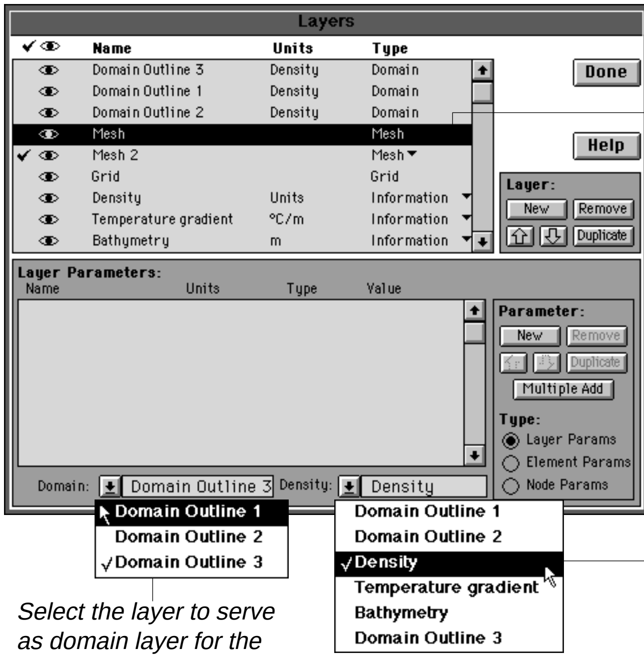
You can replace both the domain and density layers associated with a mesh layer at any time. To create different meshes in the same mesh layer, you can replace the domain and density layers it is associated with and remesh. You can also create different mesh, domain and density layers sets for the different realization you need to create meshes for.

### To set the domain and density layers associated with a mesh layer

You assign a mesh layer its domain and density layers in the layers dialog.

1. From the layers popup menu in the information ruler, choose Layers...
2. In the layers list, select the line of the mesh layer its domain and density layers you want to change.
   When you select a mesh layer, two popup menus appear at the bottom of the dialog box.
3. From the left one, named Domain, select the domain layer. Only domain layers are listed in this popup menu.
4. From the other, named Density, select the layer to serve as density layer. Listed in this popup are only information type and domain type layers.

*Set the domain and density layers for the selected mesh layer.*




*Select the mesh layer its domain and density layer you want to set.*

*Select the layer to serve as density layer for the selected mesh layer.*

*Select the layer to serve as domain layer for the selected mesh layer.*

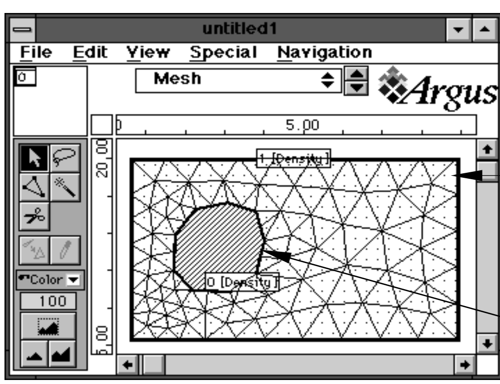
# Using Contour Objects in Domain Layers

As in information type layers, you can use close, open and point contours in a domain type layer. All contour objects can be assigned mesh density values.

## The Domain Outline Contour

A close contour in a domain type layer is referred to as a domain outline contour. It tells Argus ONE meshing module what is the domain to be meshed and the default mesh density. A domain outline contour can contain other close contours defining islands-like internal boundaries.

*The area within islands is not meshed.*




*A simple domain outline contour assigned default mesh density of 1.0.*

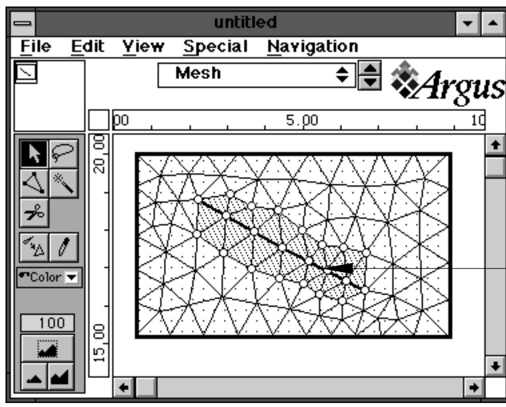
*An island.*

## The Open Contour

An open contour is used to mark internal boundaries such as rivers and faults lying along lines. When you enter an open contour within a domain outline contour, it tells Argus ONE to create elements along the open contour such that the elements' sides do not cross the open contour line. This capability allows for the solution to avoid mass transfer through internal line boundaries for instance, and to apply boundary conditions along an open contour.

Assigning an open contour a mesh density value forces the meshing engine to reduce elements' sizes to the required density in the vicinity of the contour.

*A domain outline contour with an open contour.*



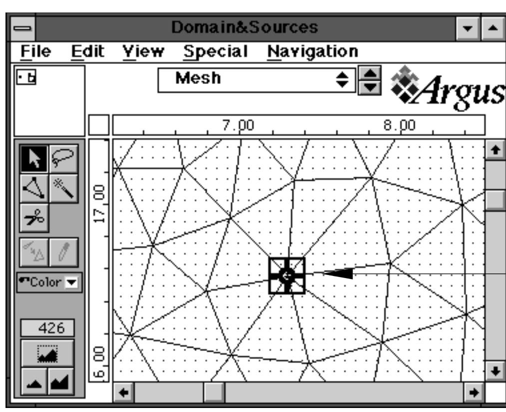

*A simple domain outline contour with an open contour. Elements lie along the open contour.*

## The Point Object (Contour)

A point object is used to mark internal point boundaries representing sources, sinks and point loads such as wells. When you enter a point object within a domain outline contour, it tells Argus ONE meshing module to create a node at the point object's location. This capability allows you to input to your simulator the required boundary conditions at the point object's location.

Assigning a point object contour a mesh density value forces the meshing engine to reduce elements' sizes to the specified density around that point.

*A domain outline contour with a point object.*




*A simple domain outline contour with a point object. A node is created at the point object's location.*

# Mesh Densities

Mesh density is defined in Argus ONE workplace as the length of the side of an element. The larger the side length you set, the larger the resulting elements, and the lower the number of elements.

You define the density by two means: the mesh density you assign to the domain outline contour, referred to as the default density, the density you assign to “islands”, open contours and point objects in the domain layer, and the density layer in which you assign different zones different mesh densities.

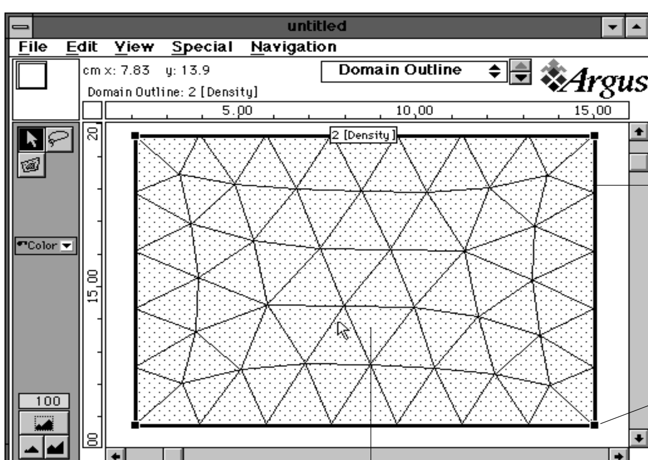
As explained later in this chapter, there are also some other factors influencing the density of the mesh, resulting from numerical considerations.

## Default Mesh Density

At the end of a domain outline contour creation, Argus ONE prompts you for the mesh density of that contour. The number you set tells Argus ONE what is the average size of the elements you would like to have in the domain. You specify the mesh density in the current units as they appear on the rulers. When Argus ONE meshes the domain it will try to produce a mesh such that all of its elements are as close as possible to equilateral triangles, and that their sides are of a size as close as possible to the specified mesh density. It does so by gradually changing the elements’ sizes from the smallest required density up to the largest density specified elsewhere in the domain. Meshing will also create a node at each of the vertices of the domain outline contour.

In the following example a simple four vertices close contour was created in the domain layer and assigned the default density of 2 centimeters.

*Automatically generated mesh using the default density assigned to the domain outline contour.*



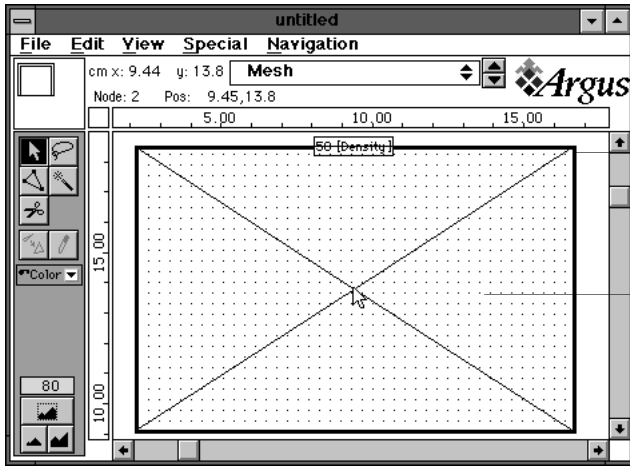

*A simple domain contour assigned a mesh density of 2 cm.*

*A node is forced at domain contours vertices.*

*The resulting mesh is made of 2 cm sided elements in average.*

If the mesh density you specify is much larger than the size of the domain, Argus ONE meshing engine will reduce the mesh density so that it will be able to create a mesh with the minimum possible number of elements, while keeping them as close as possible to equilateral triangles.

*An extreme density requirement automatically handled by Argus ONE.*




*Assign the simple domain in the preceding example a mesh density of 50 cm.*

*The resulting mesh is made of the minimum possible number of equilateral triangles.*

## Re-Meshing an Already Meshed Domain

When you click the Magic Wand in a mesh layer, Argus ONE can test the domain to be meshed against the existing elements in the mesh to check if auto mesh generation is about to create overlapping elements. If it does find such elements it alerts you and allows you to cancel. Testing the domain you clicked the Magic Wand in against all elements is very time consuming. If you are certain that you want to delete all elements in the mesh layer you can select the Delete All button to avoid these tests.

*All node and element manual overridden values are deleted as well.*


- Click the Delete All button to delete all elements in the active mesh layer.
- Click the Delete Contained button to allow Argus ONE to delete overlapping elements. Argus ONE deletes only overlapping elements.
- Click the Cancel button to remain with the current mesh.

## The Effect of the Size of Boundary Segments on Mesh Density

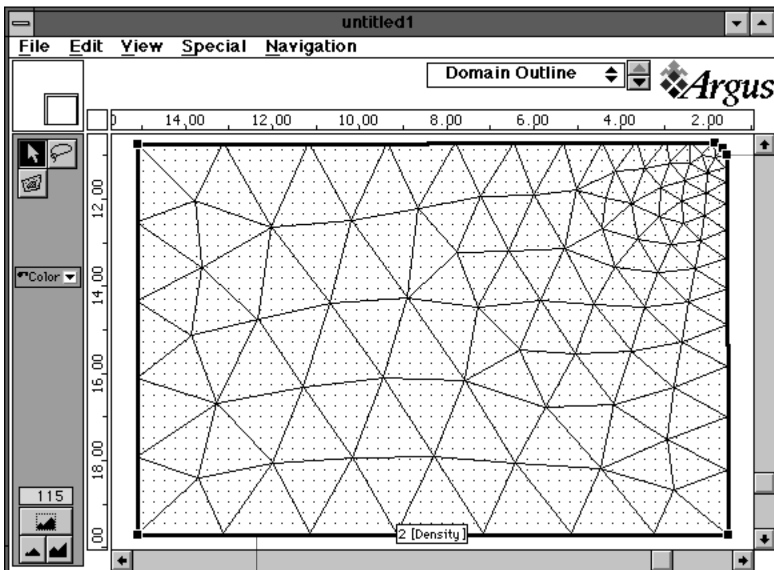
A domain outline contour, including its “islands” and open contours, are defined by their vertices, in which the auto mesh generation is forced to create nodes. In cases where contours’ segments (a segment is the line connecting two vertices) are much smaller than the enforced density, the creation of acute elements could have occurred. To avoid creating acute elements, Argus ONE meshing engine automatically reduces the density in the immediate vicinity of small segments. As you will learn later in this chapter, you can override this effect.

If you have digitized a domain outline contour in great detail, Argus ONE meshing engine uses this extra information to create a mesh that will follow the domain boundaries to their finest details.

To show this effect, create a domain outline contour of six segments, where two of the segments are much smaller than the others and smaller than the default density you set to the domain outline contour.

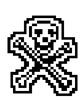
*The effect of domain outline segments smaller than the default density on the resulting mesh.*

*Notice that Argus ONE enlarges the elements gradually to obey the three discretization rules.*




*These two small domain outline segments forced the AMG to reduce the elements to their size.*

*The domain outline's density was set to 2 cm.*

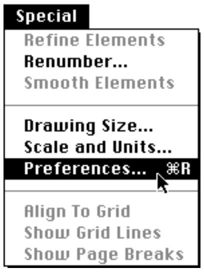


Argus ONE allows you to disable automatic density reduction resulting from small contour segments. However, this is highly unadvisable and Argus does not recommend it, since it might result in high percentage of acute elements. If in certain areas of the domain, the physical phenomena you solve for, is highly anisotropic, you might prefer to have acute elements, this is a case in which you would consider disabling the small segments effect.

## Disabling Small Segments Effect

*Beware: This feature is for the experienced user.*

You enable or disable the small segments effect in the preferences dialog. The preferences dialog is where you set all of Argus ONE preferences.
Argus ONE's preferences are document or window dependent, that is, they are saved per project. When you set the preferences in one project they do not affect other project's preferences.

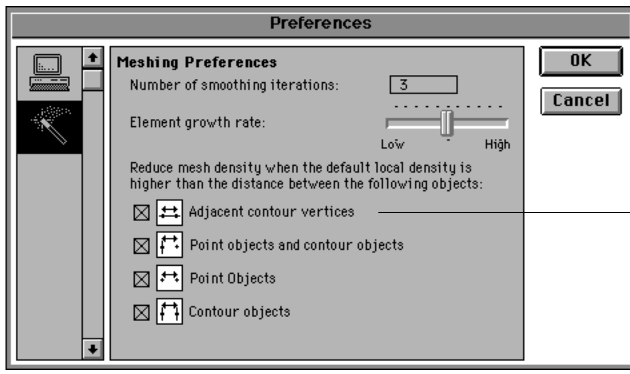


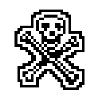

## To open the preferences dialog

- From the Special menu, choose Preferences...
  The preferences dialog opens.

## To open the mesh preferences controls

- Click the Magic Wand icon.
  The mesh preferences controls are presented.



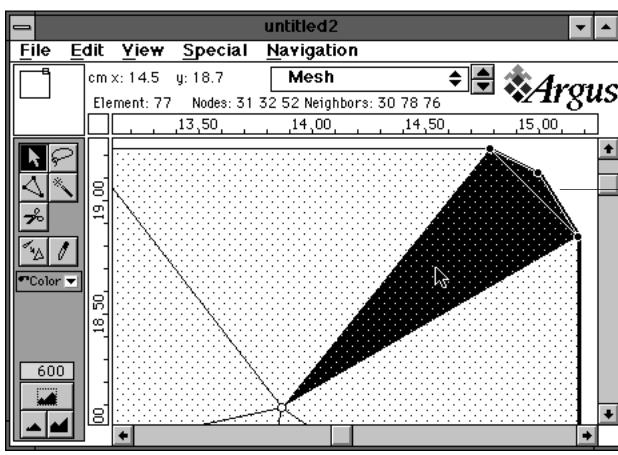
*Click this check box to disable small segments effect.*

## To disable the small segments effect

- Click the first check box.
  To enable it, click it again.

*In certain cases you need to create acute elements. If you do, turn off the small segments effect.*

*Do not forget to enable this option, after you finish creating the acute elements.*




*Meshing the domain in the preceding example when the small segments effect is disabled resulting in these two acute elements.*

## Other Domain Factors Affecting Mesh Density

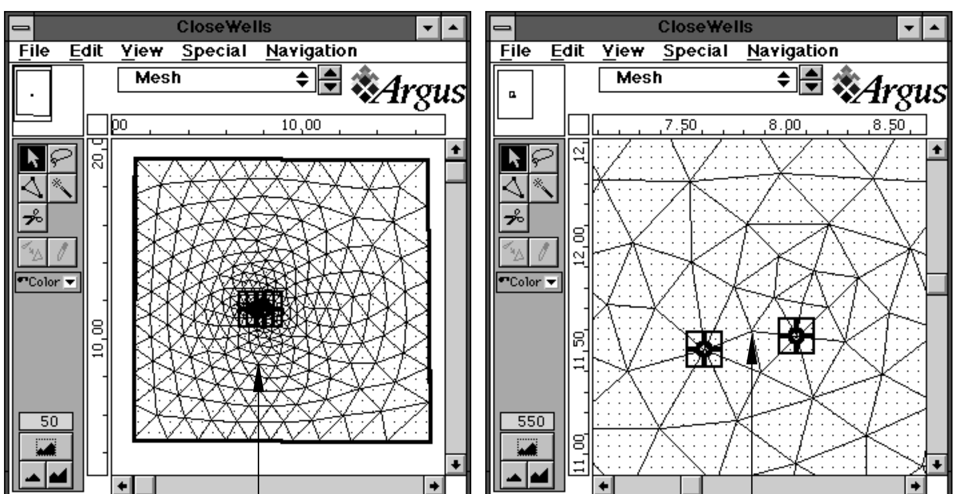
When you numerically solve a problem, you assign nodes lying along boundaries boundary conditions. Boundaries do not always lie on the domain outline. Sources and sinks for instance, can be regarded as internal boundaries, so can internal discontinuities, and internal line boundaries such as rivers, faults, etc. Usually, nodes assigned boundary conditions are not solved for. So, if two or more nodes of an element lie on boundaries, these nodes are not solved for.

For instance, if your domain outline is very concave, and some opposing domain outline segments are closer to each other than the local mesh density, traditional automatic mesh generation algorithms might place two nodes of an element on opposing boundaries, thus disabling this element from participating in the solution. This can also cause a numerically imposed discontinuity in the problem domain.

Argus ONE meshing engine automatically handles all cases that might produce such errors. It does so by automatically reducing the mesh density in the area where domain contours, close contours, open contours and points are closer to each other than the local mesh density.

The following two screen-shots illustrate how Argus ONE handles cases where two domain contours, in this case, two point objects, are closer to each other than the local mesh density.

*The default mesh density assigned to the domain outline contour is 5. To handle the two very close sources/sinks Argus ONE reduced the mesh density near them.*

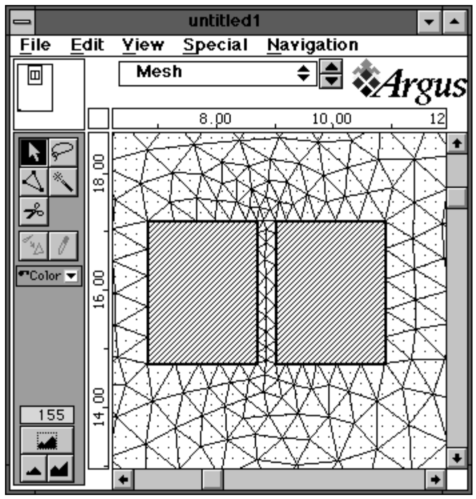



*Argus ONE automatically reduces the mesh density around the two close point objects to...*

*...create at least one node between them.*

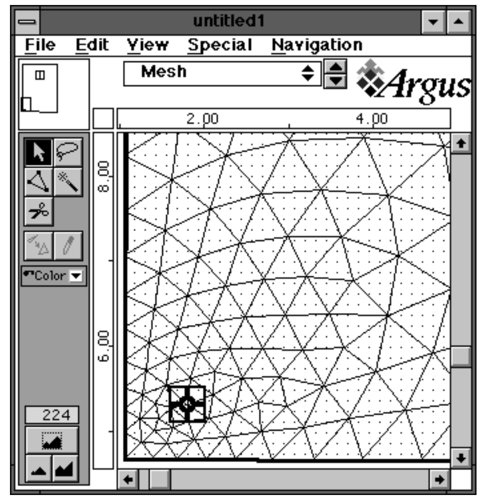
In the previous example and two following ones, the domain outline contour was assigned a default density of 5. As you can see, Argus ONE automatically reduced the mesh density around the problematic areas to avoid having elements with two nodes on different boundaries.

*Small contour to contour distance example.*




*Two close domain islands.*

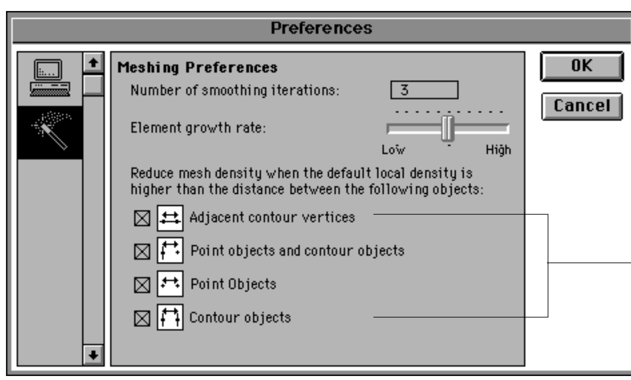
*Small point to contour distance example.*

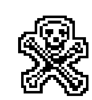



*A point object close to a domain boundary.*

You can disable each of the three effects in the preferences dialog. However, this is highly unadvisable and we do not recommend it.

*There are three distinct cases that are automatically handled by Argus ONE and that you can disable or enable:*
*1. "Point to contour distance."*
*2. "Point to point distance."*
*3. "Contour to contour distance."*

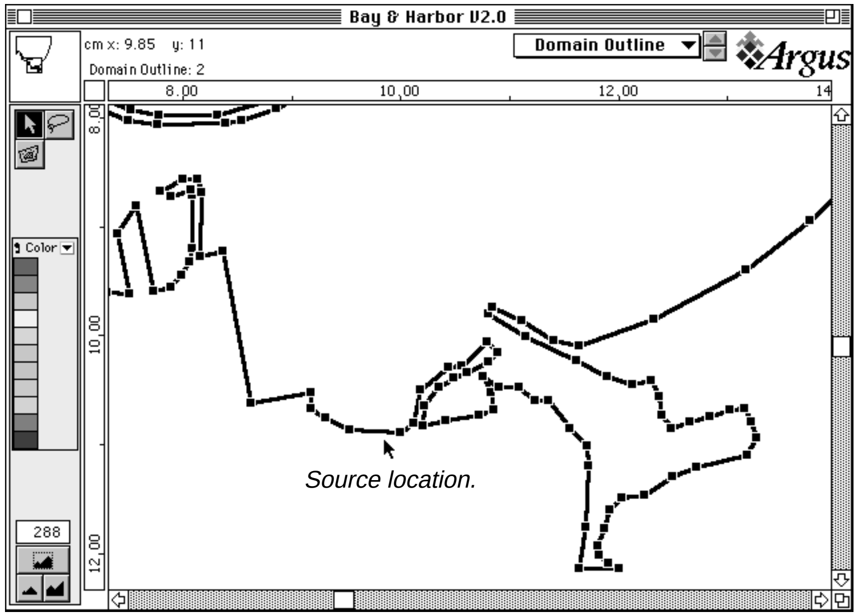


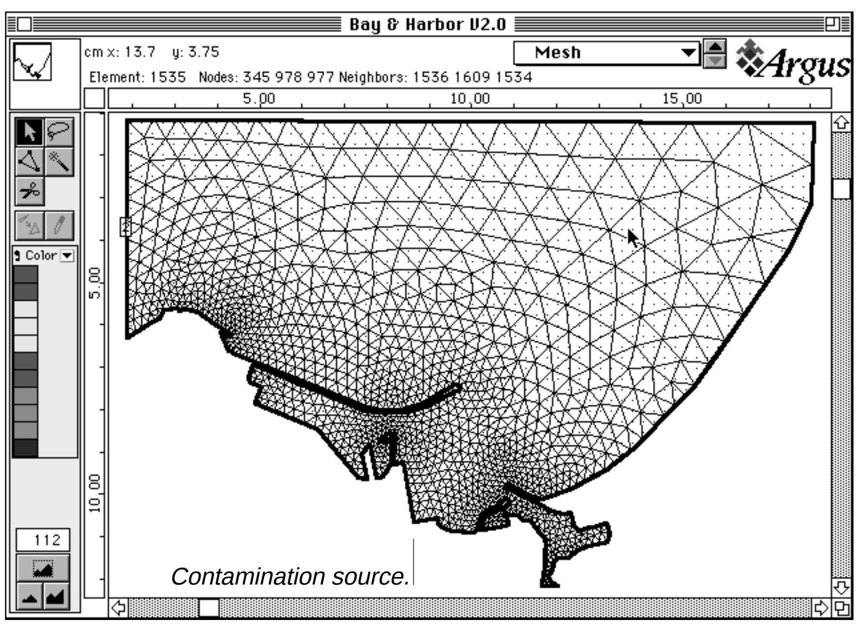
*Click the check box to disable the option.*

Argus ONE's ability to automatically handle domain outlines with complex geometries becomes very important when you try to mesh complex domains. The following snap shot is of a bay and harbor outline, digitized for modeling and predicting the contaminant distribution from a source within the harbor. The domain outline contour was digitized in much detail in the vicinity of the contaminant source.

*A detailed view of a domain outline contour at the vicinity of a contaminant source.*




As a result of Argus ONE's ability to automatically handle complex geometries, meshing was accomplished in a few minutes without the need to manually define mesh densities.



## Setting Other Meshing Preferences

Argus ONE allows you to control two other meshing preferences affecting the resulting mesh:

- The element growth rate.
- The number of smoothing iterations.

## Setting the Element Growth Rate

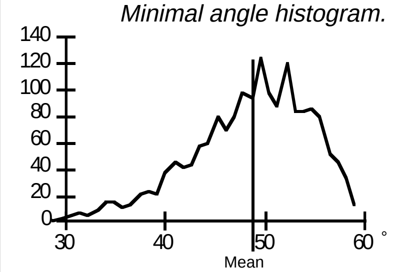
As mentioned before, there are three meshing rules Argus ONE keeps for you while meshing a domain. In a mesh that must have small elements at some areas and can have larger elements at other areas, the first and third rules contradict. The first requires that the mesh will have as little number of nodes as possible, while the third requires that the elements created are as close to equilateral triangles as possible.

To satisfy the first and third meshing rules, Argus ONE allows elements to grow at a certain rate. This rate is the element growth rate that was optimized through tens of thousands of meshing iterations to produce the best results. However, since the element growth rate has a tremendous impact on the number and shape of elements in the mesh, Argus ONE allows you some control over it.
The following tables, charts and screen-shot are of a domain meshed using the default element growth rate.

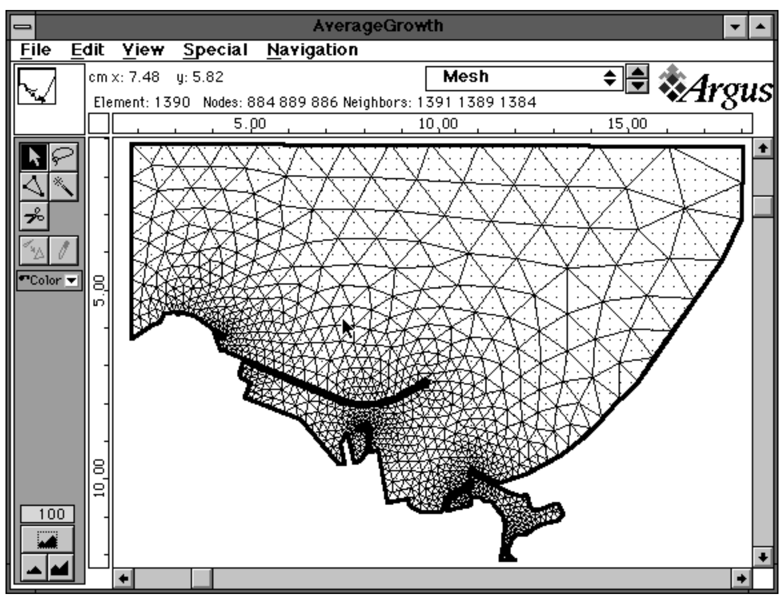
***Minimal angle descriptive statistics.***

| | |
|---|---|
| *Mean* | *48.830198* |
| *Standard Error* | *0.14489005* |
| *Median* | *49.645* |
| *Mode* | *49.01* |
| *Standard Deviation* | *6.0922777* |
| *Variance* | *37.1158476* |
| *Kurtosis* | *0.47032798* |
| *Skewness* | *-0.7506322* |
| *Range* | *36.12* |
| *Minimum* | *23.73* |
| *Maximum* | *59.85* |

*Minimal angle histogram.*




*Domain meshed with element growth rate set to average (default).*




***Number of nodes:1044***
***Number of acute elements (below 22.5°): 0***

### To set element growth rate to maximum

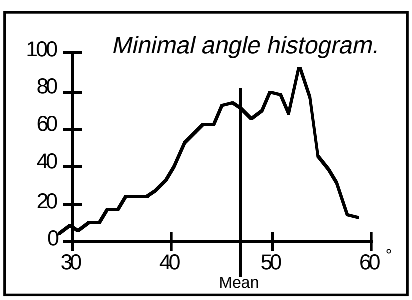
If your major concern is to keep the number of nodes to minimum, and you are prepared to accept some acute elements, you could set the element growth rate to maximum.

- In the preferences dialog, drag the Element Growth Rate dial to the right most.


The previous example re-meshed with element growth rate set to maximum yields the following mesh.

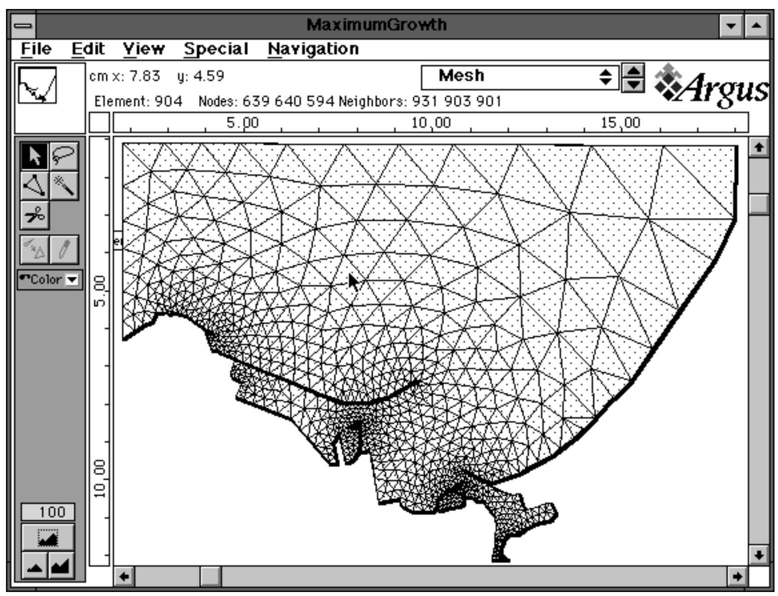
***Minimal angle descriptive statistics.***

| | |
|---|---|
| *Mean* | *46.9642363* |
| *Standard Error* | *0.18170233* |
| *Median* | *47.825* |
| *Mode* | *47.71* |
| *Standard Deviation* | *6.7694789* |
| *Variance* | *45.8258446* |
| *Kurtosis* | *0.07393574* |
| *Skewness* | *-0.6480442* |
| *Range* | *35.45* |
| *Minimum* | *24.04* |
| *Maximum* | *59.49* |




*Minimal angle histogram.*

*Domain meshed with element growth rate set to maximum.*




***Number of nodes:845***
***Number of acute elements (below 22.5°): 0***

The resulting mesh has only three quarters of the number of elements and no acute elements. However, as you can see from the descriptive statistics, the average element minimal angle is 2 degrees smaller. Also from the histogram one can deduce that more elements are further away from equilateral triangles.

### To set element growth rate to minimum

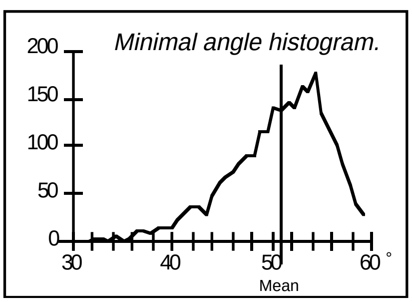
If your major concern is elements shape and you are prepared to accept higher number of nodes, you can set the element growth to minimum.

- In the preferences dialog drag, the Element Growth Rate dial to the left most.


The previous example re-meshed with element growth rate set to minimum yields the following mesh.

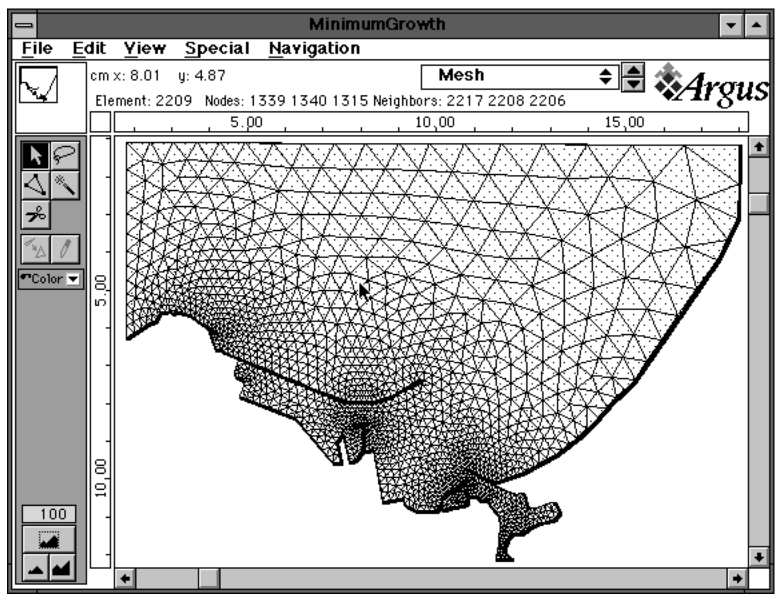
***Minimal angle descriptive statistics.***

| | |
|---|---|
| *Mean* | *50.8060756* |
| *Standard Error* | *0.1029522* |
| *Median* | *51.68* |
| *Mode* | *52.98* |
| *Standard Deviation* | *5.29478675* |
| *Variance* | *28.0347667* |
| *Kurtosis* | *1.69441378* |
| *Skewness* | *-1.0680023* |
| *Range* | *35.13* |
| *Minimum* | *24.73* |
| *Maximum* | *59.86* |



*Minimal angle histogram.*

*Domain meshed with element growth rate set to minimum.*




***Number of nodes:1507***
***Number of acute elements (below 22.5°): 0***

The resulting mesh has fifty percent more elements than the mesh created using the default element growth rate and no acute elements. However, as you can see from the descriptive statistics, the average element minimal angle is 2 degrees closer to 60 degrees. Also from the histogram one can deduce that more elements are closer to equilateral triangles.

## Setting the Number of Smoothing Iterations

As part of the automatic mesh generation, Argus ONE performs a number of smoothing iterations to smooth the mesh. The smoothing algorithm moves every node to the center of gravity of the polygon made of the elements connected to the node.

The default number of iterations is set to three, which was thoroughly tested to produce the best meshes at the minimum computing resources.

You can set the number of smoothing iterations performed as part of the AMG to any number higher than 2.

### To set the number of smoothing iterations

1. From the Special menu, open the preferences dialog.
2. In the Number of smoothing iterations text edit box enter the required number. and click the OK button to close the dialog box.


# Creating Mesh Density Contours

When meshing a domain, you usually want to have more elements at some parts of the domain and less elements at others. You may want to have a fine mesh at areas were gradients are expected to be higher. These areas may be along boundaries or at any other part of the domain. If, for example, you have a well pumping at the middle of the domain, or a source of contamination at the middle of a lake, you want to have a fine mesh around these points.

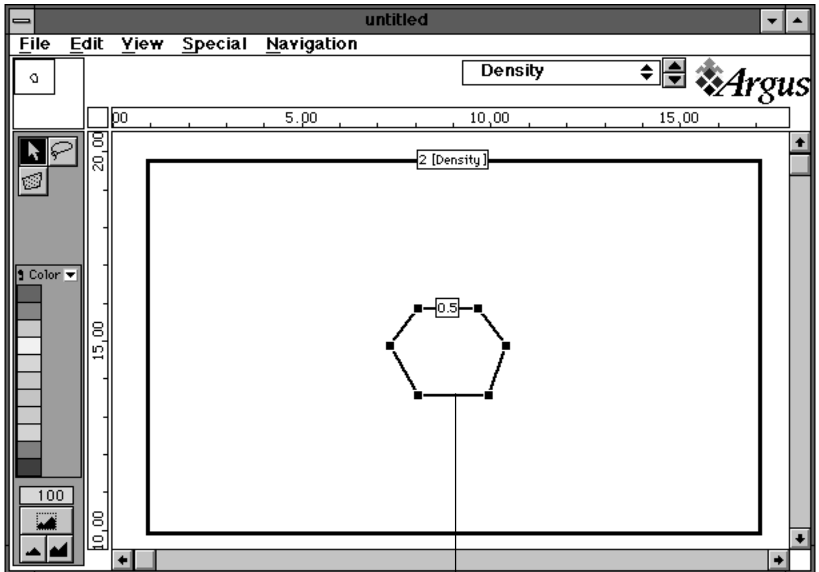
You define different mesh densities at different areas by using mesh density contours in the mesh density layer. An information type layer assigned as the mesh density layer is where you define the required mesh densities.

## To Define Mesh Density Contours

After you have created a domain outline contour and assigned it the default mesh density,

- Change the active layer to the layer associated as a mesh density layer.
- Using the close contour tool, create a contour around the area its density you want to change.
- When you finish creating the contour, Argus ONE asks you for the required density. Enter it in the text edit box. For example, if you are working in units of cm and want to have a 0.5 cm elements, type 0.5.
- Click the OK button.

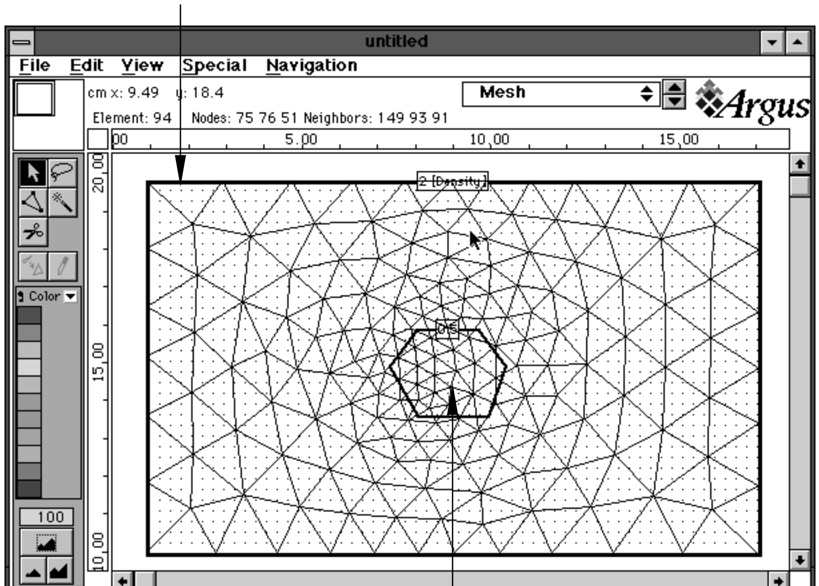
*A simple domain outline contour assigned default mesh density of 2 cm.*




*A mesh density contour telling Argus ONE that elements within this contour should have sides of 0.5 cm.*

- To mesh the domain, move to the mesh layer, and click the Magic Wand anywhere within the domain outline contour.

*Argus ONE gradually increases the element size from the smallest specified to that specified for the domain outline contour (2 cm in this example.)*

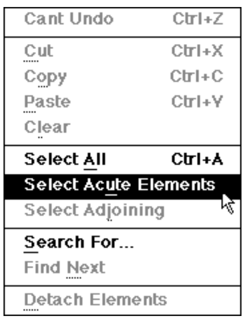



*Elements within the mesh density contour of 0.5 cm are forced to that size.*

## To Check your Mesh for Acute Elements

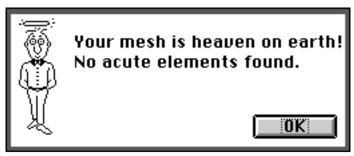
Argus ONE enables you to create very complex meshes that are free of acute elements. It also furnishes you with the tools to test the mesh validity. One of the most common criteria for testing the mesh validity is the percent of acute elements it contains. Argus ONE's default criteria for an acute element, is that one of the element's angles is less than 22.5°. You can change this criteria in the preferences dialog. To read more about this criteria, refer to the chapter “The Mathematics Beneath Argus ONE” and to chapter 7 “Editing and Refining a Mesh.”

After you have created a mesh:




From the Edit menu, choose Select Acute Elements.

If the mesh is free of acute elements, Argus ONE puts up the following message:

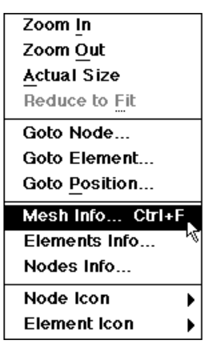


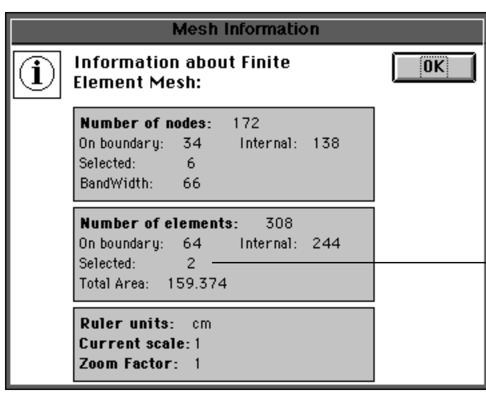

If Argus ONE locates any acute elements, it selects them.

### To find out the total number of acute elements

- From the Information menu, choose Mesh Info...

Argus ONE presents you with the following dialog:




*This field will contain the number of acute elements found, if you bring up this dialog immediately after performing the Select Acute Elements command.*

## Setting Different Mesh Densities in Different Domain Areas

You can give Argus ONE a complex description of different areas where you want to have different mesh densities.

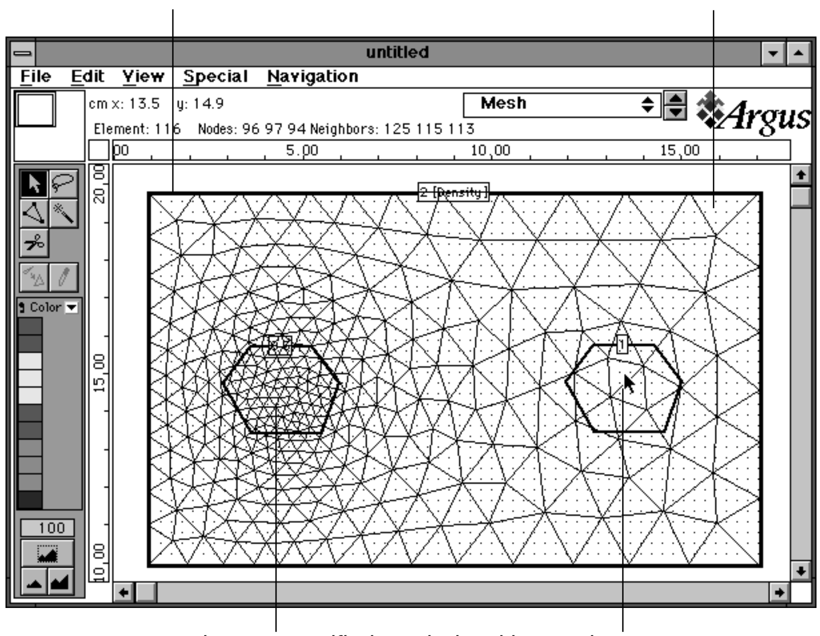
When Argus ONE interprets your mesh density requirements it follows the three following rules:

1. The smallest required mesh density in an area of the mesh, is mandatory.
2. The mesh should not contain acute elements.
3. The number of nodes in the mesh should be kept minimal.

*The concept is that the smallest mesh densities you specify are mandatory, while higher ones are upper limits for elements' growth.*

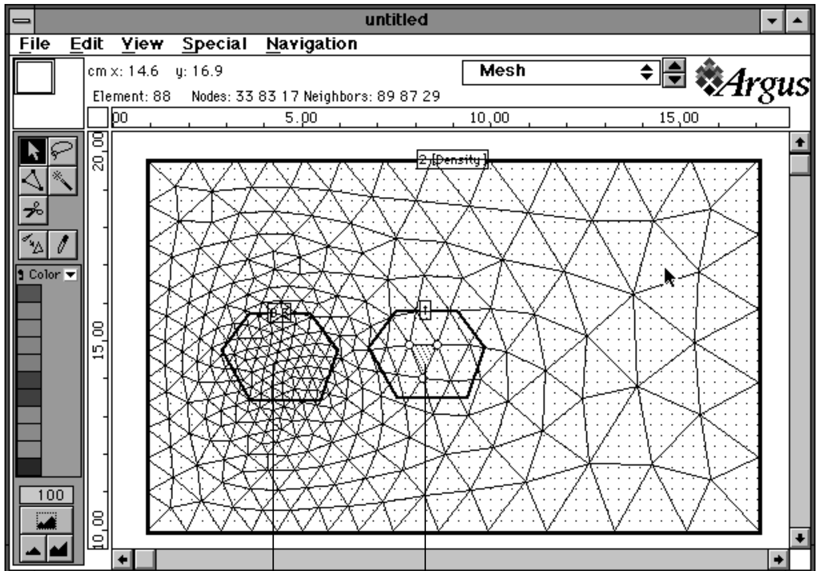
Argus ONE fills areas designated by mesh density contours with the lowest density, with elements of the size you've specified. From areas designated with the smallest mesh density, Argus ONE will gradually increase the elements' size up to the upper limit you've specified. It might however not achieve the upper limit in order to maintain the element's shape and not create acute or obtuse elements.

*Argus ONE gradually increases the element's size from the values specified by the two mesh density contours. The elements are expanded to a size as close as possible to the mesh density of the domain outline contour (in this example, 2 cm.)*




*The two specified mesh densities are kept.*

In the following examples the two mesh density contours were created close to each other.

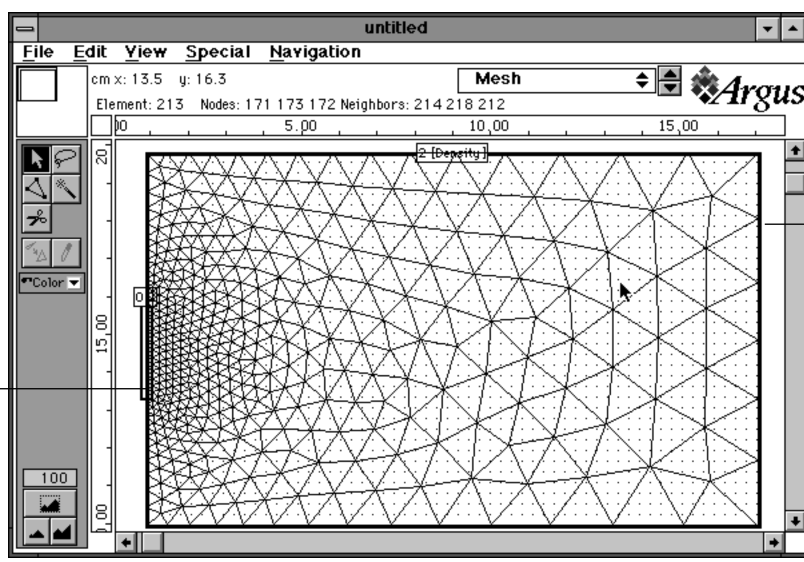



*Only the smallest (0.3 cm) density requirement is fulfilled, the elements within the 1.0 cm specified density contour are smaller than specified to allow for a smooth growth from 0.3 cm elements.*

You can create mesh density contours anywhere around a domain layer. For example, to specify the mesh density around parts of the domain boundary just create there a density contour and assign it the required element size.

*The mesh gradually expands from the lower density to the default density assigned to the domain outline contour.*

*area assigned 0.2 cm mesh density.*




*Default mesh density set to 2.0 cm.*

## Assigning Domain Contour Objects Mesh Densities

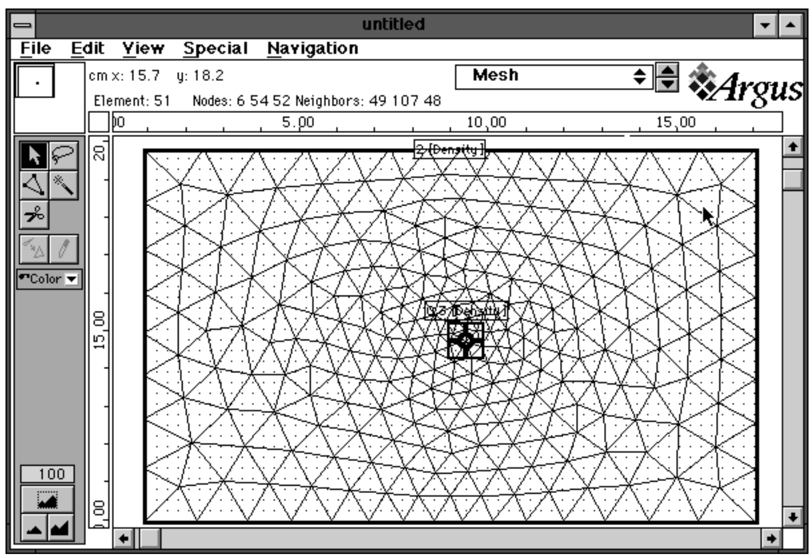
As you have read in the beginning of this chapter, domain outline contours are where you assign the mesh default density. You can also assign other domain contours mesh densities. For example, if you expect to have high gradients around a source/sink described by a point object, you can either create a mesh density contour in the mesh density layer around its location, or just assign the point object describing it a mesh density.

### To assign mesh density to other domain contour objects

At the end of a contour object creation in a domain layer, you are requested to assign it a mesh density.

- Enter the mesh density.

The following screen-shot is of a point object assigned mesh density.




Open and close contours (islands) can be also assigned mesh densities.

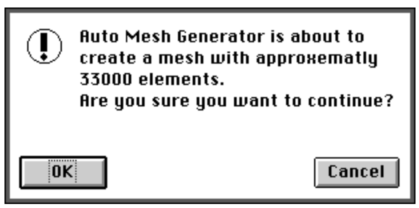
## Controlling Automatic Mesh Generation

You can terminate automatic mesh generation during meshing. However, when the message on the progress bar changes from Meshing in progress... to Smoothing mesh... you can no more stop it.

*Progress Dialog box*


*Press the keyboard sequence to halt meshing. The keyboard combination on Windows and Unix platforms is CTRL+C.*

When you click the magic wand to start meshing, Argus ONE estimates the total number of elements you are about to create. If you this estimate is over 3000 elements, Argus ONE warns you by opening a dialog informing you of the estimated number of elements it is about to create.

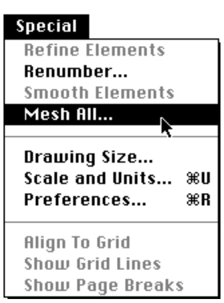


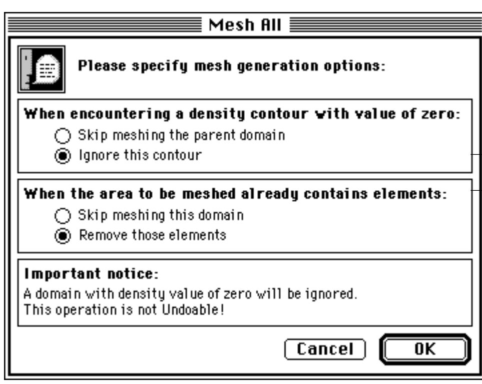

Click the OK button to continue, or the Cancel button to end meshing.

## Meshing Multiple Domains

If you need to mesh a number of domains in one mesh layer, you do not need to mesh them sequentially by clicking the magic wand in each of them. You can use the Mesh All... command under the Special menu.

When you select this command Argus ONE opens the Mesh All dialog asking you how to treat special cases while it is meshing multiple domains.




*Instruct Argus ONE what to do when one of the domain outlines to be meshed lies above a density contour assigned a zero value.*

*Instruct Argus ONE what to do when one of the domain outlines to be meshed will create elements, overlapping existing elements.*

While meshing the domains Argus ONE informs you of the domain it is currently meshing by posting its name (or number) in the progress dialog.

# Editing and Refining a Mesh

# Overview

As you have seen in the previous chapters, Argus ONE gives you a high degree of control of the mesh shape using the domain layer and the mesh density layer techniques.

Argus ONE's mesh editing capabilities enable you to control the mesh with regard to each element and node in it, thus transforming meshing into an art. In this chapter you will learn about mesh editing and refining, and about manual element creating techniques.

# Setting Your Views

When you edit your mesh, you can take advantage of Argus ONE's information hiding capabilities to concentrate on the objects you need. In the following paragraphs we will outline some of these capabilities. For a detailed explanation of showing and hiding layers refer to part 1 chapter 1 in this manual.

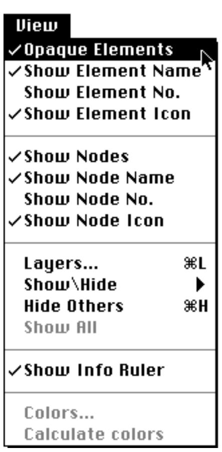
All the commands you will use to set your views are presented in the View menu.

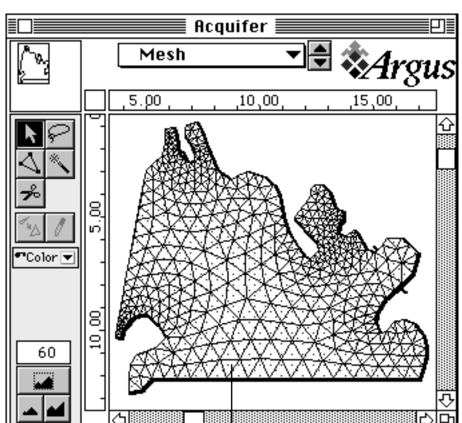
### Seeing through the mesh

To view the mesh and the other layers you have defined at the same time, you can set the mesh transparent. If you need to hide all other information you can set the mesh to opaque. Argus ONE's default is set to Opaque Elements.

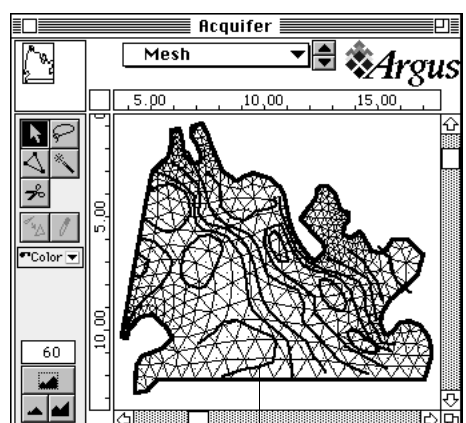
### To toggle between opaque and transparent elements

- From the View menu, choose Opaque Elements.




Opaque elements, hiding other layers.



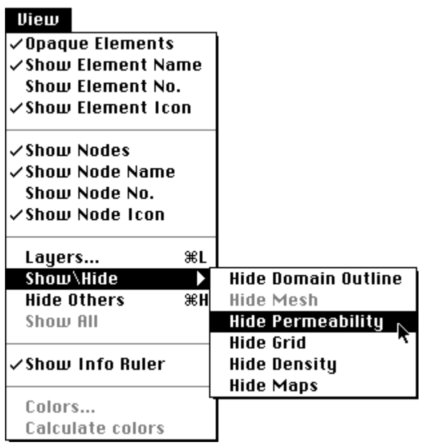

Transparent elements, all layers are visible.

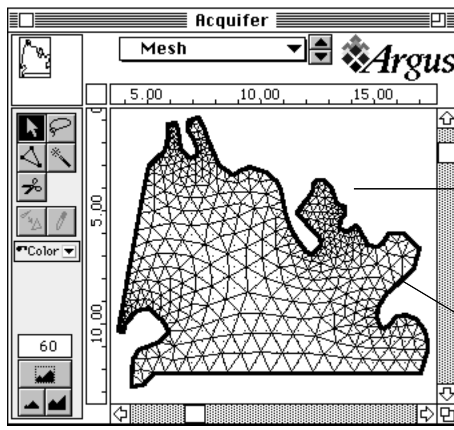
## Showing and Hiding Layers

Showing and hiding layers enables you to view the mesh with respect to only some of the layers. Hiding and showing of layers and moving between layers are explained in detail in the chapter “The Argus ONE Workplace.”

### To hide or show a layer

- From the View menu, choose Hide or Show “Layer name.”




*The contours visible on the previous snapshot are hidden.*

*The domain outline contour is visible since the domain layer is visible.*

## Zooming

*Argus ONE's default zoom level is set to 100 percent.*

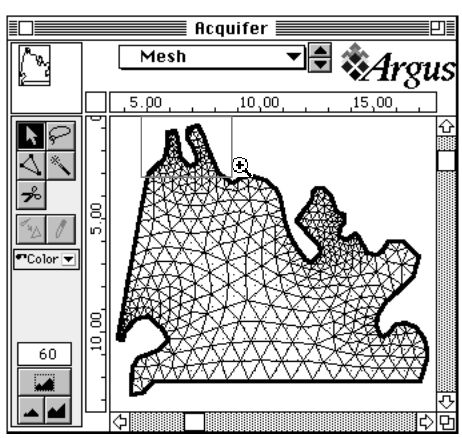
When you want to edit a part of the mesh that is too small to edit in the current zoom level, Zoom In, or, to select large areas of the mesh, Zoom Out to refrain from tedious autoscroll. A detailed discussion about zooming is presented in the chapter “The Argus ONE Workplace.”

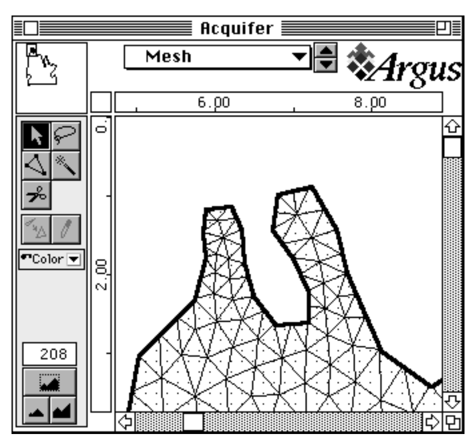
### To zoom in

1. From the tool palette, choose the Magnifying Glass.
2. Stretch the tool around the area of interest.

*When zooming using the zooming tools, Argus ONE centers the zoomed view.*

*Saving a project, saves the zoom level too.*



## Showing and Hiding Node and Element Information

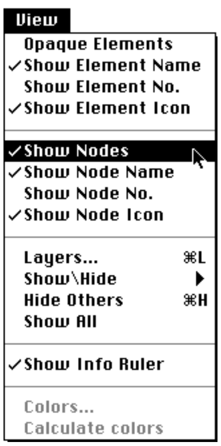
Some of the information you can assign to nodes and elements as well as information automatically assigned to them by Argus ONE, can be graphically presented on the screen. You can show and hide this information so that your workplace does not become cluttered.

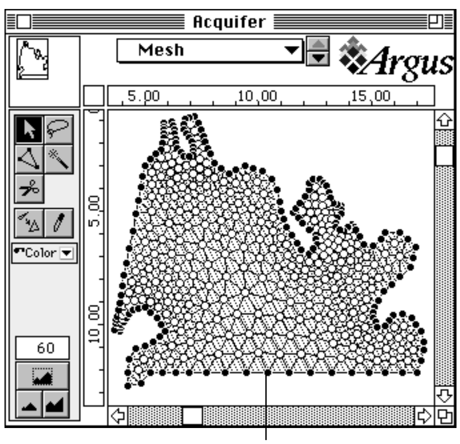
### To show and hide the nodes' highlight

When you select a node it is surrounded by a small circle (node's highlight). When you zoom out, or the mesh is very dense, and you select elements and nodes, the mesh may become cluttered with the nodes' selection. You can instruct Argus ONE to remove nodes' highlighting. However, when you do so you lose the feedback effect when selecting nodes.

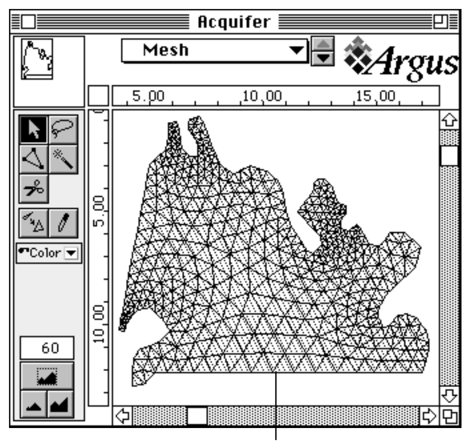
- To disable nodes' highlighting, from the View menu, choose Show Nodes.

The check mark to the left of the menu item is removed.




*All elements are selected, Show Nodes is enabled.*




*All elements are selected, Show Nodes is disabled.*

- To enable nodes' highlighting, from the View menu, choose Show Nodes.

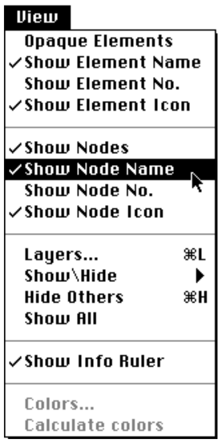
The check mark to the left of the menu item reappears.

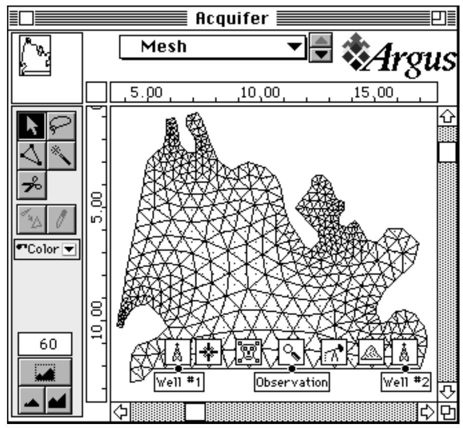
### To show and hide nodes' names

If you assign a name to a node, Argus ONE enables you to decide whether it shows on the screen or not. This is useful when the nodes' names overlap or hide other nodes. When creating a new project, Argus ONE's default is set to Show Node Name.

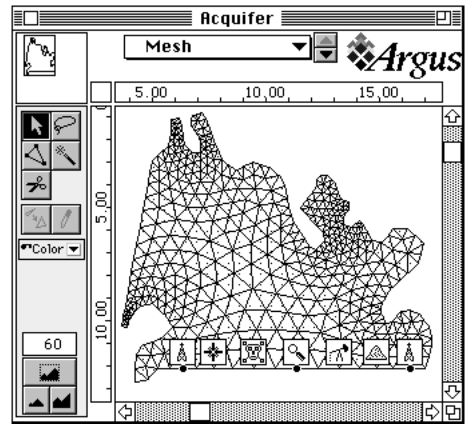
- To hide nodes' names, from the View menu, choose Show Node Name.

The check mark to the left of the menu item is removed.




*Node names and icons are shown.*




*Node names are hidden.*

- To show nodes' names, from the View menu, choose Show Node Name.

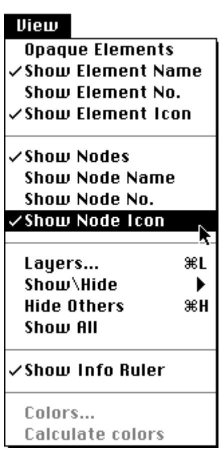
The menu item is marked with a check mark to its left.

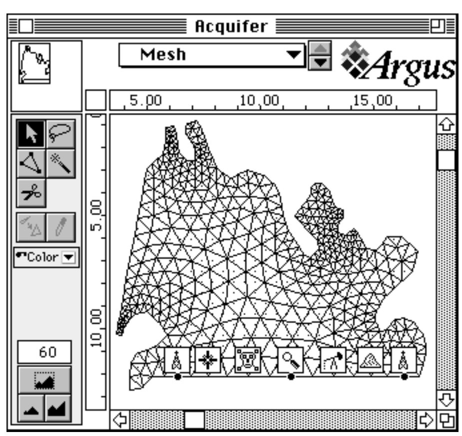
## To show and hide the node icon

If you assign an icon to a node, Argus ONE enables you to decide whether it shows on the screen. This is useful when the nodes icons overlap or hide other nodes. When creating a new project, Argus ONE's default is set to visible node icon.

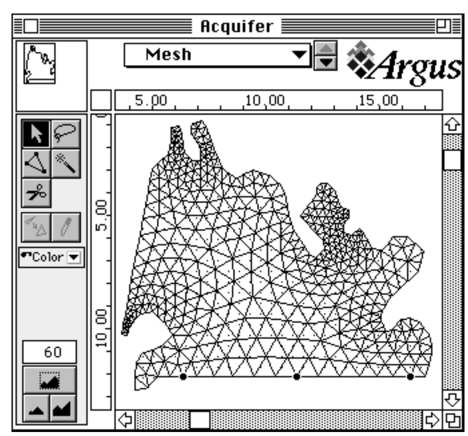
- To hide node icons, from the View menu, choose Show Node Icon.

The check mark to the left of the menu item is removed.




*Node icons are shown.*




*Node icons are hidden.*

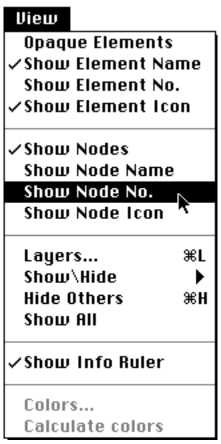
- To show node icons, from the View menu, choose Show Node Icon.

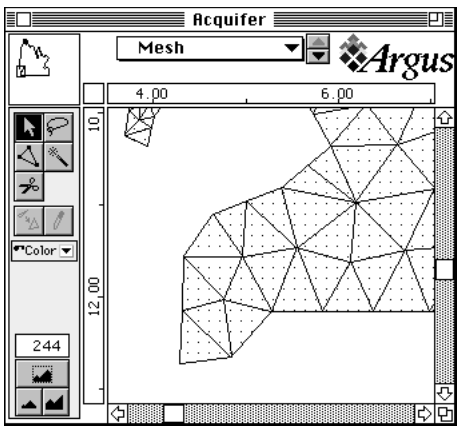
The menu item is marked with a check mark to its left.

### To show and hide nodes numbers

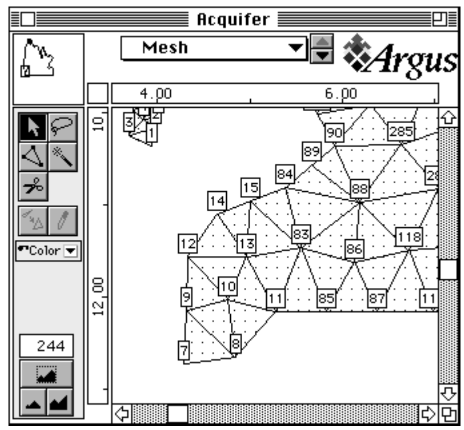
Argus ONE automatically assigns each node in a mesh layer a unique node number and enables you to decide whether node numbers are shown or hidden. This is useful when the node numbers overlap or hide other nodes. When creating a new project, Argus ONE's default is set to hide node numbers.

- To show nodes numbers, from the View menu, choose Show Node No. A check mark is added to the left of the menu.




*Node numbers are hidden.*




*Node numbers are shown.*

- To hide nodes numbers, from the View menu, choose Show Node No. The check mark to the left of the menu item is removed.

### To show and hide elements numbers

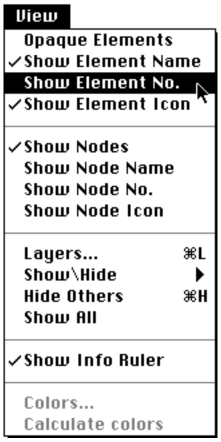
*Note: Elements numbers are dynamic. Do not count on them, unless you are through with deleting elements or renumbering the mesh.*

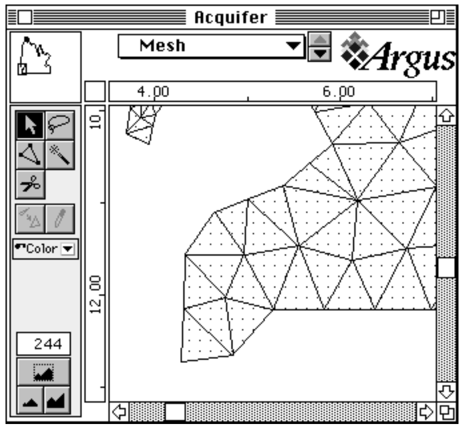
Argus ONE can mark each element with its number. Elements are numbered by Argus ONE automatically as it creates them. When you delete elements, or renumber (optimize the BandWidth) the mesh, Argus ONE changes the numbering of the elements. Although the elements' numbers are dynamic, you may still want to relate to them at times.

Argus ONE presents the number at the center of the element. If the element is too small to contain the number, Argus ONE hides the number. When you zoom in, Argus ONE tests each element again to decide whether the number can be contained within the element and if possible, presents it.

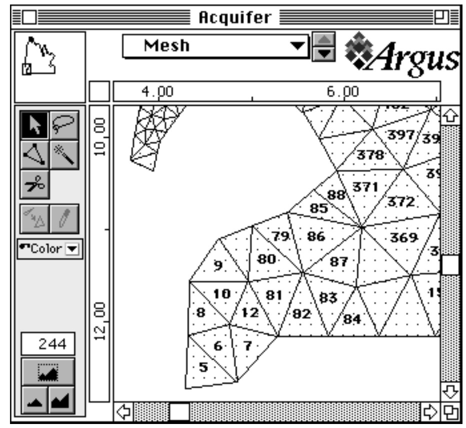
- To show elements numbers, from the View menu, choose Show Element No.

The menu item is marked with a check mark to its left.




Element numbers are hidden.




Element numbers are shown in elements large enough to contain their number. To see the other numbers, zoom in.

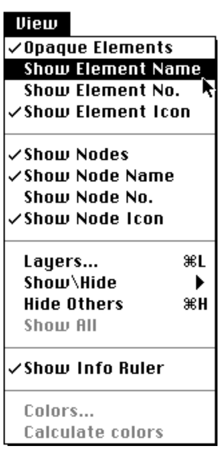
- To hide elements numbers, from the View menu, choose Show Element No. The check mark to the left of the menu item is removed.

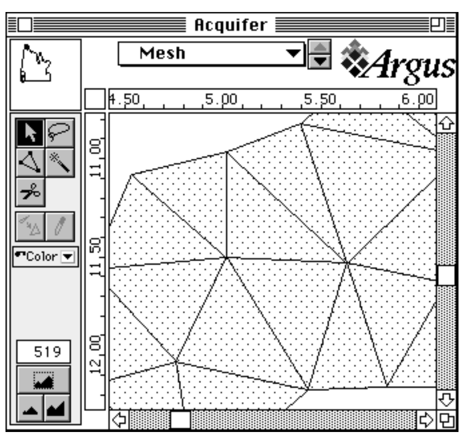
## To show and hide elements names

If you assign a name to an element, Argus ONE enables you to decide whether it shows on the screen. Argus ONE presents the name at the center of the element. If the element is too small to contain the name, Argus ONE does not show it. When you zoom in, Argus ONE tests each element again to decide whether the name can be contained within the element and if possible, presents it.

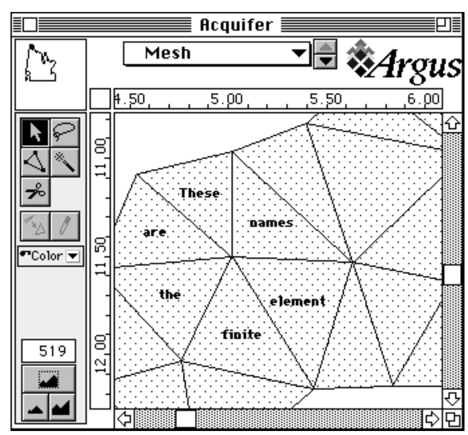
- To show elements names, from the View menu, choose Show Element Name.

A check mark is added to the left of the menu item.




Element names are hidden.




Element names are visible.

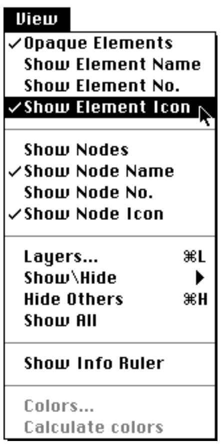
- To hide elements names, from the View menu, choose Show Element Name. The check mark to the left of the menu item is removed.

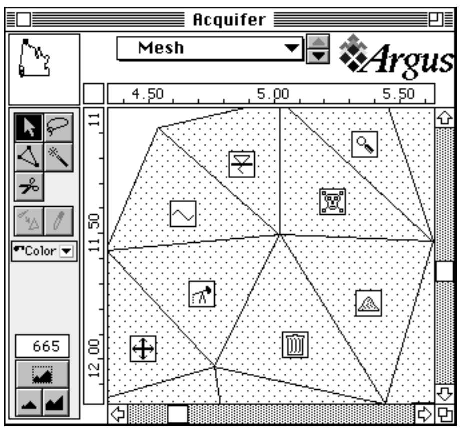
## To show and hide element icons

If you assign an icon to an element, Argus ONE enables you to decide whether it shows on the screen. Argus ONE presents the icon at the center of the element. If the element is too small to contain the icon, Argus ONE does not show it. When you zoom in, Argus ONE tests each element again to decide whether the icon can be contained within the element and if possible, presents it.

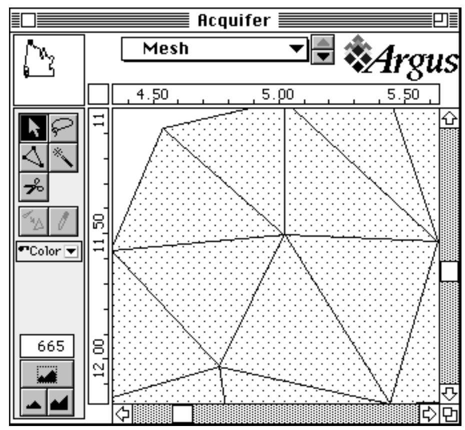
- To hide elements icons, from the View menu, choose Show Element Icon.

The check mark to the left of the menu item is removed.




Elements' icons are shown in elements large enough to contain their icon. To see the other icons, zoom in.




Elements' icons are hidden.

- To show elements icons, from the View menu, choose Show Element Icon. The menu item is marked with a check mark to its left.

To read about assigning nodes and elements names and icons, refer to the following section in this chapter.

# Selecting Elements and Nodes

To edit nodes and elements you must first select them. Selecting them tells Argus ONE that your actions are to affect the selected objects.

To learn about selection techniques, refer to the chapter “The Argus ONE Workplace.”

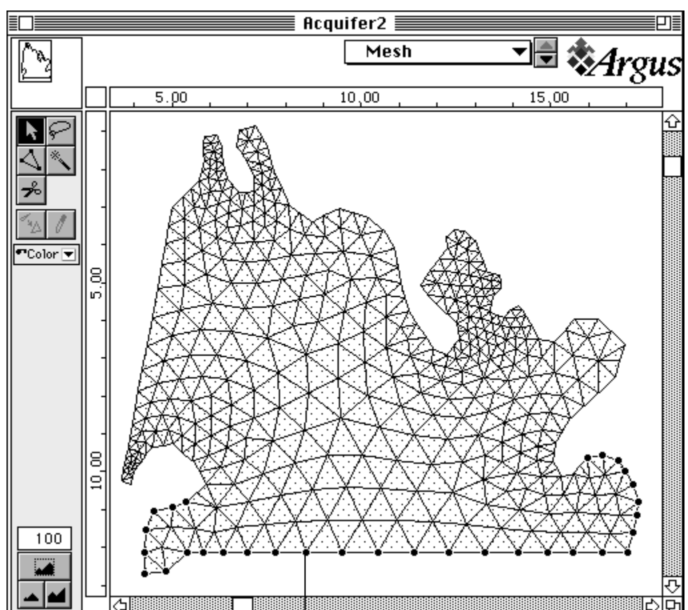
## Selecting Nodes Only

When in the mesh layer, Argus ONE enables you to select elements, nodes or both elements and nodes.

Selecting an element or a group of elements selects all nodes belonging to the selected elements. You can also select nodes only. This is very helpful if you want to assign a group of nodes the same value, for instance, to select some boundary nodes and assign them special boundary conditions.

### To select nodes only

Use any of the techniques explained in the chapter “The Argus ONE Workplace” but be sure to include only nodes. If a group of nodes you wish to select lies on a curved line, use the lasso tool.




*The selected nodes (highlighted) were selected using the Lasso tool.*

# Editing Nodes

A node is one of the three vertices of an element. A typical node in a mesh may be a vertex in few elements. Argus ONE enables you to perform extensive editing operations on a node.

You may move, detach, join and edit the information of a node. When you move a node, Argus ONE performs validity tests of the node's new location. Argus ONE will not allow you to relocate a node in one of two cases:

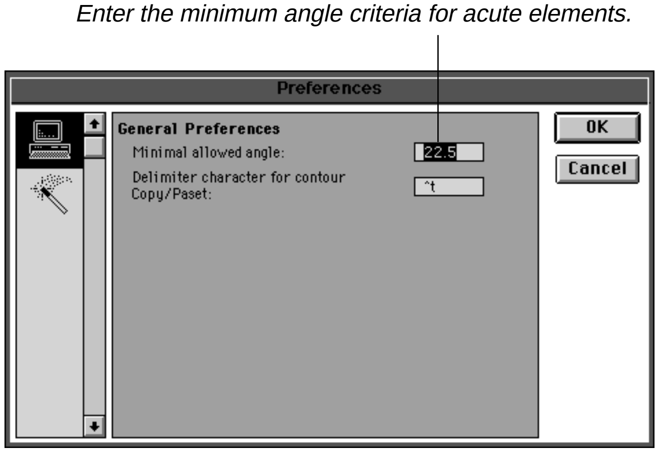
- If by relocating the node to its new position element overlapping will occur.
- If by relocating the node to its new position, one of the elements it belongs to becomes acute.

Argus ONE's default criteria for an acute element is that one of its angles is smaller than 22.5 degrees (see the chapter “The Mathematics Beneath Argus ONE.") You may change this criteria.

### To set the acute element criteria

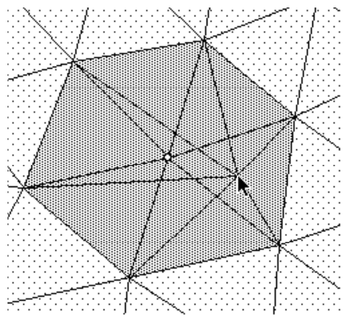
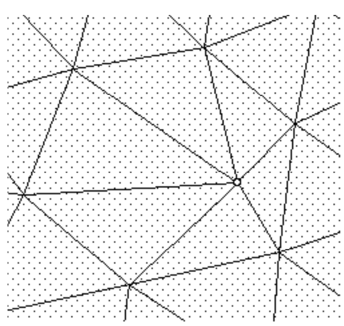
1. From the Special menu, choose the Preferences... command. Argus ONE opens the following dialog box:
2. Enter the minimal allowed angle of your choice.
3. Click the OK button.

*Enter the minimum angle criteria for acute elements.*




You may even set the minimum value to 0 degrees, but at your own risk. Argus ONE will not perform acute element validity tests. It will however still not allow you to create or reshape elements to be of zero area.

## Moving a node

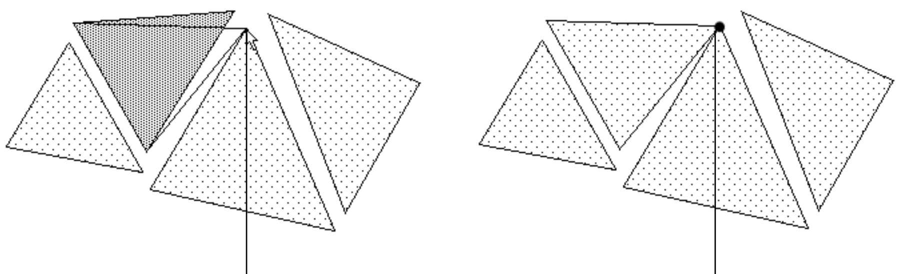
| User action and Argus ONE's reaction | Argus ONE's automatic validity testing |
|---|---|
| **1.** To move a node simply drag it.  An outline of the new shape of the connected elements follows your mouse movement, while also showing you the original node position. | |
| **2.** All elements connected to the relocated node are reshaped to accommodate for the new node location. | *If you release the node in a position where it causes element overlapping, Argus ONE beeps, and relocates the illegally moved node to its original location.* |
| **3.** Release the mouse button. The node relocates to its new position.  | *If while moving the node you stretch some elements so that their minimal angle becomes smaller than specified in the Preferences dialog box, Argus ONE presents a warning alert, and relocates the node to its original position.* |

## Joining Nodes

Argus ONE enables you to join nodes. Each time you drag a node above another node, the cursor changes into a hollow cursor, signaling you that if you release the mouse button at this location the two nodes will be joined into a single node.

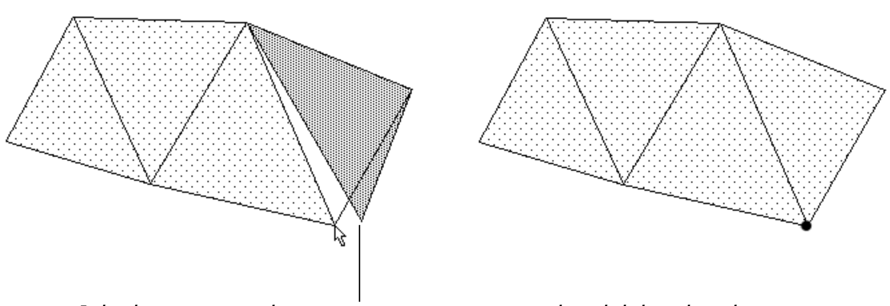
The two node will be joined only if Argus ONE validity tests are met.

*When the node being dragged is in the vicinity of another node, Argus ONE snaps the dragged node to the other node.*



*To join these elements, drag one of the nodes onto another...* *...the two nodes are joined, continue joining the rest of the nodes...*

*When you join nodes, the node being dragged is removed from the nodes list. All data you have manually set to it is lost.*



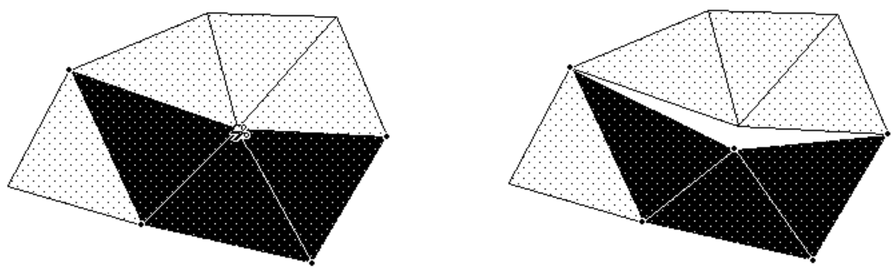
*...Join these two nodes to...* *...complete joining the elements.*

## Detaching Nodes

Detaching nodes is the reverse operation of joining nodes. Actually you detach elements at the detached node. All the selected elements will be detached from the non selected elements sharing the detached node. Detaching an element or a group of elements is explained later in this chapter.

To detach a node:

1. Select the elements their node you want to detach.
2. From the tool palette, choose the Detach tool ,The cursor changes into a detach cursor .



3. Click the node to be detached.
4. Move the node.

If the detached node was an internal node, (highlighted as a hollow circle) it changes to a boundary node (highlighted as a full circle)

- Or using a keyboard shortcut:

1. Select the element its node you want to detach.
2. Hold down the COMMAND key (Macintosh), CTRL key (other platform). The cursor changes into a detach cursor .
3. Bring the cursor over the node you wish to detach, the cursor changes into a hollow tool .
4. Drag the node you want to detach into the selected element.

If the detached node was an internal node, (highlighted as a hollow circle) it changes to a boundary node (highlighted as a full circle).

A new node is created. Argus ONE assigns the newly created node all the information you have manually assigned to the node it has been detached from, including its name and icon, but assigns it a new and unique node number.

## Assigning Node Information

A node is actually a detailed data structure. The node data structure contains its location, name, icon, and data from the other layers. Argus ONE allows you to access all data types. To read about creating node and element parameters refer to chapter 3. The following table summarizes node data.

| Node Data | Description | Access |
|---|---|---|
| Node Position | X and Y coordinates. | Move by mouse. |
| Node Name | Any string. | Nodes Info dialog. |
| Node Icon | One of ten pre-configured icons. | Nodes Info dialog and Node Icon menu item. |
| Data from linked layers | Any real number. | Through mesh layer parameters, referenced parameters, and manual override in the Nodes Info dialog. |

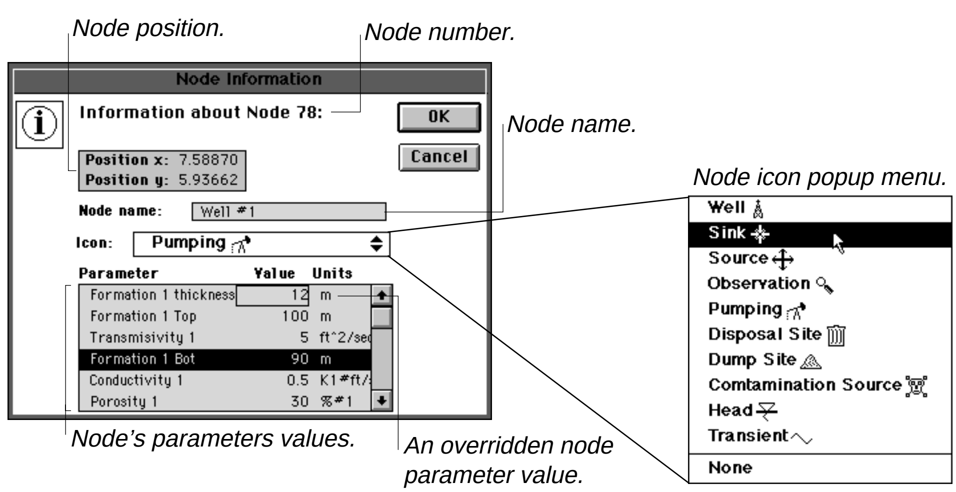
### To assign and edit node data

You can edit and assign node data at any time by opening the Node Information dialog.

*Use manual overrides carefully!*

When you manually set the value of a node parameter, the parameter is no longer evaluated for that node, instead the overridden value is used. If you create a new mesh or delete the node, the overridden value is lost.

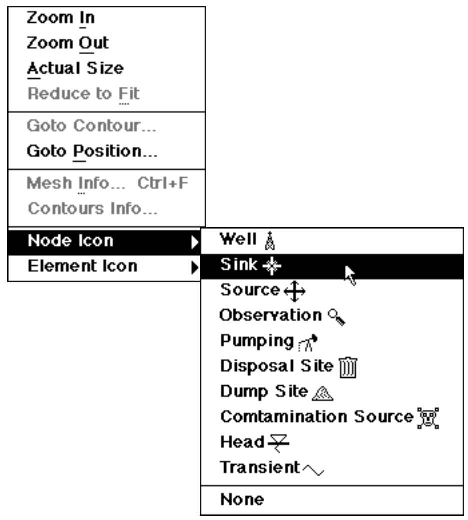
1. Select the node its data you want to edit.
2. From the Navigation menu, choose Nodes Info..., or Double-Click one of the selected nodes. The Node Information dialog box appears.



### To quickly set the node icon

You can assign a node or a group of nodes an icon without opening the Node Information dialog.

1. Select the nodes to be assigned an icon.
2. From the Information menu, choose Node Icon, a popup dialog opens.




*Select one of the icons.*

*Select None to remove the node icon.*

3. Select the icon you wish and release the mouse button.
   The selected nodes are assigned the icon.

- To remove an icon assigned to a node, select the None menu item.

# Editing Elements

## Moving Elements

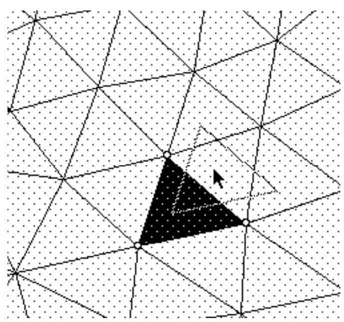
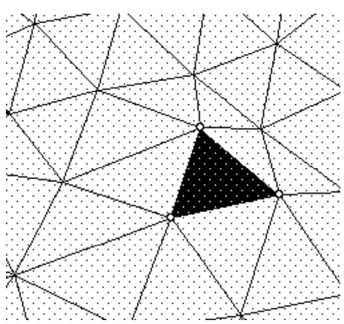
An element is the basic object of a mesh. Argus ONE enables you to perform extensive editing operations on an element.

You can move, detach, join and reshape an element, as well as edit the information associated with it. When you move an element, Argus ONE performs validity tests of the element's new location and shape. Argus ONE will not allow you to relocate an element in one of two cases:

- If by relocating the element to its new position, element overlapping occurs.
- If by relocating the element to its new position, one of the elements it is connected to becomes acute.

Argus ONE's default criteria for acute element is that one of its angles is smaller than 22.5 degrees (see the chapter “The Mathematics Beneath Argus ONE.”) You may change this criteria, as explained earlier in this chapter.

### To move an element

| User action and Argus ONE's reaction | Argus ONE's automatic validity testing |
|---|---|
| **1.** To move an element or a group of elements, select it and move the mouse while pressing down the mouse button. An outline of the elements you're moving follows your mouse movement. | |
|  | *If you release the elements in a position where they overlap other elements, Argus ONE presents a warning alert, and relocates the illegally moved elements to their original location.* |
| **2.** Release the mouse button and the elements redraw themselves in their new location. | |
|  | *If while moving the elements you stretch one of the connected elements so that its minimal angle becomes smaller than specified in the Preferences dialog, Argus ONE presents a warning alert, and relocates the elements to their original position.* |
| Connected elements are distorted to adjust for the new location of the moved element. | |

## Deleting Elements

To delete elements, all you have to do is select them and hit the DELETE key on your keyboard. If you delete all the elements connected to a node, the node is also deleted.

## Elements numbering

When you delete elements, Argus ONE does not keep track of the original numbering. After you delete an element, the last created element is assigned the number of the deleted element. The same applies to nodes being deleted when their elements are deleted.

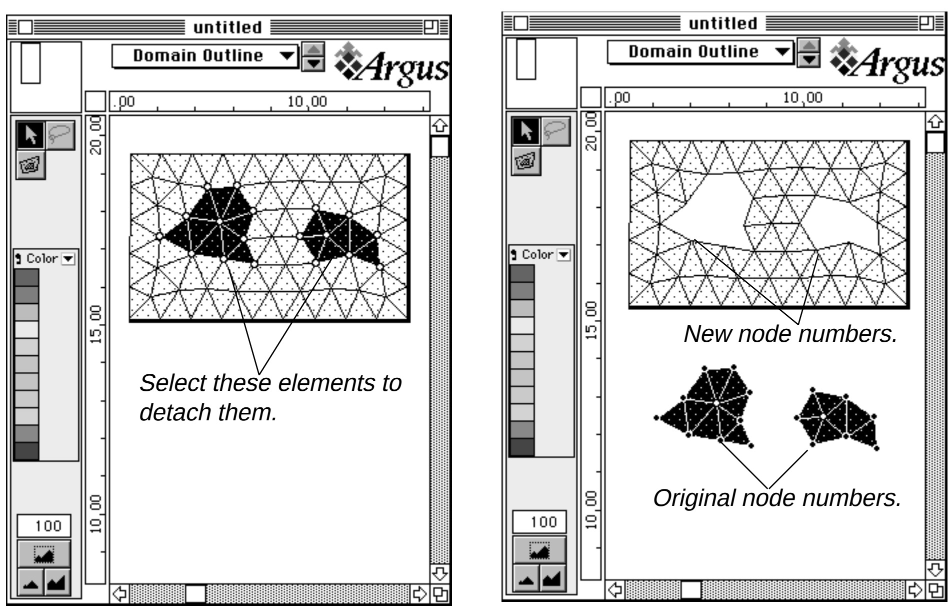
## Detaching Elements

If you want to detach an element or a group of elements from the mesh, you could use the detach tool to manually detach each of the nodes. However, since this might be a tedious task, Argus ONE enables you to detach an element or a group of elements using a single mouse operation.

### To detach an element or a group of elements

1. Select the elements you want to detach.
2. From the tool palette, choose the Detach tool, the cursor changes into a scissors cursor .

*You can also detach elements using the Detach Elements command from the Edit menu.*


3. Click the elements to be detached (one of the groups), and drag it out of the mesh.
   - Or using a keyboard shortcut:

1. Select the elements you want to detach.
2. Hold down the COMMAND key (Macintosh, other platforms - CONTROL key), the cursor changes into a scissors cursor .
3. Bring the cursor over the elements you wish to detach.
4. Drag the elements you want to detach out of the mesh.

If you release the detached elements over other elements, Argus ONE alerts you that they overlap other elements and relocates them to their original location.

### Node numbering

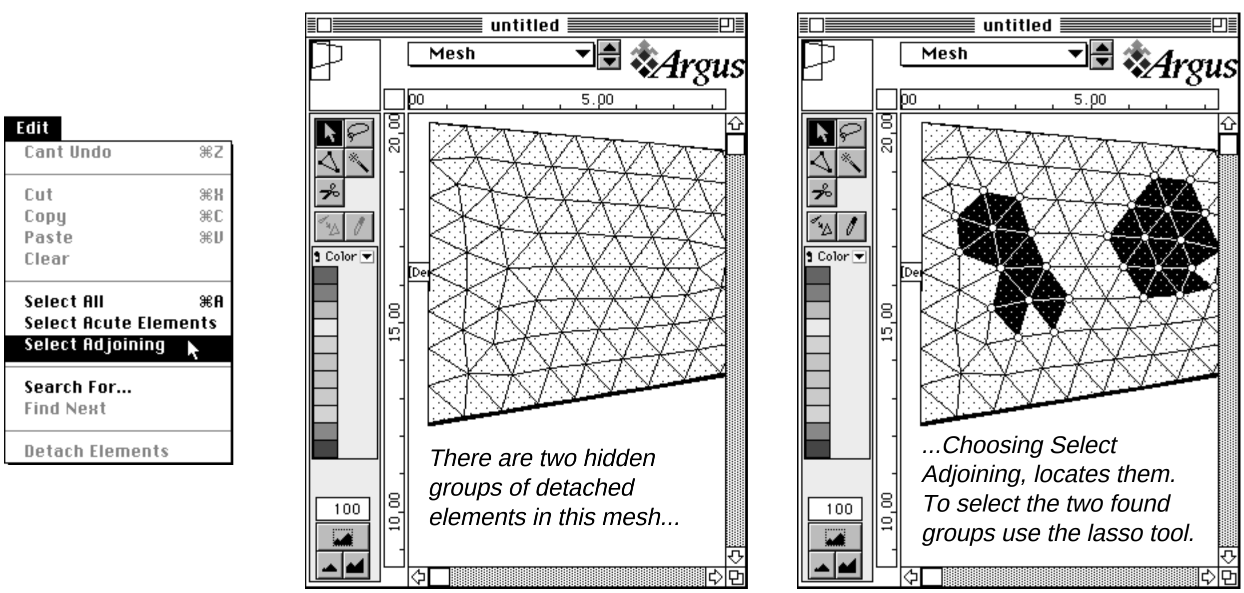
When you detach elements, Argus ONE automatically creates new nodes. The nodes on the boundary of the detached group are left with their original numbers. The new nodes within the mesh are assigned new numbers.

### Finding a detached group of elements within the mesh

If you detach a group of elements and do not move it out of the mesh, your mesh may look at first glance as a continuous one, but it is not. To find the lost group of elements:

1. Select any element in the mesh.
2. From the Edit menu, choose Select Adjoining.

Argus ONE selects all elements in the mesh that are connected to the element you have selected.



## Assigning Element Information

Like a node, an element is actually a detailed data structure. The element's data structure includes its center's location, area, minimal angle, three nodes, name, icon, and data from the other layers. Argus ONE allows you to access all data types. The following table summarizes element data.

| Element Data | Description | Access |
|---|---|---|
| Element Position | X and Y coordinates of the element's center of gravity. | Move or reshape by mouse. |
| Element Name | Any string. | Element Info dialog. |
| Element Icon | One of ten pre-configured icons. | Element Info dialog and Element Icon menu item. |
| Data from linked layers | Any real number. | Through mesh layer parameters, referenced parameters, and manual override in the Element Info dialog. |

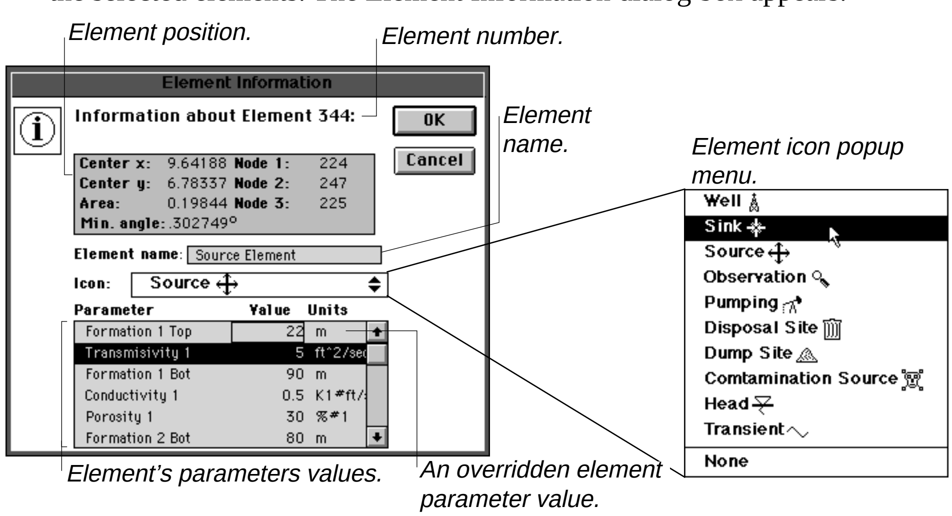
### To assign and edit element data

You can edit and assign element data at any time by opening the Element Information dialog.

*Use manual overrides carefully!*

When you manually set the value of an element parameter, the parameter is no longer evaluated for that element, instead the overridden value is used. If you create a new mesh or delete the element, the overridden value is lost.

1. Select the element its data you want to edit.
2. From the Navigation menu, choose Elements Info..., or Double-Click one of the selected elements. The Element Information dialog box appears.



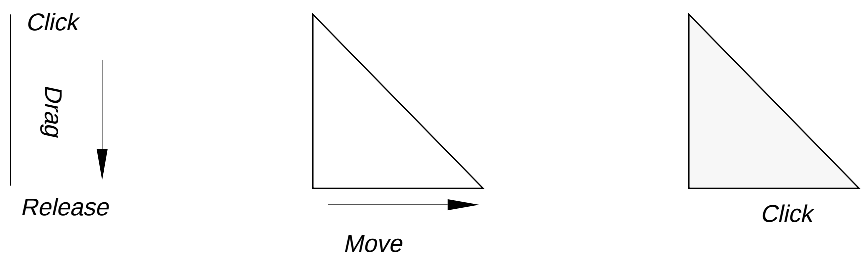
# Creating Elements Manually

Argus ONE enables you to create elements manually. This might become useful when you want to fine-tune the mesh at specific areas.

## To Manually Create an Element

1. From the tool palette, select the Manual Element Creation tool  .
2. Click the mouse button where you want to record the first node, do not release the mouse button yet.
3. While holding the mouse button, drag the mouse to the second node's location.The line following your mouse movement is the first side of the element.
4. Click the mouse button where you want to create the second node.
   The second node is created and the element’s first side is fixed.
5. Release the mouse button.
   The two remaining sides of the element follow your mouse movements.
6. Click the mouse button where you want to create the third node.

The element is created and filled with a pattern.




*To create the first two nodes.* *Move the mouse to the third node’s location.* *Click to create the third node.*

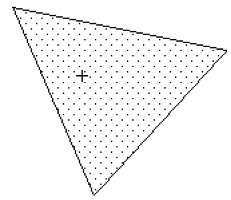
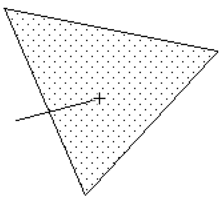
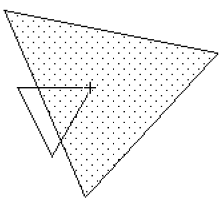
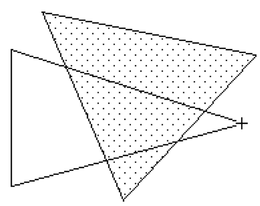
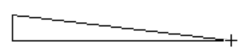
## Validity Tests

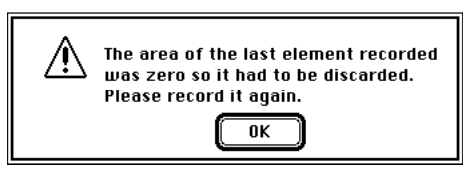
Manual element creation adheres to all the basic validity rules that apply to elements.

These rules are:

- Elements can not overlap.
- The element's smallest angle must be larger than the minimal angle criteria you set.
- An element can not be of zero area.

When you manually create elements, Argus ONE performs all the needed validity tests to ascertain the elements created comply with the above set of rules.

| **User action** | | **automatic validity testing** |
|---|---|---|
| • Click in an element to create a new element's first node. |  | *Argus ONE beeps and does not record the element.* |
| • Click in an element to create a new element's second node. |  | *Argus ONE beeps and does not record the element.* |
| • Click in an element to create a new element's second or third node. |  | *Argus ONE beeps and does not record the element.* |
| • Click across an element to create a new element's third node. |  | *Argus ONE beeps and does not record the element.* |
| • Click to create an acute element. |  | *Argus ONE brings up an alert describing the problem and does not record the element.* |
| • Click to create an empty element. |  | *Argus ONE brings up an alert describing the problem and does not record the element.* |

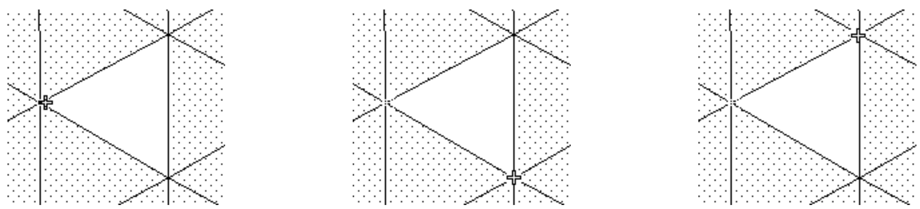



*If you set the Minimal Angle criteria in the Preferences dialog to zero degrees you may accidentally try to create zero area elements.*

### To create an element using existing nodes

When creating an element manually, you can record any of its nodes within existing nodes. When the Element Creation cursor lies over an existing node it changes its appearance into a hollow cross . If you click the tool on an existing node, Argus ONE does not record a new node.

For instance if you want to fill the hole in this mesh:



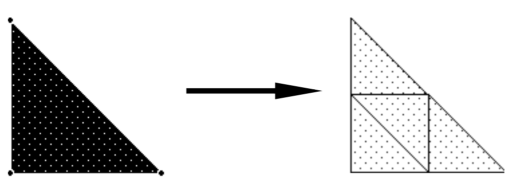
### Constraining cursor movements

Using the cursor constraining functions you can create right angle triangles, equal sided triangles, etc. The Angle constraining function is explained in detail in part 2 chapter 1.

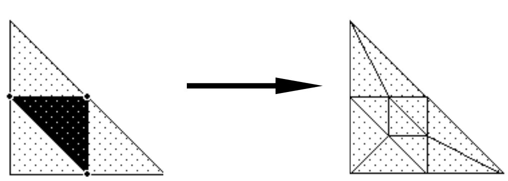
## Refining Elements

To refine the mesh at a specific zone you can either use the mesh density contour techniques and remesh, or use the techniques explained in the following paragraphs.

When using the Refinement button, Argus ONE splits each selected element into four similar elements.

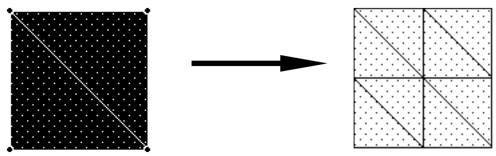


If the element to be refined has neighbors that are not to be refined, Argus ONE splits them into two elements to avoid transition elements.



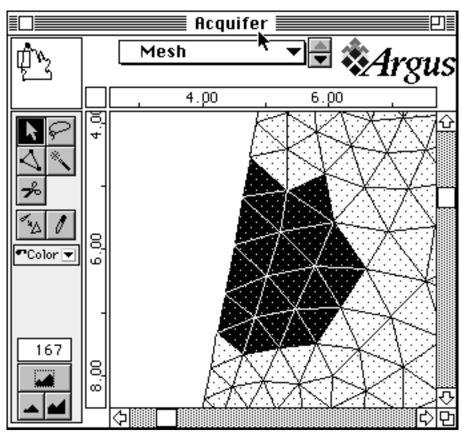
This method's disadvantage is that it tends to degrade the adjacent elements by reducing one of their angles to half it original size. This is the only operation for which Argus ONE does not perform validity tests for the minimal angle created to comply with the criteria you set in the Preferences dialog.

If you use the Refinement button to refine only complete groups of elements (including the boundary elements) refinement will not degrade any of the elements.

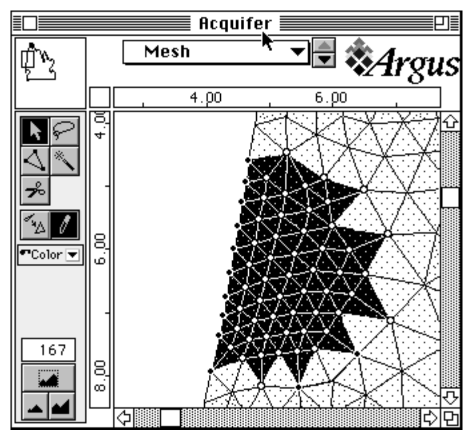


### To refine elements

**1.** Select the elements you wish to refine.




*To refine the selected elements...*




*...Click the refine button.*

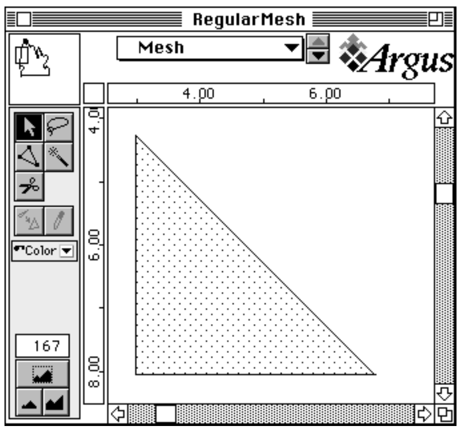
**2.** Click the Refine button  in the tool palette.

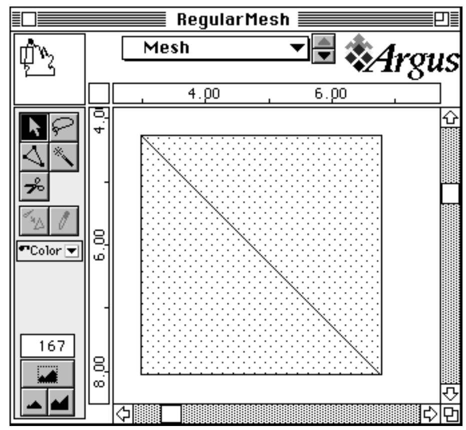
A progress bar indicates the progress of the refinement process.

The selected elements and their neighbors are refined.

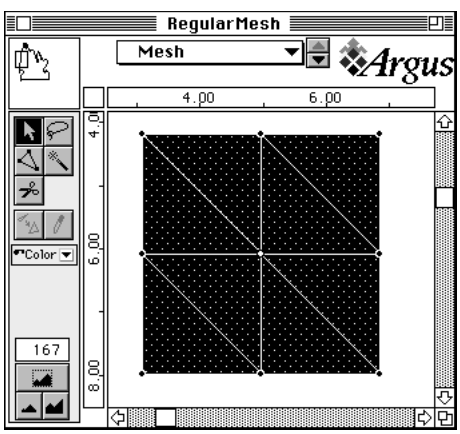
## Using manual element creation and refinement to create a regular mesh

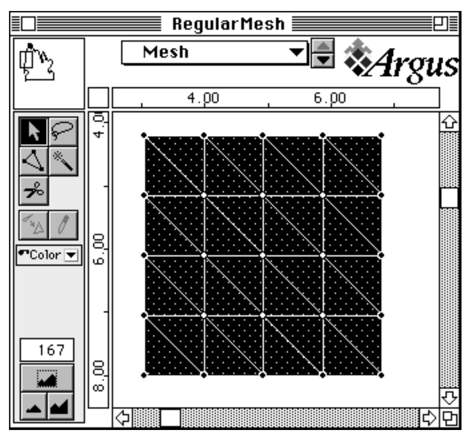
1. Create a right angle triangle.
2. Create a complementary right angle triangle connected to the first one to form a rectangular shaped mesh.






3. Select both elements.
4. Refine the selection as many times as you need.





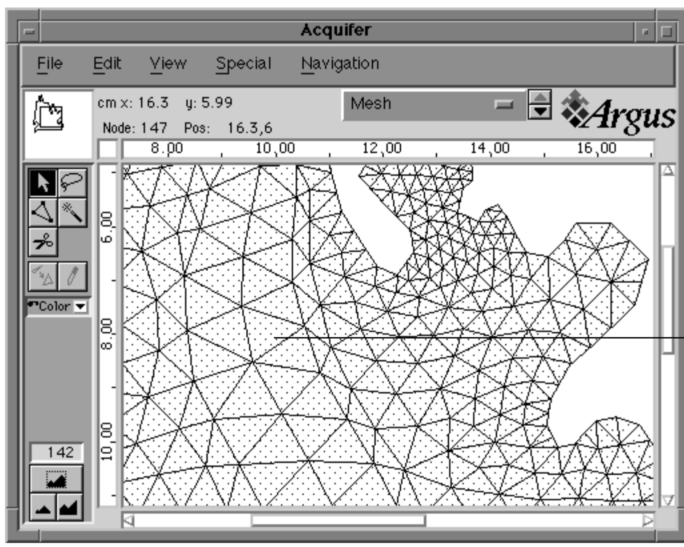
# Smoothing Elements

Mesh smoothing is carried out by Argus ONE as part of the auto mesh generation procedure. However, to smooth parts of the mesh you have created manually or edited, you can select these parts and smooth them.

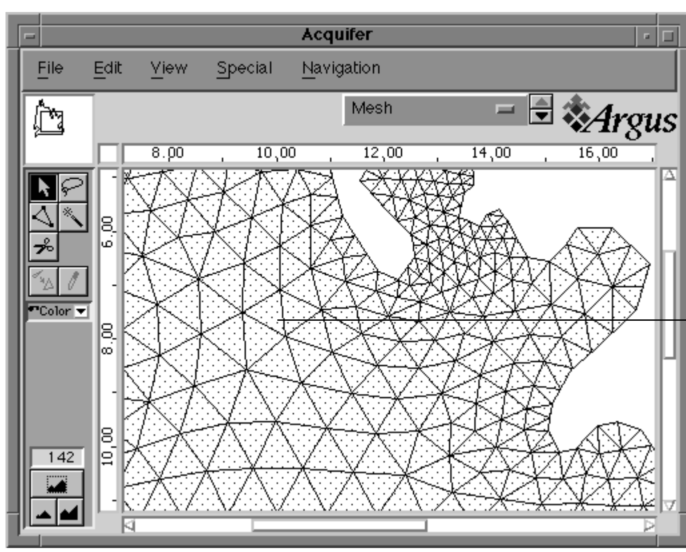
The smoothing algorithm is an iterative procedure that moves the nodes to the center of gravity of their respective polygons.

### To smooth elements

1. Select the elements you wish to smooth.
2. In the tool palette, click the Smooth button.




*To smooth these elements, select all elements or just the group to be smoothed and click the smooth button.*



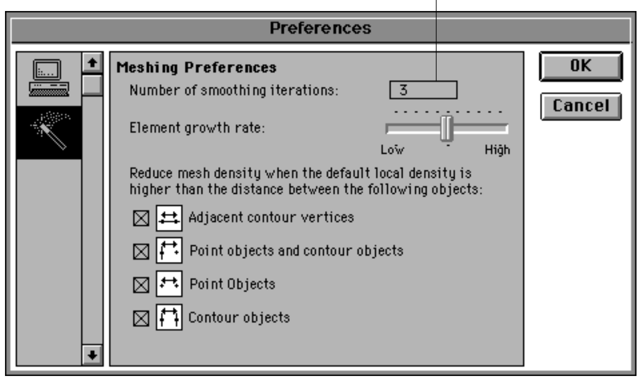

*Argus ONE smooths the elements by moving each node to the center of gravity of the elements it is connected to.*

Smoothing is achieved after a few iterations. Argus ONE's default number of iterations is set to three. Although we have found that after three iterations sufficient convergence is achieved, you can change this default to any number above two.

### To set the number of smoothing iterations

1. From the Special menu, choose Preferences...,
2. Set the preferred number of iterations.

*Enter number of smoothing iterations.*




# Renumbering (BandWidth Optimization)

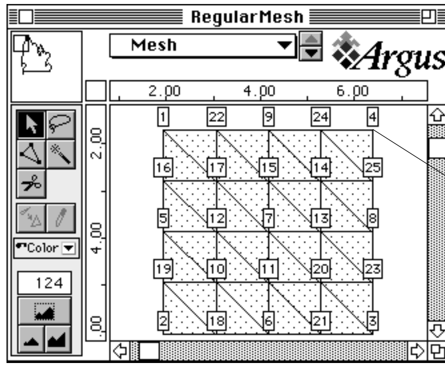
The efficiency of some numerical simulators depends on the half BandWidth of the matrix they operate on. The BandWidth of the matrix is determined by the numbering of nodes. Each row in the matrix to be solved describes the connectivity of a node (on the main diagonal) to the other nodes it is connected to. The matrix BandWidth is defined as the difference between the diagonal term and the most distant non zero term in the row, over all rows. Renumbering the mesh can results in an improved BandWidth by reassigning node numbers such that the numbers of connected nodes are as close as possible.

If you use such solvers, Argus ONE can optimize the mesh numbering to achieve smaller BandWidth. The renumbering algorithm is a fast “minimum degree” heuristic algorithm.

Argus ONE can perform the renumbering if the mesh layer contains only one complete continuous mesh.

## Renumbering a Mesh, an Example

The following mesh was created using the technique described in the paragraph “Using Manual Element Creation tool and Refinement to create regular Meshes.”




*Node number 4 is connected to node 25. The half bandwidth is equal to 21.*

The following table is the connectivity matrix of the above presented mesh. The connectivity matrix is symmetric. However, the following matrix shows the connections appearing in the export file of the mesh.

*The matrix is very sparse, the half BandWidth is 21.*

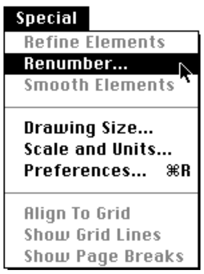
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | | | | | | | | | | | | | | | x | x | | | | | x | | | |
| 2 | | 2 | | | | | | | | | | | | | | | | x | x | | | | | | |
| 3 | | | 3 | | | | | | | | | | | | | | | | | x | x | | x | | |
| 4 | | | | 4 | | | | | | | | | | | | | | | | | | | | x | x |
| 5 | | | | | 5 | | | | | x | | x | | | | x | | | x | | | | | | |
| 6 | | | | | | 6 | | | | x | x | | | | | | | x | | | x | | | | |
| 7 | | | | | | | 7 | | | | x | x | x | | x | | x | | | x | | | | | |
| 8 | | | | | | | | 8 | | | | | x | x | | | | | | | | | x | | |
| 9 | | | | | | | | | 9 | | | | | x | x | | | | | | | x | | x | x |
| 10 | | | | | | | | | | 10 | x | x | | | | | | | | | | | | | |
| 11 | | | | | | | | | | | 11 | | | | | | | | | | | | | | |
| 12 | | | | | | | | | | | | 12 | | | | | | | | | | | | | |
| 13 | | | | | | | | | | | | | 13 | x | x | | | | | | | | | | |
| 14 | | | | | | | | | | | | | | 14 | | | | | | | | | | | |
| 15 | | | | | | | | | | | | | | | 15 | | | | | | | | | | |
| 16 | | | | | | | | | | | | x | | | | 16 | x | | | | | | | | |
| 17 | | | | | | | | | | | | | | | x | | 17 | | | | | x | | | |
| 18 | | | | | | | | | | x | | | | | | | | 18 | x | | | | | | |
| 19 | | | | | | | | | | | | | | | | | | | 19 | | | | | | |
| 20 | | | | | | | | | | | x | | | | | | | | | 20 | x | | | | |
| 21 | | | | | | | | | | | | | | | | | | | | | 21 | | | | |
| 22 | | | | | | | | | | | | | | | | | | | | | | 22 | | | |
| 23 | | | | | | | | | | | | | x | | | | | | | x | | | 23 | | |
| 24 | | | | | | | | | | | | | | x | | | | | | | | | | 24 | x |
| 25 | | | | | | | | | | | | | | | | | | | | | | | | | 25 |

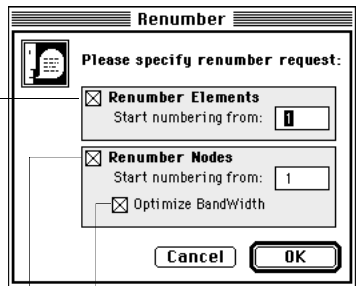
## To optimize the mesh BandWidth

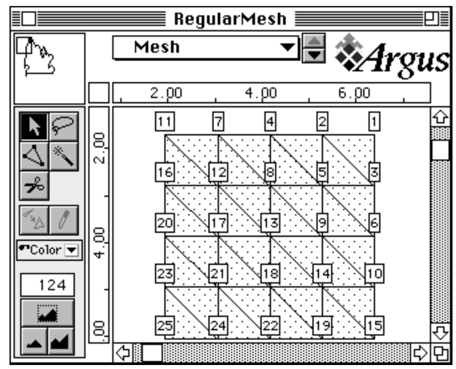
**1.** From the Special menu, choose Renumber...
The Renumber dialog opens.

The Renumber dialog enables you to set the starting node and element number at any integer, and to optimize the Band Width as well.

**2.** Check the Optimize Bandwidth check box.




*Check to optimize bandwidth.*

*Check to renumber the nodes and enter the starting integer.*

*The resulting half bandwidth equals to 5.*

*Check to renumber the elements and enter the starting integer.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | x | x | | | | | | | | | | | | | | | | | | | | | | |
| 2 | | 2 | x | | x | | | | | | | | | | | | | | | | | | | | |
| 3 | | | 3 | | | | | | | | | | | | | | | | | | | | | | |
| 4 | | x | | 4 | x | | x | x | | | | | | | | | | | | | | | | | |
| 5 | | | | | 5 | | | | | | | | | | | | | | | | | | | | |
| 6 | | | x | | x | 6 | | | x | x | | | | | | | | | | | | | | | |
| 7 | | | | | | | 7 | | | | | | | | | | | | | | | | | | |
| 8 | | | | | | | | 8 | | | | | | | | | | | | | | | | | |
| 9 | | | | | x | | | x | 9 | | | | | | | | | | | | | | | | |
| 10 | | | | | | | | | x | 10 | | | | x | | | | | | | | | | | |
| 11 | | | | | | | x | | | | 11 | x | | | | x | | | | | | | | | |
| 12 | | | | | | | x | x | | | | 12 | | | | | | | | | | | | | |
| 13 | | | | | | | | x | x | | | x | 13 | x | | | x | x | | | | | | | |
| 14 | | | | | | | | | | | | | | 14 | | | | x | x | | | | | | |
| 15 | | | | | | | | | | x | | | | x | 15 | | | | x | | | | | | |
| 16 | | | | | | | | | | | | x | | | | 16 | x | | | | | | | | |
| 17 | | | | | | | | | | | | | | | | | 17 | | | | | | | | |
| 18 | | | | | | | | | | | | | | | | | | 18 | | | | | | | |
| 19 | | | | | | | | | | | | | | | | | | | 19 | | | | | | |
| 20 | | | | | | | | | | | | | | | | x | x | | | 20 | x | | x | | |
| 21 | | | | | | | | | | | | | | | | | x | x | | | 21 | | | | |
| 22 | | | | | | | | | | | | | | | | | | x | x | | x | 22 | | x | |
| 23 | | | | | | | | | | | | | | | | | | | | | | | 23 | | |
| 24 | | | | | | | | | | | | | | | | | | | | | x | | x | 24 | |
| 25 | | | | | | | | | | | | | | | | | | | | | | | x | x | 25 |

*The resulting matrix's half Band Width equals 5. The matrix is much less sparse.*

# Searching the Mesh Database

As you add information to your project, using the Information layers and node and element information fields, you're actually building a database. Argus ONE enables you to perform extensive and complex searches of the nodes and elements, based on their data.

The Search For... command is the graphical interface to the query language you use to build your queries. It allows you to perform searches based on all the types of data assigned to a node and an element:

- Node and element number.
- Node and element name.
- Node and element icon.
- Node and element data originating from linked parameters.

It also enables you to differentiate boundary and domain nodes and elements.

You can use these searching capabilities to orientate within the project or to create selection groups for exporting specific parts of the mesh.

While iterating through the processes of model calibration and solving for different realizations, you can use the search facilities to relate numerical information (node numbers) from the solution program with the nodes' or elements' location in the mesh. For instance, if you get a singular solution at some nodes, and you want to query the boundary or initial conditions at these nodes or elements, you can search for these nodes/elements using the Search For... command. You can then use the Node Information dialog to change these nodes' and/or elements' values one by one or all at once.

The Search For dialog enables you to search for elements or nodes, define a search criteria and to decide how to treat the current selection.

After you have selected a group of nodes or elements based on some search criteria, you can extend or reduce the found group by adding or removing from it another group of nodes or elements satisfying a different search criteria.

You can also negate each search criteria.

## Using Nodes or Elements Parameters to Define Search Criteria

Since you can assign nodes and elements as many parameters as needed, you can use these parameters to create intricate search criteria. To do so, create mesh parameters containing the expressions needed or create other layers parameters and link them to mesh parameters. Then, in the Search For dialog box, choose these parameters to define the search criteria.

# To Search a Mesh

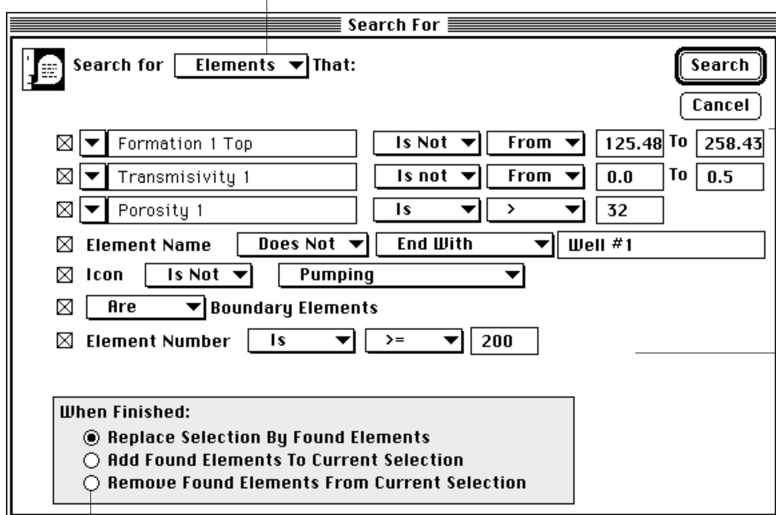
1. Make sure the active layer is a Mesh Layer.
2. From the Edit menu, choose the Search For... Command.

The following dialog appears.

3. Set all the fields you need in order to define the search criteria.
4. Press the Search button.

While searching, a progress dialog appears to indicate the search progress.

*Choose elements or nodes.*


*Set the search range.*

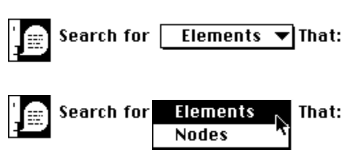
*Using several search processes together with different settings of the current found set enables you to expand or reduce the search scope.*

## Defining a search criteria

In the following paragraphs each of the fields and options in the Search For dialog are explained.

### Choosing between elements and nodes search

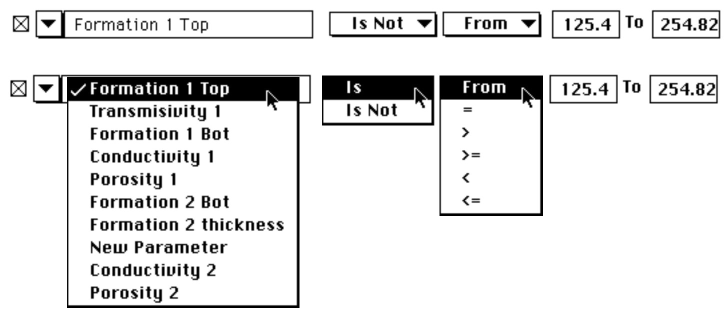
- From the Search For popup menu, choose Nodes or Elements.



### Choosing the nodes or elements parameters to be searched

You can create a search criteria containing up to three linked parameters. If you need to define a search criteria based on more linked parameters, execute one search for the first three and then define and execute more searches using the Add Found To Current Selection option.

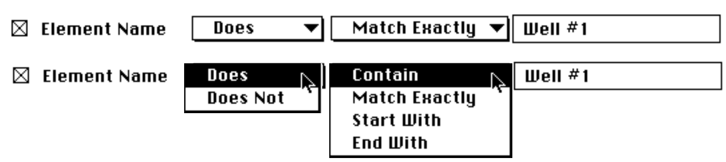
- From each of the three Parameters popup menus, choose the parameter to be searched for and assign its range.




### To search for nodes’ or elements’ names

Assigning names to very special nodes and elements can help you to orientate within the mesh.

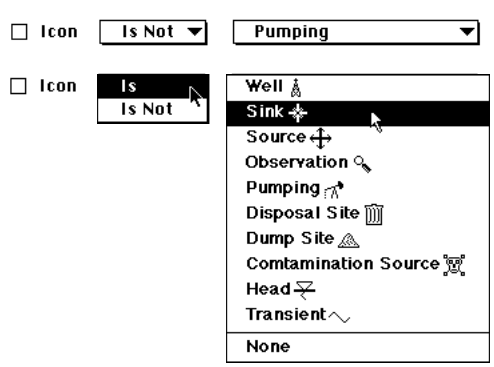
- Enter the name and specify the search criteria.




### To search for nodes’ or elements’ icons

Assigning icons to very special nodes and elements can help you to orientate within the mesh.

- From the Icon popup menu choose the icon and specify the search criteria.

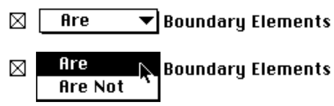


### To search for boundary nodes or elements

Boundary nodes are defined as nodes lying on the outline of the mesh, on external as well as internal domain outline contours. Boundary elements are defined as elements having at least one of their nodes lying on internal or external mesh boundaries.

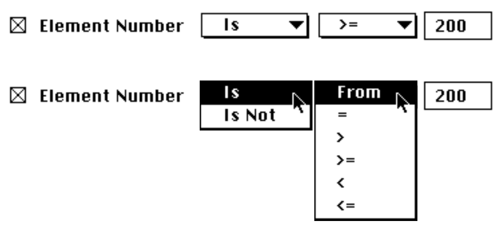
Negating the search allows you to find internal nodes and elements.

- Choose Are or Are Not.

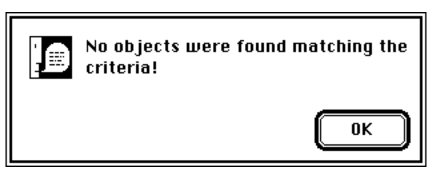



### To search for nodes or elements numbers

- Set the node or element number range.




The nodes or elements matching the search criteria are selected. If no nodes or elements matching the search criteria are found, Argus ONE reports it with the following alert:



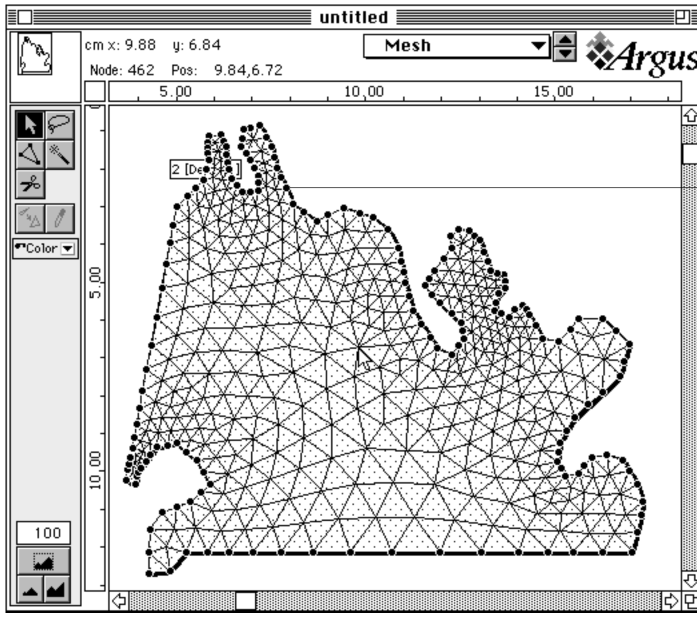

## Expanding the Search Scope

By default the dialog is set to “Replace Selection By.” If when invoking the dialog some nodes or elements are already selected, they are de-selected and replaced by the newly found set.

To expand or reduce the search scope over an already found set, change this field to “Add Found To Current Selection” or “Remove Found From Current Selection” and define the additional search.

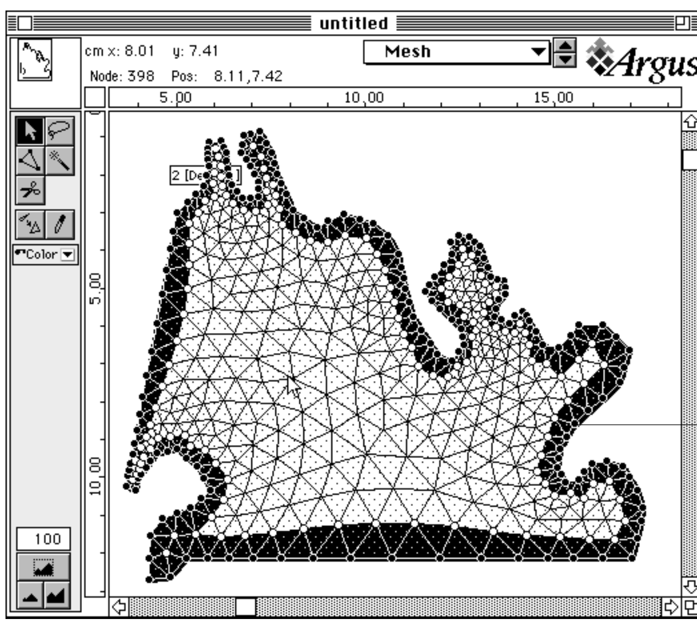
## Search Examples

- To find all nodes lying on the mesh boundary, check the Nodes On Boundary check box and initiate the search.




*Boundary nodes are selected.*

- To find all elements lying on the mesh boundary, check the Elements On Boundary check box and initiate the search.

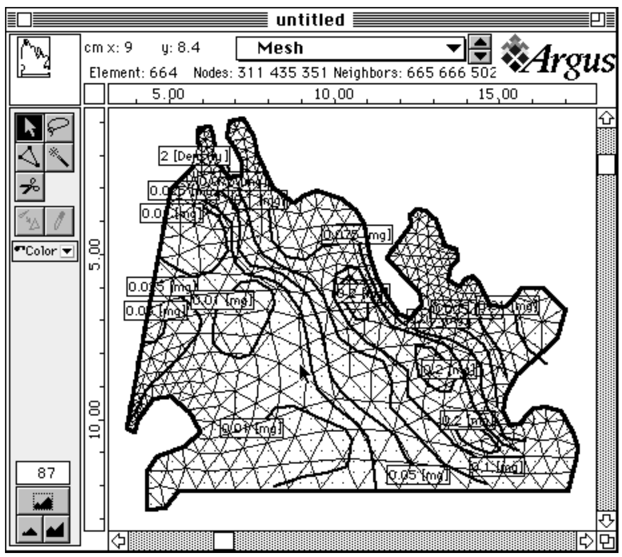



*Boundary elements are selected.*

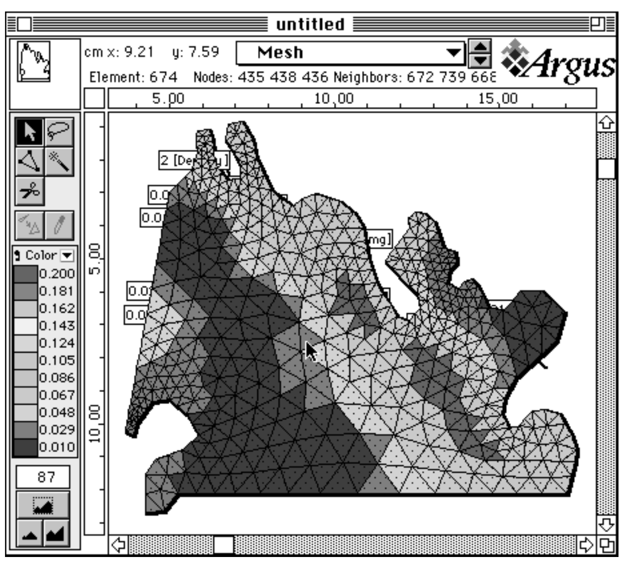
# Coloring Elements

Using the color palette you can instruct Argus ONE to evaluate the mesh elements with respect to any of the element parameters. Using colors is explained in detail in chapter 1. To color the elements in the following screenshot, with respect to their values in the information layer seen through the mesh:

1. Create a mesh or element parameter and link it to the information layer.
2. From the Colors popup menu, select the linked parameter.




3. Click the colors handle to activate coloring.



## Colors and Performance Considerations

When colors is on and a parameter is chosen in the Colors popup menu, each element in the mesh is evaluated at the evaluation context (the element center) for its value in the linked parameter.

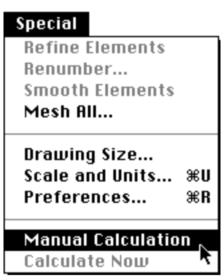
This evaluation is carried out during each redraw of the screen. If your mesh contains a large number of elements this re-evaluation process slows the redraw considerably.

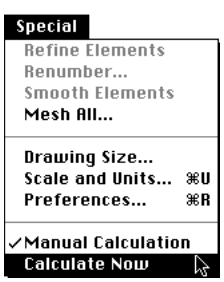
Coloring elements should be considered as a post-processing feature. You should not try to work with colors all the time but use colors when you need to preview a parameter's distribution in the elements. If however you do wish to work with colors on at all times, consider using the Manual Calculation command explained below.

### Temporarily halting color evaluation

If colors is on, and you wish to stop color re-evaluation immediately, and even before all elements are redrawn in color, press COMMAND + . (period) on the Macintosh, or CTRL + C on other platforms. Color evaluation is halted and the redraw continues, but in black and white mode.

## Manual Calculation and Calculate Now Menus




If your mesh or grid contain many elements or blocks which are linked to complex parameters, screen redraws may temper with your work. Using the Manual Calculation command under the Special menu you can now instruct Argus ONE to refrain from recalculating elements and block colors (values) at each redraw.

However, you must be aware that when Manual Calculation is on, the colors of elements and blocks, as well as the values presented on the different information dialogs will not reflect any changes you have made from the time you turned manual Calculation on. To update the values, you can either select Calculate Now from the Special menu, or turn Manual calculation off.

---

**Important Note**: To prevent you from exporting wrong information to your model, Argus ONE automatically performs a Calculate Now command while exporting the data.

---

# Exporting and Importing a Mesh

# Overview

Exporting and importing meshes is how you communicate data created within Argus ONE to and from other applications. Export and import are both based on the ASCII data format.

*To control the export format you need to have the Programmable Export Templates module*

If you use your own models, or models for which Argus ONE PIEs Interfaces do not exisit, Argus ONE enables you to create the exact export format that will fit with your simulators' import formats. You do so by using the Programmable Export Templates module. Creating and using export templates is discussed in detail in chapter 5 "Exporting a Project."

If you are using Argus ONE with a ready-made PIE, the PIE takes care of defining the appropriate export template for you.

***Important note:****Using the Argus ONE built-in export format is faster than exporting "By Template."*

In this chapter you will learn about exporting your projects using the built-in export format which is available as part of Argus ONE meshing modules. You will also find out how to import meshes you have created elsewhere into Argus ONE.

# Exporting a Mesh

Exporting the mesh you have created within Argus ONE, and its related data, enables you to input it to processing and modeling programs.

## Range and Precision of Data

All real data is internally handled by Argus ONE in double format. The following table summarizes the range of real numbers Argus ONE can work with.

| **Type identifier** | **Double** |
|---|---|
| Size (bytes:bits) | 10:64 |
| Significand precision | |
| Bits | 64 |
| Decimal digits | 15 |
| Decimal range (approximate) | |

| Type identifier | Double |
|---|---|
| Min negative | -1.1E+308 |
| Max neg norm | computing platform dependent |
| Min pos norm | computing platform dependent |
| Max positive | 1.7E+308 |

Erroneous values resulting from invalid parameter expressions, circullar reference, etc. are exported as NaN (Not a Number).

## Non Exportable Mesh Variables

Only node and element names and icons are not exportable.

## Built-in Export File Format

Argus ONE’s built-in export supports the following file format for exporting meshes. Data fields can be delimited by the Tab, Space or Comma, or any other string of characters.

### The first line of a mesh export file

| | | | |
|---|---|---|---|
| Number of Elements | Number of Nodes | Number of mesh & node parameters... | Number of mesh & element parameters |

### A line describing a node

There are “Number of Nodes” lines describing the nodes. They are ordered from node number 1 to “Number of Nodes”. Each node line starts with the character “N”, denoting Node.

| | | | | | | |
|---|---|---|---|---|---|---|
| “N” | Number | X | Y | Name | Value of first node or mesh parameter... | Value of last node or mesh parameter |

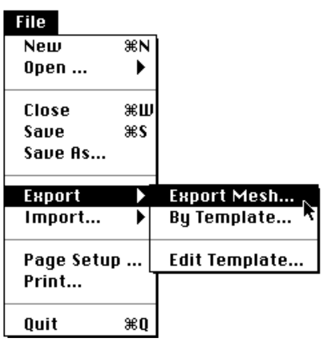
### A line describing an element

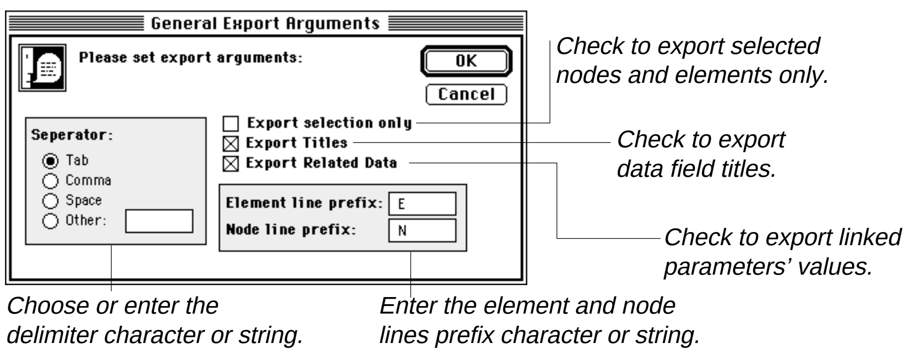
There are "Number of elements" lines describing the elements. They are ordered from element number 1 to "Number of elements". Each element line starts with the character "E", denoting Element. The connectivity list, that is the three node numbers of each element are ordered counter-clockwise in a right coordinate system and clockwise in a left coordinate system.

| "E" | Number | First Node | Second Node | Third Node | Value of first element or mesh parameter... | Value of last element or mesh parameter |
|---|---|---|---|---|---|---|

## To Export a Mesh Using the Built-in Export

1. If you have not yet linked parameters to the mesh and its objects, and do need to do so, open the Layers dialog, create the parameters and link them.
2. Make sure the active layer is a mesh layer.
3. If you want to export only parts of the mesh, select these parts.
4. From the File menu, choose Export Mesh...
   The General Export Arguments dialog opens.




The Export dialog box allows you to determine the shape of the exported file, and the delimiters between the different fields.

- To set the delimiter, check the appropriate radio button.

  Selecting Tab as the delimiter will allow you to open the file in a spreadsheet type application, where all data columns will be arranged in separate columns.

---

**Note:** Files exported with the comma character can not be re-imported by Argus ONE.

---

- To export the selection only, check the Export Selection Only check box.

The first line of an exported file of a partial mesh (Export Selection Only) contains the string: “# Partial mesh!”.

---

**Note:** If a partial selection is exported, it can not be re-imported into Argus ONE.

---

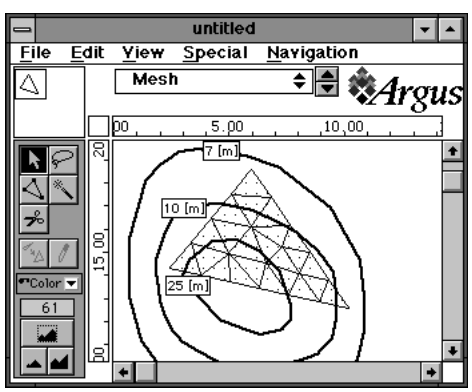
- To add titles at the top of each data line or block of lines, check the Export Titles check box.
- To export the mesh only, without parameters’ values, uncheck the Export related Data check box.
- Set the element and node lines prefix.
  If you do not need to have a prefix string at the beginning of the lines, clear these fields.
- Click the OK button to confirm your choices.
  The Save As dialog box opens.
- To export under a name different from Argus ONE's default, type it in the text edit box.
- Click the Save button.

A progress dialog box indicating the progress of the export process appears on the screen.

## An Example of an Exported Mesh

The following table contains the exported file of the project shown in the following screen-shot.

The project includes one Information layer describing depth variation. The export file was created by selecting Export Titles, Export Related Data, and setting TAB as the delimiter.



*General Information.*

| **# Elements count** | **Nodes count** | **Number of Node Parameters** | **Number of Element Parameters** |
|---|---|---|---|
| 23 | 20 | 1 | 1 |

*(Empty line)*

*Node Infromation.*

| **#** | **Node No.** | **x Pos** | **y Pos** | **Depth** |
|---|---|---|---|---|
| N | 1 | 10.2306 | 12.5272 | 7 |
| N | 2 | 8.94292 | 12.8153 | 10 |
| N | 3 | 9.39094 | 13.7267 | 10 |
| N | 4 | 2.50472 | 14.2558 | 10 |
| N | 5 | 3.38667 | 15.323 | 10 |
| N | 6 | 3.79236 | 13.9677 | 25 |
| N | 7 | 6.0325 | 18.5244 | 7 |
| N | 8 | 6.87211 | 17.325 | 7 |
| N | 9 | 5.15056 | 17.4573 | 7 |
| N | 10 | 8.10471 | 14.0059 | 10 |
| N | 11 | 7.65528 | 13.1034 | 10 |
| N | 12 | 8.55133 | 14.9261 | 10 |
| N | 13 | 6.6459 | 14.4909 | 25 |
| N | 14 | 6.36764 | 13.3915 | 25 |
| N | 15 | 7.34391 | 15.1631 | 10 |
| N | 16 | 7.71172 | 16.1256 | 7 |
| N | 17 | 5.06341 | 14.8927 | 25 |
| N | 18 | 5.08 | 13.6796 | 25 |
| N | 19 | 6.02619 | 16.0471 | 10 |
| N | 20 | 4.26861 | 16.3901 | 10 |

*(Empty line)*

*Element Information.*

| **#** | **Elem No.** | **Node-1** | **Node-2** | **Node-3** | **Depth** |
|---|---|---|---|---|---|
| E | 1 | 1 | 2 | 3 | 10 |
| E | 2 | 4 | 5 | 6 | 10 |
| E | 3 | 7 | 8 | 9 | 7 |
| E | 4 | 2 | 10 | 3 | 10 |
| E | 5 | 11 | 10 | 2 | 10 |
| E | 6 | 3 | 10 | 12 | 10 |
| E | 7 | 11 | 13 | 10 | 10 |
| E | 8 | 14 | 13 | 11 | 25 |
| E | 9 | 10 | 15 | 12 | 10 |
| E | 10 | 13 | 15 | 10 | 10 |
| E | 11 | 12 | 15 | 16 | 10 |
| E | 12 | 14 | 17 | 13 | 25 |
| E | 13 | 18 | 17 | 14 | 25 |
| E | 14 | 13 | 19 | 15 | 10 |
| E | 15 | 17 | 19 | 13 | 25 |
| E | 16 | 15 | 19 | 16 | 10 |
| E | 17 | 16 | 19 | 8 | 7 |
| E | 18 | 8 | 19 | 9 | 7 |
| E | 19 | 9 | 19 | 20 | 10 |
| E | 20 | 19 | 17 | 20 | 10 |
| E | 21 | 20 | 17 | 5 | 10 |
| E | 22 | 5 | 17 | 6 | 25 |
| E | 23 | 17 | 18 | 6 | 25 |

## Export a Selection

Exporting only parts of a mesh is a powerful tool enabling you to communicate with other programs you use.

### Export nodes only: Examples

When you change some of the nodes' values, such as boundary or initial conditions, you need not export the whole mesh in order to transfer the change into your solver (simulator). Select these nodes and check the Export Selection Only check box in the Export dialog.

If you need to transfer the coordinates of the boundary nodes into a program that draws the outline, use the Search For... command to select all boundary nodes and then Export the selection.

You could of course export any selection you choose by selecting the nodes and elements you need and exporting them.

Exported files of parts of the mesh may contain “invalid” meshes, thus Argus ONE does not import such exported files.

The first line of such a file will contain the words “# Partial mesh!”

The “Number of Elements” and “Number of Nodes” refer to the actual number of nodes and elements being exported.

# Importing a Mesh

To allow you to view, edit and manipulate meshes you've created manually or using mesh generators, Argus ONE enables you to import them.

## Import File Format

The import file format matches the export file format. However, related data fields are not supported. The data fields must be delimited by the Tab or Space characters.

### The first line of a valid import file

Number of Elements Number of Nodes

### A line describing a node

There must be exactly “Number of Nodes” lines describing the nodes. They must be ordered consecutively and in ascending order from node number 1 to “Number of Nodes”. Each node line must start with the character “N”, denoting Node.

| | | | |
|---|---|---|---|
| “N” | Number | X | Y |

### A line describing an element

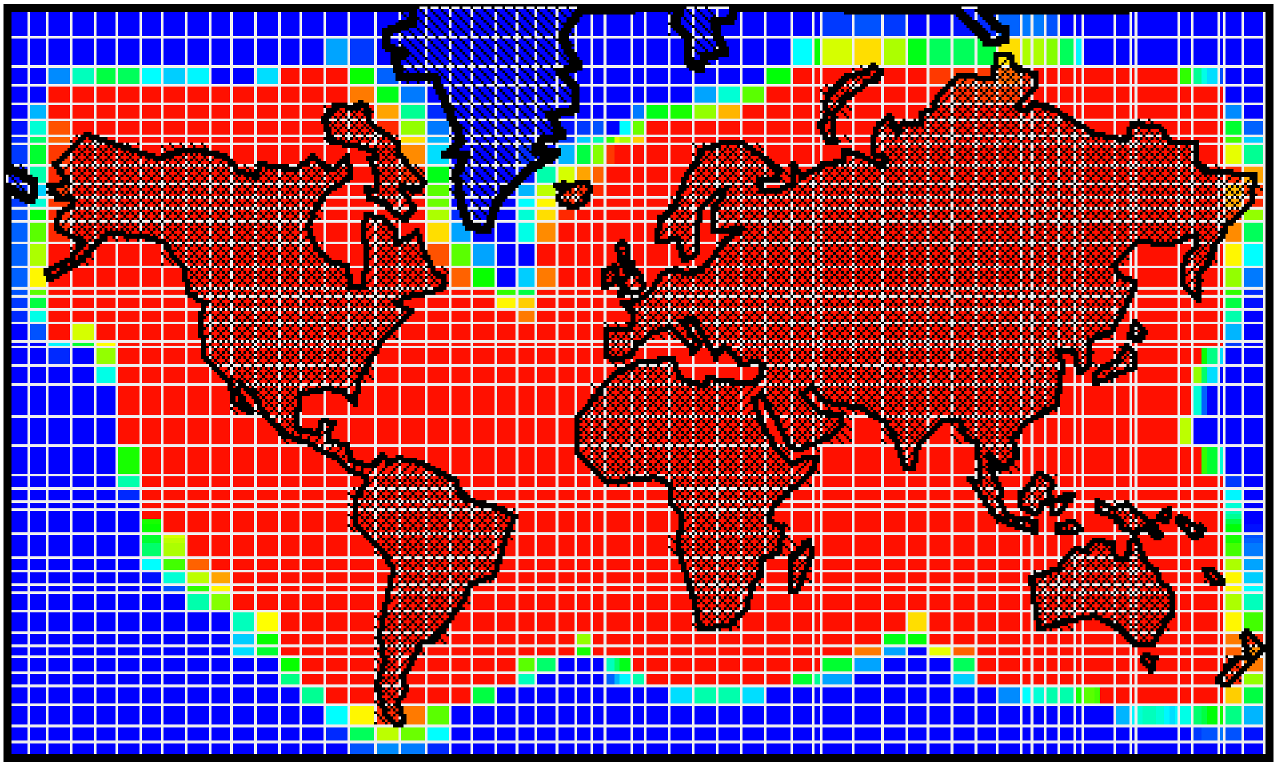
There must be exactly “Number of elements” lines describing the elements. They must be ordered consecutively and in ascending order from element number 1 to “Number of elements”. Each element line must start with the character “E”, denoting Element.

| | | | | |
|---|---|---|---|---|
| “E” | Number | Node 1 | Node 2 | Node 3 |

Any additional information stored in a line, or after the last element line is ignored.

## Data Validity

While importing, Argus ONE performs validity tests of the file format and of the mesh. If Argus ONE fails to complete the import it alerts you, states the probable cause of the problem, and if possible, suggests a solution to the problem.



# Finite Difference Grids

# Creating and Editing a Grid

# Overview

*Argus ONE's mesh and grid tools resemble each other.*

Although finite difference grids and finite element meshes serve the same purpose, namely discretisizing a domain, being derived from different theoretical approaches makes them very different. Since you might be using both of them at different projects, Argus ONE tries to keep the user interface as similar as possible between the two. This should make it easier for you to work with both methods.

If you've read part 3 of this manual, “Finite Element Meshes”, you will be able to just breeze through some of the issues explained in this part.

*Creating grids automatically is faster, easier and safer than doing so manually.*

Although you can create a grid manually, it is much easier and more benefiting to let Argus ONE automatically create it. As you might have read before, the only thing you have to do to automatically grid a domain is to define it by creating a domain outline contour, and then, click the Magic Wand in it. However, to create a grid that complies with your numerical and physical needs, you must tell Argus ONE of these needs. You do so by defining the domain, setting different grid densities to different zones in it, and finally letting Argus ONE grid it.

The three main stages in griding a domain are:

1. Defining the domain by creating a domain outline contour.
2. Setting grid densities.
3. Clicking the Magic Wand in the domain.

You define the domain in a domain outline type layer by creating an outline contour. Creating contours is explained in detail in chapter 2 “Working with Information Layers.”

In this chapter you will learn the techniques that will enable you to create a grid that will answer your numerical needs. This chapter explains the effects of a grid density layer and of the shape of the domain outline contour on the grid.

Argus ONE's grid editing capabilities enable you to control the grid with respect to each row, column and block in it, thus transforming grid creation into an art. In this chapter you will also learn about grid editing techniques.

*Auto grid generation also deactivates external blocks.*

Argus ONE’s automatic grid generation (AGG) engine performs griding in two major steps, creating the grid (rows and columns) and deactivating grid blocks lying outside the domain and in domain “islands”.

*Both grid centered and block centered grids are available.*

Two finite difference grids types are available:

- Grid centered, grid lines lie at block centers.
- Block centered, grid lines lie at blocks borders.

# Grid Centered and Block Centered Finite Difference Grids

Some finite difference based numerical simulators (numerical models) employ Grid centered grids and some employ Block Centered grids. To allow you to use the model of your choice, Argus ONE grid type layers support both. The two grid types are:

- *Grid centered*, grid lines lie at block centers.
- *Block centered*, grid lines lie at blocks borders.

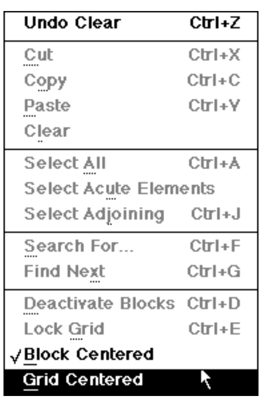
## Setting the Grid Type

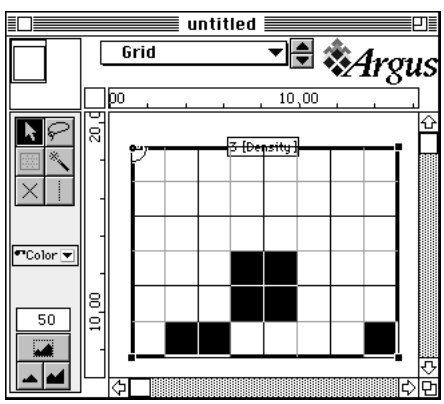
You set the grid type before you create the grid. After you've created a grid, you will have to delete it before you can create a grid of a different type. By default, a grid layer is set to the block centered grid type.

### To set the grid type to Grid Centered

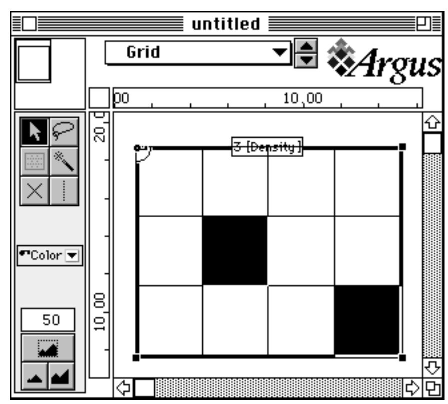
1. Make sure there is no grid in the grid layer. If there is, select it and hit the DELETE key on your keyboard to delete it.
2. From the edit menu, choose Grid Centered.

Every grid you create in this grid layer will be of the grid centered type.




In a grid centered grid, grid lines lie at blocks centers. The gray lines mark the blocks edges. Grid centered grid's blocks vary in size.




In a block centered grid, grid lines lie on block edges. Block centered grid's blocks are of equal size.

### To set the grid type to Block Centered

1. Make sure there is no grid in the grid layer, if there is, select it and hit the DELETE key on your keyboard to delete it.
2. From the edit menu, choose Block Centered.

Every grid you create in this grid layer will be of the block centered type.

# Grid Related Layers

To allow Argus ONE to automatically grid a domain, a grid layer must be associated with two other layers: a domain outline layer and a density layer.

The domain outline contours in a domain layer tell Argus ONE's AGG what is the domain it needs to grid when you click the magic wand. As you will find out soon, it also contains additional information such as the default grid density you assign to domain outline contours in the density parameter, etc.

The density layer allows you to specify different rows and columns sizes in different areas of the domain, overriding the default density defined by the domain outline contour.

## Assigning the Grid Layer Domain and Density Layers

When you create a new Argus ONE project, it is created with one grid layer, a domain layer named Domain Outline, and an information layer called Density. This predefined grid layer is automatically associated with the domain and density layers. You can change the domain and density layers a grid layer is associated with, at any time.

As you might have read before, any information type layer or a domain type layer may serve as the density layer you assign to a grid layer. However, only a domain type layer can serve as a grid layer's domain layer.

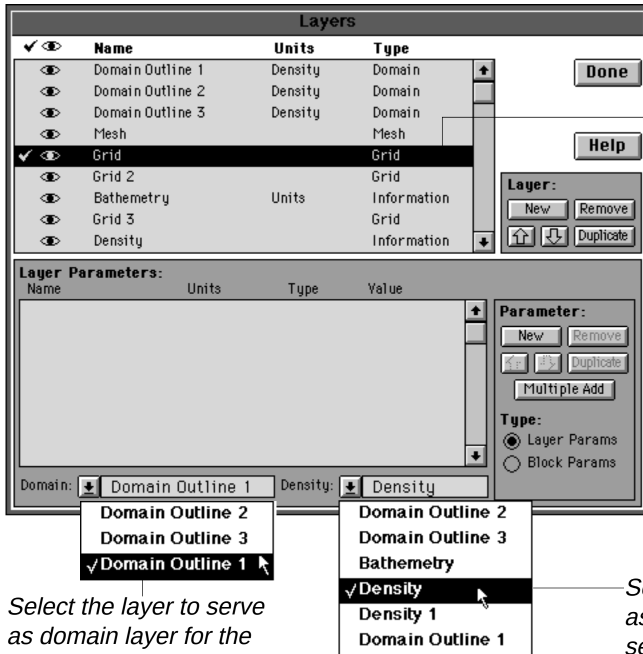
You can replace both the domain and density layers associated with a grid layer at any time. To create different grids, you can either replace the domain and density layers it is associated with, or create different grid, domain and density layer sets for the different realization you need to create grids for. This is very useful if you need to create a three dimensional grid.

### To set the domain and density layers associated with a grid layer

You assign a grid layer its domain and density layers in the layers dialog.

1. From the layers popup menu in the information ruler, choose Layers...
2. In the layers list, select the line of the grid layer its domain and density layers you want to change.
   When you select a grid layer, two popup menus appear at the bottom of the dialog box.
3. From the left one, named Domain, select the domain layer. Only domain layers are listed in this popup menu.
4. From the other, named Density, select the layer to serve as density layer. Listed in this popup are only information type and domain type layers.

*Set the domain and density layers for the selected grid layer.*




*Select the grid layer its domain and density layers you want to set.*

*Select the layer to serve as domain layer for the selected grid layer.*

*Select the layer to serve as density layer for the selected grid layer.*

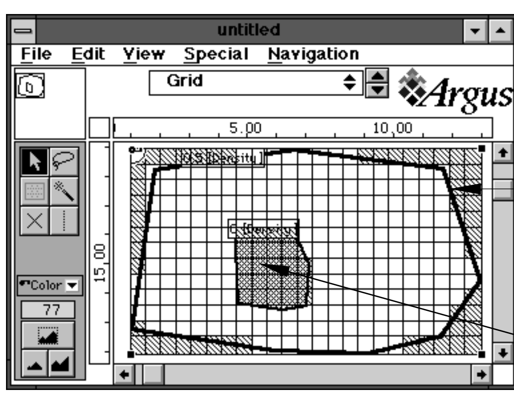
# Using Contour Objects in Domain Layers

As in information type layers, you can use close and open contours as well as point objects. All contour objects but point objects, can be assigned densities.

## The Domain Outline Contour

A close contour in a domain type layer is referred to as a domain outline contour. It tells Argus ONE what is the domain it needs to grid, that is, the enclosing rectangle of the domain outline contour, and the default grid density. A domain outline contour can contain other close contours defining islands-like internal boundaries.

*Grid blocks which their centers lie outside the domain outline contour or inside "islands", are hatched to indicate they are inactive.*




*A simple domain outline contour assigned default grid density of 0.5.*

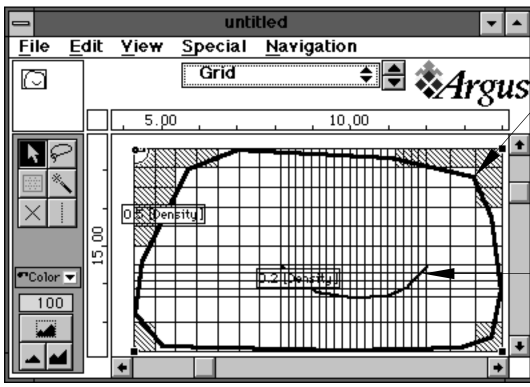
*An island.*

## The Open Contour

An open contour is used to mark internal boundaries such as rivers and faults lying along lines. You can use the block functions CountObjectsInBlock(), SumObjectsInBlock() and WSumObjectsInBlock() to assign the blocks values from all contours, but this capability is especially important when combined with open contours.

Assigning an open contour a grid density value forces the grid engine to reduce columns' and rows' sizes to the required density in enclosing rectangle of the open contour.

*A domain outline contour with an open contour.*




*A simple domain outline contour with an open contour.*
*The domain outline contour was assigned a grid density of 0.5 units, producing a regular grid of 0.5 size blocks.*

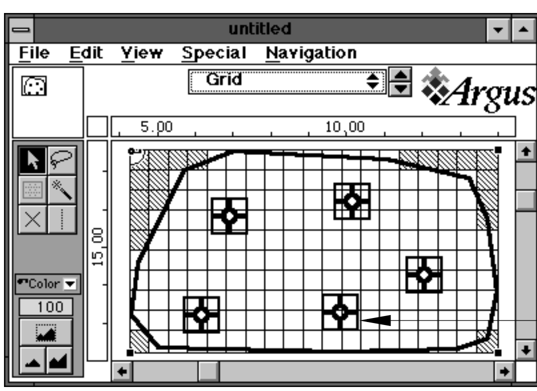
*The open contour was assigned grid density of 0.2, thus producing a denser grid around its enclosing rectangle.*

## The Point Object (Contour)

A point object is used to mark internal point boundaries representing sources, sinks and point loads such as wells. You can use the block functions CountObjectsInBlock(), SumObjectsInBlock() and WSumObjectsInBlock() to assign the blocks values from all contours, but this capability is especially important when combined with point contours.

Although you can assign grid density to a point object, the AGG ignores it. To specify a grid density around point objects use the grid density layer. Using the grid density layer is discussed in detail in the following sections.

*A domain outline contour with a Point object.*




*To instruct Argus ONE to calculate for each block the value of all point objects within it, create a block parameter and assign it the expression (SumObjectsInBlock(parameter_name,0)). It will sum up all point objects values in parameter_name layer.*

*A simple domain outline contour with point objects.*

# Grid Densities

Grid density is defined in Argus ONE workplace as the length and width of a grid block. The larger the length (width) you set, the larger the resulting blocks, and the lower the density of the grid lines.

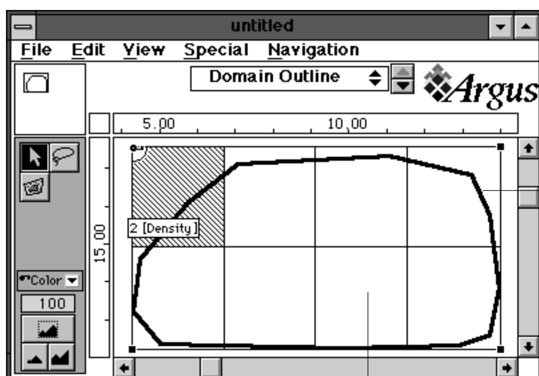
You define the density by two means: the grid density you assign to the domain outline contour, referred to as the default density, the density you assign to "islands" and open contours, and the density layer in which you assign different zones different grid densities.

## Default Grid Density

At the end of a domain outline contour creation, Argus ONE prompts you for the grid density of that contour. The number you specify tells Argus ONE what is the block size you would like to have in the domain. You specify the grid density in the current units as they appear on the rulers.

In the following example a simple close contour was created in the domain layer and assigned the default density of 2 centimeters.

*Automatically generated grid using the default density assigned to the domain outline contour.*




*A simple domain contour assigned a grid density of 2 cm.*

*The auto grid generator finds the domain's outline contour enclosing rectangle and creates a grid of the enclosing rectangle's size.*

If the grid density you specify is much larger than the size of the domain, Argus ONE creates a grid of one block.

If the enclosing rectangle's width or height can not be divided into a whole number of grid lines using the specified grid density, the AGG calculates the whole number of grid lines that can be inserted, and spaces them evenly over the width or height.

# Creating Grid Density Contours

When griding a domain, you usually want to have more grid lines (smaller blocks) at some parts of the domain and less grid lines at others. You may want to have a fine grid at areas were gradients are expected to be higher. These areas may be along boundaries or at any other part of the domain. If for example, you have a well pumping at the middle of the domain, or a source of contamination at the middle of a lake, you want to have a fine grid around these points.

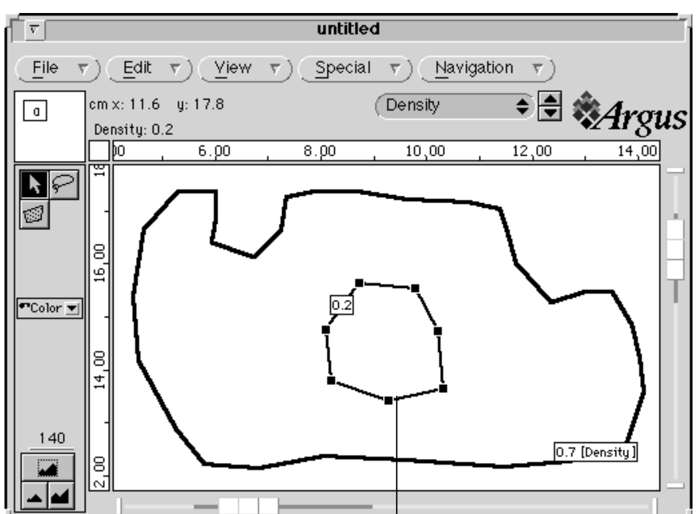
You define different grid densities at different areas by using grid density contours in the grid density layer. An information type layer assigned as the grid density layer is where you define the required grid densities.

## To Define Grid Density Contours

After creating a domain outline contour and assigning it a default grid density:

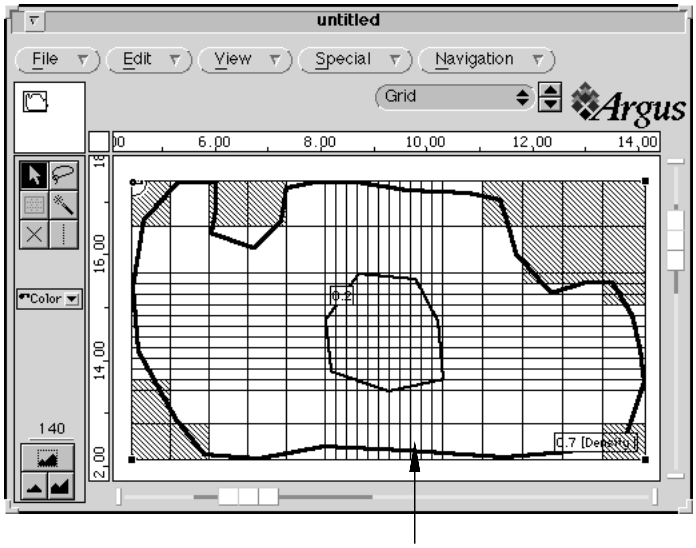
- Change the active layer to the layer associated as the grid density layer.
- Using the close contour tool create a contour around the area its density you want to change.
- When you end creating the contour, Argus ONE asks you for the required density, enter it in the text edit box. If you are working in units of cm and want to have blocks of 0.5 cm side, type 0.5.
- Click the OK button.

*A simple domain outline contour assigned default grid density of 0.7 cm.*




*A grid density contour telling Argus ONE that blocks within this contour should have sides of 0.2 cm.*

- To grid the domain, move to the grid layer, and click the Magic Wand anywhere within the domain outline contour.



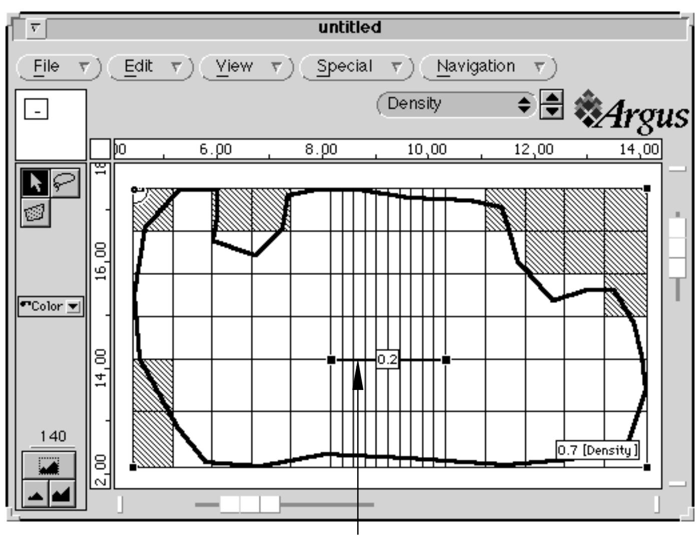

*Blocks within the grid density contour of 0.2 cm are forced to that size.*

## Controlling the Horizontal and Vertical Grid Densities

### To reduce the block size horizontally only

- Create a horizontal open density contour, and assign it the required density. Click the Magic Wand to re-grid the domain.

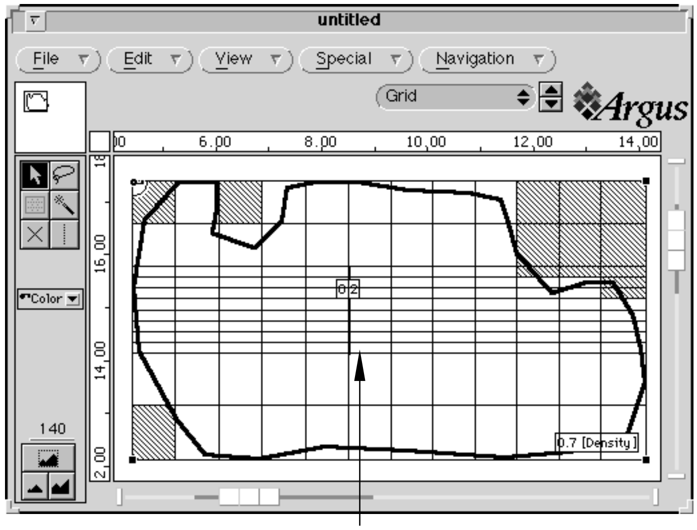
*To make sure the density open contour is horizontal or vertical hold the SHIFT key while creating it.*




*Since this density contour's enclosing rectangle is degraded (it is flat), block size is reduced only in the horizontal.*

### To reduce the block size vertically only

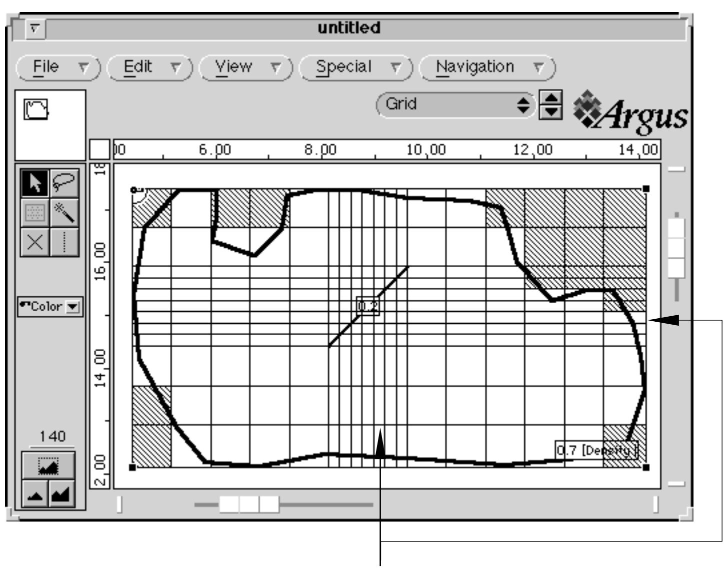
- Create a vertical open density contour, and assign it the required density. Click the Magic Wand to re-grid the domain.




*Since this density contour's enclosing rectangle is degraded (it is flat), block size is reduced only in the vertical.*

### To reduce the block size evenly over the horizontal and vertical

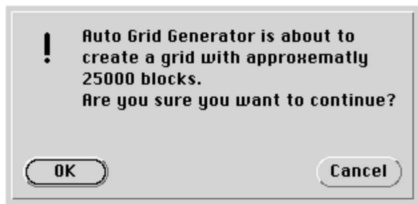
- Create an open contour at 45°, and assign it the required density. Click the Magic Wand to re-grid the domain.




*Since this density contour's enclosing rectangle is a square, block size is reduced evenly horizontally and vertically.*

## Controlling Automatic Grid Generation

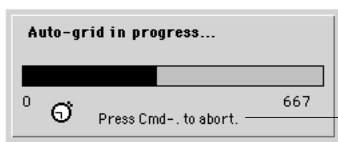
When you click the Magic Wand above a domain outline contour, Argus ONE analyzes the shape and densities of the domain outline contours and the densities in the density layer and estimates the number of grid blocks to be created. If your domain and grid densities definitions are about to create above 3000 grid blocks Argus ONE warns you by bringing up a dialog informing you of the estimated number of blocks it is about to create.




Click the OK button to continue, or the Cancel button to end auto grid generation.

When auto grid generation is in progress a progress dialog is presented.

*Progress Dialog box*



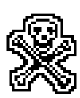

*Press the keyboard sequence to halt automatic grid generation. The keyboard combination on Windows and Unix platforms is* CTRL*+C.*

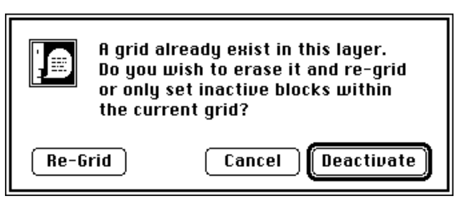
*Halting AGG will create the grid but halt automatic block deactivation.*

## Re-Griding an Already Grided Domain

When you click the Magic Wand in a grid layer, Argus ONE tests the domain to be grided for an existing grid. If it does find such a grid it alerts you and allows you to cancel.

*All block manual overridden values are deleted as well.*






- Click the Re-Grid button to allow Argus ONE to delete the existing grid and re-grid the domain.
- Click the Cancel button to remain with the current grid.
- Click the Deactivate button to automatically deactivate manually created blocks.

# Setting Your Views

When you edit your grid you can take advantage of Argus ONE's information hiding capabilities to concentrate on the objects you need. In the following paragraphs we will outline some of these capabilities. For a detailed explanation of showing and hiding layers refer to part 1 chapter 1 in this manual.

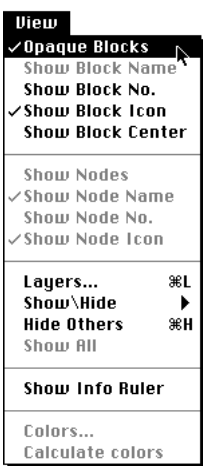
All the commands you will use to set your views are presented in the View menu.

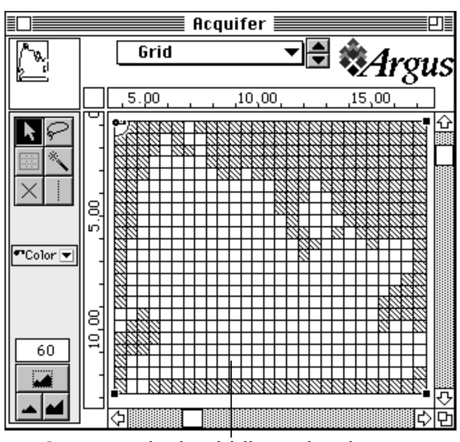
### Seeing Through the Grid

To view the grid and the other layers you have defined at the same time you can set the grid transparent. If you need to hide all other information you can set the grid to opaque. Argus ONE's default is set to Opaque Blocks.

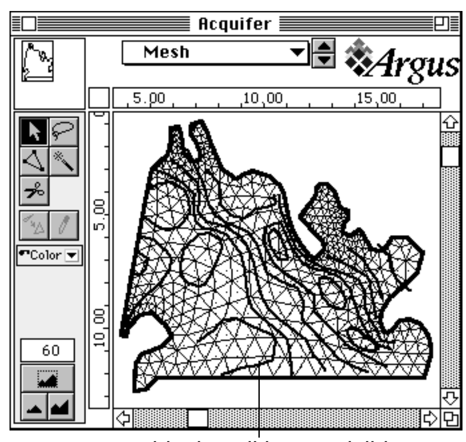
### To toggle between opaque and transparent blocks

- From the View menu, choose Opaque Blocks.
  Blocks turn transparent, and the check mark to the left of the menu item is removed.



Opaque Blocks, hiding other layers.



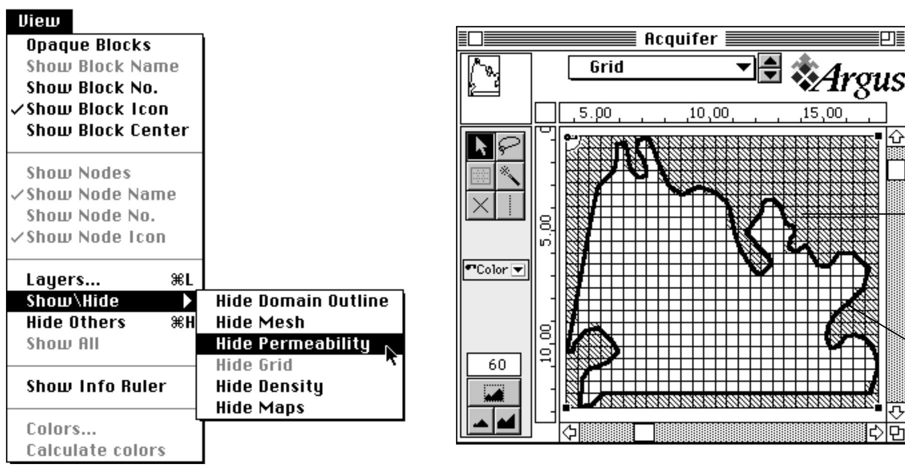
Transparent blocks, all layers visible

## Showing and Hiding Layers

Showing and hiding layers enables you to view the grid with respect to only some of the layers. Hiding and showing of layers and moving between layers is explained in detail in the chapter “The Argus ONE Workplace.”

### To hide or show a layer

- From the View menu, choose Hide or Show “Layer name.”




*The contours visible on the previous snapshot are hidden.*

*The domain outline contour is visible since the domain layer is visible.*

## Zooming

*Argus ONE's default zoom level is set to 100 percent.*

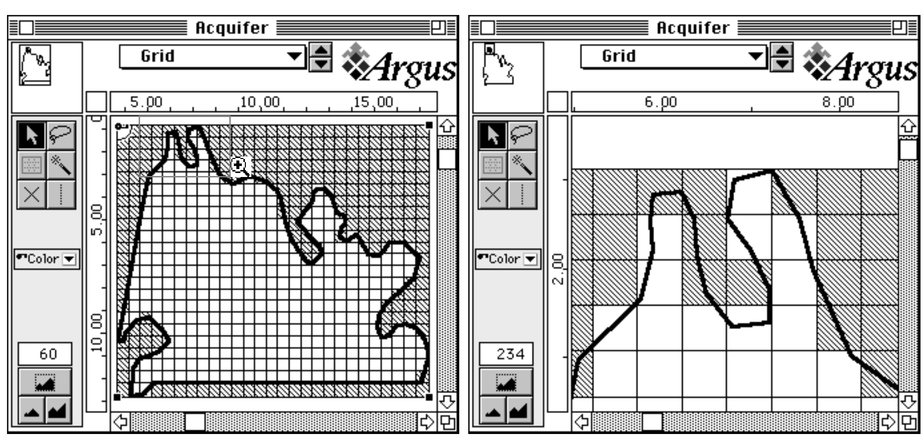
When you want to edit a part of the grid that is too small to edit in the current zoom level, Zoom In, or, to select large areas of the grid, Zoom Out to refrain from tedious autoscroll. A detailed discussion about zooming is presented in the chapter “The Argus ONE Workplace.”

### To zoom in

1. From the tool palette, choose the Magnifying Glass.
2. Stretch the tool around the area of interest.

*When zooming using the zooming tools, Argus ONE centers the zoomed view.*

*Saving a project, saves the zoom level too.*



## Showing and Hiding Block Information

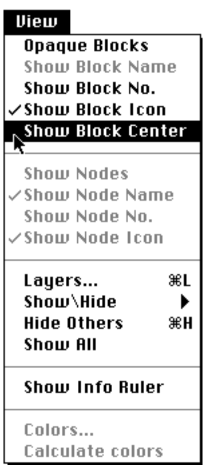
Some of the information you can assign to blocks as well as information automatically assigned to them by Argus ONE, can be graphically presented on the screen. You can show and hide this information so that your workplace does not become cluttered.

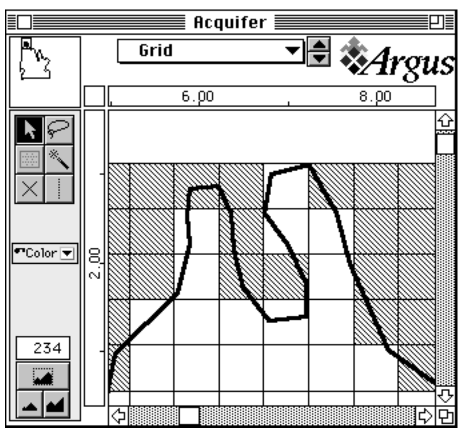
### To show and hide block centers

You can instruct Argus ONE to draw a small circle at blocks' centers. However, when you zoom out, or the grid is very dense, the grid may become cluttered with the blocks' centers circles. Also, blocks' centers circles degrade performance on some platforms, mainly on MS Windows. To make your screen clearer and improve the application response, turn this option off. Argus ONE default setting is off.

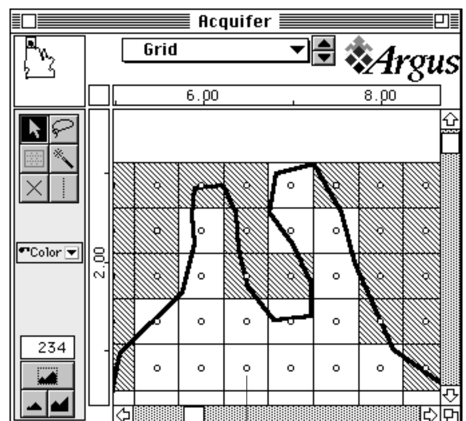
- To enable blocks' centers, from the View menu, choose Show Block Center.

A check mark to the left of the menu item is added.




*Blocks' centers circles are hidden.*




*Blocks' centers circles are shown.*

### To show and hide block numbers

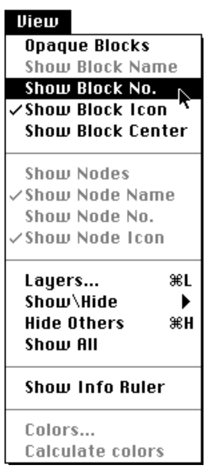
*Note: Blocks' numbers are dynamic. Do not count on them, unless you are through with deleting and adding grid lines.*

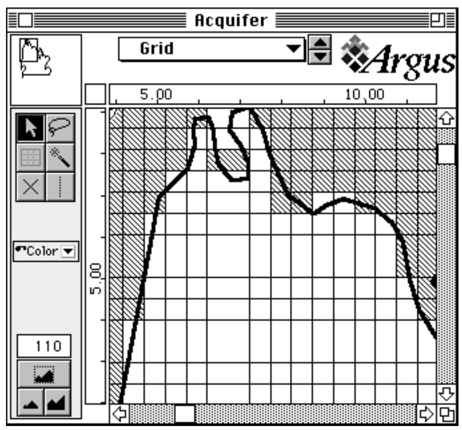
Argus ONE can mark each block with the number of the two and column it belongs to. This number is referred to as the block number. Blocks are numbered by Argus ONE automatically as it creates them. When you delete rows and column, Argus ONE changes the numbering of the blocks. Although the blocks' numbers are dynamic, you may still want to relate to them at times. Argus ONE is set by default to hide block numbers.

Argus ONE presents the block number at the top left corner of the block. If the block is too small to contain the number, Argus ONE hides the number. When you zoom in, Argus ONE tests each row and column again to decide whether the number can be contained within the block and if possible, presents it.

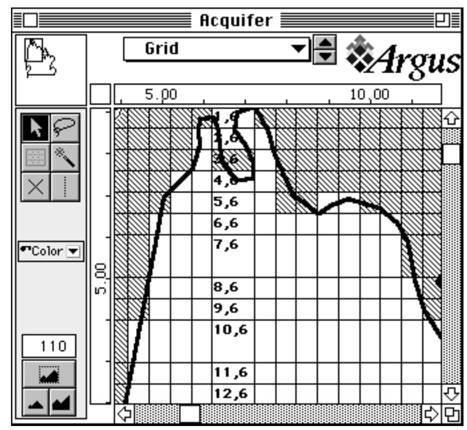
- To show block numbers, from the View menu, choose Show Block No.

The menu item is marked with a check mark to its left.




*Blocks' numbers are hidden.*




*Block numbers are shown in blocks large enough to contain their number. To see the other numbers, zoom in.*

- To hide block numbers, from the View menu, choose Show Block No. The check mark to the left of the menu item is removed.

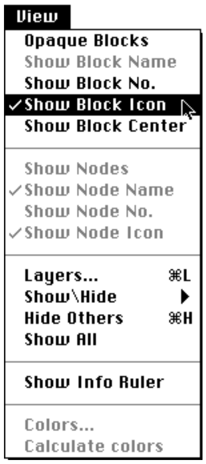
## To show and hide the block icon

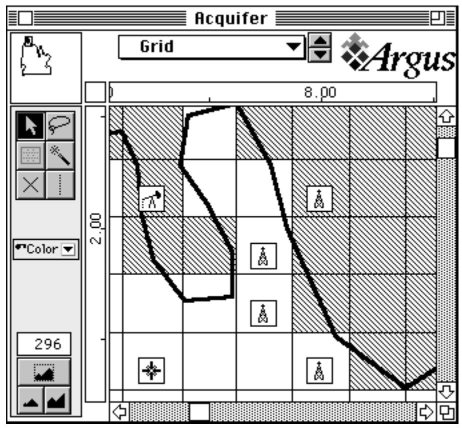
If you assign an icon to a block, Argus ONE enables you to decide whether it shows on the screen. Argus ONE presents the icon at the center of the block. If the block is too small to contain the icon, Argus ONE does not show it. When you zoom in, Argus ONE tests each block again to decide whether the icon can be contained within the block and if possible, presents it.

- To hide block icons, from the View menu, choose Show Block Icon.

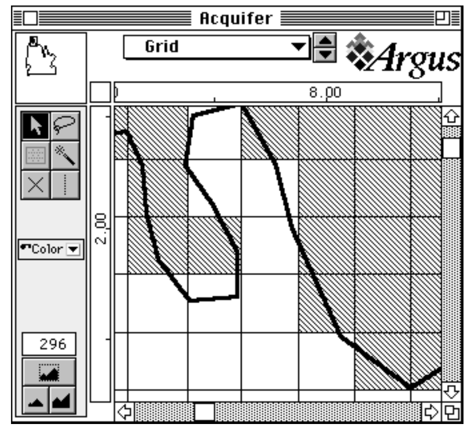
The check mark to the left of the menu item is removed.

Argus ONE is set by default to hide block icons.




*Blocks' icons are shown in blocks large enough to contain their icon. To see the other icons, zoom in.*




*Blocks' icons are hidden.*

- To show block icons, from the View menu, choose Show Block Icon. The menu item is marked with a check mark to its left.

To read about assigning blocks icons refer to the following section in this chapter.

# Moving, Resizing and Deleting a Grid

You can move, resize and delete a grid. All commands you issue within Argus ONE are undoable. To undo a command, select the Undo command form the Edit menu.

When you delete a grid all manually overridden data is lost. This is why you should try to keep manually overriding blocks' data to the minimum.

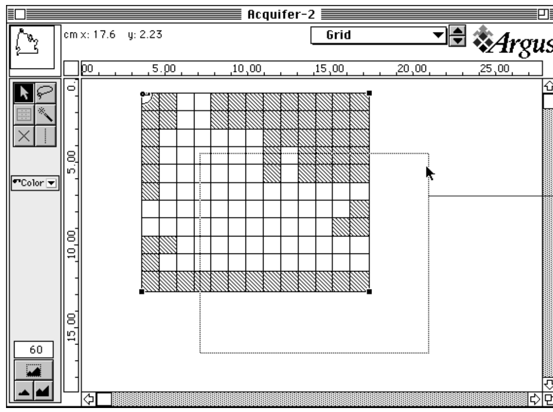
### Moving the Grid

When you move a grid you move it with respect to the current coordinate system. All blocks within the grid are automatically assigned their new X and Y coordinates. All data linked to the grid from information layers is automatically updated.

### To move the grid

1. Click-drag the grid on one of it sides.
   An outline of the grid follows your mouse movements as you move the grid.
2. Release the mouse button at the new required location.

*A grid on the move.*



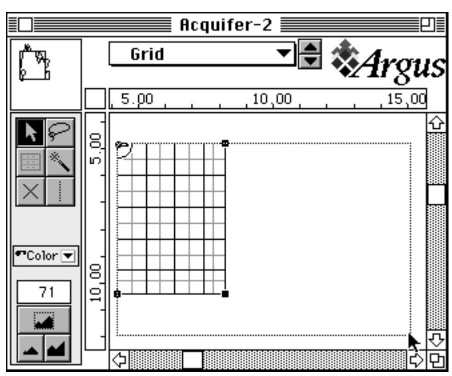

*An outline of the grid moves along with your cursor to indicate the new grid's location.*

### Resizing the Grid

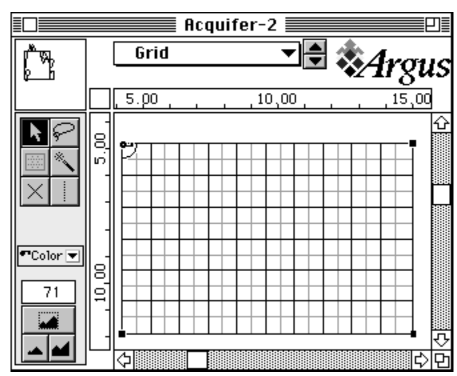
When you resize the grid, Argus ONE adds or deletes grid lines at the newly created or reduced grid area.

### To resize the grid

1. Click-drag the grid from one of its four corners.
   An outline of the resized grid follows your cursor
2. Release the mouse button to resize the grid.




*To enlarge the grid stretch it from one of its corners...*




*Grid lines are added in the newly created grid area.*

### Deleting the Grid

When you delete a grid, you delete only the grid and its blocks. All the grid layer and block parameters you have created for that grid layer are not deleted. Once you create a new grid, it assumes all the data structures you have created using the layer and block parameters.

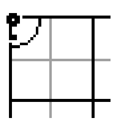
#### To delete the grid

- Hit the DELETE key on your keyboard.
  The grid is deleted.

## Locking and Unlocking the Grid

To secure your work you can lock the grid. A locked grid does not allow you to move, resize or delete it. Locking the grid also does not enable you to move grid lines.

### Locked Grid



*The handle is down indicating the grid is locked.*

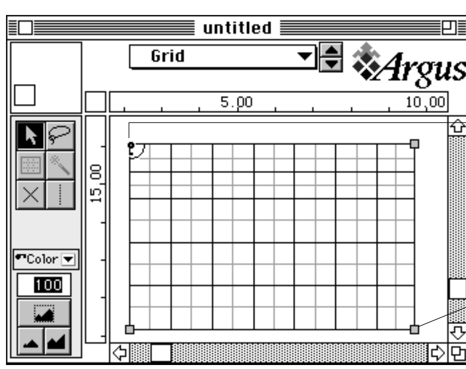
When the grid is locked the small handle at the top left corner of the grid is in a vertical position and the three other corners are grayed.

When you or Argus ONE create a grid, the grid is created in unlocked mode.

#### To lock the grid

1. Bring the cursor over the handle at the top left corner of the grid.
   The cursor changes into an open hand cursor pointing up.
2. Click the hand cursor, it changes into a closed hand to indicate the grid is locking itself. When locking is accomplished, the cursor changes into an open hand cursor , indicating that you can now use it to lift the handle to unlock the grid.
   To indicate the grid is locked, the other three corners of the grid are grayed.

*A locked grid.*




*The grid lock handle is in a vertical position, indicating the grid is locked.*

*Also note the three grid corners are grayed.*

When the grid is locked you can not perform the following actions:

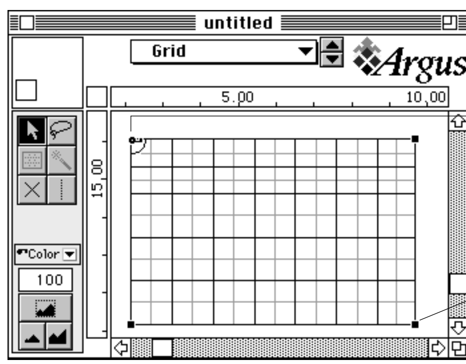
- Move the grid.
- Resize the grid.
- Delete the grid.
- Use the Magic Wand to re-grid.

### To unlock the grid

**1.** Bring the cursor over the handle at the top left corner of the grid.
The cursor changes into an open hand cursor pointing down.

**2.** Click the hand cursor, it changes into a closed hand to indicate the grid is unlocking itself. When unlocking is accomplished, the cursor changes into an open hand cursor , indicating that you can now use it to lower the handle to lock the grid.
To indicate the grid is unlocked, the other three corners of the grid are highlighted.

*An unlocked grid.*




*The grid lock handle is in a horizontal position, indicating the grid is unlocked.*

*Also note the three grid corners are highlighted.*

# Editing the Grid

You can change the grid density, that is, the blocks' height and width or the number of rows and columns, at any area you choose. There are three main capabilities to manually edit the grid: moving grid lines, adding grid lines and deleting grid lines. These capabilities are intended for final refinement of the grid. However, remember that it is much more sensible to refine the grid by using domain outline and density contours as they will remain even if you re-grid.

## Adding Rows and Columns

When you first create the grid, Argus ONE divides it into rows and columns of one (1) unit. If one of the sides of the grid is too small, it divides it and creates rows or columns of 1/10, 1/100... of that side.

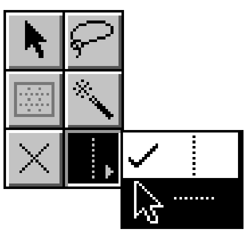
You can add grid lines one at a time or many at once.

### To add one Vertical grid line

1. From the tool palette, select the *Add Column Tool*, the tool highlights itself to designate that it is the active tool and the cursor changes its shape to a vertical line |.
2. Click the tool in the grid between any two grid lines.

A new vertical grid line is added and a new column is created.

### To add one Horizontal grid line

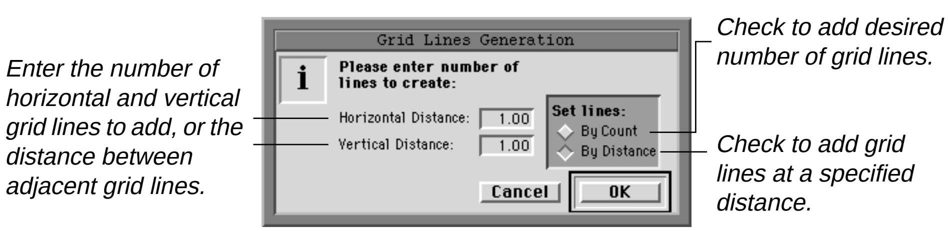


1. Click and hold the *Add Column Tool* to popup the menu and select the *Add Row Tool*, the cursor changes its shape to a horizontal line —.
2. Click the tool in the grid between any two grid lines.

A new horizontal grid line is added and a new row is created.

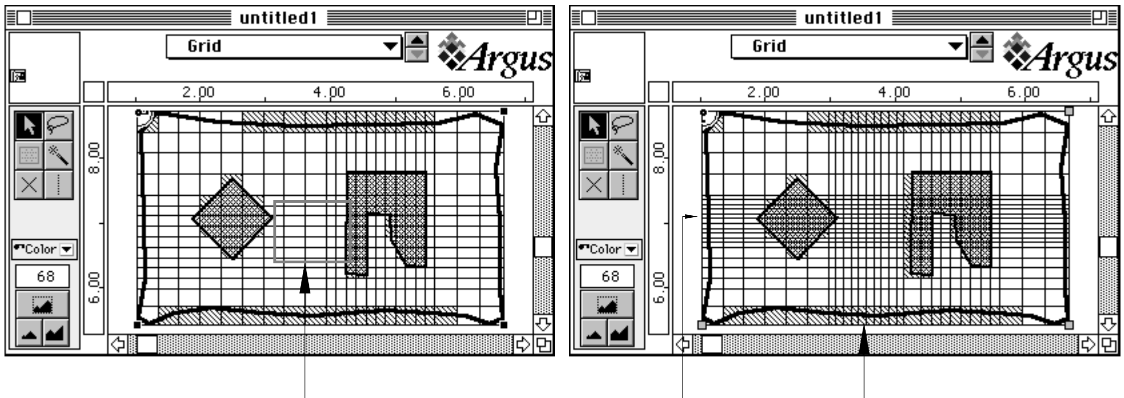
### To add a number of grid lines

1. Click the *Add Row* or the *Add Column Tool*.
2. Press the mouse button and move the cursor to create a stretch band spanning over the area you wish to add grid lines to. When you release the mouse button, the *Grid Lines Generation* dialog box appears.




3. To add a specified number of grid lines, check the *By Count* radio button. Or - to add grid lines at a specified distance from each other, check the *By Distance* radio button.
4. Specify the number of grid lines you wish to add in both directions. Or - Specify the distance between adjacent grid lines.
5. Click the OK button. Grid lines are added as specified.

Specifying zero number of lines in the Horizontal or Vertical text edit boxes, will add grid lines only in the specified direction.




*Specifying add "by count" of ten by ten grid lines in the selected rectangle...*

*...adds them to the grid.*

## Deleting Rows and Columns

You can delete grid lines one at a time or many at once.

### To delete one grid line

1. From the tool palette, select the *Delete Grid Line Tool*, the tool highlights itself to designate that it is the active tool and the cursor changes its shape to a cross ×.
2. Click the tool in the grid on a grid line you wish to delete.

The grid line is deleted. If you click in the intersection of a vertical and a horizontal grid lines, both grid lines are deleted.

### To delete a group of grid lines

1. Click the *Delete Grid Line Tool*.
2. Click-drag the mouse and move the cursor to create a stretch band spanning over the area from which you wish to delete the grid lines. When you release the mouse button, all grid lines in the marked area, horizontal as well as vertical, are deleted.

## Moving a Grid Line

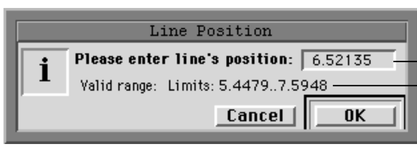
You can move a grid line by hand or re-position it at an exact location.

### To move a grid line

1. From the tool palette, select the *Arrow Tool* . When the cursor lies over a horizontal grid line its shape changes into and when it lies over a vertical grid line its shape changes into .
2. Click-drag the mouse button and move the line to its new desired location. You can not move a grid line out of the grid or above another grid line. Grid lines can be moved only between their adjacent grid lines.

### To position a grid line at an exact location

1. From the tool palette, select the *Arrow Tool* .
2. Double-Click the grid line, the *Line Position* dialog box appears.




*Type in the new grid line position.*

*The positions of the two adjacent grid lines.*

3. To re-position the grid line, type in the desired new location.

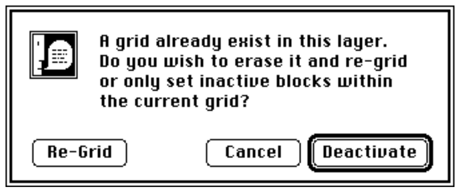
Argus ONE informs you of the valid range, that is, the locations of the two adjacent grid lines. If you try to type in a position out of the valid range, Argus ONE beeps and brings up the following alert.


## Deactivating Blocks

When you use the Magic wand to grid a domain, the automatic grid generation also automatically marks blocks that their centers lie out of the domain outline contour, or within islands, as inactive blocks and hatches them to indicate they are deactivated. You can also deactivate blocks manually. If you create a grid manually or manually refine it, you can use the Magic Wand to automatically deactivate blocks without re-griding the domain. When you export your grid, Argus ONE exports a matrix of the blocks in the grid in which deactivated blocks are assigned the integer zero (0) and others the integer one (1). The function used to obtain these values is the BlockIsActive() function.

### To automatically deactivate blocks in a manually refined grid

1. Make sure the active layer is the Grid Layer.
2. From the tool palette, choose the Magic Wand tool .
3. Click the Magic Wand anywhere within the grid.
   The following dialog box appears:




*Choosing the Re-Grid button erases the existing grid and automatically grids the domain.*

4. To only deactivate blocks, click the Deactivate button.

### To manually deactivate a selected group of blocks in the grid

1. Select the group of blocks.
2. Type the DELETE key on your keyboard (Macintosh), or BACKSPACE on all other platforms. This deactivates all blocks in the selection, active, as well as inactive.

### To manually reactivate a selected group of blocks in the grid

1. Select the group of blocks.
2. Hold the OPTION key and type the DELETE key on your keyboard (Macintosh) or ALT+BACKSPACE on all other platforms. This activates all selected blocks.

### To manually toggle active and inactive blocks

The Deactivate Blocks menu command is a toggle command.

1. Select the group of blocks.
2. From the Edit menu, choose Deactivate Blocks,
   Or
   Hold the SHIFT key and type the DELETE key on your keyboard (Macintosh) or SHIFT+BACKSPACE on all other platforms. This toggles all selected blocks.

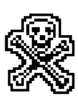
## Assigning Block Information

Like a node or an element, a block is actually a detailed data structure. The block's data structure includes its center's location, area, top, left, bottom and right position, icon, and data from the other layers. You can also access all block data that can be obtained from the block functions (chapter 3).
Argus ONE allows you to access all data types. The following table summarizes block data.

| Block Data | Description | Access |
|---|---|---|
| Block's center Position | X and Y coordinates of the block's center of gravity. | Move or change grid lines. |
| Block is Active | | Use activate/deactivate to toggle. |
| Block's Area | | Move or change grid lines. |
| Block Icon | One of ten pre-configured icons. | Block Info dialog and Block Icon menu item. |
| Data from linked layers | Any real number, which results from a link or expression. | Through grid layer parameters, referenced parameters, and manual override in the Block Info dialog. |

### To assign and edit block data

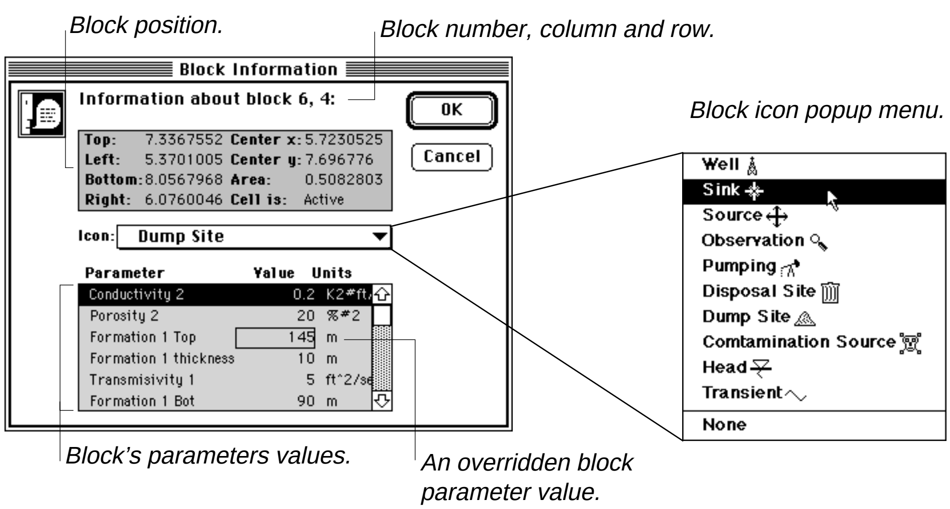
You can edit and assign block data at any time by opening the Block Information dialog.



*Use manual overrides carefully!*

When you manually set the value of a block parameter, the parameter is no longer evaluated for that block, instead the overridden value is used. If you create a new grid or delete the block, the overridden value is lost.

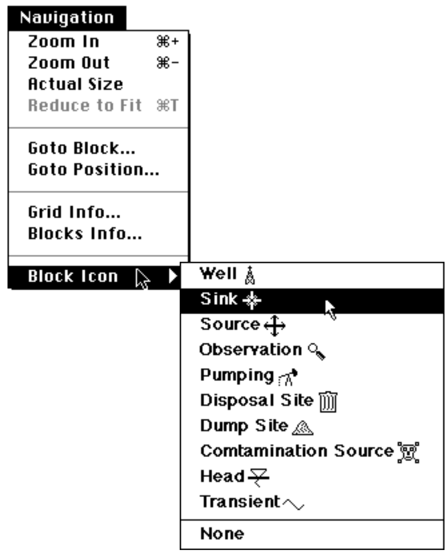
1. Select the block its data you want to edit.
2. From the Navigation menu, choose Blocks Info..., or Double-Click one of the selected Blocks.
   The Block Information dialog box is presented to allow you to edit the block's data.



### To quickly set the block icon

You can assign a block or a group of blocks an icon without opening the Block Information dialog.

1. Select the blocks to be assigned an icon.
2. From the Information menu, choose Block Icon, a popup dialog opens.




*Select one of the icons.*

*Select none to remove the block icon.*

3. Select the icon you wish and release the mouse button.
   The selected blocks are assigned the icon.

- To remove an icon assigned to a block, select the None menu item.

## Searching the Grid Database

As you add information to your project, using the Information layers and block information fields, you're actually building a database. Argus ONE enables you to perform extensive and complex searches of the blocks, based on their data.

The Search For... command is the graphical interface to the query language you use to build your queries. It allows you to perform searches based on all the types of data assigned to a block.

- Block row and column.
- Block icon.
- Block data originating from linked parameters.

It also enables you to differentiate boundary and domain blocks.

You can use these searching capabilities to orientate within the project or to create selection groups for exporting special parts of the grid.

While iterating through the processes of model calibration and solving for different realizations, you can use the search facilities to relate numerical information (block numbers) from the solution program with the blocks' location in the grid. For instance, if you get a singular solution at some blocks, and you want to query the boundary or initial conditions at these blocks, you can search for these blocks using the Search For... command. You can then use the Block Information dialog to change these blocks' values one by one or all at once.

The Search For dialog enables you to search for blocks, define a search criteria and to decide how to treat the current selection.

After you have selected a group of blocks based on some search criteria, you can extend or reduce the found group by adding or removing from it another group of blocks satisfying a different search criteria.

You can also negate each search criteria.

## Using Block Parameters to Define Search Criteria

Since you can assign blocks as many parameters as needed, you can use these parameters to create intricate search criteria. To do so, create grid block parameters containing the expressions needed or create other layers parameters and link them to grid or block parameters. Then, in the search dialog box, choose these parameters to define the search criteria.

## To Search a Grid

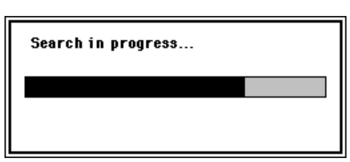
1. Make sure the active layer is a grid layer.
2. From the Edit menu, choose the Search For... Command.

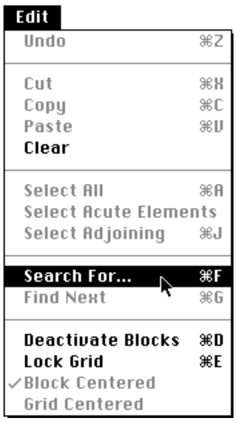
The following dialog appears.

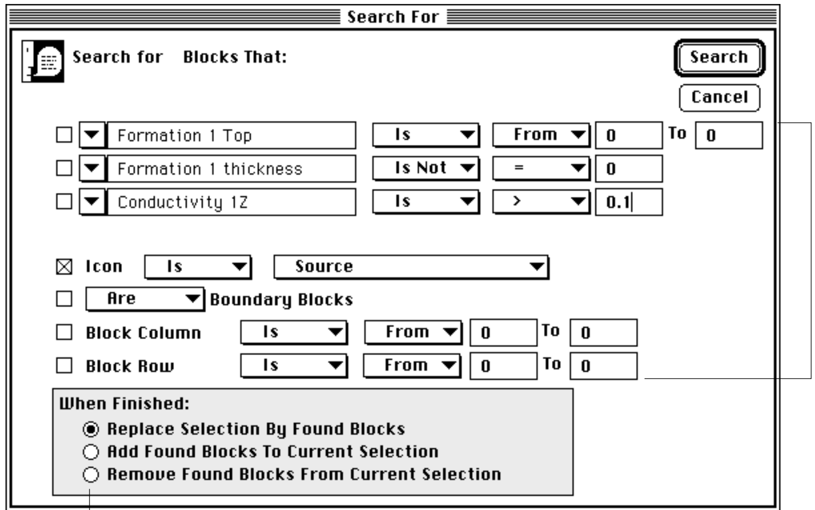
3. Set all the fields you need in order to define the search criteria.
4. Press the Search button.

While searching, a progress dialog appears to indicate the search progress.

*Search progress bar.*






*Set the search range.*

*Using several search processes together with different settings of the current found set enables you to expand or reduce the search scope.*

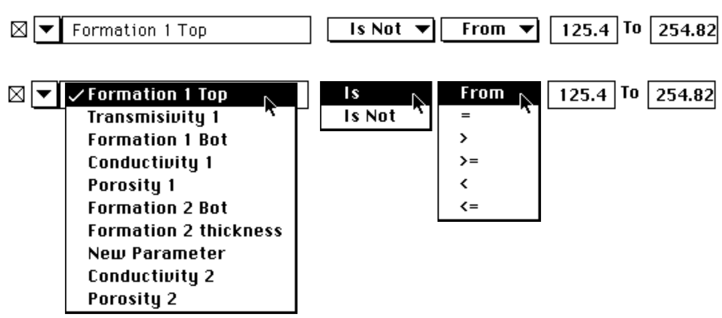
## Defining a search criteria

In the following paragraphs each of the fields and options in the search dialog are explained.

### Choosing the blocks parameters to be searched

You can create a search criteria containing up to three linked parameters. If you need to define a search criteria based on more linked parameters, execute one search for the first three and then define and execute more searches using the Add Found Blocks To Current Selection option.

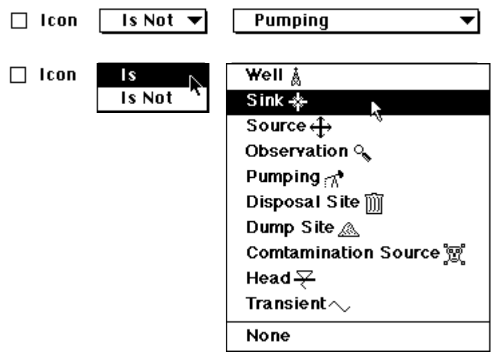
- From each of the three Parameters popup menus, choose the parameter to be searched for and assign its range.



### To search for blocks' icons

Assigning icons to very special blocks can help you to orientate within the grid.

- From the Icon popup menu choose the icon and specify the search criteria.

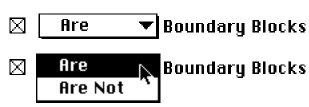



### To search for boundary blocks

Boundary blocks are defined as blocks in contact with close domain contours including "islands".

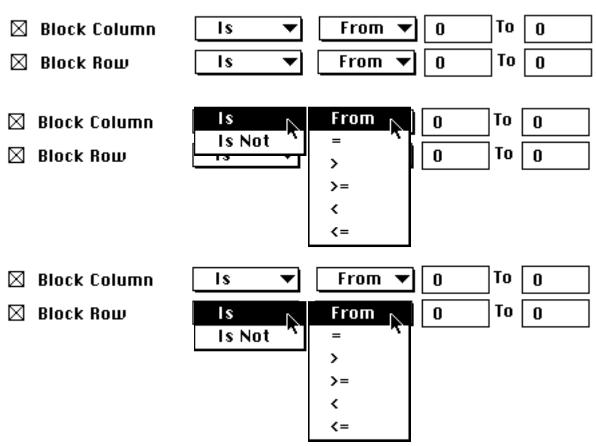
Negating the search allows you to find internal blocks.

- Choose Are or Are Not.

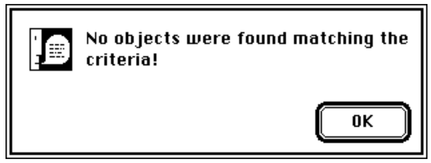



### To search for block numbers

- Set the block row and column range.



The blocks matching the search criteria are selected. If no blocks matching the search criteria are found, Argus ONE reports it with the following alert:



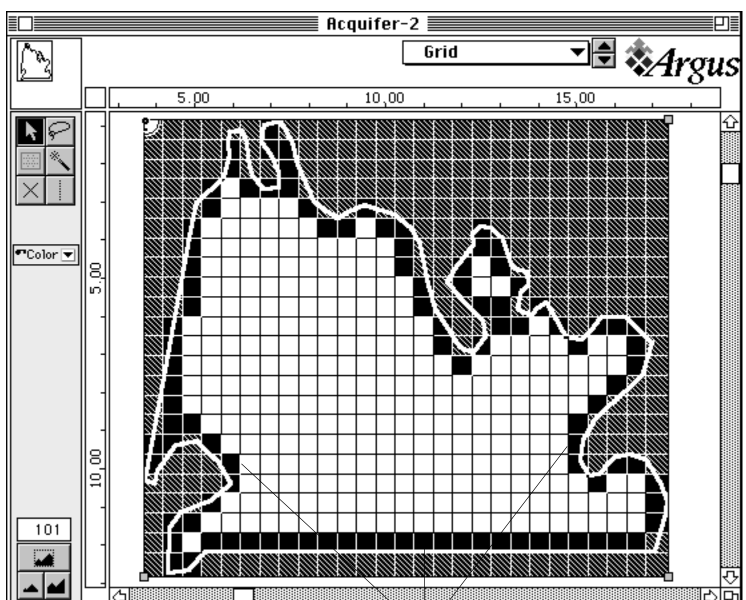

## Expanding the Search Scope

By default the dialog is set to “Replace Selection By.” If when invoking the dialog some blocks are already selected, they are de-selected and replaced by the newly found set.

To expand or reduce the search scope over an already found set, change this field to “Add Found Blocks To Current Selection” or “Remove Found Blocks From Current Selection” and define the additional search.

### Search Examples

- To find all blocks lying on the domain boundary, check the Boundary Blocks check box and initiate the search.

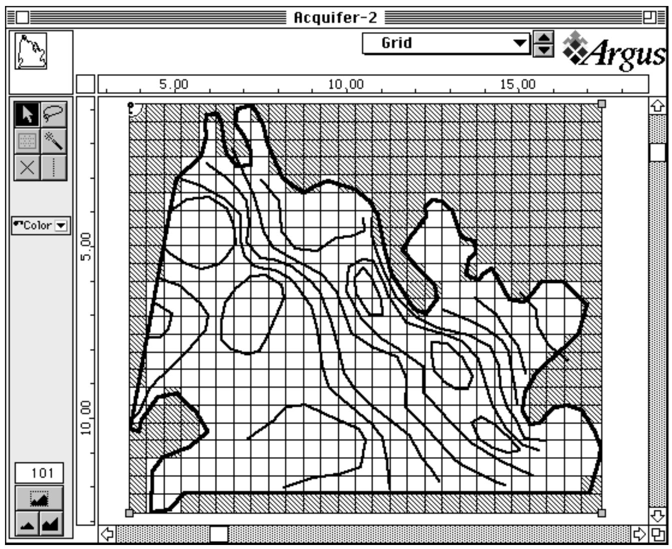



*Boundary blocks are selected.*

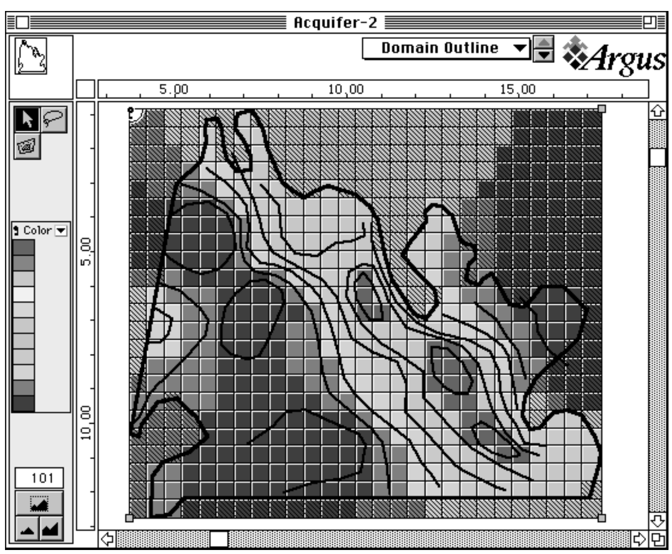
# Coloring Blocks

Using the color palette you can instruct Argus ONE to evaluate the grid blocks with respect to any of the block parameters. Using colors is explained in detail in chapter 1. To color the blocks in the following screen-shot, with respect to their values in the information layer seen through the grid:

1. Create a grid or block parameter and link it to the information layer.
2. From the Colors popup menu, select the linked parameter.




3. Click the colors handle to activate coloring.



## Colors and Performance Considerations

When colors is on and a parameter is chosen in the Colors popup menu, each block in the grid is evaluated at the evaluation context (the block center) for its value in the linked parameter.

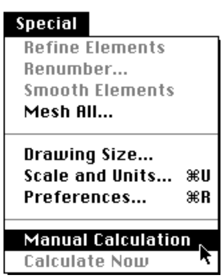
This evaluation is carried out during each redraw of the screen. If your grid contains a large number of blocks this re-evaluation process slows the redraw considerably.

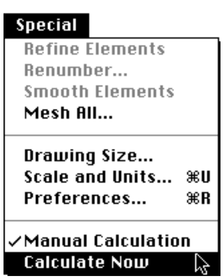
Coloring blocks should be considered as a post-processing feature. You should not try to work with colors all the time, but use colors when you need to preview a parameter's distribution in the blocks. If however, you do wish to work with colors on at all times, consider using the Manual Calculation command explained below.

### Temporarily halting color evaluation

If colors is on, and you wish to stop color re-evaluation immediately, and even before all blocks are redrawn in color, click the mouse anywhere in window. Color evaluation is halted and the redraw continues, but in black and white mode.

## Manual Calculation and Calculate Now Menus




If your mesh or grid contain many elements or blocks which are linked to complex parameters, screen redraws may temper with your work. Using the Manual Calculation command under the Special menu you can now instruct Argus ONE to refrain from recalculating elements and block colors (values) at each redraw.

However, you must be aware that when Manual Calculation is on, the colors of elements and blocks, as well as the values presented on the different information dialogs will not reflect any changes you have made from the time you turned manual Calculation on. To update the values, you can either select Calculate Now from the Special menu, or turn Manual calculation off.

---

**Important Note**: To prevent you from exporting wrong information to your model, Argus ONE automatically performs a Calculate Now command while exporting the data.

---

# Exporting a Grid

# Overview

Exporting grids is how you communicate data created within Argus ONE to and from other applications. Export is based on the ASCII data format.

*To control the export format you need to have the Programmable Export Templates module*

If you use your own models, or models for which Argus ONE PIEs Interfaces do not exisit, Argus ONE enables you to create the exact export format that will fit with your simulators' import formats. You do so by using the Programmable Export Templates module. Creating and using export templates is discussed in detail in chapter 5 "Exporting a Project."

If you are using Argus ONE with a ready-made PIE, the PIE takes care of defining the appropriate export template for you.

***Important note:****Using the Argus ONE built-in export format is faster than exporting "By Template."*

In this chapter you will learn about exporting your projects using the built-in export format which is available as part of Argus ONE meshing modules. You will also find out how to import grids you have created elsewhere into Argus ONE.

# Exporting a Grid

Exporting the grid you have created within Argus ONE, and its related data, enables you to input it to processing and modeling programs.

## Range and Precision of Data

All real data is internally handled by Argus ONE in double format. The following table summarizes the range of real numbers Argus ONE can work with.

| Type identifier | Double |
|---|---|
| Size (bytes:bits) | 10:64 |
| Significand precision | |
| Bits | 64 |
| Decimal digits | 15 |
| Decimal range (approximate) | |
| Min negative | -1.1E+308 |

| Type identifier | Double |
|---|---|
| Max neg norm | computing platform dependent |
| Min pos norm | computing platform dependent |
| Max positive | 1.7E+308 |

Errouneous values resulting from invalid parameter expressions, circullar reference, etc. are exported as NaN (Not a Number).

## Non Exportable Grid Variables

Only block icons are not exportable.

## Built-in Export File Format

Argus ONE's built-in export supports the following file format for exporting grids. Data fields can be delimited by the Tab, Space or Comma, or any other string of characters.

### The first line of a grid export file

| | | |
|---|---|---|
| Number of Row | Number of Columns | Number of grid & block parameters +1 |

The number of grid and block parameters is incremented by one to acount for the active/inactive block matrix (see follwing paragraphs).

### A line describing rows' position

There are "Number of Rows+1" lines describing the rows' Ycoordinate. They are ordered from row 1 to "Number of Rows+1".

Row Y Coordinate

After "Number of Rows+1" lines there is an empty line.

### A line describing columns' position

There are "Number of Columns+1" lines describing the columns' X coordinate. They are ordered from column 1 to "Number of Columns+1".

Column X Coordinate

After “Number of Columns+1” lines there is an empty line.

### A matrix describing active and inactive blocks

Each block is marked with either ‘one’ (1) or ‘zero’ (0). If you have deactivated a block, it is marked by a zero (0), otherwise it is marked by one (1).

| | | | | | | |
|---|---|---|---|---|---|---|
| $\text{Block}_{1,1}$ | $\text{Block}_{2,1}$ | $\text{Block}_{3,1}$ | . | . | . | $\text{Block}_{\text{"Number of Columns"}, 1}$ |
| $\text{Block}_{1,2}$ | $\text{Block}_{2,2}$ | $\text{Block}_{3,2}$ | . | . | . | $\text{Block}_{\text{"Number of Columns"}, 2}$ |
| $\text{Block}_{1,3}$ | $\text{Block}_{2,3}$ | $\text{Block}_{3,3}$ | . | . | . | $\text{Block}_{\text{"Number of Columns"}, 3}$ |
| . | | | | | | |
| . | | | | | | |
| . | | | | | | |
| $\text{Block}_{1, \text{"Number of Rows"}}$ | $\text{Block}_{2,1}$ | $\text{Block}_{3,1}$ | . | . | . | $\text{Block}_{\text{"Number of Columns"}, \text{"Number of Rows"}}$ |

After this matrix there is an empty line.

### Matrices describing grid and blocks’ parameters values

Following the active/inactive block matrix, a matrix for each of the grid and block parameters is exported.

After this matrix there is an empty line.

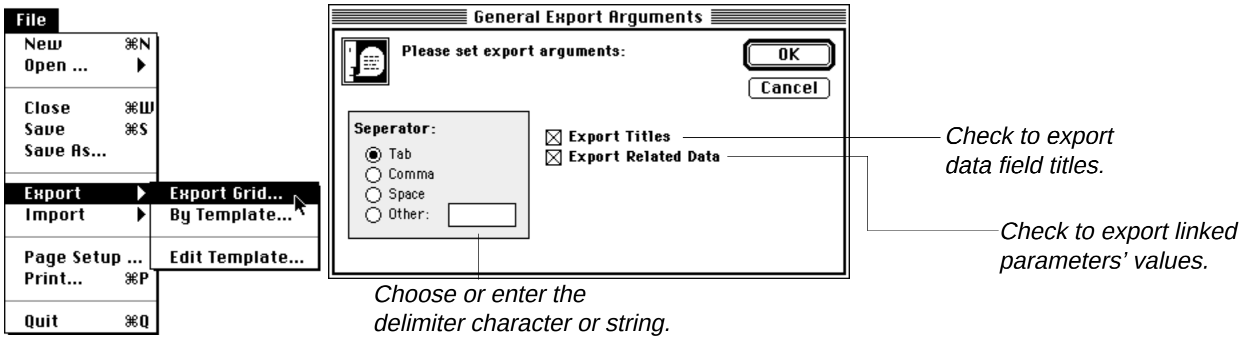
---

**Important note:** Grid $\text{Block}_{1,1}$ is the block located at the smallest X and Y coordinates, in the current coordinate system. All grid blocks are numbered in the positive X and Y direction.

---

## To Export a Grid Using the Built-in Export

1. If you have not yet linked parameters to the grid and blocks, and do need to do so, open the Layers dialog, create the parameters and link them.
2. Make sure the active layer is a grid layer.
3. File menu, choose Export Grid...
   The General Export Arguments dialog opens.




The Export dialog box allows you to determine the shape of the exported file, and the delimiters between the different fields.

- To set the delimiter, check the appropriate radio button.

  Selecting Tab as the delimiter will allow you to open the file in a spread sheet type application, where all data columns will be arranged in separate columns.

---

**Note:** Files exported with the comma character can not be re-imported by Argus ONE.

---

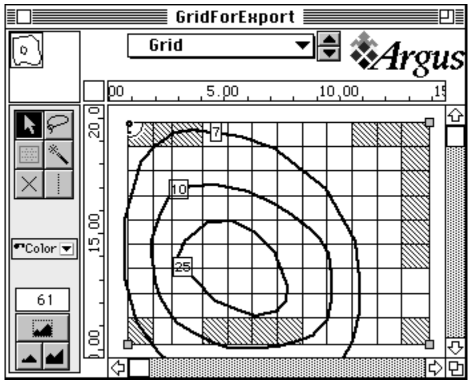
- To add titles at the top of each data line or block of lines, check the Export Titles check box.
- To export the grid only, without parameters' values, uncheck the Export related Data check box.
- Click the OK button to confirm your choices.
  The Save As dialog box opens.
- To export under a name different from Argus ONE's default, type it in the text edit box.
- Click the Save button.

A progress dialog box indicating the progress of the export process appears on the screen.

## An Example of an Exported Grid

The following table contains the exported file of the project shown in the following screen-shot.

The project includes one Information layer describing depth variation. The export file was created by selecting Export Titles, Export Related Data", and setting Tab as the delimiter.




*General Information.*

```
# Rows            Columns   Number of Sub Parameters
9                 12        2
(Empty line)
```

*Rows' Coordinates.*

```
# Rows position
10.2058
11.3175
12.3713
13.4252
14.479
15.5328
16.5867
17.6405
18.6943
19.7482
(Empty line)
```

*Columns' Coordinates.*

```
# Columns position
0.867486
1.94702
3.02656
4.1061
5.18564
6.26518
7.34472
8.42426
9.50379
10.5833
11.6629
12.7424
13.8798
(Empty line)
```

*Deactivated blocks matrix.*

*Blocks parameters' values.*

```
# Existency table:
0    1    1    0    0    0    0    1    1    1    1    0
1    1    1    1    1    1    1    1    1    1    1    1
1    1    1    1    1    1    1    1    1    1    1    1
1    1    1    1    1    1    1    1    1    1    1    0
1    1    1    1    0    0    1    1    1    1    1    0
1    1    1    1    0    1    1    1    1    1    1    0
1    1    1    1    1    1    1    1    1    1    1    0
1    1    1    1    1    1    1    1    1    1    1    0
0    0    0    1    1    1    1    1    1    0    0    0
(Empty line)
Depth
7    7    7    10   10   10   10   10   7    7    7    7
7    10   10   10   25   25   10   10   7    7    7    7
7    10   10   25   25   25   10   10   7    7    7    7
7    10   25   25   25   25   10   10   7    7    7    7
7    10   10   25   25   10   10   7    7    7    7    7
7    10   10   10   10   10   7    7    7    7    7    7
7    7    7    7    7    7    7    7    7    7    7    7
7    7    7    7    7    7    7    7    7    7    7    7
7    7    7    7    7    7    7    7    7    7    7    7
```

# Importing a Grid

See supplement version 3.

# Appendixes

# Keyboard and Mouse Shortcuts

This appendix summarizes the shortcut procedures you can use with either the keyboard or the mouse, to help speed your work with Argus ONE. The first section of the appendix lists and describes the keyboard shortcuts available to you. The second section summarizes the mouse shortcuts. The final section summarizes keyboard and mouse combination shortcuts.

## Keyboard Shortcuts

You don't always have to use the mouse to use Argus ONE; most commands and operations are accessible with key combinations—two keys pressed at the same time. Once you learn these combinations, you can perform most tasks quickly and easily with the keyboard.

If a command on a menu has an assigned key combination, the menu shows the key combination beside the command name.

### Choosing Commands with Key Combinations

Using a key combination, you can bypass the step of opening a menu. Assigned key combinations are displayed to the right of the commands on the menu.

| To Choose | Press | |
|---|---|---|
| | Macintosh | Other Platforms |
| **File** | | |
| New | ⌘ + N | CTRL + N |
| Open | ⌘ + O | CTRL + O |
| Close | ⌘ + W | CTRL + W |
| Save | ⌘ + S | CTRL +S |
| Print (visible layers) | ⌘ + P | CTRL + P |
| Quit | ⌘ + Q | CTRL + Q |
| **Edit** | | |
| Undo | ⌘ + Z | CTRL + Z |

| To Choose | Press | |
|---|---|---|
| | Macintosh | Other Platforms |
| Cut | ⌘ + X | CTRL + X |
| Copy | ⌘ + C | CTRL + C |
| Paste | ⌘ + V | CTRL +V |
| Select All | ⌘ + A | CTRL + A |
| Search For... | ⌘ + F | CTRL + F |
| Find Next | ⌘ + G | CTRL + G |
| Select Adjoining | ⌘ + J | CTRL + J |
| Deactivate Block | ⌘ + D | CTRL + D |
| UnLock Grid | ⌘ + E | CTRL + E |
| Detach Elements | ⌘ + D | CTRL + D |
| **View** | | |
| Layers | ⌘ + L | CTRL + L |
| Hide Others | ⌘ + H | CTRL + H |
| **Special** | | |
| Scale & Units... | ⌘ + U | CTRL +U |
| Preferences... | ⌘ + R | CTRL + R |
| **Navigation** | | |
| Zoom In | ⌘ + + | CTRL + + |
| Zoom Out | ⌘ + - | CTRL + - |
| Reduce To Fit | ⌘ + T | CTRL + T |

### Choosing Options in Dialog Boxes

You can use the keyboard to choose and edit some of the options in a dialog box.

| To do this | Press this key | |
|---|---|---|
| | **Macintosh** | **Other Platforms** |
| Select next text box | tab | tab |
| Select next button, check box, radio button, dial | — | tab |
| Move right or left within a text box | → or ← | → or ← |
| Move up and down in a list box | ↑ or ↓ | ↑ or ↓ |
| Press the default button (highlighted) | enter or ↵ | enter or ↵ |

## Mouse Shortcuts

Double-clicking in different layers, in certain areas and in some objects opens a dialog box for you.

| To do this | Double-click here |
|---|---|
| Open the Node Information dialog | A node |
| Open the Element Information dialog | An element |
| Open the Block Information dialog | a block |
| Open the Contour Information dialog | A contour |

# Keyboard and Mouse Combinations

| To do this | Keyboard and Mouse | |
|---|---|---|
| | **Macintosh** | **Other Platforms** |
| Detach a node | ⌘ + 🖱 | CTRL + 🖱 |
| Detach a selection of elements | ⌘ + 🖱 | CTRL + 🖱 the selected elements |
| Zoom in from the cursor position | ⌥ + 🖱 | ALT + 🖱 |
| Zoom out from the cursor position | ⇧ + ⌥ + 🖱 | SHIFT + ALT + 🖱 |
| Zoom to 100 percent zoom level | ⌃ + 🖱 in the zoom by percent box. | CTRL + 🖱 in the zoom by percent box. |

## Hiding, Showing and Moving Between Layers

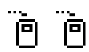
Using the following key combinations enables you to quickly move between layers and set their visibility.

| To do this | Keyboard and Mouse | |
|---|---|---|
| | **Macintosh** | **Other Platforms** |
| Move to the next visible layer. | ⌘ + ↓ | CTRL + ↓ |
| Move to the previous visible layer. | ⌘ + ↑ | CTRL + ↑ |
| Move to the next or previous visible layer and hide all other layers. | ⌘ + ⇧ + ↓ or ↑ | SHIFT + CTRL + ↓ or ↑ |
| Move to the next or previous visible layer, either hidden or visible, and show it | ⌘ + ⌥ + ↓ or ↑ | SHIFT + ALT + ↓ or ↑ |

| To do this | Keyboard and Mouse | |
|---|---|---|
| | **Macintosh** | **Other Platforms** |
| Move to the next or previous visible layer, either hidden or visible, and hide all others. | ⌘ + ⌥ + ⇧ + ↓ or ↑ | CTRL+ SHIFT + ALT + ↓ or ↑ |

# Legend

The following table summarizes the icons of the keyboard keys used in this manual.

| Icon | Meaning | |
|---|---|---|
| | **Macintosh** | **Other Platforms** |
|  | Mouse Click | Mouse Click |
|  | Double-Click | Double-Click |
| ⇧ | SHIFT Key | — |
| ⌘ | COMMAND Key | — |
| ⌥ | OPTION Key | — |
| ⌃ | CONTROL Key | |
| tab | TAB Key | TAB Key |
| ↩ | RETURN Key | RETURN Key |
| enter | ENTER Key | ENTER Key |
| ↓ | Down Cursor Key | Down Cursor Key |
| ↑ | Up Cursor Key | Up Cursor Key |
| → | Right Cursor Key | Right Cursor Key |
| ← | Left Cursor Key | Left Cursor Key |

# Subject Index

# Subject Index

# D

# E

# F



# G

# H



# I



# J

# K



# L



# M

# N

# O



# P

# Q



# R



# S

# T

## U



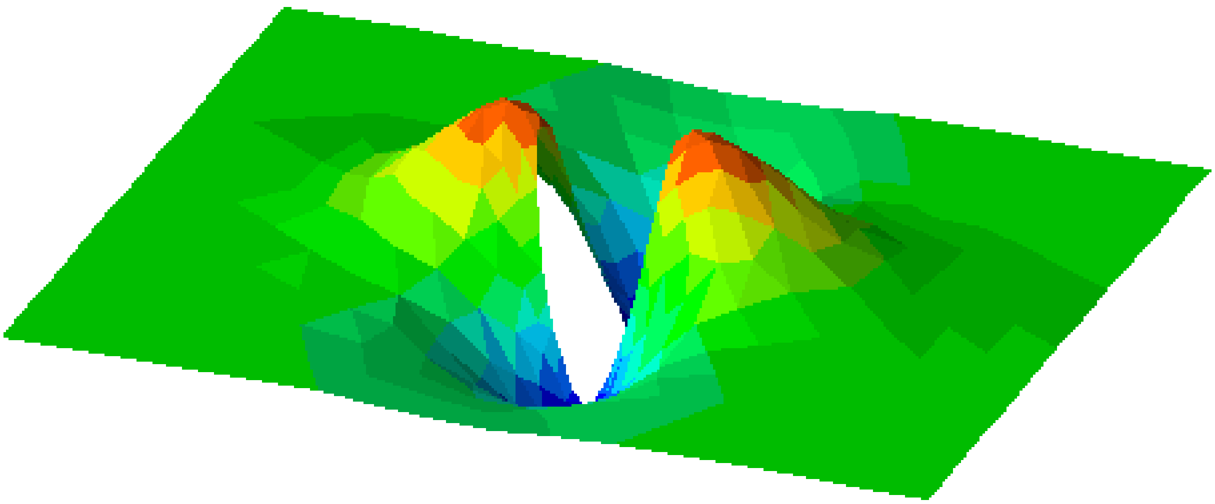
## V



## W



## Z



# Argus ONE

## New Features Supplement

For version 2.5

# Argus ONE Version 2.5

This supplement describes in detail the new capabilities added in version 2.5, among which you will find:

- **Post-Processing and Scientific Visualization Tools.**
- **Data Layers - A new layer type supporting many new data formats.**
- **Quadrilateral Mesh Layer- Quadrilateral finite element meshes.**
- **Additional CAD tools and capabilities.**
- **Additional GIS functions**

If your User's Guide revision is 7 or later, some of the capabilities described here are also documented in the User's Guide itself.

# Contents

# Working with Data Layers

## Overview

*Data type layers serve as containers of discrete information.*

In some cases your data is created or sampled on discrete points in space. Discrete data might be created by a sampling procedure such as when ocean bathymetry is measured from a ship, or when the data was created on a grid or on a mesh, by some kind of a numerical procedure. The latter is very common when you work with numerical models. The model's results are calculated, and are thus available, on a grid or a mesh.

*Data type layers are very fast in handling large numbers of data points.*

When your discrete information is available to you on hundreds and thousands of points, the use of point objects in Information type layers is not adequate and may prove too slow. In these cases, you should use Data type layers to import your information into Argus ONE's workplace.

*Information in data type layers is static.*

The newly introduced Data type layers are a natural extension of Argus ONE's Information type layers. Data type layers differ from Information type layers in that the data you read into them is **static**. You can not edit this data and if you change your source data, you need to re-read it into the Data type layer.

Data can be read into Data type layers from Mesh and Grid layers and can be imported from external files.

Discrete information can be imported into Argus ONE's data layers in many formats. The three main types of data formats are:

- Scattered Data - Data at X, Y locations.
- Gridded Data - Data on regular or irregular orthogonal grids.
- Meshed Data - Data on triangular meshes.

Information is stored in data layers at data points. Each data point, defined by its X and Y coordinate may have as many additional data fields as required.

*Use Data type layers to visually investigate your model results.*

Using Data type layers you can import your model's results back into Argus ONE's workplace to present and investigate your solution using the scientific visualization tools that are available in Maps type layers.

Reading information from mesh and grid layers into Data type layers enable you to use the Maps type layers visualization tools to study your model input. It also enables you to interpolate data from grid layers onto mesh layers and vise a versa.

*Information in data type layers can be linked to other layers and used in expressions.*

One of the strongest points of data type layers is that information stored in them can be linked to other layers, and can be used in all of Argus ONE expressions.

## Data Types and Formats Supported by Data Layers

Introducing information into data type layers can be accomplished by reading it from mesh nodes and grid blocks, or by importing it from data files. Data in the following ten formats can be read or imported into a data layer:

1. Data from mesh layers - Triangular elements
2. Data from grid layers - Line centered grid
3. Data from grid layers - Block centered grid
4. Data from file - Scattered points
5. Data from file - Points on a line centered grid with grid topology
6. Data from file - Points on a block centered grid with grid topology
7. Data from file - Points on a line centered grid associated with a grid layer
8. Data from file - Points on a block centered grid associated with a grid layer
9. Data from file - Points on a triangular element mesh with mesh topology
10. Data from file - Points on a triangular element mesh associated with a mesh layer

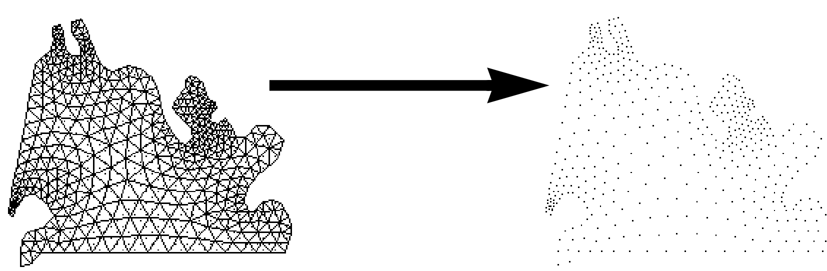
## How is Information Stored in Data Layers

When you read or import information into a data type layer, the information is stored in a data structure. This structure constitutes of data point locations, data layer parameters storing the various values at the points, and triangulation information.

The data is stored at information points, and the triangulation is either read or created. If for instance, you read data from a mesh layer, the number and locations of data points read will be that of the mesh nodes, and the mesh topology constitutes the triangulation information. This chapter describes in detail the data formats and triangulation specifications.

## How is Information Presented in Data Layers

When you read information into data layers the layer presents the locations at which data is stored. If for instance, the data was read from a mesh layer, data points locations are at the exact locations of the mesh nodes.

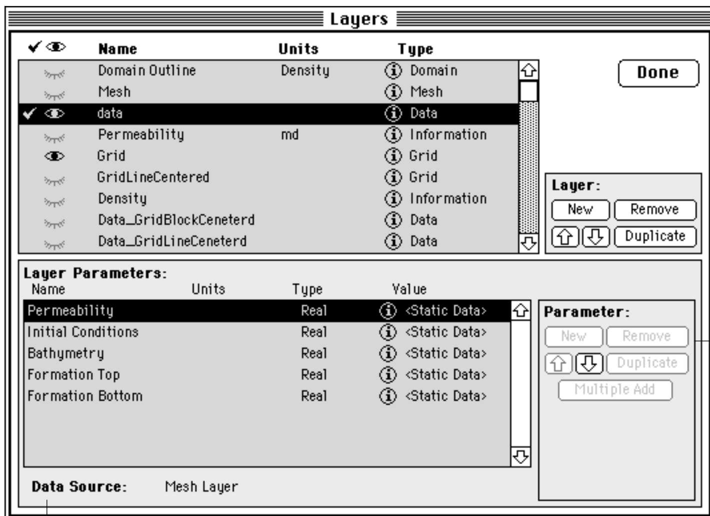


*When reading information into a data layer from this mesh...*

*...Data points are created at the locations of mesh nodes.*

## Data Layer Parameters

When you read information from mesh or grid layer parameters or from a file, these parameters are added to the data layer parameter list. Data layer parameters do not differ from other layers' parameters. You can link data layers parameters to other layers and use them in expressions. To read more about using data layer parameters refer to "Linking Data layers Parameters to Other Layers" on page 17 and to "Using Data Layer Parameters in Expressions" on page 17.




*Data layer parameters are automatically created when information is read or imported into the layer. The source of data is indicated in this field.*

*Data layer parameters can not be manually added, duplicated or removed. You can only change their order of appearance by clicking the up and down arrows.*

## Naming Data Layer Parameters

Data layer parameters are named automatically. When you read them from mesh and grid layer parameters they are named after the parameters that they were read from. When you read data from a disk file the parameters are named sequentially as "parameter0", "parmaeter1", etc. The line at the bottom of the Layers dialog presents the name of the source layer of file.

## Information in Data Layers is Static

***Important Note:***
*Data in data layers is static. If you change the source mesh or grid, the data layer is not updated to reflect these changes*

When you read information into data layers it is detached from its source. If you read information from a mesh or grid layer, and later on refine or change that mesh or grid, the information stored in the data layer is not updated to reflect these changes. To update the data layer you need to read again the data from the mesh or grid layer.

# Reading Information from Mesh and Grid Layers

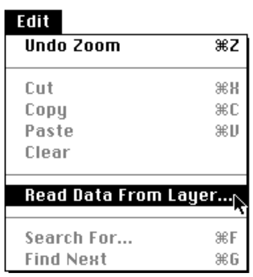
Reading mesh node and grid block parameters into a data layer enables you to investigate the distribution of these parameters on your mesh and grid using the visualization methods available in maps layers.

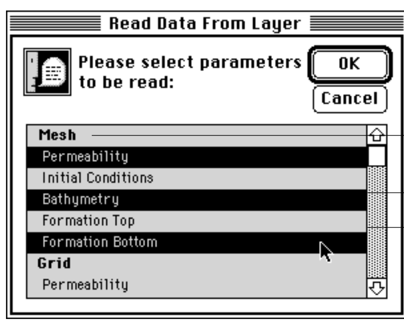
It also enables you to interpolate data from grid blocks onto mesh nodes or elements, and vise a versa, by linking the data layer parameters to mesh nodes, elements and grid blocks.

You can read as many parameters as required from a mesh or grid layer. However, you can not read parameters from different mesh or grid layers into the same data layer since each mesh or grid layer might have a different triangulation.

### To read information from mesh nodes or grid blocks

1. In the Layers dialog, create a data layer.
2. Make sure a data layer is the active layer.
3. From the Edit menu, choose Read Data From Layer...
   The Read Data From dialog opens.




***Important Note:***
*You can not add parameters from different layers since each layer might have a different triangulation.*

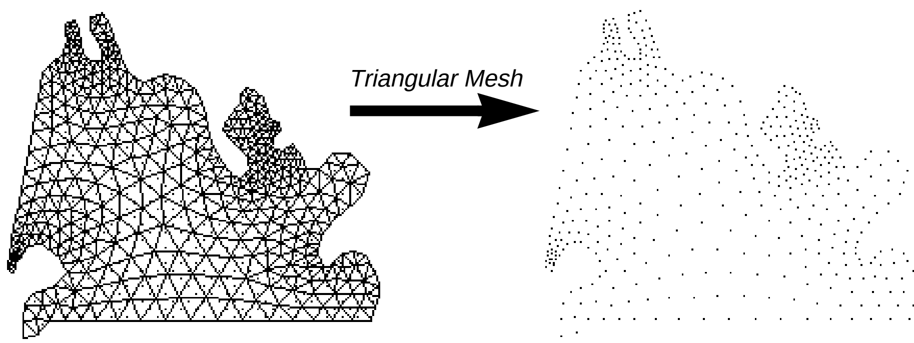
4. Click the layers parameters that you wish to read into the data layer.
   - Or -
   Click an already selected parameter to deselect it.
   - Or -
   Click the layer's name to select all of it's parameters
5. Click the OK button.

## How is Mesh and Grid Information Stored in Data Layers

All of the mesh or grid parameters that you select are read at each mesh node or grid block. When mesh parameters are read, the mesh triangulation is also read by the data layer. When reading grid parameters, the data layer creates the appropriate triangulation based on the grid topology.

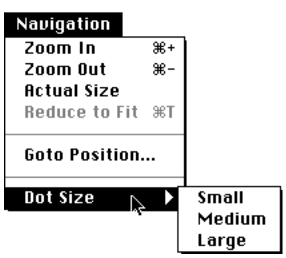
## How is Mesh and Grid Information Presented in Data Layers

When you read information into data layers, the layer presents the locations at which data is stored. If the data was read from a mesh layer, the data points locations are at the exact locations the of mesh nodes. In case the data was read from a grid layer the data points locations are at the exact locations of block centers.

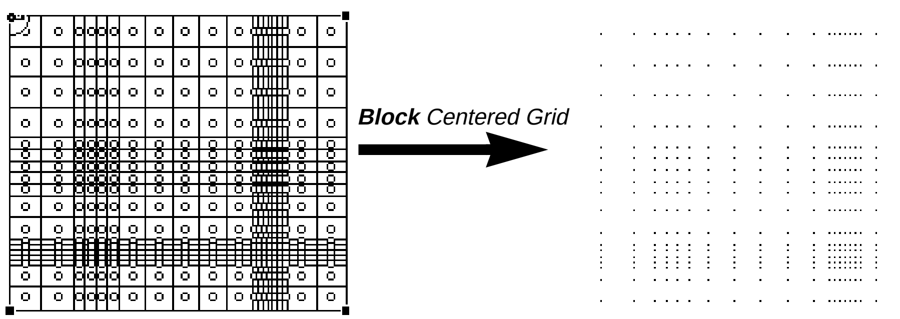



*When reading information into a data layer from this mesh...*

*...Data points are created at the locations of mesh nodes.*



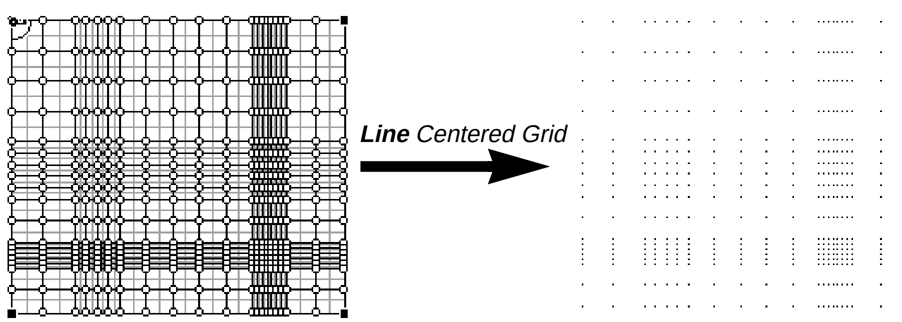

*The default dot size is set to medium. To set the dot size smaller or larger, select the appropriate menu item from the Navigation menu.*




*When reading information into a data layer from this block centered grid...*

*...Data points are created at the locations of block centers.*

***Important note:***
*The smaller the dot size the faster it draws on the screen. If you have many data points, and redraw is slow, set the dot size to small.*




*When reading information into a data layer from this line centered grid...*

*...Data points are created at the locations of block centers.*

# Importing Information from Files

If some of your information is available to you at scattered, grid or mesh points, stored in a file, you can import this information into a data layer. This enables you to interpolate such data onto your meshes and grids, and to use Maps layers visualization objects to investigate model results that you import from your model output files.

## Supported Data Formats

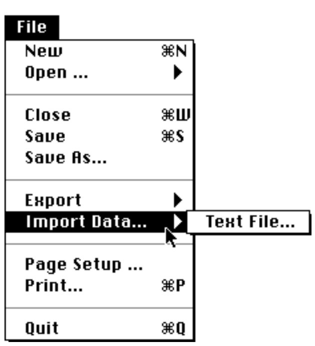
There are seven data formats that are available for importing information from files. These are:

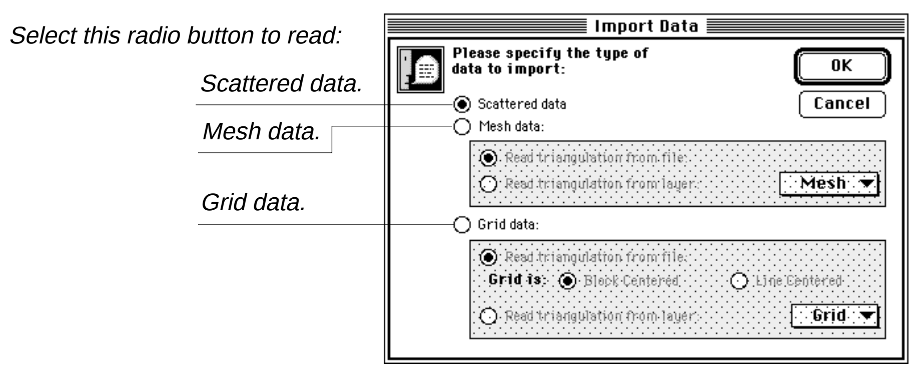
1. Scattered points
2. Points on a line centered grid with grid topology
3. Points on a block centered grid with grid topology
4. Points on a line centered grid associated with a grid layer
5. Points on a block centered grid associated with a grid layer
6. Points on a triangular element mesh with mesh topology
7. Points on a triangular element mesh associated with a mesh layer

You specify the data format you intend to read in the Import Data dialog.

### To import data into a data layer

1. Make sure the active layer is a data type layer.
2. From the File menu select Import Data...
   The Import Data dialog opens to allow you to specify the data format.




3. Specify the data format.
4. Click the OK button,
   Use the standard Open File dialog box to locate and open the desired file.

# Scattered data

## To import scattered data

1. In the Import Data dialog click the ◉ Scattered data radio button to set the data format to Scattered data.
2. Click the OK button,
   Use the standard Open File dialog box to locate and choose the file containing the data.

## Scattered data file format

### The first line of a file describing scattered data

| | |
|---|---|
| Number of Data Points | Number of Parameters |

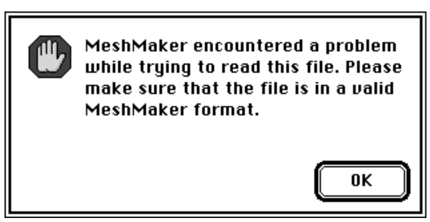
Fields must be delimited by the TAB or SPACE characters. The line is terminated by a CARRIAGE RETURN. The number of parameters is the number of values assigned to each data point (X,Y location) in the file. The number of parameters is unlimited.

### A line describing a data point

| | | | | | |
|---|---|---|---|---|---|
| X | Y | Value of first data point parameter | Value of second data point parameter... | Value of nth data point parameter... | Value of last data point parameter |

There are "Number of Data Points" lines describing the data points. Fields must be delimited by the TAB or SPACE characters. Each line is terminated by a CARRIAGE RETURN.

Empty fields and sequential delimiters are not allowed. If the file does not fully adhere to the above format, Argus ONE will fail to read the file and will notify you by presenting the following alert:




# Mesh data

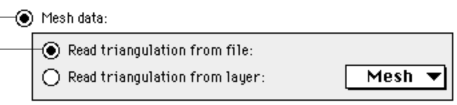
Data created on a mesh can be read into a Argus ONE data layer in two formats. The first format includes the triangulation information (connectivity list) and node coordinates within the file. The second format contains only node numbers and parameter values, while the triangulation information and node locations are read from a Argus ONE referenced mesh. The latter is very convenient when you need to visualize your model results and when the mesh on which you simulated the problem was created in Argus ONE. You must however be sure that you did not change or edit that mesh since the time you exported it to your model.

### To import mesh data with triangulation information

1. In the Import Data dialog click the “Mesh Data” radio button to set the data format to Mesh data.
2. Click the “Read Triangulation from file” radio button.
3. Click the OK button,
   Use the standard Open File dialog box to locate and choose the file containing the data.

*Click this radio button to select mesh data.*

*Click this radio button to specify that the triangulation information as well as node coordinates are to be read from the file.*




## Mesh data file format (Triangulation included)

### The first line of a file describing mesh data

| Number of Elements | Number of Nodes | Number of Element Parameters (ignored) | Number of Node Parameters |
|---|---|---|---|

Fields must be delimited by the TAB or SPACE characters. The line is terminated by a CARRIAGE RETURN. The number of element parameters is the number of values assigned to each element and is ignored. The number of node parameters is the number of values assigned to each node. The number of parameters is unlimited.

### A line describing a node

| Node Number | X | Y | Value of first node parameter | Value of second node parameter | Value of nth node parameter... | Value of last node parameter |
|---|---|---|---|---|---|---|

There are “Number of Nodes” lines describing the nodes. Fields must be delimited by the TAB or SPACE characters. Each line is terminated by a CARRIAGE RETURN.

### A line describing the element connectivity

| Element Number | 1st Node Number | 2nd Node Number | 3rd Node Number |
|---|---|---|---|

There are “Number of Elements” lines describing the element connectivities in a counter-clockwise order. Fields must be delimited by the TAB or SPACE characters. Each line is terminated by a CARRIAGE RETURN.

Empty fields and sequential delimiters are not allowed. If the file does not fully adhere to the above format Argus ONE alerts you and halts import.

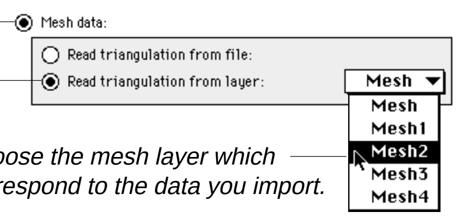
You may also read files exported from a Argus ONE mesh layer using the default export format, provided you used a SPACE or TAB as the delimiters. To read more about the mesh export default format refer to “Chapter 8”, “Built-in Export File Format” on page 251.

### To import mesh data without triangulation information

1. Make sure that the data you are about to import correspond to a Argus ONE mesh, and that you did not change or edit that mesh since you exported it to your model.
2. In the Import Data dialog click the “Mesh Data” radio button to set the data format to Mesh data.
3. Click the “Read Triangulation from Layer” radio button.
4. Click the OK button,
   Use the standard Open File dialog box to locate and choose the file containing the data.

*Click this radio button to select mesh data.*

*Click this radio button to specify that the triangulation information as well as node coordinates are to be read from a mesh layer*




*Choose the mesh layer which correspond to the data you import.*

## Mesh data file format (Read triangulation from a mesh layer)

### The first line of a file describing mesh data

Number of Node Parameters

The line is terminated by a CARRIAGE RETURN. The number of node parameters is the number of values assigned to each node. The number of parameters is unlimited.

### A line describing a node

| | | | | |
|---|---|---|---|---|
| Node Number | Value of first node parameter | Value of second node parameter | Value of nth node parameter... | Value of last node parameter |

There are “Number of Nodes” lines describing the nodes. Fields must be delimited by the TAB or SPACE characters. Each line is terminated by a CARRIAGE RETURN.

## Possible Pitfalls

Argus ONE does not check that the data you read correspond to the mesh the triangulation is read from. To avoid problems you must be certain that the information you import fully correspond to the referenced mesh.

## Grid data

Both block centered and line centered grids can be read into a data type layer. Data created on a grid can be read into a Argus ONE data layer in two formats. The first format includes the grid triangulation information within the file, while the second contains only matrices of parameter values. In the latter, the grid triangulation (row and column positions) is read from a Argus ONE referenced grid. The latter is very convenient when you need to visualize your model results and when the grid on which you simulated the problem was created in Argus ONE. You must however be sure that you did not change or edit that grid since the time you exported it to your model.

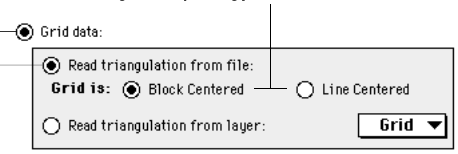
### To import grid data with grid triangulation

1. In the Import Data dialog click the “Grid Data” radio button to set the data format to Grid data.
2. Click the “Read Triangulation from file” radio button.
3. Click the OK button,
   Use the standard Open File dialog box to locate and choose the file containing the data.

*Click the Block or Line centered buttons to select the grid topology to be read.*

*Click this radio button to select grid data.*

*Click this radio button to specify that the triangulation information (row and column positions) is to be read from the file.*




### Grid data file format (Triangulation included)

#### The first line of a file describing mesh data

| | | |
|---|---|---|
| Number of Row | Number of Columns | Number of block parameters |

Fields must be delimited by the TAB or SPACE characters. The line is terminated by a CARRIAGE RETURN. The number of block parameters is the number of values assigned to each block. The number of parameters is unlimited.

### A line describing rows’ position

For block centered grids, there are “Number of Rows+1” lines describing the rows’ Y coordinate, ordered from row 1 to “Number of Rows+1”. For line centered grids, there are “Number of Rows” lines describing the rows’ Y coordinate, ordered from row 1 to “Number of Rows”.

Row Y Coordinate

After “Number of Rows+1” or “Number of Rows” lines there is an empty line.

### A line describing columns' position

For block centered grids, there are "Number of Columns+1" lines describing the columns' X coordinate, ordered from column 1 to "Number of Columns+1". For line centered grids, there are "Number of Columns" lines describing the columns' X coordinate, ordered from row 1 to "Number of Columns"

Column X Coordinate

After "Number of Columns+1" or "Number of Columns" lines there is an empty line.

### Matrix describing block parameters

For each parameter (value), a matrix of "Number of Rows" by Number of Columns" describing the blocks' parameter value.

| | | | |
|---|---|---|---|
| BlockParam $_{1,1}$ | BlockParam $_{2,1}$ | . . . | BlockParam $_{\text{"Number of Columns"}, 1}$ |
| BlockParam $_{1,2}$ | BlockParam $_{2,2}$ | . . . | BlockParam $_{\text{"Number of Columns"}, 2}$ |
| BlockParam $_{1,3}$ | BlockParam $_{2,3}$ | . . . | BlockParam $_{\text{"Number of Columns"}, 3}$ |
| . | . | . . . | . |
| . | . | . . . | . |
| . | . | . . . | . |
| BlockParam$_{1, \text{"Number of Rows"}}$ | BlockParam $_{2,1}$ | . . . | BlockParam $_{\text{"Number of Columns"}, \text{"Number of Row"}}$ |

After this matrix there is an empty line.

There are "Number of block parameters" matrices.

---

**Important note:** Grid Block $_{1,1}$ is the block located at the smallest X and Y coordinates, in the current coordinate system. All grid blocks are numbered in the positive X and Y direction.

---

Empty fields and sequential delimiters are not allowed. If the file does not fully adhere to the above format Argus ONE alerts you and halts import.

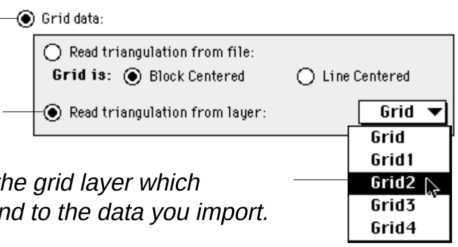
You may also read files exported from a Argus ONE grid layer using the default export format, provided you used a SPACE or TAB as the delimiters. To read more about the grid export default format refer to "Chapter 10","Built-in Export File Format" on page 293.

### To import grid data without triangulation information

1. Make sure that the data you are about to import correspond to a Argus ONE grid, and that you did not change or edit that grid since you exported it to your model.
2. In the Import Data dialog click the “Grid Data” radio button to set the data format to Grid data.
3. Click the “Read Triangulation from Layer” radio button.
4. Click the OK button,
   Use the standard Open File dialog box to locate and choose the file containing the data.

*Click this radio button to select grid data.*

*Click this radio button to specify that the triangulation information (row and column positions) is to be read from a grid layer.*




*Choose the grid layer which correspond to the data you import.*

## Grid data file format (Read triangulation from a grid layer)

### The first line of a file describing grid data

Number of Block Parameters

The line is terminated by a CARRIAGE RETURN. The number of block parameters is the number of values assigned to each block. The number of parameters is unlimited.

## Matrix describing block parameters

For each parameter (value), a matrix of “Number of Rows” by Number of Columns” describing the blocks’ parameter value.

| | | | | | |
|---|---|---|---|---|---|
| BlockParam $_{1,1}$ | BlockParam $_{2,1}$ | . | . | . | BlockParam $_{\text{"Number of Columns"}, 1}$ |
| BlockParam $_{1,2}$ | BlockParam $_{2,2}$ | . | . | . | BlockParam $_{\text{"Number of Columns"}, 2}$ |
| BlockParam $_{1,3}$ | BlockParam $_{2,3}$ | . | . | . | BlockParam $_{\text{"Number of Columns"}, 3}$ |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| BlockParam$_{1, \text{"Number of Rows"}}$ | BlockParam $_{2,1}$ | . | . | . | BlockParam $_{\text{"Number of Columns"}, \text{"Number of Row"}}$ |

There are “Number of block parameters” matrices, after each there is an empty line.

## Possible Pitfalls

Argus ONE does not check that the data you read correspond to the grid the triangulation is read from. To avoid problems you must be certain that the information you import fully correspond to the referenced grid.

# Creating Additional Data Type Layers

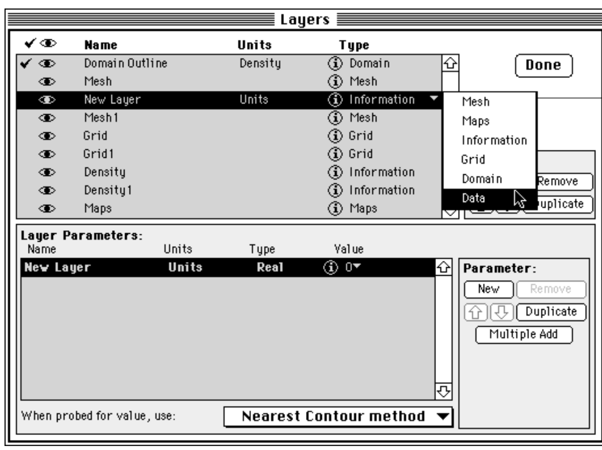
You can create as many data type layers as you need. As you will learn later in this chapter, a data layer can have as many layer parameters as you need.

## To Create a New Data Layer

Creating a new data layer is no different then creating other Argus ONE layers.

1. Follow the instruction in section “Creating an Information Type Layer” on page 31.
2. After you’ve clicked the New button and a new layer was created, From the Layer type popup menu choose Data.

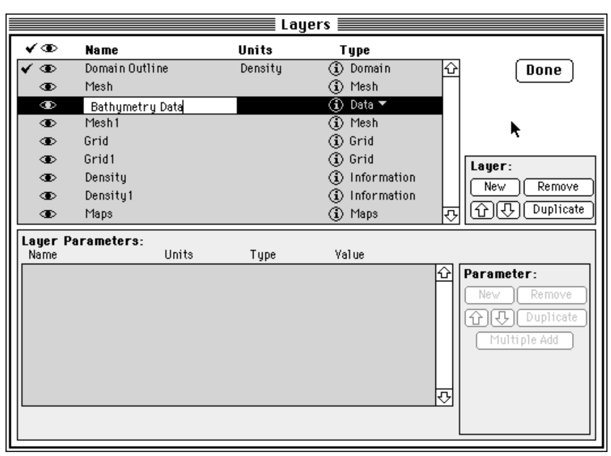
*You can create as many data type layers as you need*



*1. Click and hold the mouse button in the Type field to pop up the Layer type menu.*

*2. From the menu select Data.*

When you create a new data type layer it is created with no layer parameters. You also can’t create data layer parameter yourself. Data layer parameters are automatically created as you read or import information into a them.



*Data layers are not assigned a layer parameter when you create them.*

*When you read or import information into a data layer, Argus ONE automatically creates the required number of layer parameters*

# Linking Data layers Parameters to Other Layers

You link data layer parameters in the same manner you link other parameters. You can link data layer parameters to mesh and grid parameters.

To learn more about parameter linking refer to “Chapter 3”,“Linking Parameters” on page 132.

# Interpretation of Data in Data Layers

Interpreting information in data layers onto mesh nodes and elements, and onto grid blocks, is very useful when your data was sampled or created on a large number of scattered, grided or meshed points. Such cases might occur when your data was sampled during an automated survey, or by a numerical procedure.

When a data layer parameter is linked to a mesh or a grid layer parameter, the value returned by the data layer, when probed, is an interpolated value of a number of neighboring data points. Argus ONE interpolation is based on an “Inverse Distance weighted Interpolation” algorithm.

The point of evaluation is part of the “Evaluation Context” which is explained in detail in “Chapter 3”, section “Expressions” on page 135.

# Using Data Layer Parameters in Expressions

Data layer parameters can be used in expressions as any other Argus ONE parameter. To read more about using parameters in expressions refer to “Chapter 3”, section “Expressions”, starting on page 135.

# Using Data Layers for Post-Processing and Visualization

As you will find out later in this chapter, post-processing objects can be created only from data stored in data type layers.

# Exporting Data Layers Information

Argus ONE enables you to export any data stored in a data type layer. Although all data stored in data layers can be also exported otherwise, for instance directly from a mesh or grid layer it was read from, Argus ONE allows you to export directly from a data layer to let you have full access to your data.

## To export from data layers

1. Make sure that the active layer is the data layer from which you wish to export data.
2. From the File menu select Export.
3. Use the standard Save File dialog box to name and select where to save the file.

All of the data layer parameters and data point locations are exported to the file.

## Export Data File Format

The file format is identical to the input file format of scattered data. For a detailed listing of that format refer to the section“Scattered data” on page 10 in this chapter.

# Enhanced Maps Layers

## Overview

The Maps type layer has been enhanced to enable you to perform many CAD and Post-Processing tasks. These enhancements include additional drawing tools, scientific visualization methods, enhanced color for DXF objects and scalable fonts.

Using these tools Argus ONE enables you to complete a full cycle of a modeling project within the same workplace. You can import your data into Argus ONE's workplace, create your meshes and grids, interpolate information onto them, export to all the models you use and develop, import the model results and visualize them, and print images for your reports.

The visualization methods are created as maps layer objects that can be resized, moved and arranged so that you can create a meaningful presentation that overlay your project's region.

## Color and Values in DXF Files

When you import objects from DXF files, Argus ONE imports the objects colors and "Z" values (attributes) as well. This is extremely useful when you create contour maps in AutoCad, or store digitized maps in DXF format. When you import such files into Argus ONE maps and information type layers, the contours (or objects) values are imported as well and are automatically assigned to Argus ONE's objects.

## Vector Fonts - Scalable and Rotatable Text Objects

Text that is imported from DXF files or created in Argus ONE's visualization objects is now fully scalable and rotatable. If the source text is rotated, Argus ONE rotates it as well. When you zoom in or out, all text objects are scaled to the appropriate size.

# Drawing Tools

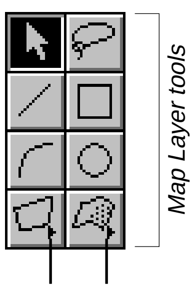
## Maps layers tool palette (this page should be placed after page 57 in the User's Guide)

The map layer tool palette contains all the tools you need to easily create and edit map objects. To operate on map objects, select a tool and use it on the appropriate object. To create objects select the tool and start creating the object.

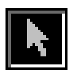
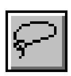
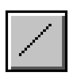
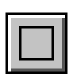
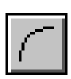
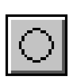
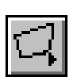
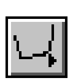
A tool is active only when you can use it. When a tool is not available it is dimmed to indicate you can not use it. When you select a tool it highlights itself to indicate it is the active tool, and the cursor changes its shape to suit the selected tool.

Tools that contain a number of optional tools are marked with a popup arrow.

### New Drawing Tools




*Tools' popup arrow*

| Tool | Tool's name and function | Tool can be activated when |
|---|---|---|
|  | The Arrow tool. The default tool in all layers. Allows you to select, edit and move objects. | Always. |
|  | The Lasso tool. Allows you to select all objects within an arbitrary shape. | There are one or more objects in the current layer. |
|  | The Line tool. Allows you to draw a line. | A maps type layer is the active layer. |
|  | The Rectangle tool. Allows you to draw a rectangle. | A maps type layer is the active layer. |
|  | The Arc tool. Allows you to draw an arc. | A maps type layer is the active layer. |
|  | The Circle tool. Allows you to draw a circle. | A maps type layer is the active layer. |
|   | The Polygon tool. Allows you to draw a close polygon,<br>- Or -<br>The open polygon tool. Allows you to draw an open polygon. | A maps type layer is the active layer. |

## Drawing graphic objects in maps type layers (the following two page should be placed after page 68 in the User's Guide)

You can now draw graphic objects directly in Argus ONE's Maps type layers. This allows you to annotate your project's backgrounds and to create contours of exact shapes. You can copy these objects from a maps type layer and paste them in other maps type layers or paste them in information type layers. If you study an analytic problem, or a problem with an exact geometry, this allows you to define the exact geometry within the Argus ONE workplace.

### To draw a line

1. From the tool palette select the line tool.
2. Click-drag the mouse where you want the line to start.
   an image of the line follows your cursor.
3. Stretch and rotate the line as you create it.
4. Release the mouse button to set the line's last point.

### To draw a rectangle

1. From the tool palette select the rectangle tool.
2. Click-drag the mouse where you want the rectangle to start.
   an image of the rectangle follows your cursor.
3. Stretch the rectangle in any direction as you create it.
4. Release the mouse button where you want the rectangle to end.

### To draw an arc

1. From the tool palette select the arc tool.
2. Click-drag the mouse where you want the arc to start.
   an image of the arc follows your cursor.
3. Stretch the arc in any direction as you create it.
4. Release the mouse button where you want the arc to end.

### To draw a circle

1. From the tool palette select the circle tool.
2. Click-drag the mouse where you want the circle to start.
   an image of the circle follows your cursor.
3. Stretch the circle in any direction as you create it.
4. Release the mouse button where you want the circle to end.

### To draw a close polygon

1. From the tool palette select the polygon tool.
   If the open polygon tool is selected, click the mouse in its icon until the menu pops up, and then select the close polygon tool.
2. Click the mouse where you want the polygon to start.
   an image of the first segment follows your cursor.
3. Click the mouse to create an additional vertex.
4. Double click the mouse to end the polygon creation and close it.

### To draw an open polygon

1. From the tool palette select the open polygon tool.
   If the close polygon tool is selected, click the mouse in its icon until the menu pops up, and then select the open polygon tool.
2. Click the mouse where you want the polygon to start.
   an image of the first segment follows your cursor.
3. Click the mouse to create an additional vertex.
4. Double click the mouse to end the polygon creation.
   If you click the last vertex in the first one, a close polygon is created.

## To copy graphic objects from a Maps layer

You can now copy all graphic objects from a maps type layer and paste them into other maps or information type layers, either in the same project or in another project.

1. Select the graphic object or objects you want to copy.
2. From the File menu, choose Copy.
   The objects are stored in Argus ONE's internal clipboard.

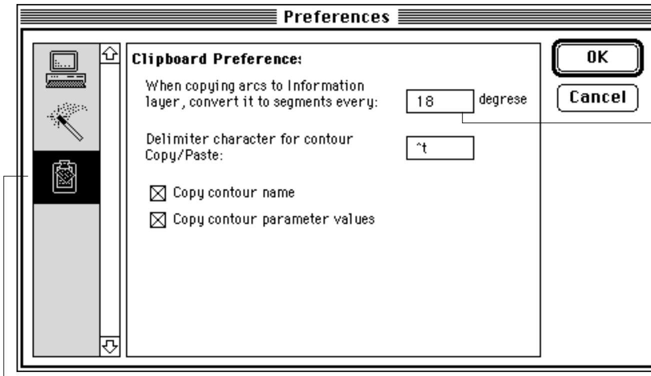
## Setting the segment length for copying arcs and circles

When you copy arcs and circles from maps type layers, Argus ONE subdivides them into line segments to transform them to information layers contours. This way you can paste them into information, domain outline and density layers.

You can control the number of segments Argus ONE will subdivide an arc and a circle into. You do so by telling Argus ONE what is the angle by which it should subdivide arcs and circles.

1. From the Special menu choose Preferences...
   The Preferences dialog appears.
2. Click the clipboard preferences icon to select clipboard preferences.
3. Select the current angle in the angle text edit box.
4. Type in the angle by which Argus ONE will subdivide arcs and circles.
5. Click the OK button.

The Preferences dialog.




Type in the required angle.

The clipboard preferences icon.

# Post-Processing Tools and Objects

With the introduction of scientific visualization tools, Argus ONE now allows you to perform all of your post-processing tasks in the same workplace you prepare your data for your model. This enables you to easily investigate your model output with respect to your model input.

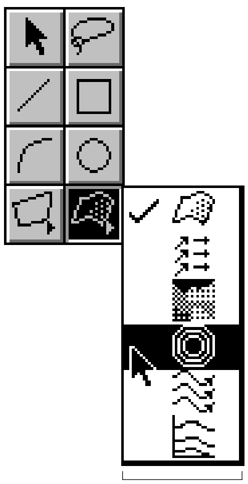
Post-Processing objects also enable you to visually study the data you assigned to meshes and grids before you export them to you numerical model. If for instance, you interpolated the distribution of a physical parameter stored in an information layer, onto your mesh nodes, and want to investigate the nodal distribution of that parameter, you can use the post-processing tools to visualize that distribution.

The six available visualization methods are:

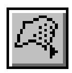
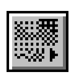
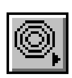
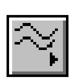
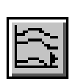
1. 3D-Surface
2. Vector Diagram
3. Color Diagram
4. Contour Diagram
5. Path-lines Diagram
6. Cross-Section Diagram

- Post-processing objects can only use data stored in data type layers.
- Post-processing objects can be moved and deleted as any other graphic object in maps layer.
- Post-processing objects are created and edited in maps type layers.

## Post-Processing Tools



*Post-Processing tools popup menu*

| Tool | Tool's name and function | Tool can be activated when |
|---|---|---|
| | The Visualization tools are grouped in a tool popup menu. To select a tool, click and hold the mouse button until the menu pops up. Select the required tool as you select a menu item by dragging the mouse onto it. | |
|  | The 3D Surface tool. Allows you to create a 3D surface visualization of scalar data stored in a data type layer. | There is at least one, non-empty data type layer. |
|  | - Or - The Vector tool. Allows you to create a vector diagram of two components of a vector field stored in a data type layer. | There is at least one, non-empty data type layer. |
|  | - Or - The Color Diagram tool. Allows you to create a color map visualization of scalar data stored in a data type layer. | There is at least one, non-empty data type layer. |
|  | - Or - The Contour Diagram tool. Allows you to create a contour map visualization of scalar data stored in a data type layer. | There is at least one, non-empty data type layer. |
|  | - Or - The Path-Line tool. Allows you to create a path-lines diagram of two components of a vector field stored in a data type layer. | There is at least one, non-empty data type layer. |
|  | - Or - The Cross-Section tool. Allows you to create a cross-section diagram of any number of data sets stored in a data type layer. | There is at least one, non-empty data type layer, and a graphic object such as an open polygon, a close polygon, or a line is selected. |

## Creating and Editing Post-Processing Objects

Visualizing your model results or the distribution of a parameter are as easy as stretching a rectangle to specify the plot size and location, and choosing the parameters to be displayed. All post-processing objects share common controls that allow you to specify their look, location and size.

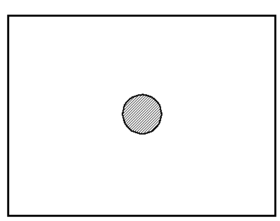
### An Example

The following pages presents you with all six visualization methods using a simple flow example. The example used is of a flow around a cylinder solved on a mesh. The solution was then read back into a Argus ONE data layer. The three variables solved for are the pressure field, and the X and Y velocity components.

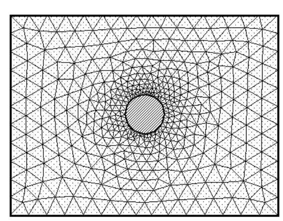
#### Steps performed

1. Defining the problem domain in a Argus ONE Domain layer.
2. Application of boundary conditions and other parameters using Information layers.
3. Auto mesh generation of the domain.
4. Linking parameters required by the model to the mesh.
5. Export the data to ASCII files in the format required by the model.
6. Run the model and import the resulting pressure distribution and two velocity components into a Argus ONE data layer.
   To find out how to import data into a Argus ONE data layer refer to “To import mesh data with triangulation information” on page 11 or to “To import mesh data without triangulation information” on page 12 earlier in this chapter.

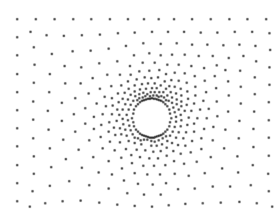
The following three screen shots represent the problem domain, the mesh, and the data points at which the solution was obtained.



*A cross section of a flow region around a cylinder.*



*The finite element mesh on which the solution was obtained.*



*The data points at which solution was obtained and imported into a data layer.*

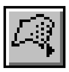
## Creating and Editing a 3D Surface

The 3D Surface post-processing object enables you to visualize the distribution of a scalar field in space, while letting you control the size, location, color, shading, view angle, and legend.

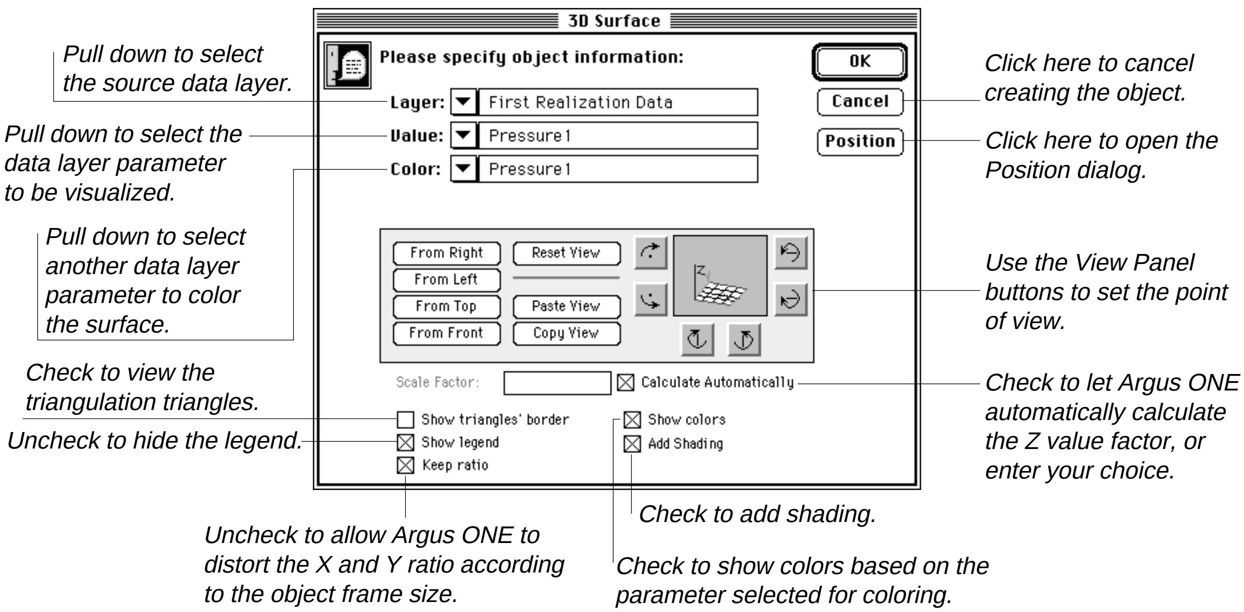
Argus ONE creates a 3D Surface by plotting the “Z” values at the vertices of the triangulation read from the data layer.

## To create a 3D Surface

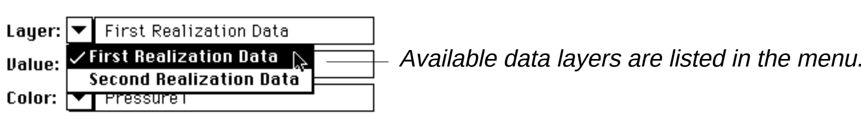
1. Make sure that there is at least one data layer into which a data set was read.
2. From the Layers menu select the maps layer in which you wish to create the post-processing object.



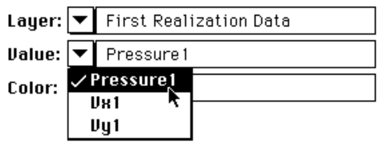
3. From the tool palette select the 3D Surface tool.
4. Click the cursor where you want the object to start and stretch a rectangle to where you want it to end.
   The 3D Surface dialog opens to let you specify additional parameters.




5. From the Layer popup menu select the data layer containing the parameter you wish to plot.
   All data layers into which you have read or imported information are listed in the menu.



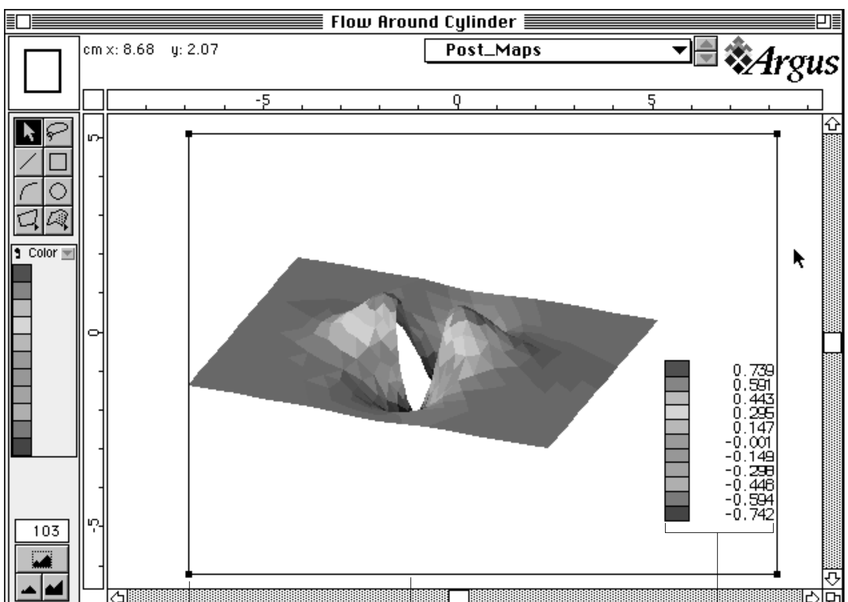
**6.** From the Value popup menu select the parameter to be visualized.
All the parameters that have been read or imported into the selected data layer are presented.




**7.** If you wish to use another parameter to color the surface select it from the Color popup menu. Again, all the parameters that have been read or imported into the selected data layer are presented.
This feature enables you to plot the intensity distribution of a parameter on top of the elevation distribution of another parameter. You could for instance use the concentration distribution as the coloring parameter, over the elevation distribution of a geological formation or the bathymetry of a lake.

**8.** Click the OK button.

*A 3D Surface post-processing object representing the pressure distribution in a flow filed around a cylinder.*

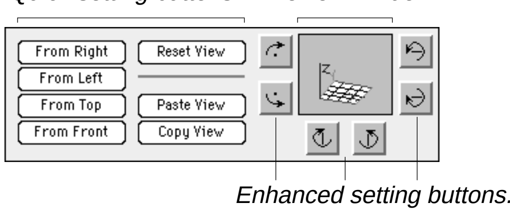



## To set your point of view

You can instruct Argus ONE to change the 3D Surface appearance by changing your point of view. You set your view by turning the object with respect to your view. The three available movements are:

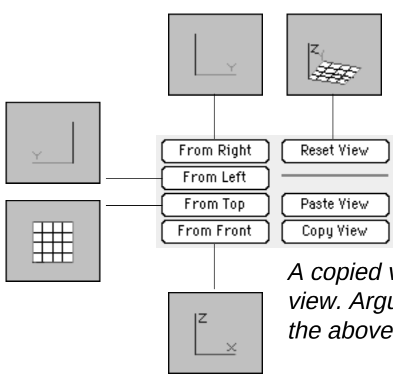
- Pitch - Turning the object around your horizon.
- Yaw - Turning the object around the vertical to your horizon.
- Roll - Turning the object around the line of sight (the line connecting your eye and the center of the object).

The View Panel is where you set your point of view and the object appearance. The View Panel contains quick setting buttons and the extended setting buttons. Using the quick setting buttons you can set your view to a number of predefined views, reset to the default view, and copy and paste views. Using the enhanced setting buttons you have full control over the view using the three movements buttons. At all times the preview window presents you with the selected view.




### The quick setting buttons

- From Right - The object’s X coordinate is aligned with the line of sight, looking at the right face of the object.
- From Left - The object’s X coordinate is aligned with the line of sight, looking at the left face of the object.
- From Top - The object’s Z coordinate is aligned with the line of sight, looking at the top face of the object.
- From Front - The object’s Y coordinate is aligned with the line of sight, looking at the front face of the object.




*The graphical and textual representations of the Default view.*

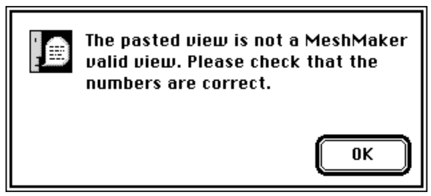
| | | | |
|---|---|---|---|
| 0.925417 | -0.163176 | 0.34202 | 0.0 |
| 0.378522 | 0.44097 | -0.813798 | 0.0 |
| -0.0180283 | 0.882564 | 0.469846 | 0.0 |
| 0.0 | 0.0 | 0.0 | 1.0 |

*A copied view can be pasted in any 3D surface object view. Argus ONE copies the view into the clipboard in the above text (tab delimited format).*

## Copying and pasting views

You can use the copy and paste view buttons to set the view of a number of 3D surface objects to be identical. When you copy a view Argus ONE copies the current Rotation Matrix into the clipboard as a tab delimited text.

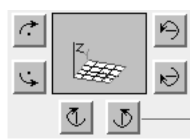
You can edit the rotation matrix yourself in a text editor and paste it in the 3D Surface dialog. If the rotation matrix is not "distance-conserving"
Argus ONE presents you with the following dialog and does not change the current view.




## The enhanced setting buttons

You can also fully control the object's view using the three sets of Pitch, Yaw and Roll buttons.

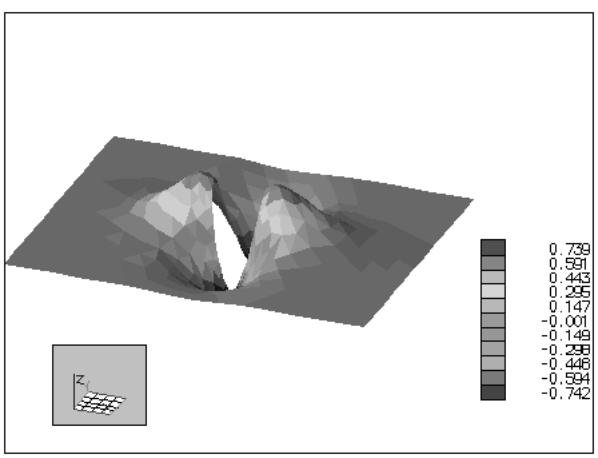
Roll buttons.



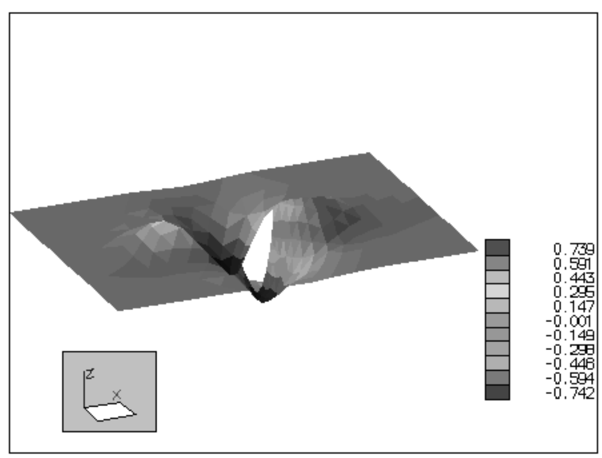
Pitch buttons.

Yaw buttons.

With these buttons you can set your view to the exact required one. For instance, if you want to set the view to "From Bottom", just turn the Pitch buttons to the required angle.




*Pressure field viewed in default view.*

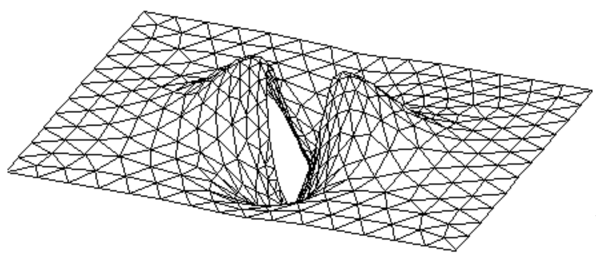



*Viewed using the default Yaw and Roll while changing the Pitch to view the surface from underneath.*

## Controlling other parameters

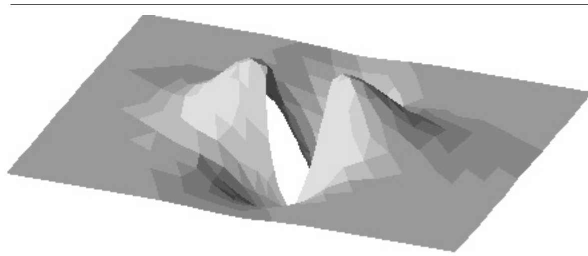
You can also control the following parameters:

- Show/Hide triangulation - View wire-frame, with or without Color and Shading.
- Show/Hide legend.
- Show/Hide Colors - Triangles coloring according to the selected coloring parameter.
- Add/Remove shading.
- Keep/Ignore ratio - When keep ratio is off the objects frame horizontal/vertical ratio overrides the coordinate system ratio.
- Scale Factor - A factor multiplying the “Z” value.



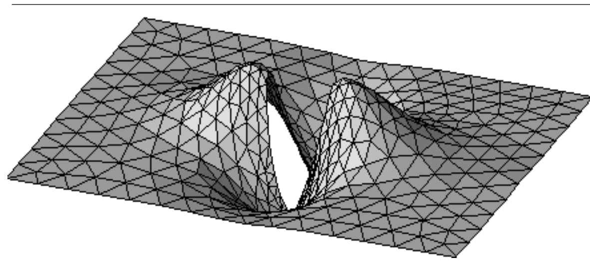
*Checked options:*
*Triangle Borders*
*Keep ratio*
*Scale Factor*

*Unchecked options:*
*Show colors*
*Add Shading*
*Show legend*



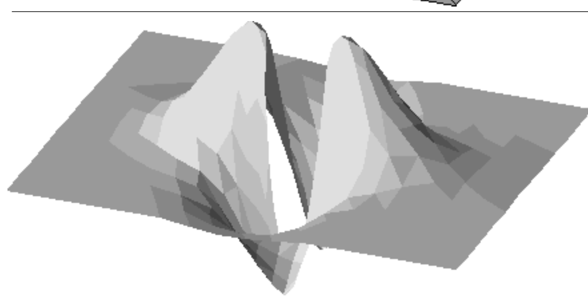
*Checked options:*
*Add Shading*
*Keep ratio*
*Scale Factor*

*Unchecked options:*
*Triangles’ borders*
*Show colors*
*Show legend*



*Checked options:*
*Triangle Borders*
*Add Shading*
*Keep ratio*
*Scale Factor*

*Unchecked options:*
*Show colors*
*Show legend*



*Checked options:*
*Add Shading*
*Keep ratio*

*Unchecked options:*
*Triangles’ borders*
*Show colors*
*Show legend*
*Scale Factor=5*

## Creating and Editing a Vector Diagram

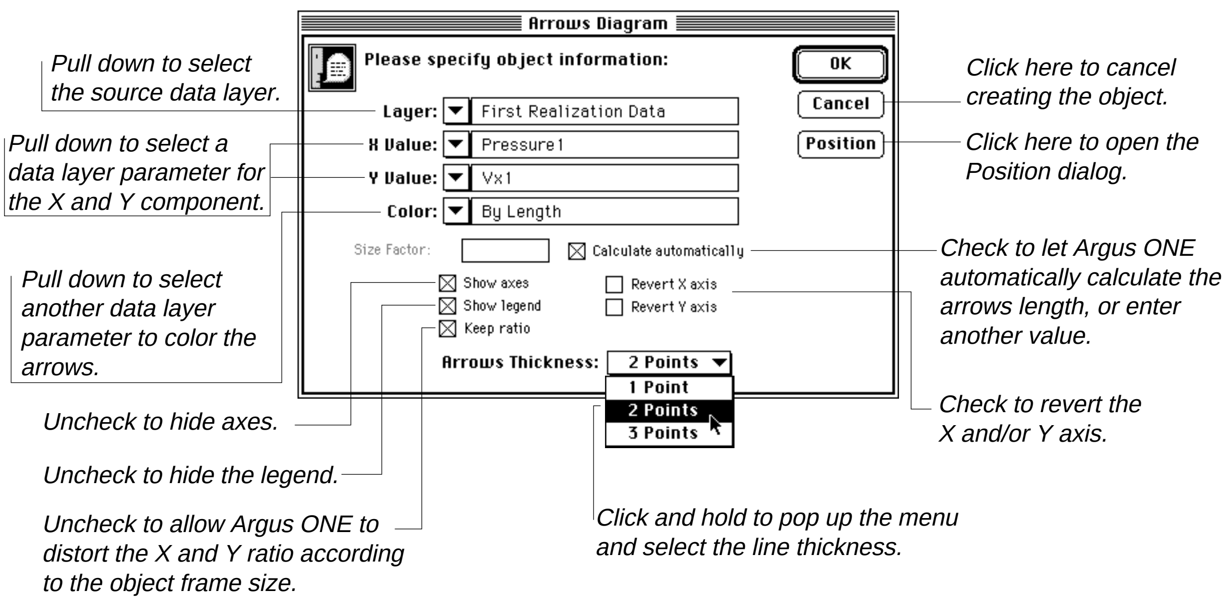
The vector diagram post-processing object enables you to visualize the a vector field in a plane, while letting you control the size, location, color, line thickness, axes, and legend.

Argus ONE creates a vector diagram by plotting an arrow of length $\sqrt{Vx^2 + Vy^2}$ at an $\tan\frac{Vy}{Vx}$ angle at the vertices of the triangulation read from the data layer.

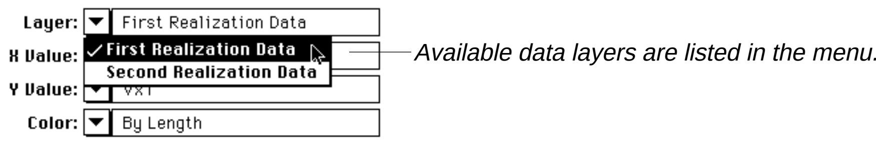
### To create a Vector Diagram

1. Make sure you have at least one data layer into which a data set was read. Also the data should have at least two parameters representing the Vx and Vy components.
2. From the Layers menu select the maps layer in which you wish to create the post-processing object.

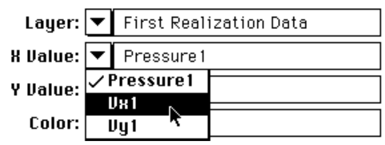
3. From the tool palette select the vector diagram tool.
4. Click the cursor where you want the object to start and stretch a rectangle to where you want it to end.
   The Vector Diagram dialog opens to let you specify additional parameters.



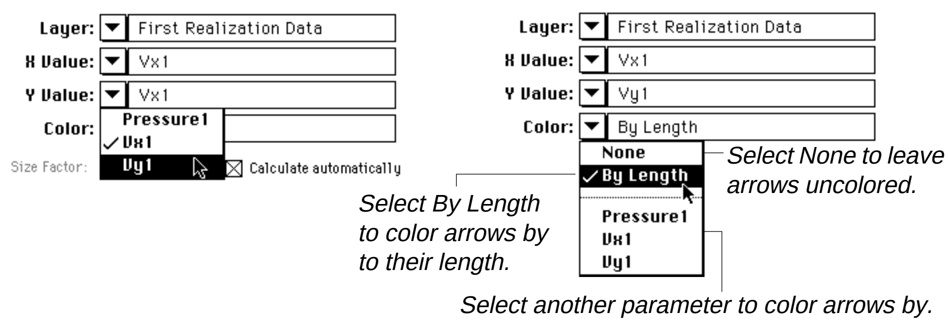

5. From the Layer popup menu select the data layer containing the parameters you wish to plot.
   All data layers into which you have read or imported information are listed in the menu.




6. From the X Value popup menu select the parameter for the X component. All the parameters that have been read or imported into the selected data layer are presented.



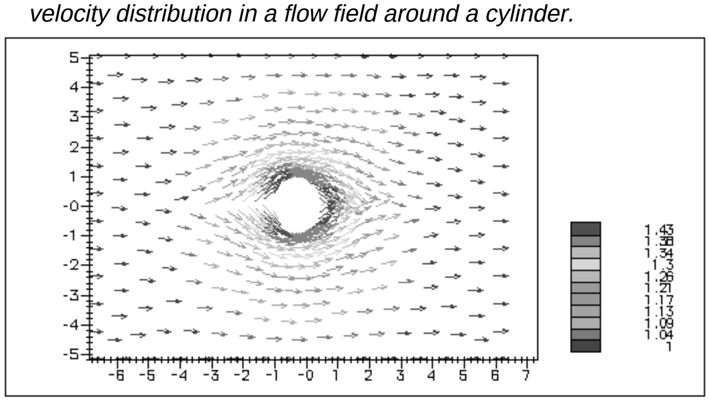

7. From the Y Value popup menu select the parameter for the Y component. All the parameters that have been read or imported into the selected data layer are presented.




8. By default Argus ONE colors the arrows according to their length which represents their magnitude. If you wish to use another parameter to color the arrows select it from the Color popup menu. You can also choose not to color the arrows by choosing None. Again, all the parameters that have been read or imported into the selected data layer are presented.

9. Click the OK button.

*An Arrow Diagram post-processing object representing the velocity distribution in a flow field around a cylinder.*



### Controlling other parameters

You can also control the following parameters:

- Show/Hide axes.
- Show/Hide legend.
- Keep/Ignore ratio - When keep ratio is off the objects frame horizontal/vertical ratio overrides the coordinate system ratio.
- Revert the X axis.
- Revert the Y axis.
- The arrow thickness.
- Scale Factor - A number by which the $\sqrt{Vx^2 + Vy^2}$ magnitude is multiplied to change the arrows' length.

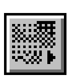
## Creating and Editing a Color Diagram

The color diagram post-processing object enables you to visualize the intensity distribution of a parameter in the plane, while letting you control the size, location, axes, and legend.

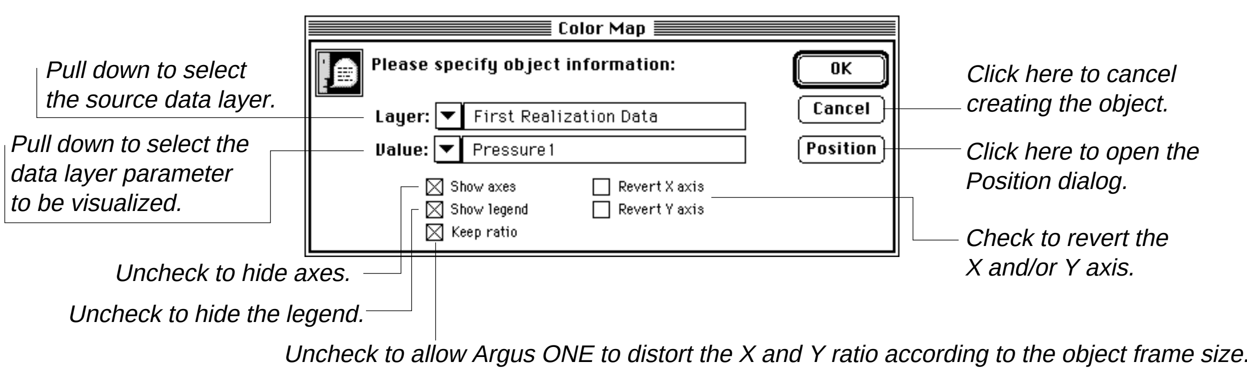
Argus ONE creates a color diagram by coloring each triangle in the triangulation read from the data layer according to the average value at its three vertices.

### To create a Color Diagram

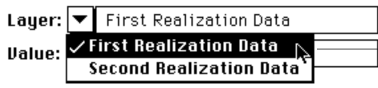
1. Make sure you have at least one data layer into which a data set was read.
2. From the Layers menu select the maps layer in which you wish to create the post-processing object.



3. From the tool palette select the color diagram tool.
4. Click the cursor where you want the object to start and stretch a rectangle to where you want it to end.
   The Color Diagram dialog opens to let you specify additional parameters.

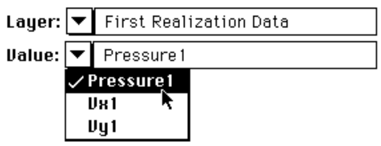


**5.** From the Layer popup menu select the data layer containing the parameters you wish to plot.
All data layers into which you have read or imported information are listed in the menu.



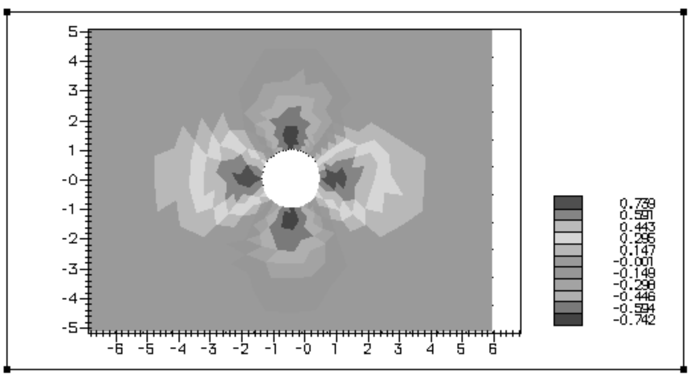

*Available data layers are listed in the menu.*

**6.** From the Value popup menu select the parameter its distribution you wish to plot.
All the parameters that have been read or imported into the selected data layer are presented.




**7.** Click the OK button.

*A Color Diagram post-processing object representing the pressure distribution in a flow field around a cylinder.*



## Controlling other parameters

You can also control the following parameters:

- Show/Hide axes.
- Show/Hide legend.
- Keep/Ignore ratio - When keep ratio is off the objects frame horizontal/vertical ratio overrides the coordinate system ratio.
- Revert the X axis.
- Revert the Y axis.

## Creating and Editing a Contour Diagram

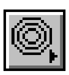
The Contour diagram post-processing object enables you to contour the intensity distribution of a parameter in a plane, while letting you control the size, location, minimum, maximum, contour value, axes, and legend.

Argus ONE creates a contour diagram by performing the following sequence of operations:

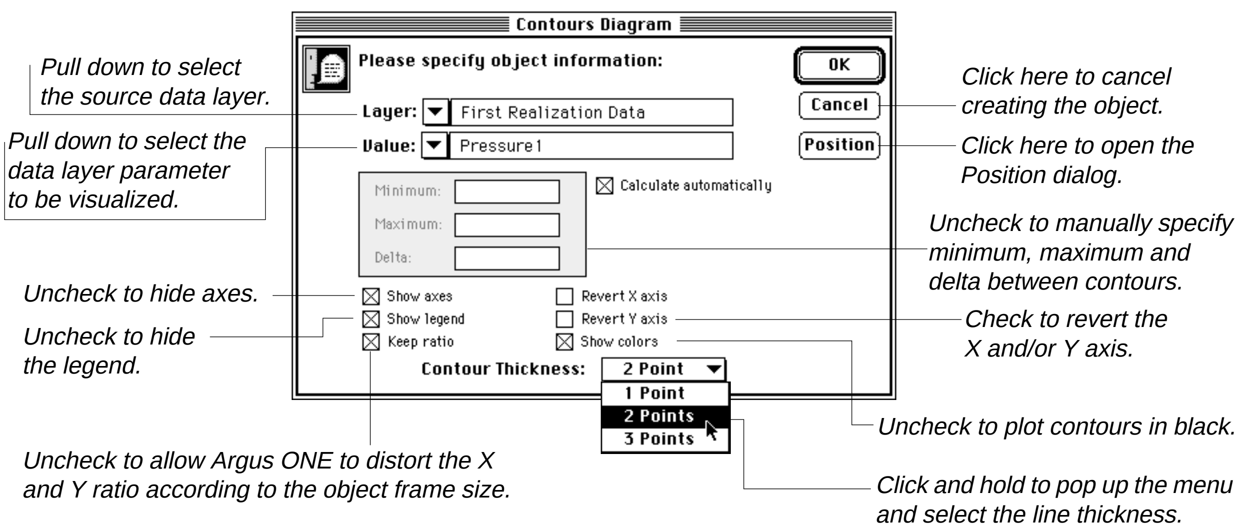
1. It automatically calculates, or reads your specifications for the minimum, and maximum values, and the interval at which contours should be plotted.
2. From the above information it calculates the contour values that should be plotted.
3. It then loops over the triangulation triangles (read from the source data layer) to find which contours intersect each of the triangles and records the intersection points. Finding the intersection points is performed using a linear interpolation scheme along the triangulation triangles' sides.

### To create a Contour Diagram

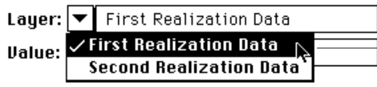
1. Make sure you have at least one data layer into which a data set was read.
2. From the Layers menu select the maps layer in which you wish to create the post-processing object.



3. From the tool palette select the contour diagram tool.
4. Click the cursor where you want the object to start and stretch a rectangle to where you want it to end.
   The Contour Diagram dialog opens to let you specify additional parameters.

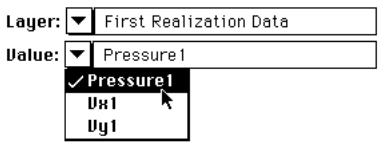



5. From the Layer popup menu select the data layer containing the parameters you wish to plot.All data layers into which you have read or imported information are listed in the menu.



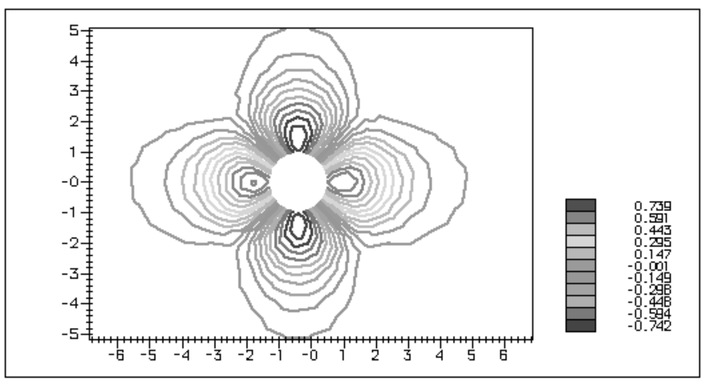

*Available data layers are listed in the menu.*

**6.** From the Value popup menu select the parameter its distribution you wish to plot.
All the parameters that have been read or imported into the selected data layer are presented.




**7.** Click the OK button.

*A Contour Diagram post-processing object representing the pressure distribution in a flow field around a cylinder.*



## Controlling other parameters

You can also control the following parameters:

- Show/Hide axes.
- Show/Hide legend.
- Keep/Ignore ratio - When keep ratio is off the objects frame horizontal/ vertical ratio overrides the coordinate system ratio.
- Revert the X axis.
- Revert the Y axis.
- Set the contour line thickness.
- Turn off contour coloring.
- Set the minimum and maximum values to be contoured.
- Set the contours’ interval.

## Creating and Editing a Path-Line Diagram

The path-line diagram post-processing object enables you to visualize particle path-lines, while letting you control the size, location, color, line thickness, axes, and legend.

To create a particle path line diagram you specify the X and Y velocity components, the type of tracking, the intervals at which Argus ONE will seed path lines start points, and the size factor. Argus ONE then performs the following sequence for each start point:

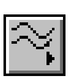
1. Creates path lines start points at the specified intervals on the data boundaries.
2. At each start point it interpolates the Vx and Vy components at that point, from the data in the source data layer.
3. Using Vx and Vy at that point, the magnitude of the velocity is calculated as $\sqrt{Vx^2 + Vy^2}$ and the angle as $\tan\frac{Vy}{Vx}$ .
4. It then plots a line segment of length $SizeFactor \times \sqrt{Vx^2 + Vy^2}$ in the direction $\tan\frac{Vy}{Vx}$ .
5. Steps 2 through 4 are repeated at the end of each new line segment created, until the last line segment crosses the data boundaries or until maximum number of iterations is exceeded.

If back-tracking is selected, the line segments are plotted from a point at - $\tan\frac{Vy}{Vx}$ .

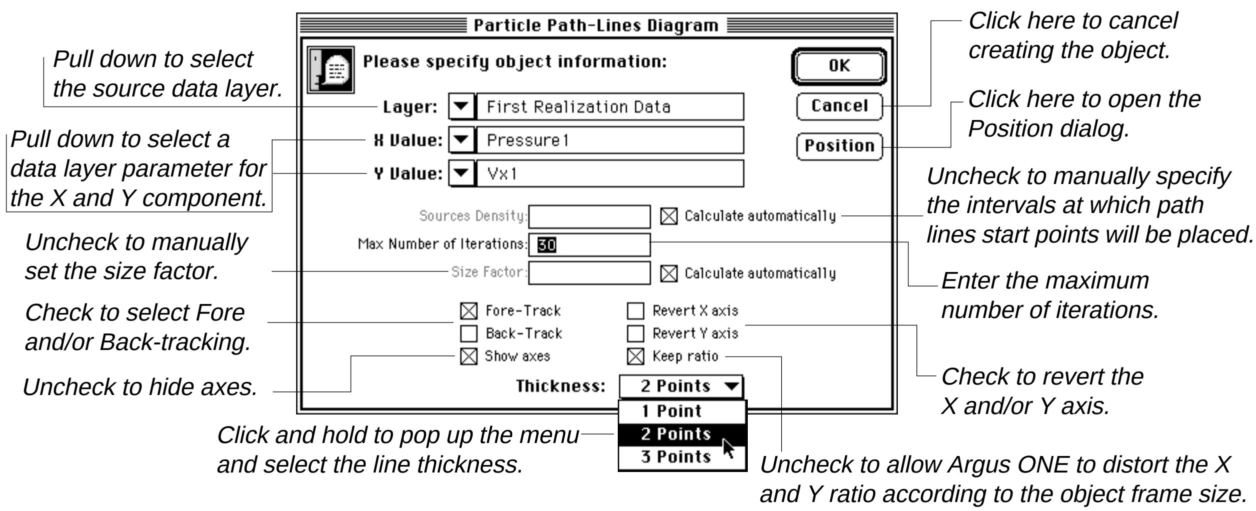
If both fore-tracking and back-tracking are selected, two lines segments are created at $\tan\frac{Vy}{Vx}$ and - $\tan\frac{Vy}{Vx}$ angles.

### To create a Path-Line Diagram

1. Make sure you have at least one data layer into which a data set was read. Also, the data should have at least two parameters representing the Vx and Vy components.
2. From the Layers menu select the maps layer in which you wish to create the post-processing object.

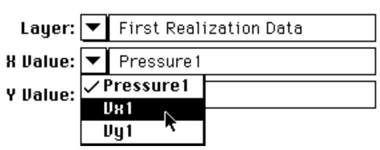


3. From the tool palette select the path-line diagram tool.
4. Click the cursor where you want the object to start and stretch a rectangle to where you want it to end.
   The Path-Line Diagram dialog opens to let you specify additional parameters.

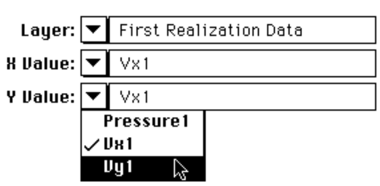



**5.** From the Layer popup menu select the data layer containing the parameters you wish to plot.
All data layers into which you have read or imported information are listed in the menu.


**6.** From the X Value popup menu select the parameter for the X velocity component.
All the parameters that have been read or imported into the selected data layer are presented.

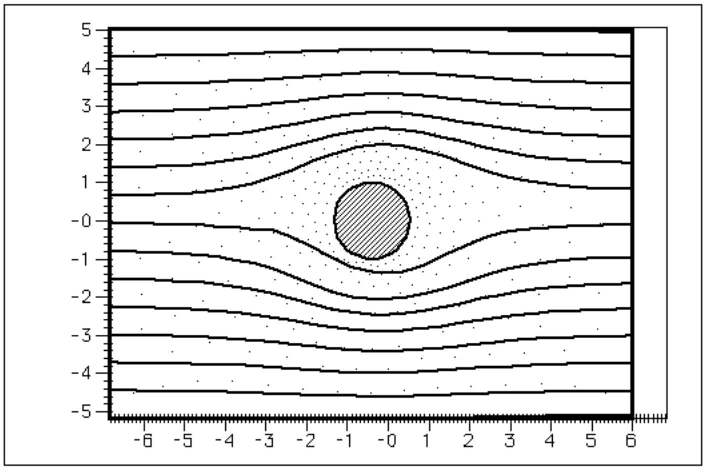



**7.** From the Y Value popup menu select the parameter for the Y velocity component.
All the parameters that have been read or imported into the selected data layer are presented.



**8.** Click the OK button.

*An Path-line Diagram post-processing object representing particle path lines in a flow field around a cylinder.*




## Controlling other parameters

You can also control the following parameters:

- Show/Hide axes.
- Keep/Ignore ratio - When keep ratio is off the objects frame horizontal/ vertical ratio overrides the coordinate system ratio.
- Revert the X axis.
- Revert the Y axis.
- The line thickness.
- Fore-Tracking.
- Back-Tracking.
- Scale Factor - A number by which the $\sqrt{Vx^2 + Vy^2}$ magnitude is multiplied. $SizeFactor \times \sqrt{Vx^2 + Vy^2}$ defines the length of line segments making up the path line.
- Maximum number of iterations.
- Source density - The intervals, in current ruler units, at which path lines start points will be created on data boundaries.

## Limitations

If the flow does not intersect the data boundaries no path lines will be created. Since path lines start points are only created on the data boundaries, path lines can not be created if the flow does not cross these boundaries.

## Creating and Editing a Cross-Section Diagram

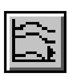
The cross-section diagram post-processing object enables you to visualize the distribution of a parameter or parameters along a cross-section, while letting you control the size, location, color, axes, and legend.

To create a cross-section diagram you must first select the cross-section polygon and then specify the parameters you wish to plot. Argus ONE then performs the following sequence for each of the parameters you select:

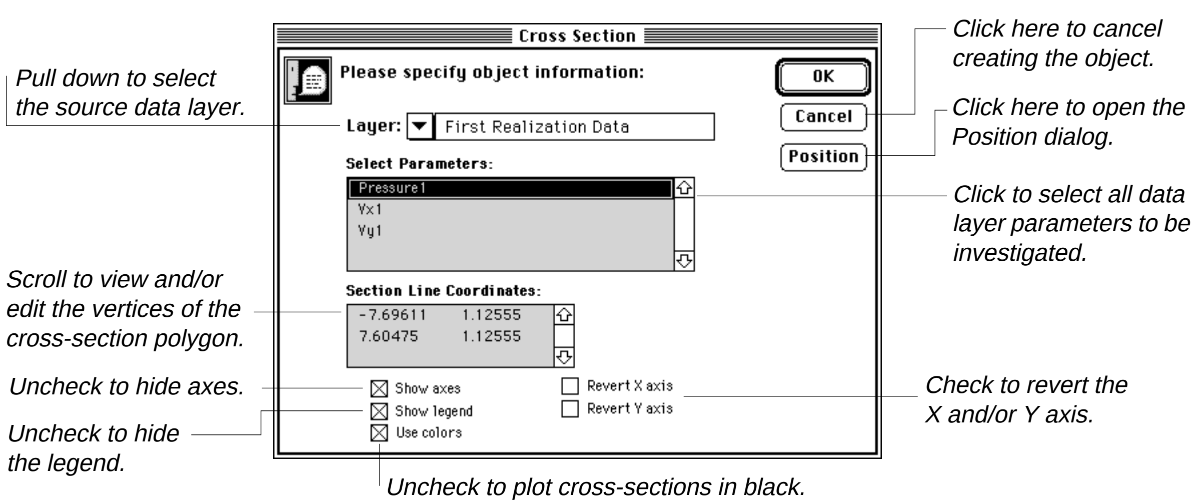
1. It finds the intersections of the cross-section polygon with the triangulation.
2. At each intersection it interpolates the value of the parameters investigated.
3. It then plots these values while connecting them by straight line segments.

### To create a Cross-Section Diagram

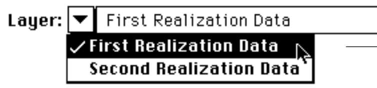
1. Make sure you have at least one data layer into which a data set was read.
2. From the Layers menu select the maps layer in which you wish to create the post-processing object.
3. Select a graphic object to serve as the cross-section polygon.



4. From the tool palette select the cross-section diagram tool.
   If a graphic object is not selected then you can not choose the cross-section tool from the palette.
5. Click the cursor where you want the object to start and stretch a rectangle to where you want it to end.
   The Cross-Section Diagram dialog opens to let you specify additional parameters.

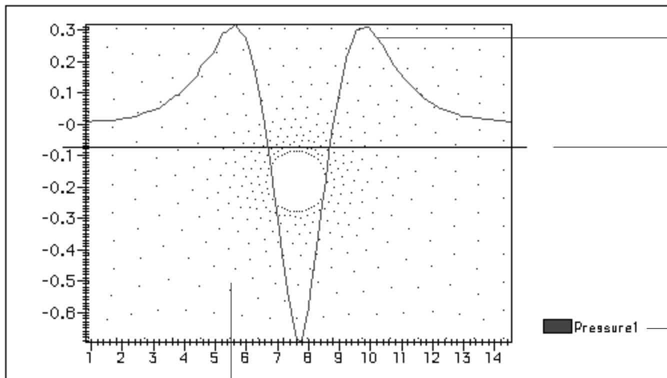



6. From the Layer popup menu select the data layer containing the parameters you wish to plot. All data layers into which you have read or imported information are listed in the menu.




7. Click the OK button.

A Cross-Section Diagram post-processing object representing the pressure distribution along a section line in a flow field around a cylinder.




### Identifying the cross-section object and its cross-section polygon

When you select a cross-section object, it highlights an image of the cross-section line it is defined for. Note that after a cross-section object is created it is detached from the polygon you used to define it. Moving that polygon does not change the cross-section diagram.

### Controlling other parameters

You can also control the following parameters:

- Add parameters to be plotted.
- Edit the cross section polygon.
- Show/Hide axes.
- Revert the X axis.
- Revert the Y axis.
- Line Color - assisting in identifying the various parameters plotted.

### Using graphic objects as cross-section polygons

You can use any of the graphic objects as a cross-section polygon. When you use the arc or circle objects Argus ONE translates them into polygons as it does when you copy such objects. You can control the angle and hence, the segment length, into which Argus ONE will subdivide these objects. To read more about the subdivision of arcs and circles refer to “Setting the segment length for copying arcs and circles” on page 22 in this chapter.

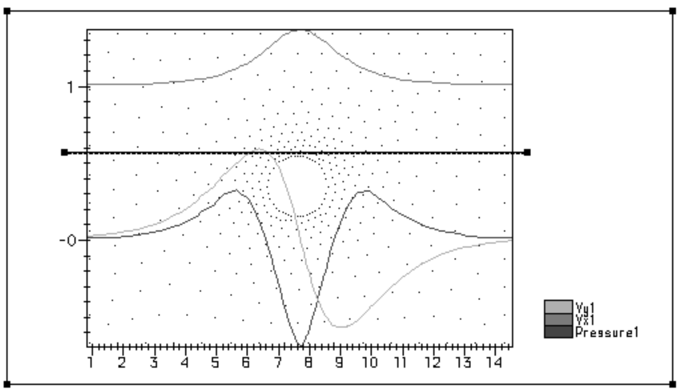
## Editing Post-Processing Objects

Since all post-processing plots are real objects you can edit them at any time.

### To edit a post-processing object

1. Select the object by clicking on it.
2. From the Navigation menu select Object Info...
   - Or -
   Double-click the object.
   The object's dialog opens to allow you to change the object appearance, location and any other parameter.
3. When you click the OK button, the objects reads the associated data layer and recalculates itself.

*The Cross-Section Diagram on the previous page changed to present the velocity components as well as the pressure distribution.*




Note that only if you re-open a post-processing object it will re-read the source data layer. If you changed the source data layer all post-processing objects associated with that layer will not change until you open their dialogs and confirm the changes.

### To delete a post-processing object

1. Select the object by clicking on it.
2. Hit the delete key on your keyboard.
   The object is deleted.

If you accidently deleted a post-processing object use the Undo command from the Edit menu to undelete it.

## Arranging Post-Processing Objects

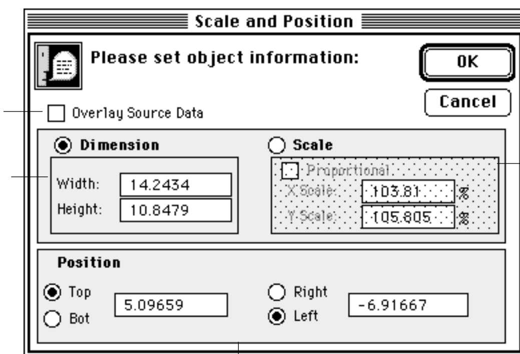
Since all post-processing plots are real objects you can always move them around the project as you can move any other object. You can also use the position dialog to position and size your post-processing objects.

## To position and size post-processing objects

1. Select the object by clicking on it.
2. From the Navigation menu select Object Info...
   - Or - Double-click the object.
   The object's dialog opens to allow you to change the object appearance, location and any other parameter.
3. Click the Position button.
   The Position Dialog box opens.

*Check to overlay the object and source data enclosing rectangle.*

*Specify the object's dimensions.*

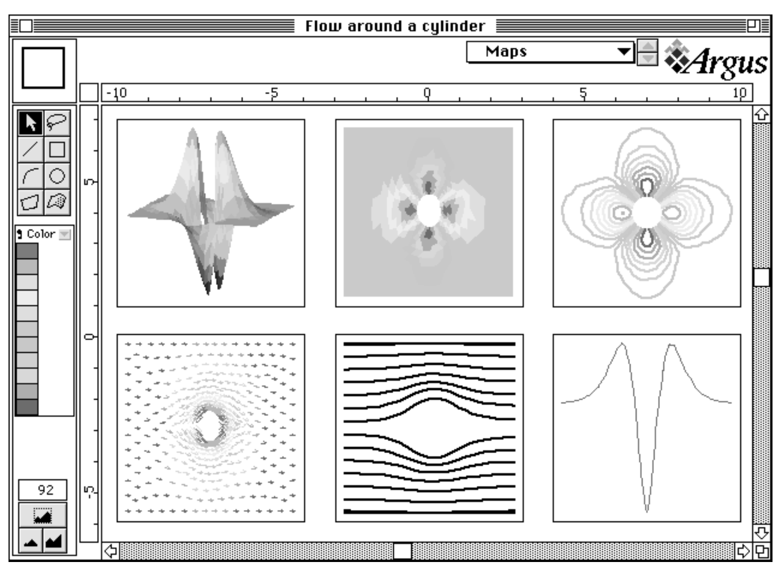



*Scale the object with respect to the source data enclosing rectangle.*

*Specify the object's frame top or bottom and left or right borders.*

The Overlay Source Data option is not available for the 3D-Surface and Cross-Section post-processing objects.

Moving post-processing objects by click and drag, and resizing and re-positioning them using the position dialog enables you to easily arrange your objects for presentations or reports.



# New Functions (the following functions are new additions to the functions described in chapter 3)

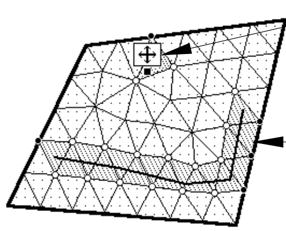
A number of new functions that have been added in version 2.5 are listed here by category.

## New Element Functions

The element functions that have been added are analogues to the block functions **CountObjectsInBlock**, **SumObjectsInBlock** and **WSumObjectsInBlock.**

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **CountObjectsInElement(** ***layer_name,*** *contour_type)* | ***layer_name*** | The name of the layer for which you need to count the number of objects of type *contour_type* intersecting the element or contained in it. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the number of objects of type *contour_type* in layer ***layer_name***, intersecting or contained in the element. If *contour_type* is omitted, returns the total number of all objects intersecting or contained in the element. |

Example: The function `CountObjectsInElement(Domain Layer)` counts the number of objects in the layer "Domain Layer" intersecting or contained by the element and returns:
0 — if the element is not intersected by, or contains no objects.
1 — if the element is intersected by, or contains one object of any type.
N — if the element is intersected by, or contains N objects of any type.

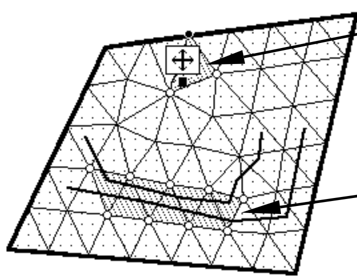


*This highlighted element on top of the point object will return the value 1 for the function*
`CountObjectsInElement(Domain Layer,0)`

*These highlighted elements along the open contour will return the value 1 for the function*
`CountObjectsInElement(Domain Layer,1)`

Example: The function `CountObjectsInElement(Domain Layer,1)` counts the number of open contours in the layer "Domain Layer" intersecting or contained by the element and returns:
0 — if the element is not intersected by, or contains no open contours.
1 — if the element is intersected by, or contains one open contour.
N — if the element is intersected by, or contains N open contours.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **SumObjectsInElement(*parameter_name,*** *contour_type*) | ***parameter_name*** | The name of an information type layer parameter for which you need to sum the values of objects of type *contour_type* intersecting the element or contained in it. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the sum of the objects' values of type *contour_type* in parameter ***parameter_name***, intersecting or contained in the element. If *contour_type* is omitted, returns the sum of values of all objects intersecting or contained in the element. |



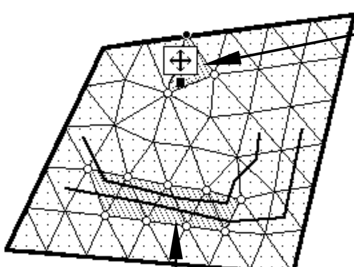
*If contour_type is 0 or if it is omitted and a point object is encountered,* `SumObjectsInElement` *returns the sum of values of the point objects.*

*In these highlighted elements the function* `SumObjectsInElement(Fractures Layer,1)` *returns the sum of the two contour values.*

Example: The function `SumObjectsInElement(Well Layer,0)` can be used to sum the values of the specified variable at the point objects which are in the layer "Well Layer" and that are contained by the element. If divided by `CountObjectsInElement(Well Layer,0)` it shall return the average value of the specified function.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **WSumObjectsInElement*(* *parameter_name,*** *contour_type* ***)*** | ***parameter_name*** | The name of an information type layer parameter in which you need to calculate the weighted sum of objects of type *contour_type* intersecting the element or contained in it. The weighted sum is defined as the sum of the products of each intersecting segment length, multiplied by its value. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the weighted sum of objects' values of type *contour_type* in parameter ***parameter_name***, intersecting or contained in the element. If *contour_type* is omitted, returns the weighted sum of values of all objects intersecting or contained in the element. |

Example: The function `WSumObjectsInElement(Fractures Layer,1)` computes the product of each intersecting or contained segment length by its value and sums them.

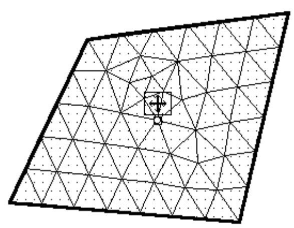


*If contour_type is 0 or if it is omitted and a point object is encountered,* `WSumObjectsInElement` *returns the value of the point object.*

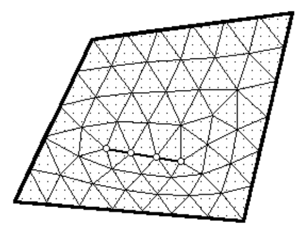
*In these highlighted elements the function* `WSumObjectsInElement(Fractures Layer,1)` *returns the sum of the two products of the length of intersecting segments multiplied by their contour value.*

## New Node Functions

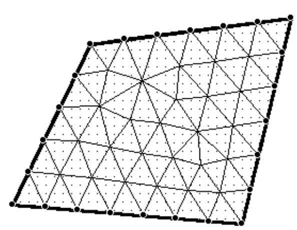
| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NodeAboveCntr*( layer_name)*** | ***layer_name*** | The name of the layer in which you want to check if a node lies above any contour object.<br><br>Returns the type of object the node lies above or within *layer_name*:<br>0 - If the node does not lie above or within any contour.<br>1 - If the node lies above a point object.<br>2 - If the node lies above an open contour.<br>3 - If the node lies above a close contour.<br>4 - If the node lies inside a close contour. |



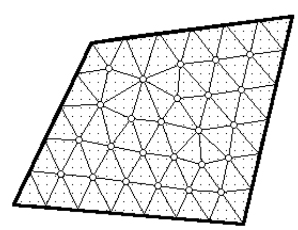
`NodeAboveCntr(Domain)` *returns 1 for the node above the point object.*



`NodeAboveCntr(Domain)` *returns 2 for nodes above the open contour.*



`NodeAboveCntr(Domain)` *returns 3 for all nodes above the close contour.*



`NodeAboveCntr(Domain)` *returns 4 for all nodes inside the close contour.*

Example: The function `if(NodeAboveCntr(BC)=1,BC,DefaultValue(BC))` returns the value of the boundary condition assigned to the point object that lies below a node and if the node does not lie above a point object, the default value of the BC layer. This is useful for allocating boundary conditions types and values to node elements.

## Action Taking Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **DefaultValue*(expression (Information Layer parameter))*** | ***expression*** | An expression containing any number of Information type layer parameters.<br><br>Returns the default value of all Information type layer parameters contained in the expression. |

Example: `DefaultValue(Rainfall*Concentration)` returns the default value of the information layer parameter “Rainfall” multiplied by the default value of the information layer parameter “Concentration” disregarding any value assigned to contours in these layers (the default value of a parameter is set in the Layers Dialog.

An expression can hold as many information layer parameters as needed, all parameters’ values will be set to default before the expression is evaluated.

## Logical Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **IsNA*(expression (Information Layer parameter))*** | ***expression*** | An expression containing any number of Information type layer parameters.<br><br>Returns True (1) if expression evaluates to $n/a and False (0) if expression does not evaluate to $n/a. |

Example: `isNA(if(NodeOnBoundary()|NodeAboveCntr(BCType)=1,BCType,$n/a)).` The function isNA returns True (1) if the if statement evaluates to false and returns $n/a, and False (0) if the if statement evaluates to true and returns BCType which is not equal to $n/a. This is useful for creating Export Template Script which will export only nodes with boundary conditions values.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **IsNAN*(expression (Information Layer parameter))*** | ***expression*** | An expression containing any number of Information type layer parameters.<br><br>Returns True (1) if expression evaluates to $NaN and False (0) if expression does not evaluate to $NaN. |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **IsNumber*(expression (Information Layer parameter))*** | ***expression*** | An expression containing any number of Information type layer parameters.<br><br>Returns True (1) if expression evaluates to a number and False (0) if expression evaluates to $n/a or $NaN. |

Example: These two functions can be used to test expressions returning illegal numbers when evaluated. Nest any expression within one of these functions and search the mesh or grid for nodes, elements or blocks that are NaN or Numbers.

# Quadrilateral Mesh Layer and Additional Mesh Layers Tools

## New Quadrilateral Element Mesh Layer

The Argus Numerical Environments now support a quadrilateral element mesh type. The new element type is supported in the Quad Mesh layer which is available as a separate module, or as a combined triangular and quadrilateral module.

Automatic mesh generation, manual creation, editing, exporting and all other mesh functions are identical to those used in the triangular element module and described in detail in the User's Guide Part 3, "Finite Element Meshes".

### The Quadrilateral Auto Mesh Generation Engine

The quadrilateral AMG Engine is a two stage engine. At first, a triangular mesh is created and then the triangular mesh is transformed into a quadrilateral mesh. Argus chose to first implement this quadrilateral AMG because it enables you to mesh a domain of any shape and complexity. The resulting element shapes and other characteristics, such as domain shape similarity, may differ from classic "transformation type" meshing engines.

### Differences Between Triangular and Quadrilateral Meshes

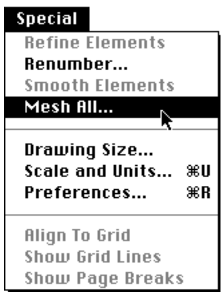
Working with quadrilateral elements only differs from working with triangular ones as a result of the quadrilateral's element additional node.

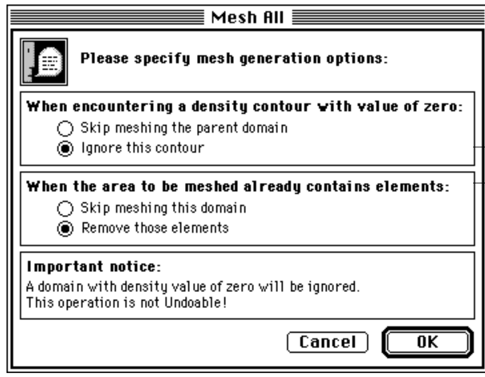
- The manual element creation tool has quadrilateral shape.
- While manually creating a quadrilateral element, you can use the delete key to remove the last recorded node.
- Maximal angle can be set in the Preferences dialog.
- Element functions that require the number of the nth element's side or node can refer to the forth side or node.
- While exporting the element-node connectivity list, four nodes are exported.
- The file format of all imported files which are quadrilateral element related, includes additional line or column to describe the fourth node.
- The information ruler and Element Info dialog list the additional node.

## New Tool for Meshing Multiple Domains

If you need to mesh a number of domains in one mesh layer, you do not need to mesh them sequentially by clicking the magic wand in each of them. You can now use the new Mesh All command under the Special menu.

When you choose this command Argus ONE brings up the following dialog asking you how to treat special cases while it is meshing the multiple domains.

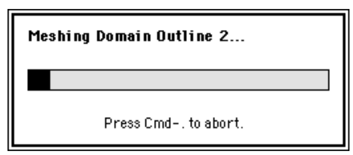



*Instruct Argus ONE what to do when one of the domain outlines to be meshed lies above a density contour assigned a zero value.*

*Instruct Argus ONE what to do when one of the domain outlines to be meshed will create elements, overlapping existing elements.*

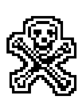
While meshing the domains Argus ONE informs you of the domain it is currently meshing by showing the progress dialog box of each of the domains as they are meshed.

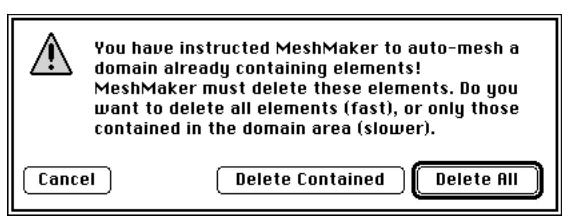



## Re-Meshing an Already Meshed Domain

When you click the Magic Wand in a mesh layer, Argus ONE can test the domain to be meshed against the existing elements in the mesh to check if auto mesh generation is about to create overlapping elements. If it does find such elements it alerts you and allows you to cancel. Testing the domain you clicked the Magic Wand in, against all elements is very time consuming. If you are certain that you want to delete all elements in the mesh layer you can select the Delete All button to avoid these tests.

*All node and element manual overridden values are deleted as well.*






- Click the Delete All button to delete all elements in the active mesh layer.
- Click the Delete Contained button to allow Argus ONE to delete overlapping elements. Argus ONE deletes only overlapping elements.
- Click the Cancel button to remain with the current mesh.

# New Contours Capabilities

To allow you to customize your Argus ONE workplace you can now also assign icons to contours. You can also search for contours based on the icon assigned to them.

## Assigning icons to contours

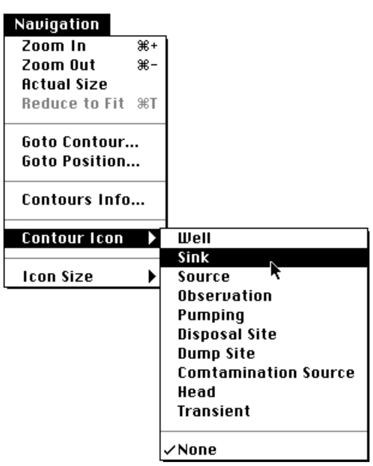
You can now assign icons to contours as you can assign them to elements, nodes and blocks. You can also set a search criteria based on the icons the contours are assigned.

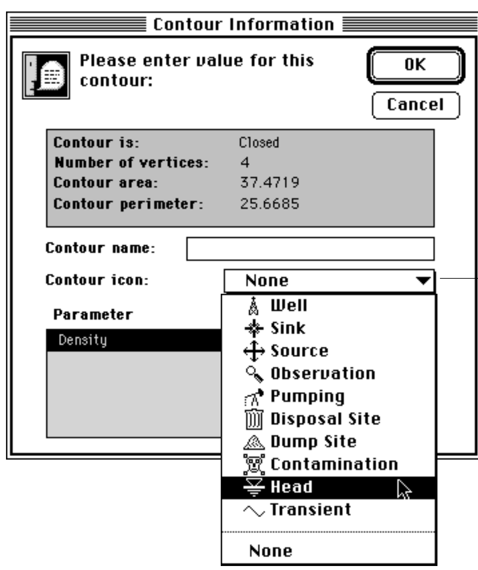
**To assign an icon to a contour** (this page should be placed after page 98 in the User's Guide)

1. Double-Click the contour,
   - Or -
   Select it, and from the Navigation menu select the Contours Info menu item. The Contour Information dialog box opens.
2. In the dialog click and hold the Icon popup menu to select the required icon.

Or

1. Click the contour to select it.
2. From the Navigation menu select the Contour Icon popup menu and choose the required icon.



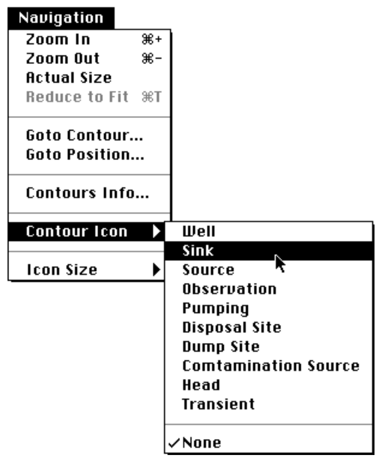




*Click the icon popup menu to choose an icon.*

## To assign an icon to a contour or to a selection of contours

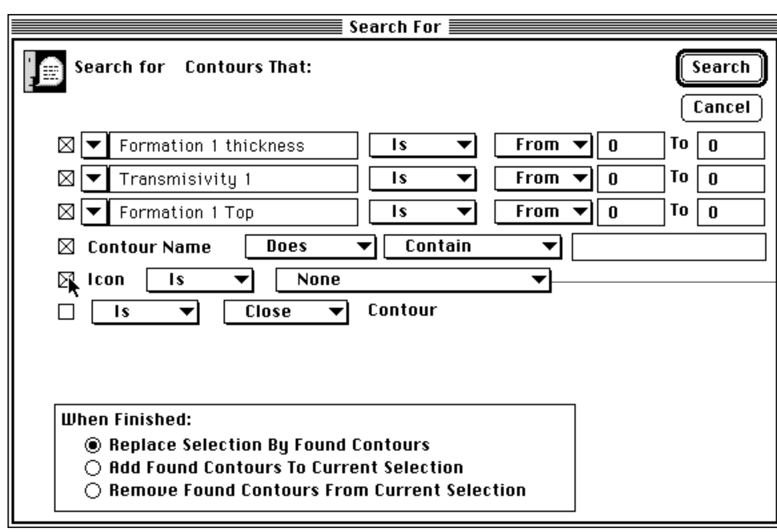
You can also assign an icon to a selected contour or a group of contours by using the Contour Icon menu item under the Navigation menu.

1. Use the stretch rectangle, the lasso tool or shift-click to add contours to the selection.
2. From the Navigation menu select the Contour Icon popup menu and choose the required icon.




## To search contours by their icon (this page should be placed after page 121 in the User's Guide)

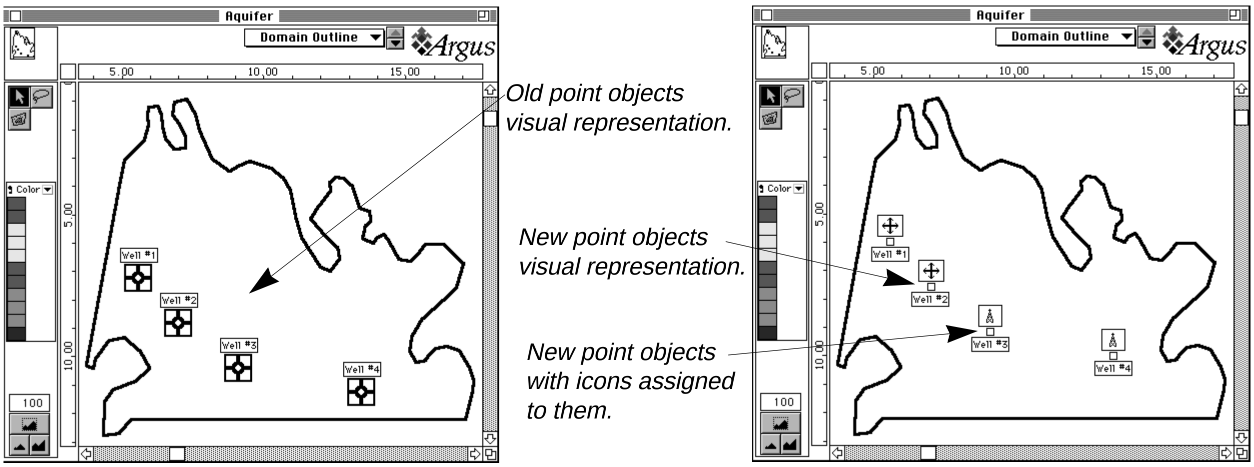
You can instruct Argus ONE to search for contours by the icon assigned to them. The Icon check box and appropriate popup menus have been added to support this capability.




*Check the box and select an icon.*

### Point object (one point contour) visual representation

Since you can now assign icons to contours, Argus ONE assigns a point object a default icon of a source. However, using the new Contour Icon menu you can assign the point object any icon of your choice.



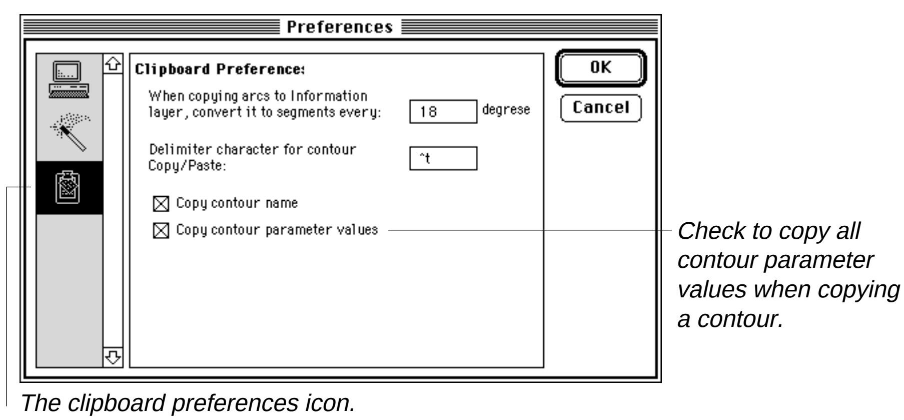

## New Contours Copying and Pasting Capabilities

### Setting contours' copying preferences (this page should be placed after page 110 in the User's Guide)

You can now choose if you want Argus ONE to copy the contours parameters values when copying contours.

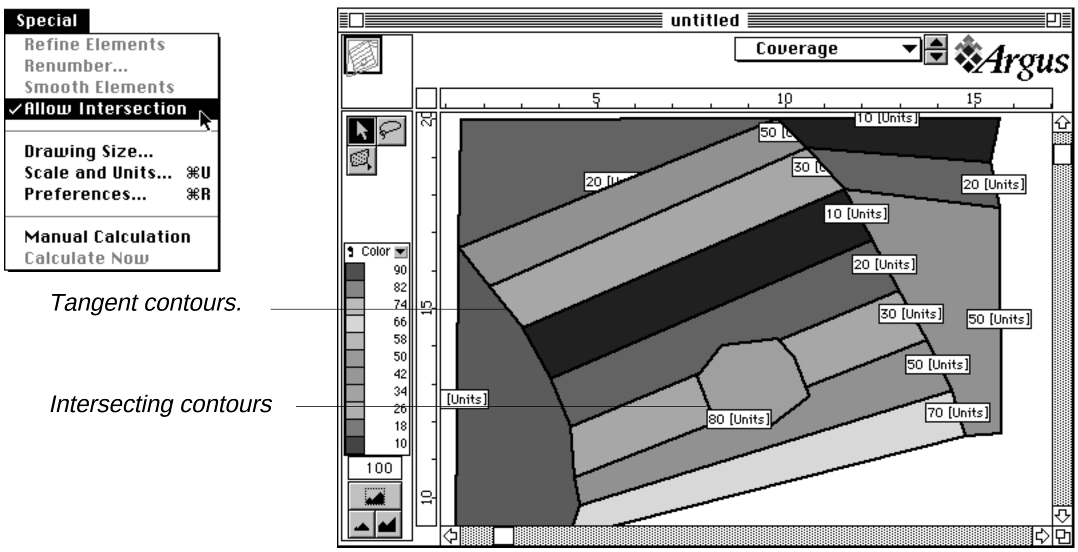
1. From the Special menu choose Preferences...
2. Click the clipboard preferences icon.
3. Check the "Copy contour parameter values" check box.

*The Clipboard Preferences dialog.*



## Tesslated and Overlapping Contours

Information layers now support tessellated and overlapping contours. This feature is extremely useful for coverage information such as man-made bathymetry, and land usage type. - To use: From the Special menu, select Allow Intersection. Notice that a contour may still not intersect itself. Also note that an Information type layer containing no intersecting contours can be transformed to Allow Intersection mode, but not vise a versa.




*An Information type layer with enabled Allow Intersection option.*

## New Options for Setting Icon Sizes

To allow you to customize your Argus ONE workplace you can now also set the icon’s sizes.

### Setting icon sizes

You can now choose the sizes of icons.

- From the Navigation menu choose Small or Large icon size.


*Large Icon* *Small Icon*

# New Parameters’ Tags

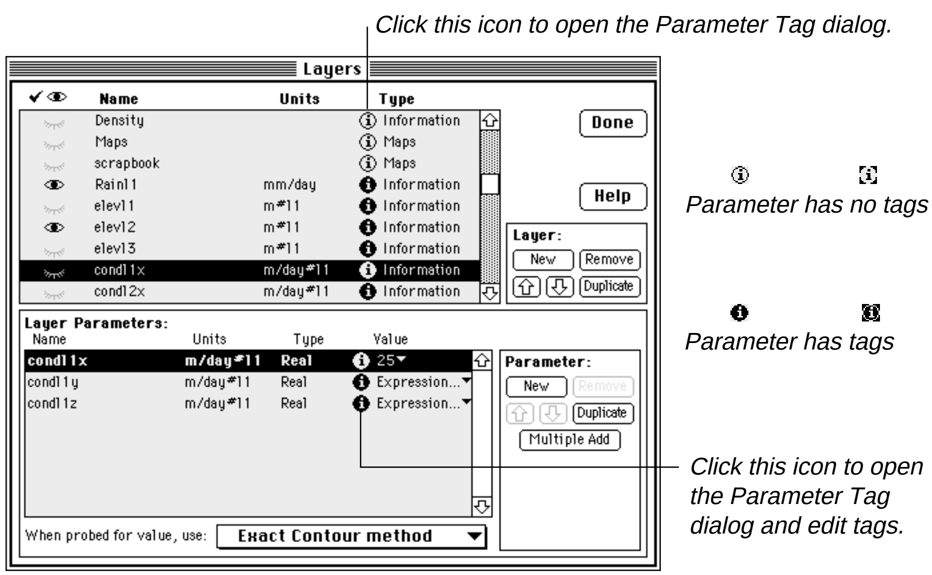
To allow for adding model specific information Argus ONE layers and parameters can now be assigned additional information. You store this additional information in parameters tags. Each Argus ONE layer and parameter can be assigned as many tags as you need.

Parameter Tags can also help you better control your export template by using the tags as means for identifying special layers and parameters. This way you can create templates that export different parameters in different formats while still leaving your template general purpose.

A parameter tag is defined by its name, the TagName and its value, the TagValue, where both are strings.

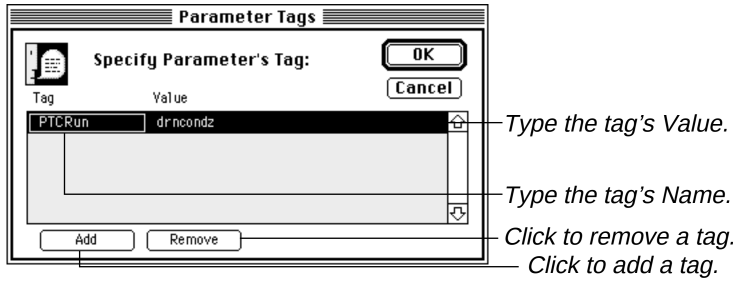
## Assigning tags to parameters

Parameters tags are assigned in the Parameter Tag dialog which is invoked from the Layers dialog. A layer’s main parameter as well as its other parameters can be assigned tags. Every layer and parameter line in the Layers dialog is marked with an ⓘ icon standing for additional information. Clicking that icon brings up the Parameter Tag dialog.



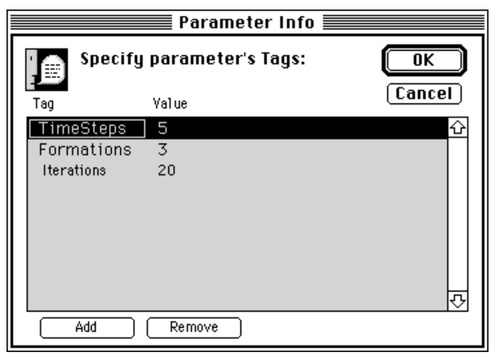

### To assign tags to a parameter

1. Select the line of the layer its tags or its parameters’ tags you wish to edit.
2. Click the ⓘ icon. The Parameter Tags dialog box appears.




Although both the TagName and TagValue can include any character, you should avoid using special characters. The $ character must not be used in the TagName.

**Example**: In the following example three parameter tags have been added to a layer. These tags are assigned the information regarding the number of time steps the model should run, the number of formations described and the number of iterations for each time step.




To export the information stored in these tags use the following Export Template script. The if statements insure that the tags values will be exported only if you have created such tags.

```
Start a new line
  If: $HasTag(TimeSteps)$
    Export expression: $Tag(TimeSteps)$; [I5]
  End if
  If: $HasTag(Formations)$
    Export expression: $Tag(Formations)$; [I5]
  End if
  If: $HasTag(Iterations)$
    Export expression: $Tag(Iterations)$ [I5]
  End if
End line
```

# New Argus ONE Export Macros

Three new export macros have been added. Also, the File script command is enhanced to allow calling it from within other commands and to nest it within other File commands.

## New Export Macros

### The $HasTag(TagName)$ Macro

This macro returns 1 (true) if it finds a tag having TagName, and 0 (false) if it does not. In an export loop over parameters each parameter is tested for TagName.

### The $Tag(TagName)$ Macro

This macro returns the TagValue of TagName. In an export loop over parameters each parameter is tested for TagName. If the TagName is found, TagValue, is returned. If TagName is not found the macro is replaced with its string representation (unresolved) to notify you that a problem occurred.

Example: A Tag name `PTCRun` has a TagValue `drnelev` in a certain parameter. The macro `$Tag(PTCRun)$` returns `drnelev` for that parameter. If TagName `PTCRun` is not found the string `$Tag(PTCRun)$` is returned.

### The $BaseNameNoExt$ Macro

This is a natural extension to the `$BaseName$` macro, which might produce unwanted results on PC's.

## Resolving $Basename$

While exporting a mesh layer using the following template, the `$BaseName$` macro is replaced by the file name supplied in the Save As... dialog.

```
Redirect output to: $Basename$.nod
     Node Command
     Node Command
End File
Redirect output to: $Basename$.elm
     Element Command
     Element Command
End File
```

**Macintosh:** For instance, if you supply the name MESHEXP, the two occurrences of the argument `$Basename$` will be replaced by MESHEXP, to create two files named MESHEXP.nod and MESHEXP.elm.

**PC and UNIX:** For instance, if you supply the name MESHEXP, the two occurrences of the argument `$Basename$` will be replaced by MESHEXP, to create two files named MESHEXP.nod.exp and MESHEXP.elm.exp. Since this is not allowed on PC and usually not needed you could use the `$BaseNameNoExt$` macro instead. In this case the two occurrences of the argument `$BasenameNoExt$` will be replaced by MESHEXP, to create two files named MESHEXP.nod and MESHEXP.elm.

### Enhanced File command

| *Name* | *Syntax* | *Arguments* |
|---|---|---|
| **File** | **Redirect output to:**<br>**End File** | ***Filename*** **or** ***$Basename$*** |

- This command can now be nested within any other script command, including the File command.
- This command is a block command.

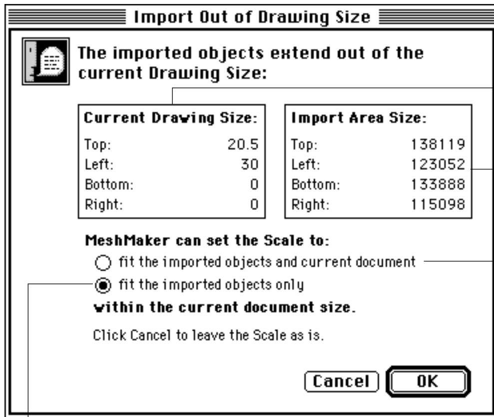
# Miscellaneous New Features

## New Tool for Geographical Alignment of Data

### Importing and coordinate system considerations

When you import and paste graphic objects from sources external to the current project, you must take into account the possibility that the source and target coordinate systems might differ. They might differ in scale, units, coordinate system direction, drawing size, and origin. If they do, the objects you bring in might look different from what you expect, or too small to be noticed in the current zoom level.

If the objects you import or paste are partially or completely out of the current drawing size, Argus ONE will now alert you of the situation by bringing up the following dialog:




*Current drawing size.*

*Size of imported objects' enclosing rectangle.*

*Check to force Argus ONE to scale and change the origin such that imported objects and the current drawing size fit within the current drawing size.*

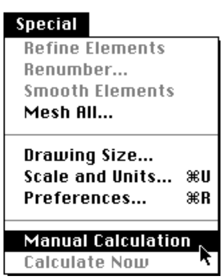
*Check to force Argus ONE to scale and change the origin such that imported objects fit within the current drawing size.*

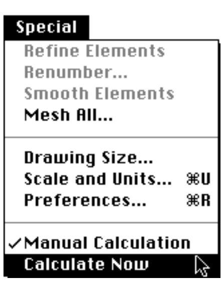
You can choose to either:

- Join the two areas and scale them to fit to the current drawing size.
  Or -
- Scale the area defined by the imported objects' enclosing rectangle to fit the current drawing size.

Note: If the project already contains any objects, only the first option will appear in the dialog. If you choose OK Argus ONE will set the scale to fit both the existing and imported objects within the current drawing size. If you choose Cancel the scale and origin are maintained, thus possibly causing imported object to be drawn out of the canvas.

## Manual Calculation and Calculate Now Menus




As explained in "Chapter 7" under "Colors and Performance Considerations" on page 247, color evaluation of elements takes place on each redraw of the screen. This is also true for grid blocks.

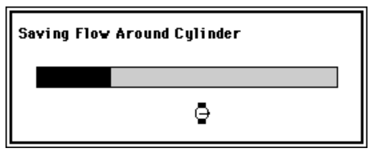
If your mesh or grid contain many elements or blocks which are linked to complex parameters, these screen redraws may temper with your work. Using the Manual Calculation menu under the Special menu you can now instruct Argus ONE to refrain from recalculating elements and block colors at each redraw.

However, you must be aware that when Manual Calculation is on, the colors of elements and blocks, as well as the values presented on the different information dialogs will not reflect any changes you have made from the time you turned manual Calculation on. To update the values, you can either select Calculate Now from the Special menu, or turn Manual calculation off.

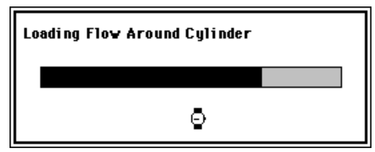
**Important Note**: To prevent you from exporting wrong information to your model, Argus ONE automatically performs a Calculate Now command while exporting the data.

## Save and Load Progress Bars

Saving and loading large Argus ONE project files may take a long time. When loading and saving, Argus ONE presents you with a progress bar indicating the status of the save and load operations and specifies the file name.




*Save progress bar.*




*Load progress bar.*

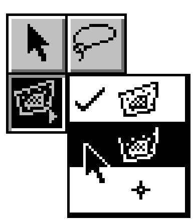
**Important Note**: While a progress bar is active you can send Argus ONE to the background and work with a different application. This is the case with all operations that are posting a progress bar, which include, save, load, import, export, auto mesh generation, auto grid generation and search.

## New Popup Menus in Tool Palettes

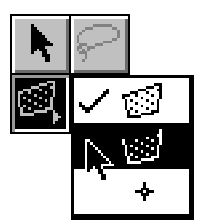
Domain and information type layers contour tools, as well as the Add Grid Line and Column tools, are arranged in a tool palette popup menus.

### To select a contour tool

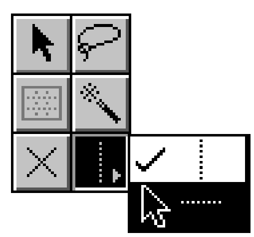
1. Click and hold the mouse button until the popup menu appears.
2. Move the cursor to the item you wish to select.
3. Release the mouse button.



*The contour tool popup menu for domain type layers.*



*The contour tool popup menu for information type layers.*

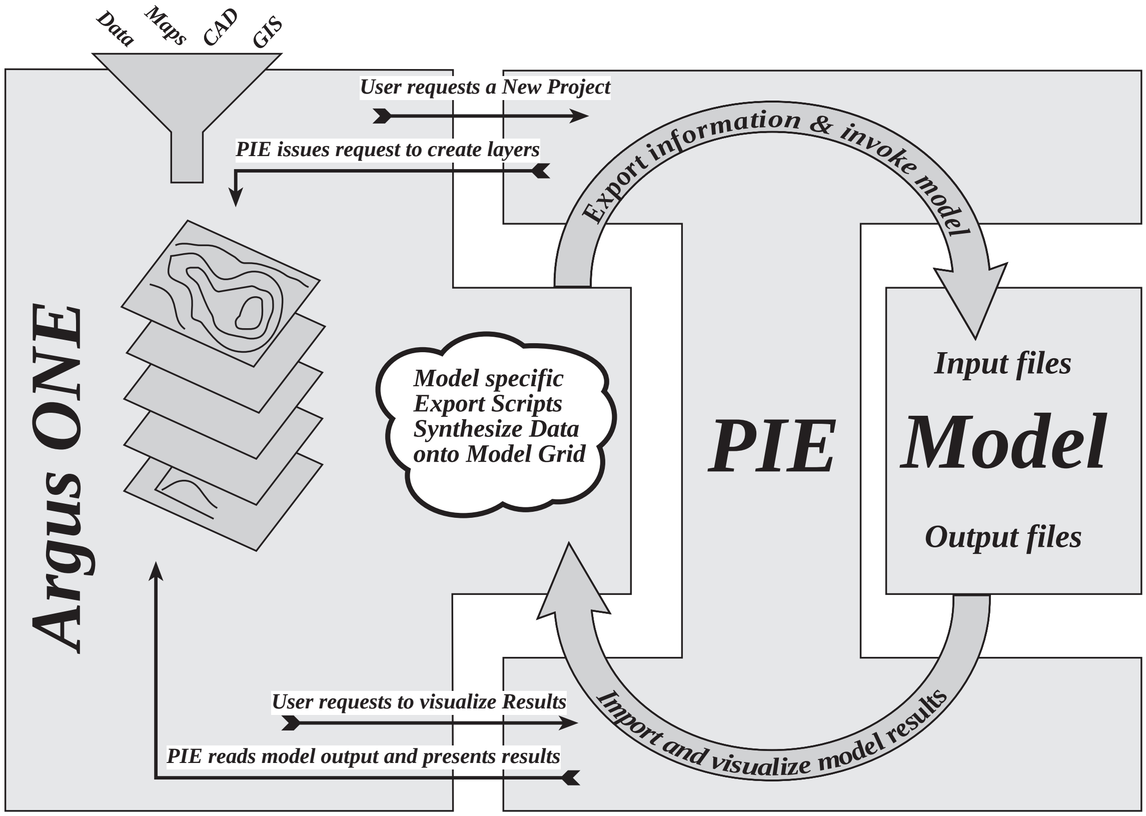


*The add grid rows or columns tool popup menu.*

# New Examples on Disk

After installing Argus ONE you will find within the Argus ONE directory or folder an Examples directory. In this directory we placed a number of Argus ONE projects, DXF files, and information files containing contours. The DXF and information files are the files used to create the examples projects. Also, please find a new example file named POSTPROC.MMB demonstrating some of the post-processing capabilities.

| ***PC File names*** | ***Mac and Unix File names*** | |
|---|---|---|
| Examples | Examples | |
| BayExmp.mmb | Bay & Harbor Mesh | *Argus ONE project files. Open them to learn about linking parameters to grids and meshes.* |
| GridExmp.mmb | Grid Example | |
| MeshExmp.mmb | Mesh Example | |
| Morectr1 | Additional Contour files | *Files containing domain outline contours. Create a new project, import them to a domain layer and mesh them.* |
| BayCtr.exp | Bay & Harbor Geo.exp | |
| WorldCtr.exp | World Domain Outline.exp | |
| Contours | Contours files for Exaples | *Files containing information contours used in creating the Grid Example project and Mesh Example project.* |
| Density.exp | Density.exp | |
| Domain.exp | Domain Outline.exp | |
| Perm1.exp | Permeability 1.exp | |
| Perm2.exp | Permeability 2.exp | |
| Poro1.exp | Porosity 1.exp | |
| Poro2.exp | Porosity 2.exp | |
| Rainfall.exp | Rainfall.exp | |
| Topo1Bot.exp | Topography 1 Bot.exp | |
| Topo1Top.exp | Topography 1 Top.exp | |
| Topo2Bot.exp | Topography 2 Bot.exp | |
| DXF | DXF files | *DXF files. Import them into Maps type layer.* |
| NY.dxf | NY.dxf | |
| RainCntr.dxf | Rainfall contours.dxf | |
| USAStats.dxf | USA StateLines.dxf | |
| Export | Exported Files | *Files containing information on a mesh and a grid exported using the default Argus ONE export.* |
| BayExmp.exp | Bay & Harbor Mesh.exp | |
| GridExmp.exp | Example Grid.exp | |
| MeshExmp.exp | ExampleMesh.exp | |




# Argus ONE

## New Features Supplement

For version 3

This product was developed using Neuron Data Open Interface Elements.

Sales and Services offices at:

Argus Interware, Inc.

11 Merit Lane

Jericho, NY 11753

USA

# Argus ONE Version 3

The many new features and capabilities introduced in Argus Open Numerical Environments (Argus ONE) version 3 are described in detail in this supplement. We suggest that you go through the table of contents and locate items that relate to your work.

Although not described in this supplement, the most important addition to Argus ONE version 3 is the introduction of the Plug-In Extension (PIE) technology. The PIEs make Argus ONE a fully open system, to which our users can easily interface state of the art models and technologies. The ArgusNE PIE technology is fully documented and accompanied by many useful examples contributed by us and our users. The technology is an open architecture technology and is made available to you free of charge. For availability of documentation, technical notes and examples contact Argus. All PIE related materials are also posted on the Argus ftp site.

Many companies, organizations and universities are developing an abundance of scientific tools and fully interfaced numerical models using the PIE technology. PIEs for Ground Water flow and transport models such as MODFLOW, SUTRA, HST3D, PTC and MOC3D are and will be offered to you as commercial products and some as public domain products. Other geostatistical PIEs including state of the art interpolation methods are also being developed.

## Contents

# Compatibility Issues

Full backwards compatibility exists between version 3 and version 2.5c. This means that you can open Argus ONE files created with version 2.5c in Argus ONE version 3. However, you can not open files which will be saved under version 3 with version 2.5c.

We suggest that until you are certain that all functions operate as you were accustomed to in version 2.5c you save backup copies of your files created with version 2.5c.

Also, if you have enough disk space, do not erase your copy of version 2.5c until you are certain that all is fine with results you get from version 3.

## Change in Custom Grid Export Templates

If you developed a custom grid export template you should edit such templates under version 3 to change all loops over parameters to loops over block parameters. For a detailed discussion of the new features requiring this change refer to the section “New Export Template Dialog and New Script Commands” on page 61.

## Using Plug-In Extensions (PIEs)

### Installing PIEs

You can only use PIEs with Argus ONE version 3 and later. When you install version 3 a special directory is automatically placed on the disk. The directory is named ArgusPIE and contains one example PIE developed by Argus. If you want to add a PIE you should place it in the this directory. You can group PIEs and place them in sub-directories within the ArgusPIE directory. When you run Argus ONE all PIEs located in the ArgusPIE directory are loaded and become available.

If you purchase a product which uses PIEs follow the instructions provided by the PIE developer.

### Disabling PIEs

If you do not want a certain PIE to load when you use Argus ONE, you need to take them out of the ArgusPIE directory. We suggest you create a directory called UnusedPIEs in the root directory and move PIEs you do not want to load into that directory.

# Miscellaneous New Features

## The Layers Floating Window

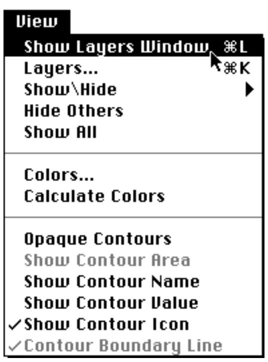
The Layers floating window lists all layers and allows you to quickly show, hide and activate layers. When open, it floats above the active window of the Argus ONE project it belongs to. This window does not replace other methods available for controlling layers visibility, but adds new ones.

### Opening the Layers Window

When you open the Layers window it appears above the main window. Since it floats above its parent window you can work in both the main window and the Layers window without having to first bring one of them to the front.

#### To open the Layers Window

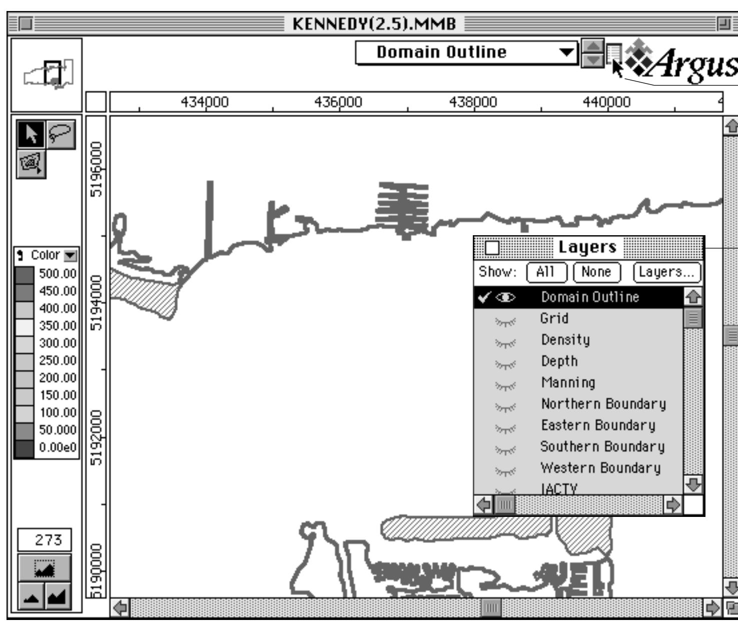
- Click the Layers Window button located to the right of the Layers menu. -Or
- From the View menu, select the Show Layers Window menu item. - Or -
- From your keyboard type CTRL + L (⌘ + L on the Macintosh).




---

**Important Note:** The keyboard shortcut for opening the Layers Dialog was changed from CTRL + L to CTRL + K (⌘ + K on the Macintosh).

---




*Click this button to open the Layers Window.*

*The Layers Window floating above the main window.*

*Maps courtesy of Adele Militello, Conrad Blucher Institute, Texas A&M*

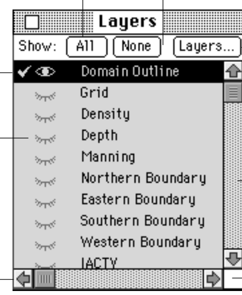
## Layout of the Layers Window

Click the All button to show all layers.

Check mark indicating the active layer. Click in any layer to activate it.

Open or shut eyes indicating layers' visibility status, click the eye icon to show/hide a layer.

If your layer names are long, scroll to view them.




Click the None button to hide all layers but the active one.

Click the Layers... button to open the Layers Dialog.

If you have many layers, scroll to view them, or resize the window.

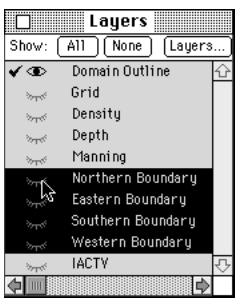
## Showing and hiding sets of layers

You can show and hide sets of layers. This enables you to easily control the visibility of information.

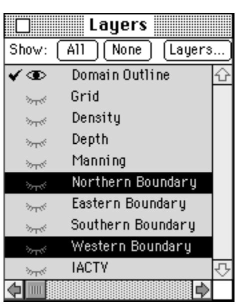
### To show and hide a group of layers

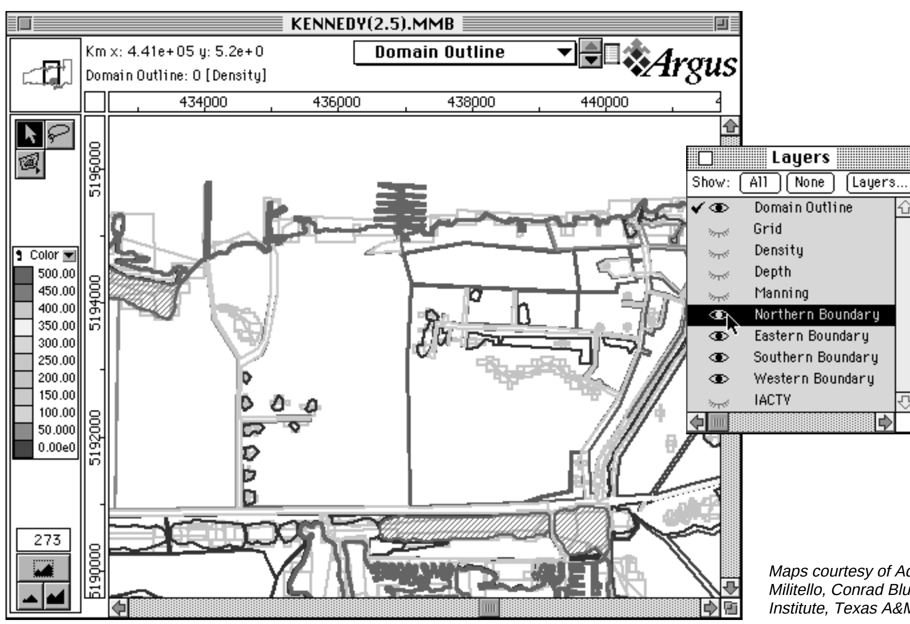
**1.** Select the group of layers you wish to show or hide by shift-clicking them.

**2.** Click the eye icon on one of the lines of the selected layers to toggle the layers visibility.
All layers in the selection are toggled.



*You can also select discontinuous sets of layers.*






Maps courtesy of Adele Militello, Conrad Blucher Institute, Texas A&M

### To hide all layers

Click the None button to hide all layers but the active layer.

### To show all layers

Click the All button to show all layers.

### To open the Layers dialog

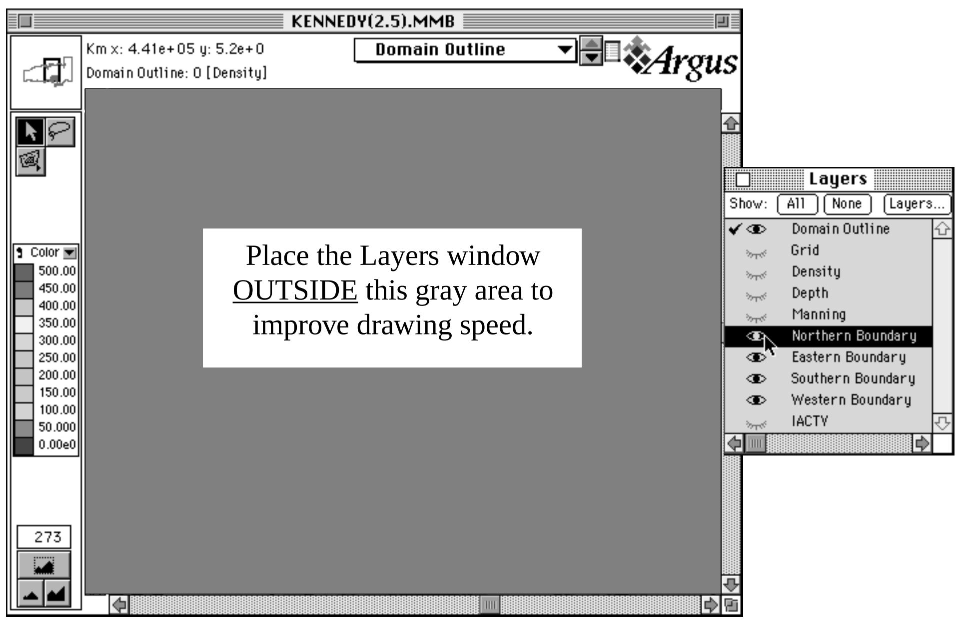
Click the Layers... button to open the Layers Dialog.

### To move, resize and close the Layers window

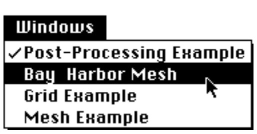
To move the window drag it in the window title. To resize the window use the resize box. To close it, click the close box at the top-left corner of the window.

## Performance considerations

If the Layers window floats above the main widget of the Argus ONE window, scrolling and redraws may become very slow. To avoid this, either place the Layers window outside the main drawing area, or close it all together.




## New Window Menu



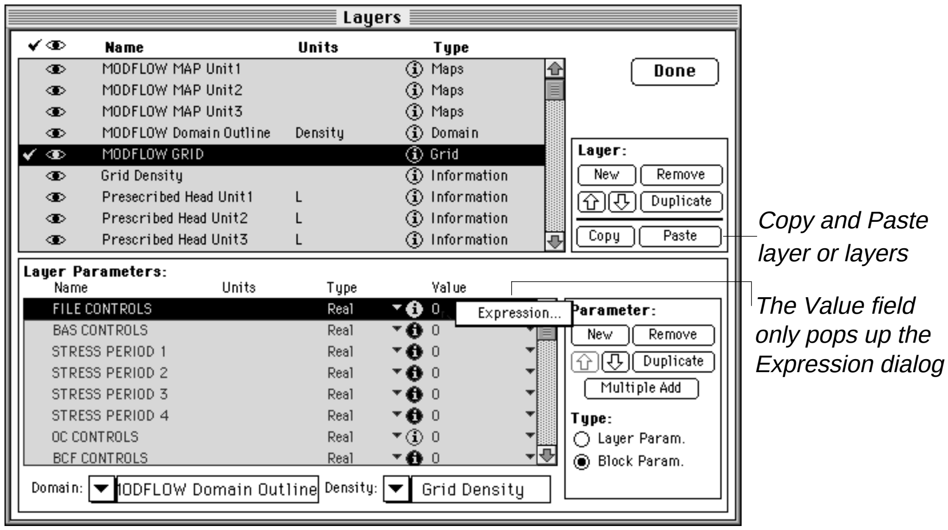

The new Window menu, appearing as the right most menu, lists all Argus ONE open windows. Both project windows and Data Layers-Show Values windows are listed.

### To bring a window to the front using the Window menu

- To activate a window select its name from the window menu.
  If the Layers floating window for that window was open at the time you sent it to the back, it is restored when you bring its project window to the front.

# Layers Dialog - New Appearance and Functionality

The Layers dialog has been slightly changed to add functionality and to improve performance.




## Linking Layers - The Parameter List Value Field

Linking layers and parameters can now be only performed from within the Expression dialog. When you click and hold the mouse above the Value field in the parameter list the Expression popup menu appears to allow you to enter the Expression dialog. This new function was implemented to avoid the very long menus resulting from the use of many layers. To link layers and or parameters you can also use the Multiple Add button.

## Copying and Pasting Layers

Copying and pasting of layers is a powerful tool both for the end-user and the Argus ONE PIE developer.

### Copy and Paste a single layer

Copying and pasting a layer allows you to duplicate a layer but have it inserted below any layer you choose. It is a shortcut for duplicating the layer and promoting or demoting it manually.

When you copy or paste a layer it is copied together with its parameters, expressions, tags, visibility, and activation status and export templates.

#### To copy a layer

1. In the layers list, select the layer line.
2. Click the Copy button.

The layer definition is copied into the clipboard.

#### To paste a layer

1. In the layers list, select the line of the layer below which you want to paste a layer.
2. Click the Paste button.

The layer appears below the selected layer.

If the name of the layer already exists (that is if you did not delete the layer you copied) then it is changed to a unique new name by a number suffix.

### Copy and Paste the Complete Layer Structure

You can also copy the complete layer structure by holding the SHIFT key while clicking the Copy button. This option is especially valuable for Argus ONE PIE developers.

### The Layer Definition Language

*Important Note: LDL's main use is in the development of PIEs. If you are not familiar with LDL be sure not to paste layer definitions which you have edited outside of Argus ONE.*

Copying and pasting layers is performed by parsing the layers using the Layers Definition Language (LDL). The LDL is a complete text representation of Argus ONE layer system. It includes all the fields and status flags required to define a layer. To find more about LDL refer to Argus ONE PIE documentation.

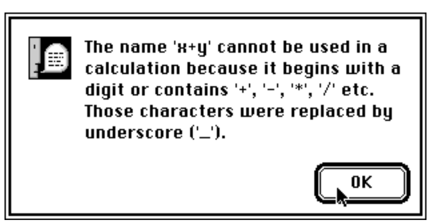
#### LDL Examples

```
Layer:
{
    Name: "Domain Outline"
    Units: "Density"
    Type: "Domain"
    Visible: Yes
    Interpretation Method: Nearest

    Parameter:
    {
        Name: "Density"
        Units: "Density"
        Value Type: Real
        Value: "0"
        Parameter Type: Layer
    }
}
Layer:
{
    Name: "Grid"
    Units: ""
    Type: "Grid"
    Visible: Yes
    Domain Layer: "Domain Outline"
    Density Layer: "Density"

    Template:
    {
```

```
         Line: "File: $BaseName$"
         Line: "\tLine"
         Line: "\tExpr: NumRows(); [I8]"
         Line: "\tExpr: NumColumns(); [I8]"
         Line: "\tExpr: NumBlockParameters()+1 [I8]"
         Line: "\tEnd line"
         Line: "\tLoop: Rows"
         Line: "\tLine"
         Line: "\tExpr: NthRowPos($Row$) [F8.2]"
         Line: "\tEnd line"
         Line: "\tEnd loop"
         Line: "\tLoop: Columns"
         Line: "\tLine"
         Line: "\tExpr: NthColumnPos($Column$) [F8.2]"
         Line: "\tEnd line"
         Line: "\tEnd loop"
         Line: "\tMatrix: BlockIsActive() [I1]"
         Line: "\tLoop: Block Parameters"
         Line: "\tLine"
         Line: "\tEnd line"
         Line: "\tLine"
         Line: "\tExpr: \"# $Parameter$\""
         Line: "\tEnd line"
         Line: "\tMatrix: $Parameter$ [F8.2]"
         Line: "\tEnd loop"
         Line: "\tEnd file"
         Line: ""
    }
}
```

## Naming of Layers and Parameters

Argus ONE does not allow layer and parameter names to include illegal characters anymore. When you name a layer or a parameter Argus ONE checks the name and automatically replaces any such characters by the underscore (_) character. Before doing so an alert opens to notify you of the change.



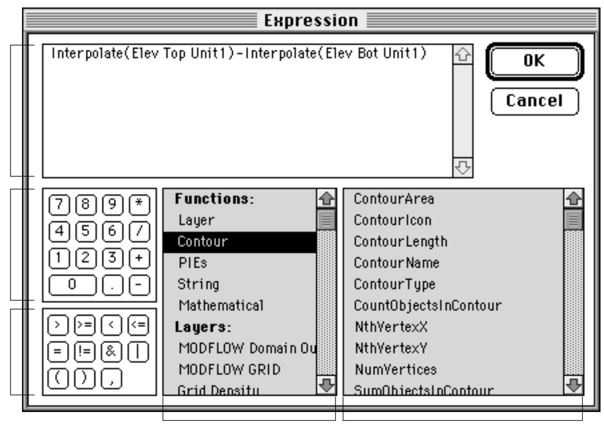
# Expression Dialog - New Appearance and Functionality

Due to the large number of Argus ONE functions and Argus ONE PIE functions available, and due to the large number of layers and layer parameters being used routinely by Argus ONE users, the functions and layer parameters have been grouped. These groups appear on the Group list. The functions and layer parameters are listed in the Functions/Parameters list located to the right of the Groups list. This new arrangement replaces the Funcs and Layers popup menus.

*The expression editor panel.*

*The calculator panel.*

*The logical operators panel.*




*The Group List lists all Function groups and Layer parameters*

*The Functions / Parameters list.*

## Entering Functions and Parameters in the Expression Editor

Functions and Layers are separated into two different groups appearing one below the other in the Groups list. Instead of selecting a function or a parameter from the shortcut menus Funcs and Layers (not available anymore), you should double-click them in the Functions/Parameters list.

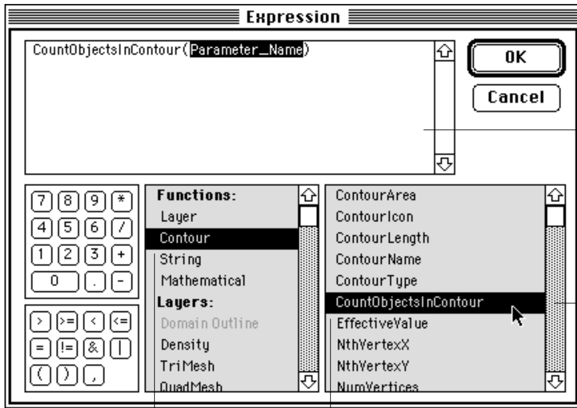
### Locating and Entering a Function

Functions are grouped into their respective groups. All function groups appear in the Groups List below the word Function:

#### To locate and enter a function

1. Select the group the function belongs to. The group’s functions are listed in the Function/Parameters list to the right.
2. Scroll through the Function/Parameters list to locate the function.
3. Double-click the function.

The function is entered in the expression editor panel at the insertion point.




*The function is entered at the insertion point with the appropriate arguments.*

*3. Double-click the function to enter it into the expression editor.*

*1. Select the function group. The list to the right presents the functions available for the selected group.*

*2. Scroll to locate a function. Use the scroll-bar or the arrow keys.*

If you do not know which group the function belongs to, just scroll through the function groups and through the Functions/Parameters list.

### Locating and Entering a Layer Parameter

Layer parameters are grouped under their layers. All layers appear in the Groups List below the word Layers:

#### To locate and enter a parameter

1. Select the layer in which the parameter is defined. The layer's parameters are listed in the Function/Parameters list to the right.
2. Scroll through the Function/Parameters list to locate the parameter.
3. Double-click the parameter.

The parameter is entered in the expression editor panel at the insertion point.

If you do not know which layer contains the parameter, just scroll through the layers and through the Functions/Parameters list.

## Function Arguments

When you double-click a function in the Function/Parameter list it is now entered in the expression editor with its arguments. Optional arguments are not entered.

You should replace the arguments by their actual values.

#### Examples

```
If(Condition, True_Value, False_Value)
```

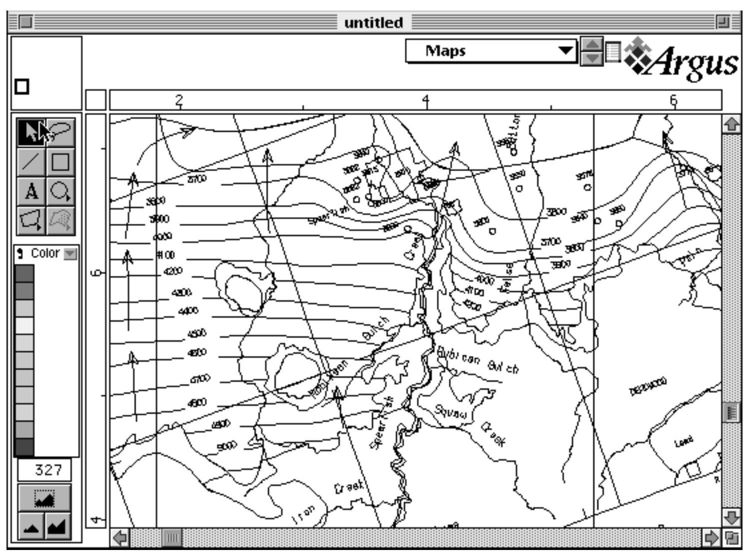
```
Index(Index, Numbers...)
```

```
CountObjectsInContour(Parameter_Name)
NthParamName(Parameter_Number)
StrIndex(String, Substring)
CountElements(Condition)
CountObjectsInElement(Parameter_Name)
NodeAboveCntr(Parameter_Name)
CountBlocks(Condition)
CountObjectsInBlock(Parameter_Name)
```

# Support of DXF Blocks

DXF files may use a macro language (named block) to allow for the description of scaled and rotated objects. It also allows for the description of objects which appear many times in a DXF file, thus reducing the amount of redundant information in the file.

Argus ONE now supports DXF blocks which enables you to import complex drawings from all applications supporting DXF format.



# Scaling of Object, Layers and Projects

You can now scale all of Argus ONE objects. The scale command allows you to scale selected objects, a whole layer or a complete project. This might prove useful in instances where some of the information imported into Argus ONE is in a different scale than other information already present in the project.

The following layer types allow you to either scale selected objects or scale the layer as a whole:

- Map type layers

You can scale the following layer types but can not scale individual objects within them to refrain you from creating overlapping elements or illegal contours in information layers:

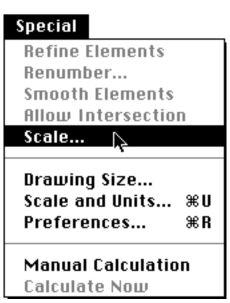
- All Information type layers (including domain and density layers).
- Grid type layers
- Mesh type layers
- Data type layers

## Scaling options

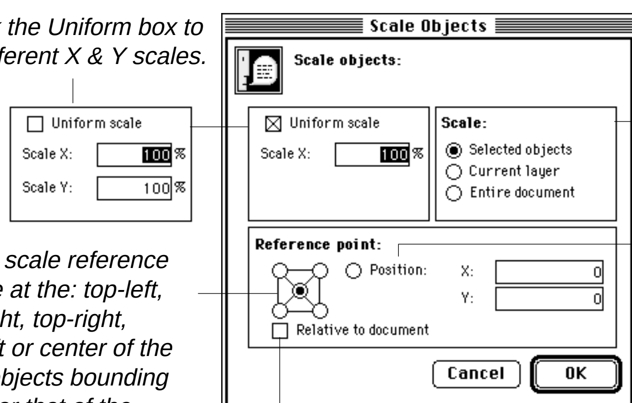
The many scaling options available allow you full control of what and how to scale. You set the scale options in the Scale Objects dialog. If you are not satisfied with the results use the Undo command and try again.

### To open the Scale Objects dialog

- From the Special menu select Scale...
  The Scale Objects dialog opens.




*Uncheck the Uniform box to apply different X & Y scales.*




*Check to select which objects are to be scaled.*

*Select the scale reference point to be at the: top-left, bottom-right, top-right, bottom-left or center of the selected objects bounding rectangle or that of the document's rectangle.*

*Click the Position radio button to specify an absolute reference point.*

*If unchecked the scale is relative to the objects' bounding rectangle. Check to make the scale relative to the document's bounding rectangle.*

### Uniform/Nonuniform scale

By default the scale is uniform. The scaled items are scaled proportionally both in X and Y.

- Type the required scale in the Scale X: text edit box.
  Or -

1. Uncheck the Uniform scale check box.
2. Type the required scale in both the X and Y edit boxes.

### Selecting the objects to be scaled

- Click the Selected objects radio button to scale only selected objects.
- Click the Current layer radio button to scale the currently active layer.
- Click the Entire document radio button to scale all project layers and objects at once.

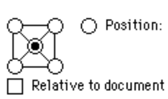
### Selecting the reference point for scaling

The reference point is the point around which the scale will occur. There are altogether eleven (11) possible reference points which are grouped into three groups: relative to objects, relative to the document, and relative to an absolute point.

After a reference point has been selected the selected objects are scaled by applying the following equation to each of the points defining the objects:

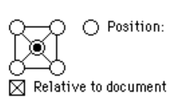
```
X_after_scale = (X_before-Xrefernce)*Xscale+Xreference

Y_after_scale = (Y_before-Yrefernce)*Yscale+Yreference
```

#### Reference points relative to the selected objects bounding rectangle



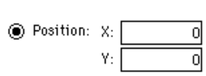

You can select one of five available reference points located at the four corners and the center of the selected objects bounding rectangle. The bounding rectangle is defined as the minimum rectangle enclosing all objects in the selection.

#### Reference points relative to the selected objects bounding rectangle




You can also select one of five available reference points located at the four corners and the center of the document's drawing size.

#### An absolute reference point




You can also define an absolute reference point at any X and Y coordinates. All scaling will be performed relative to that point.

### To scale an object or a selected group of objects

1. Select the object or objects to be scaled.
2. From the Special menu select Scale... The scale dialog opens.

3. Click the Selected objects radio button.
4. Select other required options and click the OK button.

### To scale a layer

1. Activate the layer to be scaled.
2. Click the Current layer radio button.
3. Select other required options and click the OK button.

### To scale a project

1. Click the Entire document radio button.
2. Select other required options and click the OK button.

## Using Information Dialogs with Multiple Selected Objects

Until version 3 you could have manually changed the values of objects such as contours, nodes, element and blocks only one at a time. All object information dialogs, which popup when you double-click objects, are now enhanced to allow you to change all the selected objects' values at once.

If the selection contains more than one object you can now:

- Change all of the parameters' values of all the selected objects at once.
- Change values of only some parameters of all the selected objects at once.
- Scroll through the selected objects one by one, and change only some.
- Locate the current object in the main window.

The three areas where you can derive the most benefit from these new capabilities are:

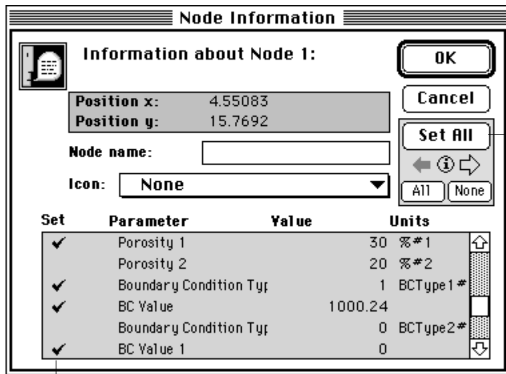
- Assigning a value to a group of objects which are grouped based on their geographical location. If you need to assign a value to a set of elements, nodes and or blocks and you can not do so using the automatic GIS functionality, you can visually select them and assign them the required value.
- Assigning a value to a group of objects which are grouped based on the range of values assigned to them. If you need to change a group of objects' values, which are within a known range, to a single value, you can use the search command to select them and then change their values.
- If you need to quickly clear all manually entered (override) values and reinstate the default values assigned in the layers dialog.

### Enhanced Object Information Dialogs

If more than one object is selected when you open the contour, block, node and element information dialogs, additional buttons are presented to allow you to assign values to all or some of the objects in the selection.

## Information dialogs for multiple selected objects

The screen-capture below is of the node information dialog. The contour, block and element information dialogs are all enhanced in the same way.




*The node information dialog as opened when multiple nodes are selected.*

*Use this panel to control the objects and parameters you assign changes to.*

*Check the parameters for which you want the change in value to be assigned to all selected object when you click the Set All button.*

### To apply your changes to all objects and all parameters

1. Click the All button to check all parameters.
2. Change the parameters' values.
3. Click the Set All button.
4. Click the OK button. If you change your mind and do not want to apply the changes to all objects and parameters, click the Cancel button.

### To apply changes to all objects but only to some of their parameters

1. Click the None button to uncheck all parameters.
2. Check all the parameters for which changes are to be applied to all objects in the selection.
3. Change the parameters' values.
4. Click the Set All button.
5. Click the OK button.

### To apply changes only to some of the selected objects

1. Click the None button to uncheck all parameters.
2. Click the left or right arrow buttons to scroll through the selected objects.
3. To visually locate the current object move the dialog away from the main drawing area and click the i icon. The window scrolls to center the object in the drawing area and a visual effect marks the object.
4. Change the values of the required parameters.
5. Repeat steps 2-4 to locate other objects whose values are to be changed.
6. Click the OK button.

# New Parameter Types

Starting with version 3 all constants, functions, expressions and parameters can be set to one of four types:

1. String - any string of characters, not including quotation marks
2. Real - the default type for all ArgusNE variables up to version 3
3. Integer - any number between -2,147,483,648 and 2,147,483,647
4. Boolean - zero (0) False, any number other than zero, True (usually 1)

All ArgusNE functions now return their values in one of the above four types.

## Setting Parameter Types

Setting a parameter's type enables you to declare parameters which will return a value of a certain type. For instance, a parameter holding the expression `ContourName()`, which returns the contour's name (String), should be set to type String, while a parameter holding the expression `NodeAboveContour(layer_name)`, which returns (boolean) True (1) only if a node is located exactly above a contour in the layer `layer_name`, could be assigned boolean, integer or real type.

You should set the parameter type when you first create it. When you first create a layer or a layer parameter, Argus ONE sets it by default to type Real. You can either change the type immediately after creating the parameter or later on.

### To set a parameter's type

1. In the Layers Dialog, select the layer for which you need to change a parameter's type.
2. In the parameters list (bottom part of the layers dialog) select the line of the parameter whose type you want to change.
3. Click and hold the mouse button in the Type column to popup the Type menu.
4. Select the required type and release the mouse button.

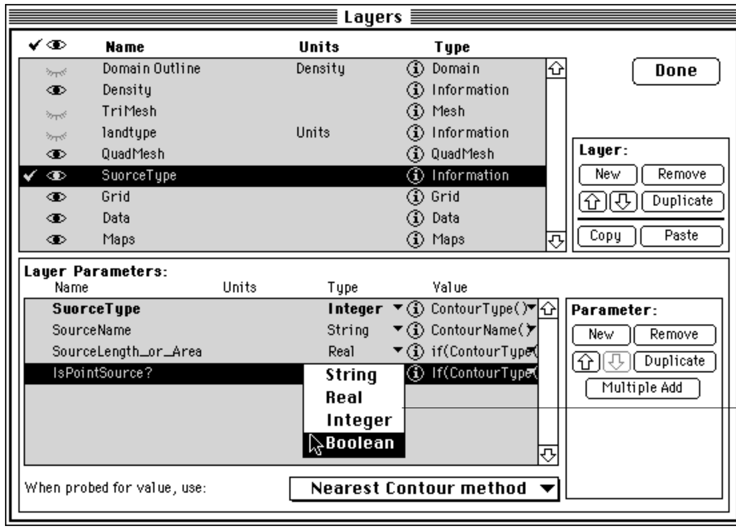
### To change a parameter's type

Changing a parameter's type is performed in the same way as setting its type.

***Important Note:***
*You can not change a parameter type if one of the objects in the layer it belongs to is assigned a manual override value.*

However, if you already manually assigned a value (override value) to that parameter for some of the objects in the layer, the small arrow in the Type column will not be present, indicating that you can not change the type while such manual values are assigned.

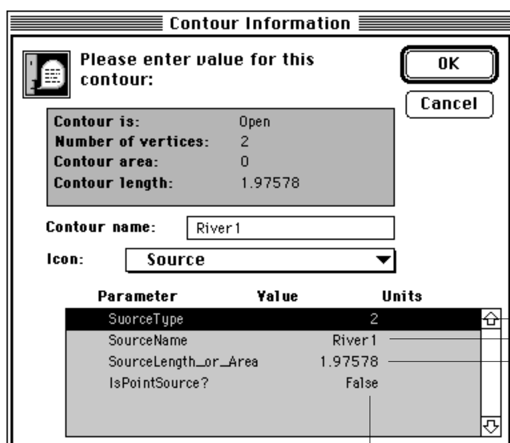
In this case you need to first close the layers dialog, go to the layer and there remove all such values from object parameter's whose type you wish to change. Then, open the layers dialog and change the parameter's type.




*The Type popup menu.*

## Viewing Parameters in Information Dialogs

When you double-click any Argus ONE object such as contours, nodes, element or blocks, an information dialog box appears to allow you to view the values assigned to the objects. With the introduction of parameter types, objects values now reflect their types. For instance, if a parameter is of type boolean, a value of True or False will be presented. If a parameter is of type string, the string assigned to the object will be presented.




*The* `SourceType` *parameter contains the ContourType() function which returns an integer.*

*The* `SourceName` *parameter contains the ContourName() function which returns a string.*

*The* `SourceLength` *parameter containing the ContourLength() or ContourArea() function returning a real.*

*The* `IsPointSource?` *parameter containing the If(ContourType()=0, 1, 0) expression, is set to boolean to convert the integer return value, to a boolean.*

## Expressions and Parameter Types

Since expressions might contain other expressions of different types, it is you that have to decide what the type of the return-value of a parameter will be. If the actual value returned is of a different type than the one you have set, then Argus ONE tries to convert the actual type returned to the one you initially set.

### Explicit conversion of real numbers

If an expression within a parameter returns a Real (i.e. 5.43) and you set the parameter type to Integer, all digits after the decimal will be truncated (i.e. 5).

**Conversions of real numbers**

| *Parameter Type* | *Expression or Value Type* | *Conversion* | *Example* |
|---|---|---|---|
| Real | Real | None | 5.43 ➜ 5.43 |
| Integer | Real | Real ➜ Integer (truncation) | 5.43 ➜ 5 |
| Boolean | Real | Real ➜ Boolean (False if 0, otherwise, True) | 5.43 ➜ True<br>0.00 ➜ False |
| String | Real | Real ➜ String (all digits, signs, decimal and exponent are converted to characters) | 5.43➜ “5.43” |

### Explicit conversion of Integer numbers

If an expression within a parameter returns an Integer (i.e. 52) and you set the parameter type to String, all digits are converted to characters (i.e. “52”)

**Conversions of integer numbers**

| *Parameter Type* | *Expression or Value Type* | *Conversion* | *Example* |
|---|---|---|---|
| Integer | Integer | None | 52 ➜ 52 |
| Real | Integer | Integer ➜ Real | 52 ➜ 52 |
| Boolean | Integer | Boolean ➜ Integer (False if 0, otherwise, True) | 52 ➜ True<br>0 ➜ False |
| String | Integer | Integer ➜ String (all digits and signs are converted to characters) | -52 ➜ “-52” |

### Explicit conversion of Booleans

Booleans are internally represented within ArgusNE as Integers, where False is always represented as zero (0) and True can be any other integer. For instance if the expression `CountObjectsInBlock(layer_name)` returns `False` (that is, no objects were found) a zero (0) is returned. If any other number of objects is found within a block, the number found is returned and is presented as `True`.

Booleans are converted to Reals as Integers, and to Strings as "True" or "False".

### Explicit conversion of Strings

Strings are converted to other representations according to C language conventions. When the conversion returns a non-number a value of zero (0) or negative zero (-0) is returned. Since ArgusNE does not give any special indication to such a situation, it is your responsibility to make sure strings are converted to other types in a controlled manner.

If a string starts with digits and continues with characters, and is then converted to Real or Integer, all digits up to the first character are converted to a number. If the string starts with digits and contains the "e" character, all digits up to the "e" character are converted to a number in scientific notation.

**Conversions of Strings**

| *Parameter Type* | *Expression or Value Type* | *Conversion* | *Example* |
|---|---|---|---|
| String | String | None | abcd ➜ abcd<br>12.4 ➜ 12.4<br>1.e+1➜ 1e+1 |
| Real | String | String ➜ Real | abcd➜0 or -0<br>-12.3 ➜ -12.3<br>1e8 ➜ 1e+08<br>12abc ➜ 12 |
| Integer | String | String ➜ Integer | abcd➜0 or -0<br>52 ➜ 52<br>12abc ➜ 1 |
| Boolean | String | String ➜ Boolean (the string is first converted to Integer) | 0 ➜ False<br>!=0 ➜ True |

## Operators

A number of Arithmetic and Comparison operators now have different meaning when performed on strings.

### Operators valid for string manipulations

| Type of operator | Example | Operator | Description |
|---|---|---|---|
| **Concatenation** Combines strings and returns strings. | The expression “Hello”+“ World” returns the string “Hello World”. | + | Concatenate |
| **Comparison** Compares two strings and returns logical value `True (1)` or `False (0)`. | String comparisons are performed as if the strings are sorted alphabetically. Hence, “a”< “b” returns True, “ab”> “a” returns True, and “a”= “b” returns False. | = | Equal |
| | | > | Greater than |
| | | < | Less than |
| | | >= | Greater than or equal to |
| | | <= | Less than or equal to |
| | | != | Not equal to |

## Order of Evaluation

Operators applied to mixed expressions, that is, expressions containing variables returning different types, convert the variables to different types automatically.

Evaluation of expressions is performed according to the following rules:

1. The expression is first evaluated according to the order of evaluation of operators as explained in chapter “Layer Parameters”, “Order of Evaluation” on page 22..
2. It is then evaluated from left to right. Implicit conversions are automatically performed by Argus ONE to resolve undefined operations.

Examples:

- `(3>1)+1+3.2+ a”=”5.2a”`
- `”a”+(3>1)+1+3.2=”aTrue13.2”`
- `”a”*2=0` - Since the * operator is an undefined string operator, the string “a” is first converted to an integer.
- `“10a”*2=20`

## Search Options for Supporting the Various Parameter Types

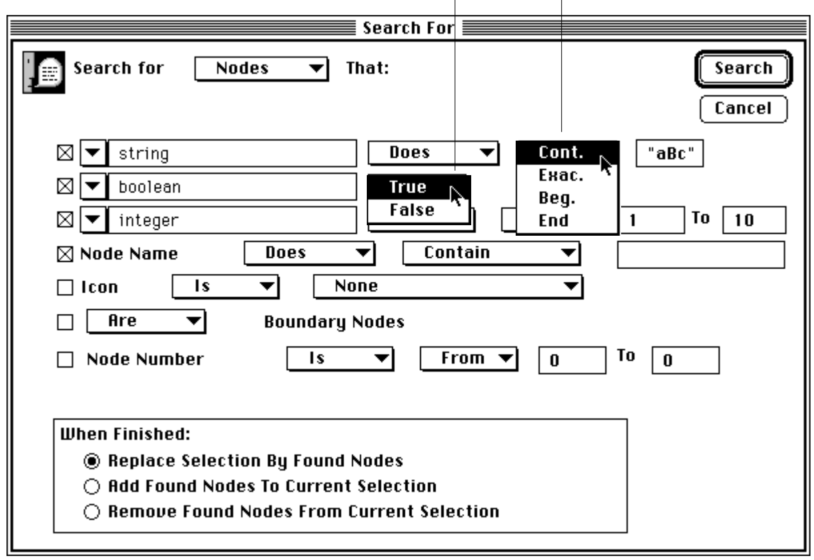
The search dialog supports all the new parameter types available in Argus ONE. The new Boolean and String types require special treatment in the search dialog. When you select a parameter to be searched Argus ONE automatically sets the search criteria to fit the parameter type.

### To define a search criteria for a string type parameter

1. Select a parameter. If the parameter is a string parameter the “From” and “to” range popup menus automatically change to the “Cont.” range popup menu and the “Is” negation popup menu changes to a “Does” popup menu.
2. In the text edit box type the string or substring between quotation marks. The search is case sensitive.


3. From the negation popup menu select “Does” or “Does not”.
4. From the range popup menu select “Cont.” (contain), “Exac.” (exact), “Beg,” (begin with), or “End” (end with).
5. Select all other appropriate parameters and search criteria required.
6. Click the Search button.

*The Boolean range popup menu.* *The String range popup menu.*




### To define a search criteria for a boolean type parameter

1. Select the parameter to be searched. If the parameter is of type boolean, the “From” and “to” range popup menus automatically disappear and a single “True” popup menu appears.
2. From the “True’ popup menu select “True” or “False”.
3. Select all other appropriate parameters and search criteria required.
4. Click the Search button.

# New Functions

Many new functions, as well as function families, which have been added in version 3.0 are listed here by category.

## New String Functions

String functions are a new family of functions available in ArgusNE. With the introduction of string values some string function and operators were added to allow manipulation of strings. If you have suggestions for additional string functions please contact us.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Contains*(string, substring)*** | ***string*** <br> ***substring*** | ***string*** is the string of characters in which ***substring*** is to be searched for. Both arguments are case sensitive, and thus the function is also case sensitive. <br><br> Returns True (1) if ***substring*** was found within ***string***. <br><br> Returns False (0) if ***substring*** was not found within ***string***. |

Example:

`Contains("Argus","Arg")` returns True (1)
`Contains("Argus","Args")` returns False (0)

`Contains(ContourName(Sources),"Well")` will return True (1) for all contours (in layer Sources) assigned a string containing the substring "Wells".

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Length*(string)*** | ***string*** | ***string*** is the string whose number of characters you wish to count. <br><br> Returns the number of characters in ***string*** (Integer) |

Example:

`Length("Argus")` returns 5.
`Length("Argus"+"Numerical Environments")` returns 27.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Lower*(string)*** | ***string*** | ***string*** is the string whose characters are to be converted to lower case.<br><br>Returns ***string*** in lower case. |

Example:

`Lower("ARGUS")` returns `argus`.
`Lower("Argus")` returns `argus`.
`Lower("argus")` returns `argus`.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **StrIndex*(string, substring)*** | ***string***<br>***substring*** | ***string*** is the string of characters in which you wish to find where ***substring*** starts. Both arguments are case sensitive.<br><br>Returns the location (Integer) of the first character of ***substring*** within ***string***.<br><br>If ***substring*** is not present within ***string***, value returned is (0). |

Example:

`StrIndex("Argus","gus")` returns `3`.
`StrIndex("Argus","argus")` returns `0`.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Substring*(string, from,count)*** | ***string***<br>***from***<br>***count*** | Returns the string of characters of length ***count*** starting at (and including) ***from*** position in ***string***.<br><br>If ***from*** is smaller or larger than ***string*** returns an empty string (“”).<br>If ***from*** is negative, it is counted from the length of ***string***.<br>If ***count*** is negative returns the string starting at ***from*** minus ***count***. |

Example:

`Substring("Argus",3,3)` returns the string `"gus"`.
`Substring("Argus",-1,-2)` returns the string `"us"`.
`Substring("Argus",-1,2)` returns the empty string `""`.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Upper*(string)*** | ***string*** | ***string*** is the string whose characters are to be converted to upper case.<br>Returns ***string*** in upper case. |

Example:

`Upper(“argus”)` returns `ARGUS`.
`Upper(“Argus”)` returns `ARGUS`.
`Upper(“argus”)` returns `ARGUS`.
`Upper(Lower(“Argus”))` returns `ARGUS`

## New Layer Functions

Layer functions are a new family of functions available in ArgusNE. They currently include the following functions:

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **LayerName*()*** | ***no arguments*** | Returns the name (string) of the layer. The return value is case sensitive. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NumParameters*()*** | ***no arguments*** | Returns the number of parameters assigned to the layer. Previously this function was only available for mesh and grid layers. It is now available for all layer types. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthParamName (Parameter_Number)** | **Parameter_Number** | The number of the layer parameter in order of appearance in the layer dialog.<br><br>Returns the name (string) of the **Parameter_Number** layer parameter. The return value is case sensitive. |

## New Contour Functions

Contour functions are a new family of functions available in ArgusNE. These functions, available in Information type layers, allow you to access contour information such as the contour length, area, and type. They currently include the following functions:

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **ContourArea*()*** | ***no arguments*** | Returns the area (real) of a contour.<br>If the contour is an open or point contour the value returned is zero (0). |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **ContourIcon*()*** | ***no arguments*** | Returns the index (integer) of the icon assigned to the contour. If the contour is not assigned an icon, returns 0 (zero). Icons are indexed from 1, according to their order of appearance in the Contour Icon pop-up menu. |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **ContourLength*()*** | ***no arguments*** | Returns the length (real) of a contour. |

Example: If an open contour object represents a line boundary condition such as a river, a fault or just a line boundary, and the mass contributed by the object is given as the total mass (and not mass per unit length) then the mass per unit length can be calculated by the expression:

```
Rivers.Mass/ContourLength()
```

If the information layer contains point objects as well, the following expression prevents accidental division by zero:

```
Rivers.Mass/If(ContourType=1,ContourLength(),1)
```

This function can also be used to calculate a quantity per unit length on closed contours representing for instance, internal boundaries.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **ContourName*()*** | ***no arguments*** | Returns the name (string) assigned to the contour. If the contour is not assigned a name, returns an empty string. The return value is case sensitive. |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **CountObjectsIn Contour*( layer_name,*** *contour_type* ***)*** | ***layer_name*** | The name of the layer for which you need to count the number of objects of type *contour_type* intersecting the contour or contained in it. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the number (integer) of objects of type *contour_type* in layer ***layer_name***, intersecting or contained in the contour. If *contour_type* is omitted, returns the total number of all objects intersecting or contained in the contour. |

Example: The function `CountObjectsInContour(Domain Layer)` counts the number of objects in the layer “Domain Layer” intersecting or contained by the contour and returns:
0 — if the contour is not intersected by, or does not contain any object.
1 — if the contour is intersected by, or contains one object of any type.
N — if the contour is intersected by, or contains N objects of any type.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **ContourType*()*** | ***no argument*** | Returns the type of the contour:<br><br>1 for a point contour<br>2 for an open contour<br>3 for a close contour |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthVertexX *(number)*** | ***number*** | Returns the X coordinate of the contour's ***number*** vertex. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NthVertexY *(number)*** | ***number*** | Returns the Y coordinate of the contour's ***number*** vertex. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NumVertices*()*** | ***no argument*** | Returns a contour's number of vertices. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **SumObjectsInContour*( parameter_name,*** *contour_type* ***)*** | ***parameter_name*** | The name of an information type layer parameter for which you need to sum the values of objects of type *contour_type* intersecting the contour or contained in it. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects |
| | | Returns the sum of the objects' values of type *contour_type* in parameter ***parameter_name***, intersecting or contained in the contour. If *contour_type* is omitted, returns the sum of values of all objects intersecting or contained in the contour. |

Example: The function `SumObjectsInContour(Wells Layer,0)` can be used to sum the values of the specified variable of the point objects which are in the layer "Wells Layer" and that are contained by the contour. If divided by `CountObjectsInContour(Wells Layer,0)` it shall return the average value of the specified function.

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **WSumObjectsInContour(*parameter_name,*** *contour_type*) | ***parameter_name*** | The name of an information type layer parameter in which you need to calculate the weighted sum of objects of type *contour_type* intersecting the contour or contained in it.<br>The weighted sum is defined as the sum of the products of each intersecting segment length, multiplied by its value. |
| | *contour_type (optional)* | 0 for Point objects<br>1 for Open Contour objects<br>2 for Close Contour objects<br><br>Returns the weighted sum of objects' values of type *contour_type* in parameter ***parameter_name***, intersecting or contained in the contour. If *contour_type* is omitted, returns the weighted sum of values of all objects intersecting or contained in the contour. |

Example: The function `WSumObjectsInContour(Fractures Layer,1)` computes the product of each intersecting or contained segment length by its value and sums them.

For additional examples refer to the functions `WSumObjectsInElement`, and `WSumObjectsInBlock`.

## New Grid Layer Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **CountBlocks** ***(condition)*** | ***condition*** | ***condition*** can be any logical expression returning True (1) or False (0). Returns the number of blocks in a grid layer which satisfy the ***condition***. |

Example: Some models require that you provide the number of blocks satisfying a condition. These may be the number of blocks in which you specify boundary conditions. Use this function in an export template to provide the model with the information required.

For instance, if the model requires the number of blocks through which rivers are flowing, create the expression:
`CountBlocks(CountObjectsInBlock(River,1)!=0)`

Note: Try not to use this function from within a grid parameter as it performs blocks^2 times.

## New Block Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockIcon*()*** | ***no arguments*** | Returns the index (integer) of the icon assigned to a block. If the block is not assigned an icon, returns 0 (zero). Icons are indexed from 1, according to their order of appearance in the Block Icon pop-up menu. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **EffectiveValue** ***(Parameter_name)*** | ***Parameter_Name*** | Intersects block boundary with all contours in the layer of ***Parameter_Name*** to create a set of polygons, and returns the sum of polygons areas multiplied by their respective values. |

Example: See example about EffectiveValue for mesh elements on “New Element Functions” on page 33s

## New Mesh Layer Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BandWidth*(*** *[with_diagonals]****)*** | *with_diagonals* | Returns the half-bandwidth of a mesh.<br><br>If the optional argument *with_diagonals* is used, returns the half-bandwidth including diagonals in a Quadrilateral mesh. The argument *with_diagonals* is not used for Triangular meshes.<br><br>If more than one continuous mesh is present in a mesh layer, returns the largest half-bandwidth of all meshes.<br><br>For a detailed discussion of bandwidth refer to “Renumbering (BandWidth Optimization)” on page 238. |

Example: Some models require that you provide the bandwidth as part of their input. This may serve the model to dimension arrays and matrices. Use this function in an export template to provide the model with the information required. For instance, if the model requires the full bandwidth create the expression: `Bandwidth()*2+1`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **CountElements*(*** ***condition)*** | ***condition*** | ***condition*** can be any logical expression returning True (1) or False (0).<br><br>Returns the number of elements in a mesh layer which satisfy the ***condition***. |

Example: Some models require that you provide the number of elements satisfying a condition. These may be the number of elements in which you specify boundary conditions. Use this function in an export template to provide the model with the information required.

For instance, if the model requires the number of elements through which rivers are flowing, create the expression:
`CountElements(CountObjectsInElement(River,1)!=0)`

Note: Try not to use this function from within a mesh parameter as it performs elements^2 times.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **CountNodes( *condition)*** | ***condition*** | ***condition*** can be any logical expression returning True (1) or False (0). Returns the number of nodes in a mesh layer which satisfy the ***condition***. |

Example: Some models require that you provide the number of nodes satisfying a condition. These may be the number of nodes in which you specify boundary conditions. Use this function in an export template to provide the model with the information required.

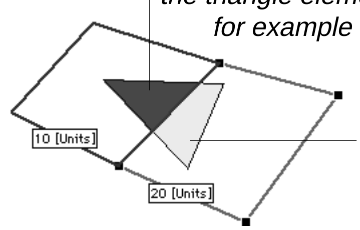
For instance, if the model requires the number of nodes acting as sources, create the expression: `CountNodes(NodeAboveCntr(Wells,0))`

Note: Try not to use this function from within a mesh parameter as it performs nodes^2 times.

## New Element Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **EffectiveValue *(Parameter_name)*** | ***Parameter_Name*** | Intersects element boundary with all contours in ***Parameter_Name*** to create a set of polygons, and returns the sum of polygons areas multiplied by their respective values. |

*The dark gray polygon is created by the intersection of the triangle element and the left contour of value 10. for example area=0.8*




*The light gray polygon is created by the intersection of the triangle element and the right contour of value 20. for example area=0.5*

*The value returned for the element (=21) is the sum of the light polygon area (0.8) multiplied by 20 (16) and that of the dark one (=0.5) multiplied by 10 (=5).*

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **ElementIcon*()*** | ***no arguments*** | Returns the index (integer) of the icon assigned to the element. If the element is not assigned an icon, returns 0 (zero). Icons are indexed from 1, according to their order of appearance in the Element Icon pop-up menu. |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **ElementName*()*** | ***no arguments*** | Returns the name (string) assigned to the element. If the element is not assigned a name, returns an empty string. The return value is case sensitive. |

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NthNodeX*(node_num)*** | ***1, 2, 3, or 4 (4 only in quad mesh layers)*** | Returns the X position of the element's ***node_num*** node. |

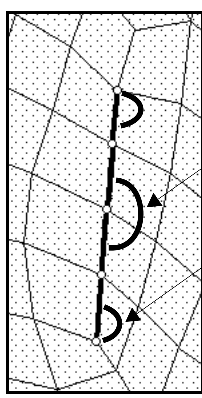
| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NthNodeY*(node_num)*** | ***1, 2, 3, or 4 (4 only in quad mesh layers)*** | Returns the Y position of the element's ***node_num*** node. |

## New Node Functions

| Name & Syntax | Arguments | Description & Return Value |
|---|---|---|
| **NodeAboveCntr*(* *layer_name*** *[,epsilon]***)*** | ***layer_name***<br><br>*epsilon* | The name of the layer in which you want to check if the node lies above any contour object within an *epsilon* range.<br><br>Returns the type of object the node lies above or within *layer_name*:<br>0 - If the node does not lie above or within any contour.<br>1 - If the node lies above a point object.<br>2 - If the node lies above an open contour.<br>3 - If the node lies above a close contour.<br>4 - If the node lies inside a close contour. |

Example: See example in Supplements for version 2.5 pp.47s. (note that the epsilon is a new option and that the correct function definition is the one above.

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NodeEffectiveLength(** ***layer_name*** *[,epsilon]* **)** | ***layer_name*** | The name of the layer containing the contour object for which you want to calculate the NodeEffectiveLength. |
| | *epsilon* | *epsilon* is the distance between the object and the node within which the node will still be considered to lie above a contour.<br><br>Returns half of the length of the two segments connected to the node which lies above the contour object.<br><br>If the node is the first or the last node above an open contour, it returns the half length of only the last segment. |



Example: `NodeEffectiveLength(Rivers)` returns half the length of segments connected to the node which also lie above the open contour object in `Rivers` layer.

For middle nodes, it returns half the length of both segments connected to it and which also lie above the object.

For an end node, it returns half the length of the single segment connected to it and which also lies above the object.

Can be used for boundary conditions. For instance, to calculate the mass contributed to a node from a river or a fracture, create the following node parameter expression: `Rivers*NodeEffectiveLength(Rivers).`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NodeIcon*()*** | ***no arguments*** | Returns the index (integer) of the icon assigned to the node. If the node is not assigned an icon, returns 0 (zero). Icons are indexed from 1, according to their order of appearance in the Node Icon pop-up menu. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **NodeName*()*** | ***no arguments*** | Returns the name (string) assigned to the node. If the node is not assigned a name, returns an empty string. The return value is case sensitive. |

# New Functionality in Information Layers

Information type layers (including domain and density type layers) and the contours they store have been enhanced to support the following capabilities:

- Contour vertices editing tools.
- The contour’s icon can now be copied and pasted with the contour.
- New preferences for contour copying.
- The Information ruler presents the value of the parameter selected in the Colors legend.

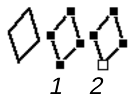
## Editing Contours and Contour Vertices

Argus ONE version 3 allows you to select and edit any number of contour vertices at once. This include the ability to move, delete, copy and paste vertices and contour segments. These new capabilities are very important for assigning boundary conditions using GIS functions.

### Selecting contours and contour vertices

To allow for vertices selection the selection of contours is now a two stage operation. The first time you click a contour it is selected and all its vertices are highlighted (as in pre version 3). If you click a vertex when the contour is selected the vertex is selected and appears as a hollow bullet.

### To select a vertex




1. Click to select the contour the vertex belongs to.
2. Click the vertex.
   The vertex is selected and appears as a hollow rectangle.

### To add vertices to the selection

- While holding the SHIFT key click the vertex to be added to the selection.

### To add multiple vertices to the selection all at once - using the arrow tool

1. From the tool palette select the arrow tool.
2. While holding down the SHIFT key click and drag the mouse to create a stretch rectangle around the vertices to be added to the selection.
3. When all vertices are within the stretch rectangle, release the mouse button.

### To add multiple vertices to the selection all at once - using the lasso tool

1. From the tool palette select the lasso tool.
2. While holding down the SHIFT key click and drag the mouse around the vertices to be added to the selection.
3. When all vertices are within the lasso, release the mouse button.

### To deselect a vertex

- While holding the SHIFT key click the vertex to be deselected.

### To deselect multiple vertices all at once - using the arrow tool

1. From the tool palette select the arrow tool.
2. While holding down the SHIFT key click and drag the mouse to create a stretch rectangle around the vertices to be deselected.
3. When all vertices to be deselected are within the stretch rectangle, release the mouse button.

### To deselect multiple vertices at once - using the lasso tool

1. From the tool palette select the lasso tool.
2. While holding down the SHIFT key click and drag the mouse around the vertices to be deselected.
3. When all vertices to be deselected are within the lasso, release the mouse button.

### To deselect all vertices in a contour at once

In case only one contour is selected, or you want to deselect all contours:

- Just click outside the contour to deselect its vertices.

In case more than one contour is selected and you do not wish to deselect other contours or vertices:

- SHIFT-click within the contour.

### To add vertices from another contour to the selection

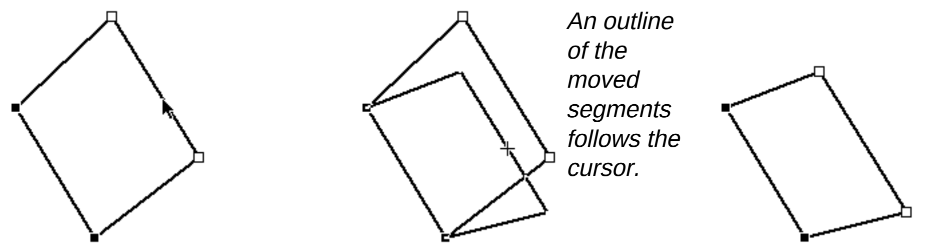
1. SHIFT click to select the other contour.
2. Use any of the above methods to add vertices to the selection.

## Reshaping contours

Reshaping contours can be performed by moving vertices and segments or by deleting vertices.

### To move vertices in a contour

1. Select the vertices to be moved.
2. Click-drag the mouse on any of the segments connected to a selected vertex or on one of the selected vertices.
   All selected vertices follow your cursor's position.
3. Release the mouse button in the desired location.




*1. Select the vertices to be moved.*

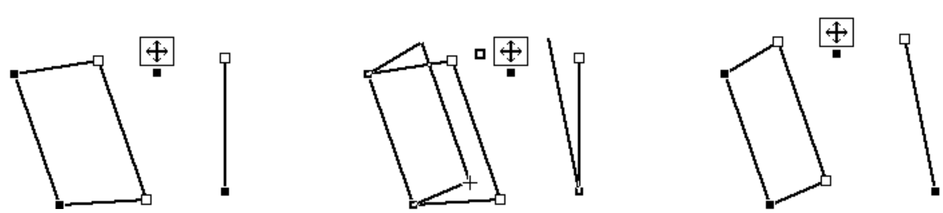
*2. Click-drag one of the selected vertices or one of the segments connected to the selected vertices.*

*3. Release the mouse button to reposition the vertices.*

### To move vertices in several contours all at once

Vertices from several contours as well as complete contours can be moved together. If all vertices within a contour are selected, or the contour is selected, it will be moved together with all other selected vertices.

1. Select the vertices and contours to be moved.
2. Click-drag the mouse on any of the segments connected to a selected vertex or on one of the vertices selected.
   All selected vertices and contours follow your cursor's position.
3. Release the mouse button in the desired location.



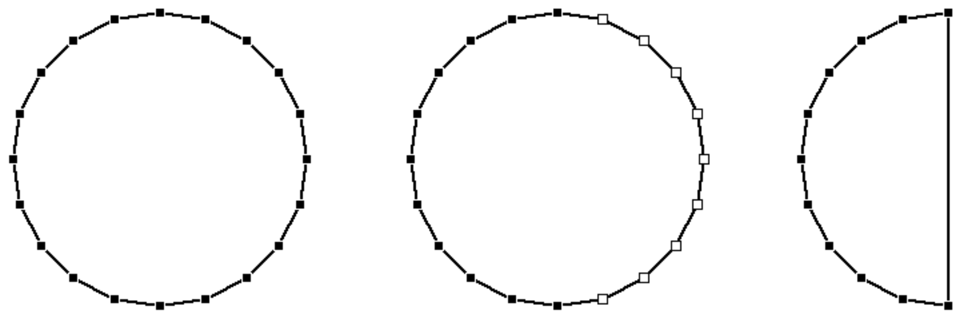
*1. Select the vertices/ contours to be moved.*

*2. Click-drag a selected vertex, segment or contour.*

*3. Release the mouse button.*

#### To delete or cut vertices

1. Select all contours from which vertices are to be deleted.
2. Select all vertices to be deleted.
3. Press the DELETE key on your keyboard,
   -Or-
   From the Edit menu select Clear or Cut.



*1. Select contours from which vertices are to be deleted.* *2. Select all vertices to be deleted.* *3. Press the delete key or select Clear or Cut from the Edit menu.*

All segments which are connected to a selected vertex are removed.

#### To copy or cut vertices

1. Select all contours from which vertices are to be copied or cut.
2. Select all vertices to be copied or cut.
3. From the Edit menu select Copy or Cut.

The contours and contour parts copied or cut are placed in the clipboard. You can now paste them in any Information or Maps type layers, or in any text editor.

### Undoing Contour Editing

All new editing commands are undoable.

### Contour validity tests

All contour validity tests (User's Guide, "The Rules" on page 95) are performed while you edit the contours. To avoid contour validity tests you can either work in Maps type layers or, if necessary and appropriate, use the Allow Intersection option in Information type layers.

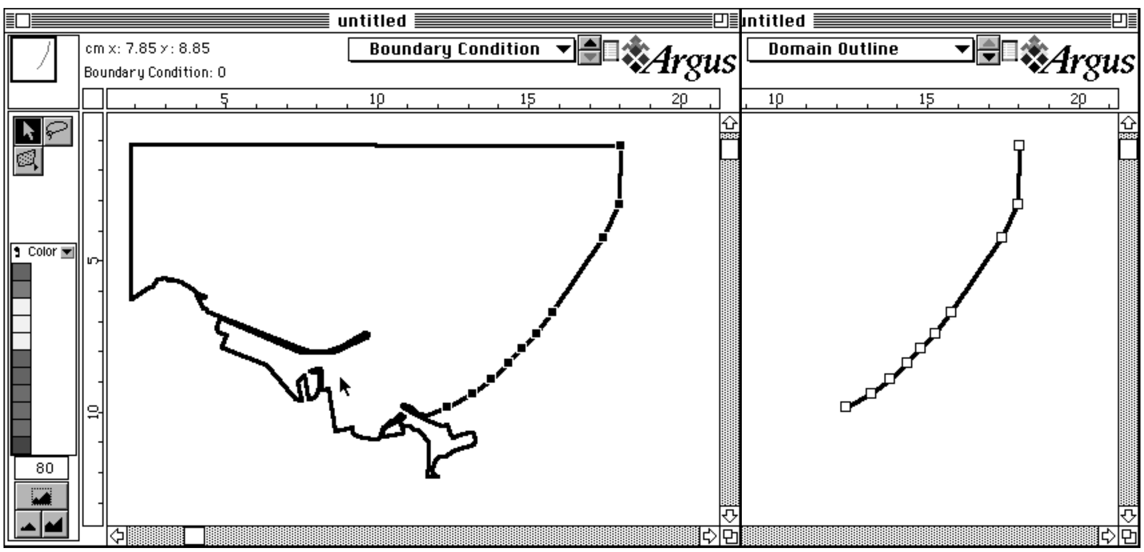
### Using copy of parts of contours for setting boundary conditions

Using the mesh function `NodeAboveCntr(layer_name)` to specify line type boundary conditions for nodes which lie along domain boundaries required the use of the epsilon distance optional parameter. This was true since one could not ascertain that an open contour object in a boundary condition layer would lie precisely under the domain outline contour. The disadvantage of using the epsilon option is that it is scale dependent.

Using the new contour copying and pasting capabilities you can now copy the part of the domain outline on which boundary conditions are to be defined and paste it in the boundary condition layer such that it is positioned precisely below the domain outline.

**To copy parts of a domain outline into a boundary condition layer**

1. In a domain layer, select all vertices you want to copy and paste in the boundary condition layer.
2. From the File menu select Copy.
3. Activate the boundary condition layer (and hide the domain layer).
4. From the File menu select Paste.




If in the above screen-capture the part of the domain outline is copied to an information layer you can use the node mesh function `NodeAboveCntr(layer_name)` to automatically locate all nodes which lie above the part of the contour copied to the information layer.

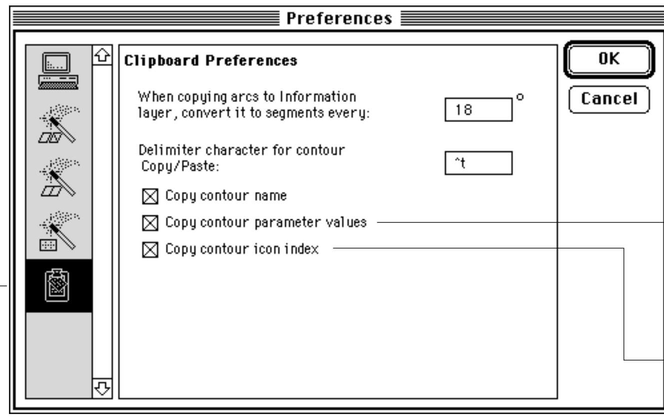
If the value of a boundary condition needs to be applied with respect to the length of the boundary which effects each node (i.e. half of the distance, in both directions, between the node and its neighbor nodes) you can use the `NodeEffectiveLength(layer_name)` without specifying an epsilon.

## New Contour Copying and Pasting Capabilities

You can now choose if you want Argus ONE to copy the contours' parameters values and icons when copying contours.

1. From the Special menu choose Preferences...
2. Click the clipboard preferences icon.
3. Check the “Copy contour parameter values” check box.
4. Check the “Copy contour icon index” check box.

*The Clipboard Preferences dialog.*




*Check to copy all contour parameter values when copying a contour.*

*Check to copy contour icon index when copying a contour.*

*The clipboard preferences icon.*

### Example of a contour copied with both values and icon index

## Name:
## Icon:1

| | |
|---|---|
| # Points Count | Value |
| 10 | 2 |
| # X pos | Y pos |
| 18.04134661 | 1.20436226 |
| 18.00521574 | 3.13134187 |
| 17.46325273 | 4.22731152 |
| 15.76510194 | 6.69625415 |
| 15.2592698 | 7.40682788 |
| 14.77752489 | 7.87652916 |
| 14.31986724 | 8.38236131 |
| 13.74177335 | 8.92432432 |
| 13.15163585 | 9.38198198 |
| 12.30858227 | 9.83963964 |

## Information Ruler in Information Layers

The information ruler can now present values on any of the layer parameters and not only the first parameter. To show the values of other parameters select the required parameter from the Layer’s parameter popup menu in the color palette.

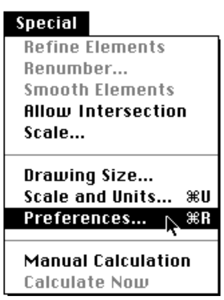
# New Functionality in Mesh Layers

A number of new options for controlling Auto Mesh Generation (AMG) have been added. These and some other new mesh layer capabilities include:

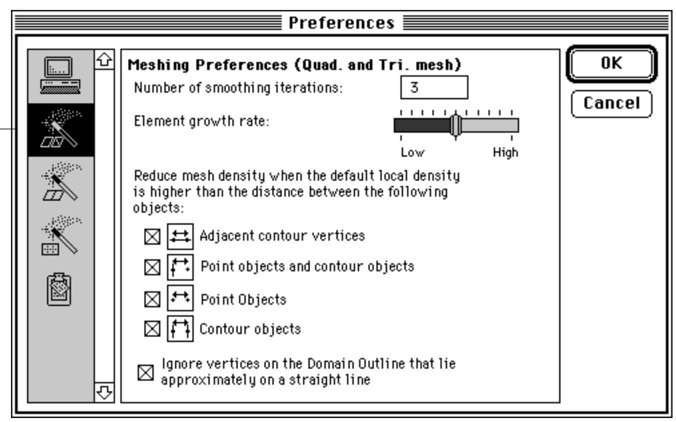
- The meshing preferences have been separated into two groups.
- Control of node creation above domain outline vertices which lie along straight, or almost straight domain outline lines.
- Enhance cleanup. Applies to quadrilateral meshes only.
- Elimination of triangular shaped elements within quadrilateral meshes. Applies to quadrilateral mesh types only.
- Control of bandwidth calculation in quadrilateral meshes.
- Control of element side visibility.
- New tool for creating orthogonal quadrilateral meshes.

## Meshing Preferences

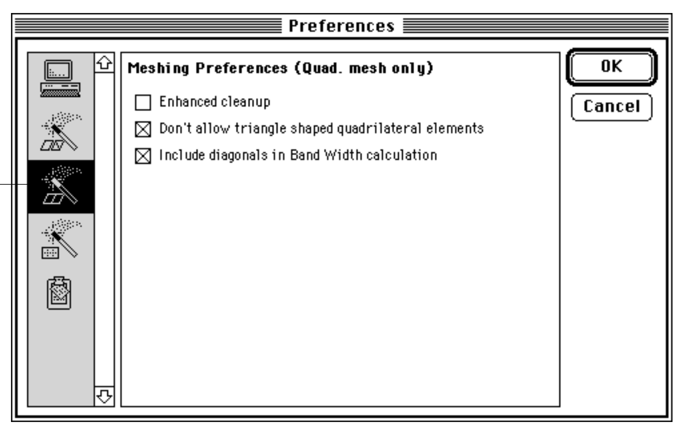
The preferences dialog now contains two mesh preferences panels, the general meshing and the quadrilateral meshing panels.




*The General Meshing preferences panel. Controls affect both triangular and quadrilateral meshes.*



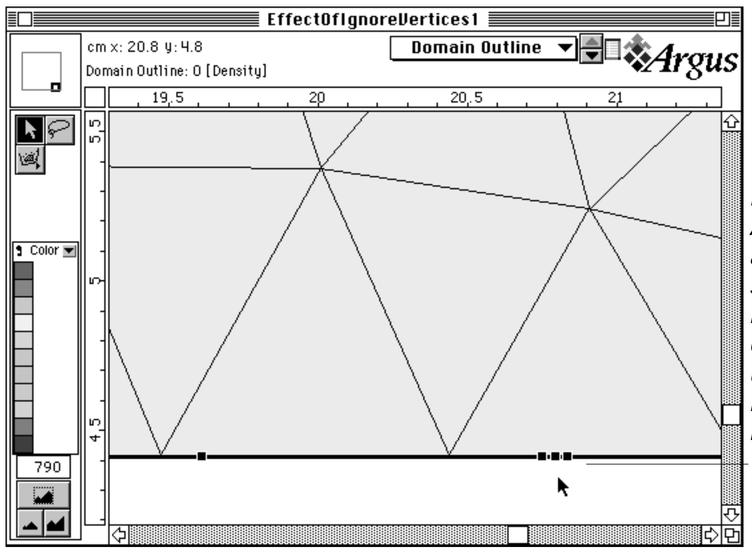

*The Quadrilateral Meshing preferences panel. Controls affect only quadrilateral meshes.*



## Control of Node Creation above Domain Outline Vertices

The Argus ONE meshing algorithm places nodes above the domain outline according to two rules; where a vertex exists, and between existing vertices according to the computed density. If the distance between two adjacent vertices along the domain outline is smaller than the user specified density, the computed density in that area is reduced to fit this small distance.

Sometimes however, such vertices although very close to each other, are located along a straight line or an approximately straight line. In such cases, such vertices might be redundant. By default, the Argus ONE AMG algorithm ignores such vertices thus reducing the creation of unnecessary nodes along and inside the domain.




*By default, the Argus ONE AMG algorithm ignores such vertices thus reducing the creation of unnecessary nodes along and inside the domain.*

In other instances however, it might be required that such vertices shall force nodes above them. You can now control whether you wish the AMG to ignore such vertices or not.

### If vertices lie along lines which are approximately straight

If the angle between two consecutive segments in the domain outline is small and the segments are long, then such a vertex might be ignored and nodes might not be placed above the domain outline contour.

If you must have all boundary nodes placed exactly above the domain outline (so that the areas of the domain outline and the mesh are equal), or if you intend to use the `NodeAboveCntr(layer_name)` function with open contours, it is imperative that you turn this option off.

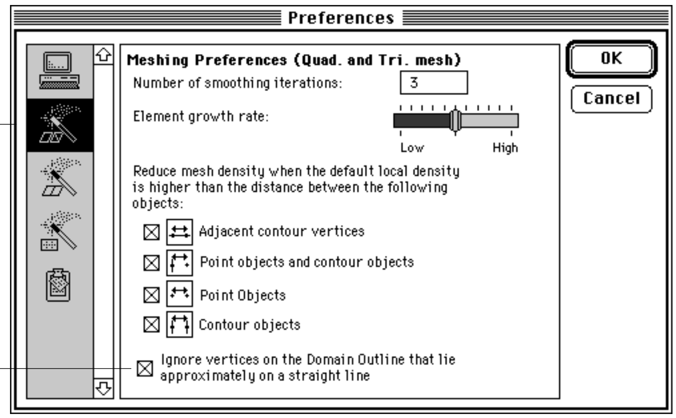
To quickly find out if boundary nodes do not lie exactly above the domain outline, compare the mesh area (appears in the Mesh Info dialog) and the domain outline area (double-click the domain outline to open the contour info dialog).

### To force node creation above all domain outline vertices

**1.** From the Special menu select Preferences...

2. Select the General Meshing preferences panel.

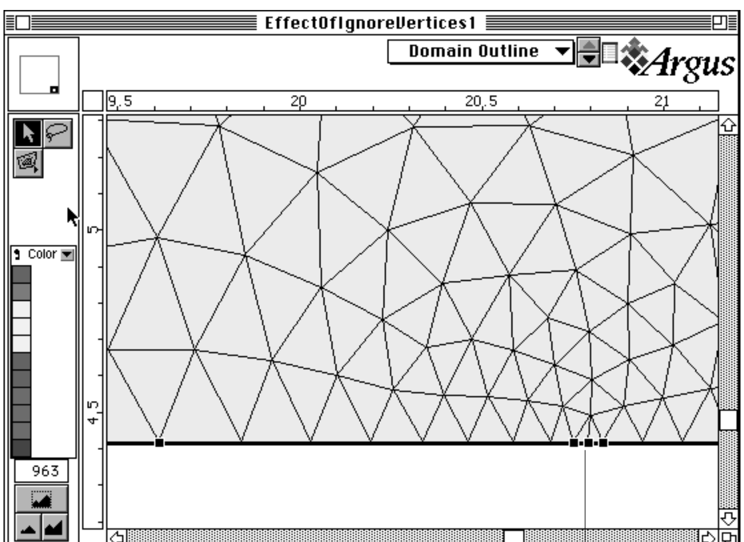
3. Uncheck to force creation of nodes above all domain outline vertices.




**2.** Click the General Meshing preferences panel.

**3.** Uncheck the “Ignore vertices...” button.

**4.** Close the preferences dialog.




Density is reduced and nodes are placed above all domain vertices.

## Enhanced Cleanup of Quadrilateral Meshes

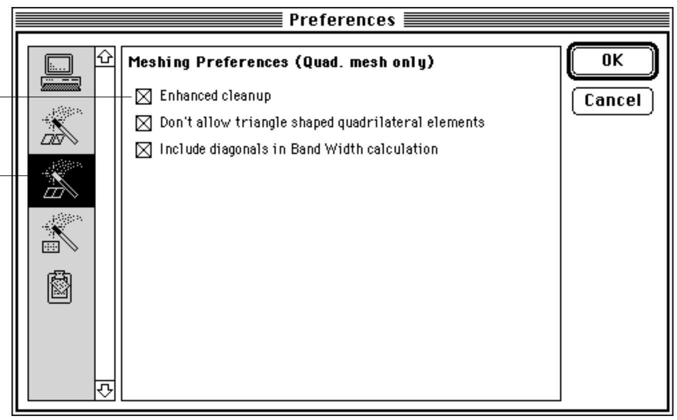
Under certain circumstances the smoothing algorithm might create ill shaped quad elements. If such elements are created try to re-mesh when the enhanced cleanup option is turned on.

### To turn enhanced cleanup on

1. From the Special menu select Preferences...
2. Click the Quadrilateral Meshing preferences panel.
3. Check the Enhanced cleanup check box.

*3. Check the Enhanced cleanup.*

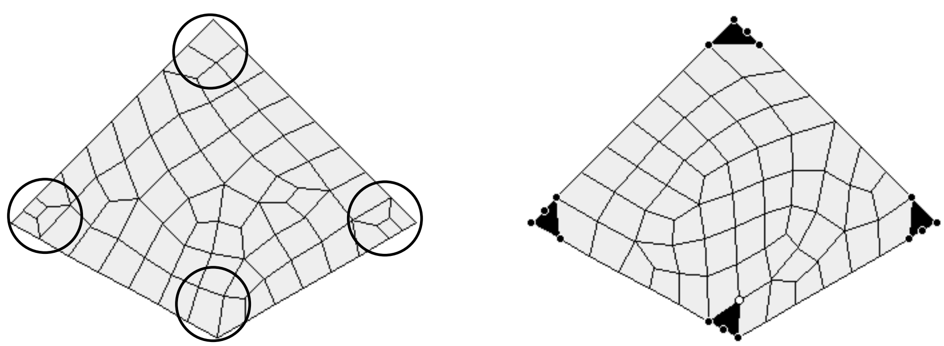
*2. Select the Quad Meshing preferences panel.*




## Eliminating Triangular Shaped Elements in Quad Meshes

Since many numerical codes assume that no triangular shaped elements (3 nodes along an element side) are present in a mesh, Argus ONE is set by default to eliminate such elements.

If however, your code allows for such elements, you can change the default setting and allow the quadrilateral meshing algorithm to create triangular shaped elements. In areas where triangular elements are created the ratio between adjacent elements might be superior to that of the mesh where triangles were eliminated.



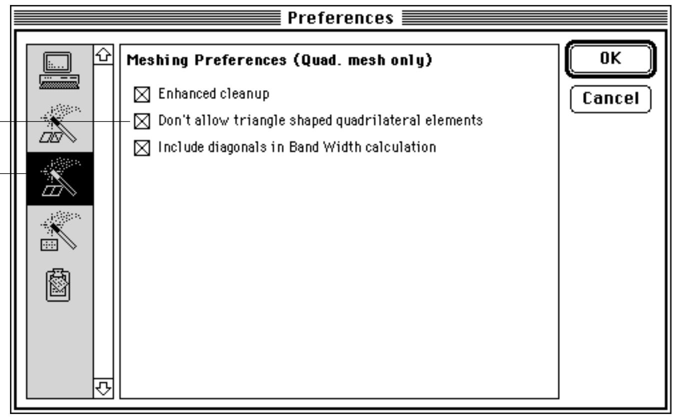
Triangular elements eliminated - Default. Triangular elements retained.

### To allow for triangular shaped elements within a quad mesh

1. From the Special menu select Preferences...
2. Click the Quadrilateral Meshing preferences panel.
3. Uncheck the “Don’t allow...” check box to retain triangular elements.

*3. Uncheck to retain triangular elements.*

*2. Select the Quad Meshing preferences panel.*




## Bandwidth Calculation in Quadrilateral Meshes

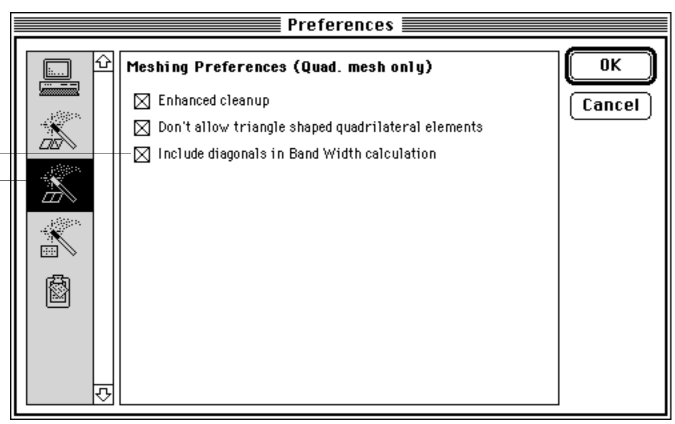
Bandwidth calculation in Quad mesh layers now takes into account all connectivity combinations including node connectivities along diagonals. For compatibility sake you can turn this option off to have the bandwidth calculated as in triangular elements.

### To calculate triangular bandwidth for a quad mesh

1. From the Special menu select Preferences...
2. Click the Quadrilateral Meshing preferences panel.
3. Uncheck the “Include diagonals...” check box to calculate triangular bandwidth.

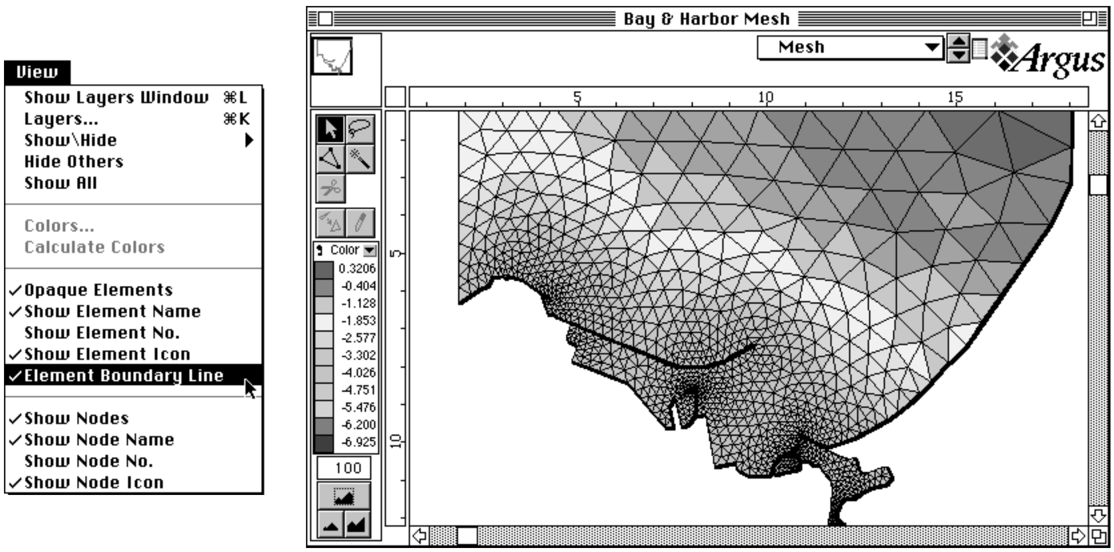
*3. Uncheck to calculate triangular bandwidth.*

*2. Select the Quad Meshing preferences panel.*



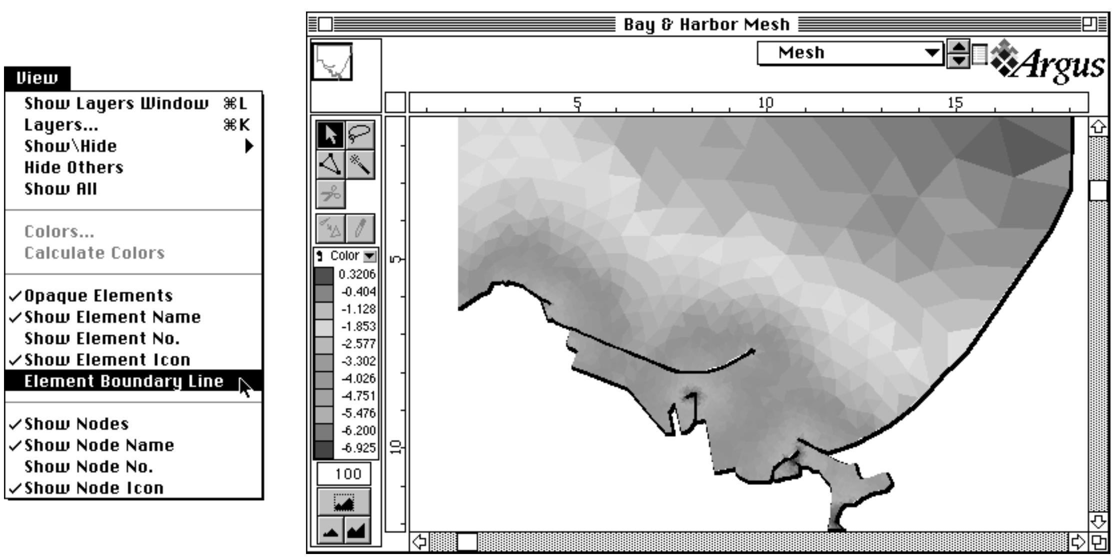
## Showing and Hiding Element Boundary Lines

You can now turn off the drawing of element boundary lines (element sides). This might prove handy if you evaluate the mesh elements by color using the color palette and popup menu.




### To show/hide element boundary lines

- From the View menu select Element Boundary Line.

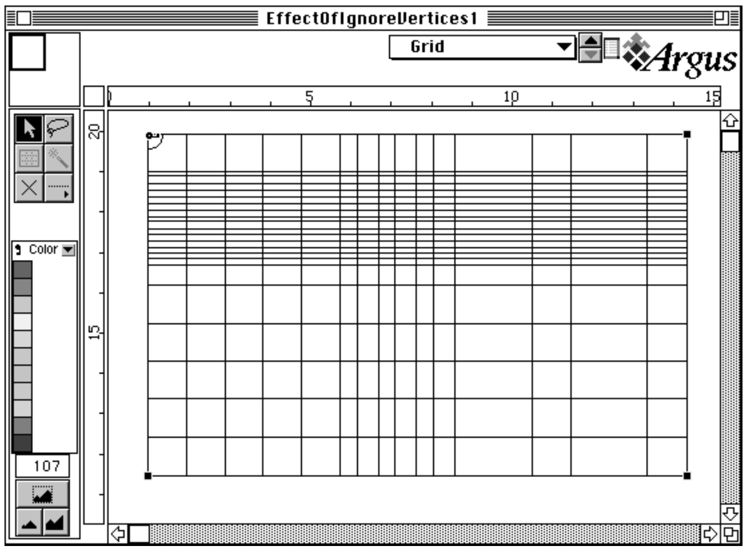


## Creating Orthogonal Quadrilateral Meshes

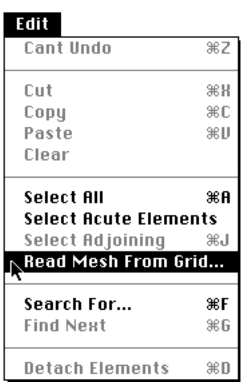
To quickly create an orthogonal quad mesh you can import a grid from a grid layer into a quad mesh layer. The grid is automatically exported by Argus ONE as a quad mesh and then imported back into the active quad mesh layer.

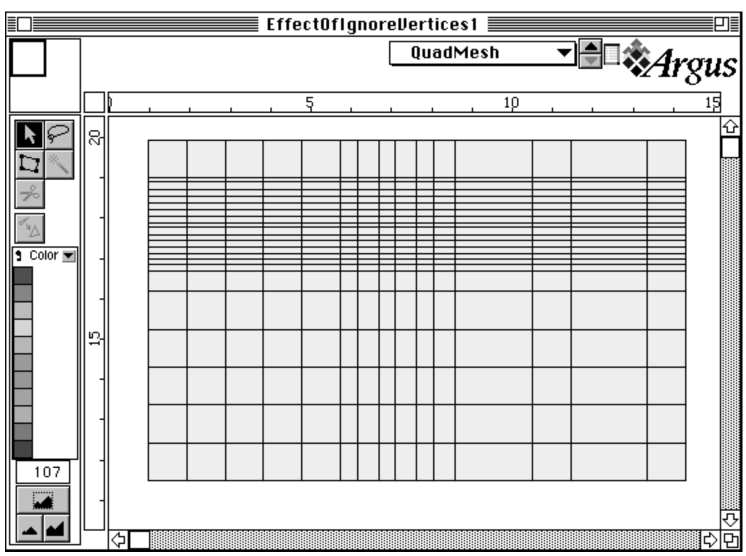
### To create an orthogonal quad mesh from a grid

1. In a grid layer, create a grid and refine it as you wish.



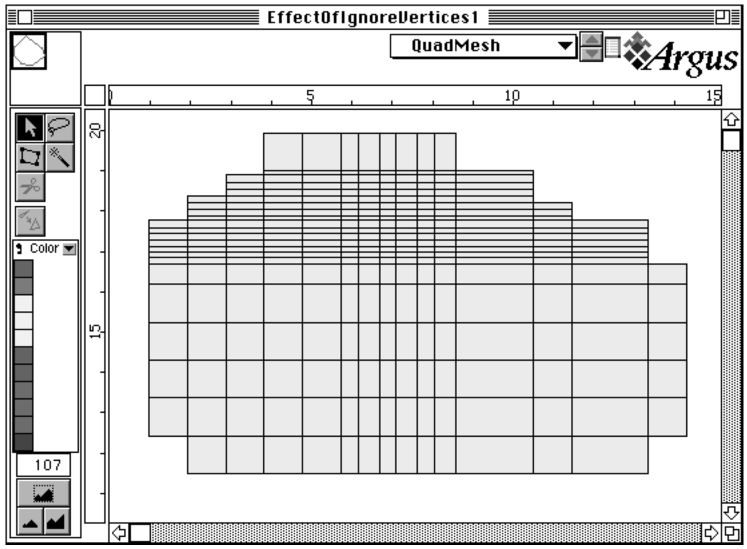

2. Activate a quad mesh layer.
3. From the Edit menu select Read Mesh from Grid...



The quad mesh created can be edited and operated on as any other mesh.

If the grid contains deactivated blocks they are not imported into the mesh layer.




## Additional Meshing Engines through the PIE Technology

Additional meshing engines can be incorporated using the Argus ONE PIE technology. An example of such a PIE is available free of charge from the Argus ftp site.

# New Grid Layer Capabilities

A number of new grid capabilities have been added. These include:

- Import a grid from a text file - Allows you to import existing grids from text files.
- Control of grid numbering - Allows you to control the direction of rows and column numbering with respect to the coordinate system.
- Additional block deactivation method - Allows you to select one of two methods of automatic deactivation of grid blocks.
- Export row heights and column widths using the default grid export.
- Hide grid lines - Allows you to show/hide grid lines.

## Import a Grid from a Text File

To allow you to view and manipulate projects which were not created in Argus ONE you can now import grids from external text files. Combined with the capabilities of importing data from text files you can now import complete grid-based models including their data into Argus ONE for visualization and manipulation purposes. The format of such a text file must be that of the Argus ONE default grid export template (see chapter 10).

### To import a grid from a text file

1. Activate a grid layer.
2. From the File menu select Import Grid... and from the sub menu select Text File...
3. Locate and open the file containing the grid to be imported.
   The grid is transformed into an active Argus ONE grid to which you can apply all Argus ONE grid functions and commands.

To practice importing a grid, import the file GridExmp.exp from the on disk examples (Macintosh file name: Example Grid.exp).

### To import grid related data

If the grid you import carries additional information you can read it into data type layers. For a detailed discussion of reading information to data layers refer to “Supplements for version 2.5” page 9s.

1. Activate a data type layer.
2. From the file menu select Import Data... and from the sub menu select Text File... The Import Data dialog opens.
3. In the Import Data dialog check the “Grid data” check box and the “Read triangulation from file” check box. Also check the appropriate grid type (Block or Line centered).
4. Locate and open the same file...

## Controlling Grid Numbering

Up to version 3 grid numbering was fixed to the coordinate system. Block number 1,1 was always fixed to the coordinate origin, and row and column numbers increased with the direction of positive Y and X respectively.

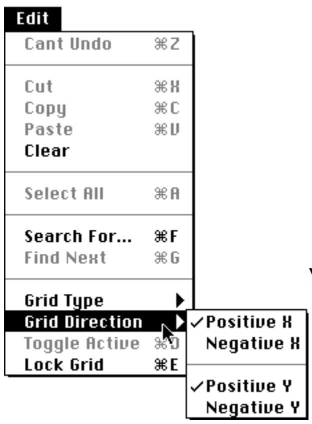
You can now set the rows and columns numbering to increase with either positive or negative X and Y coordinates. The grid numbering setting is layer specific. The default setting has remained unchanged.

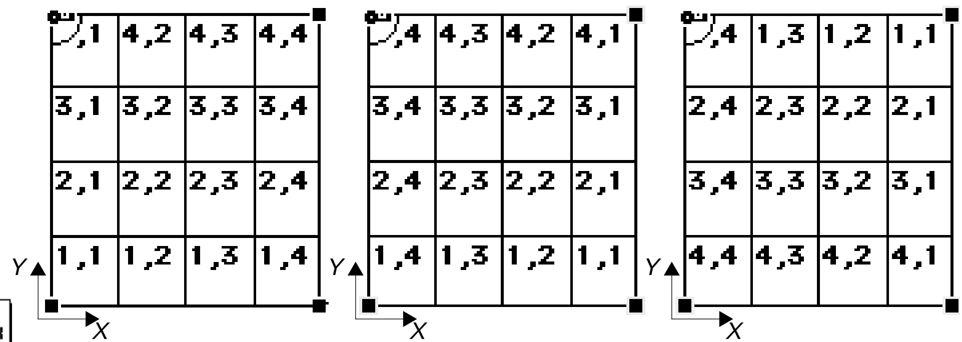
### To change row numbering

1. Activate the grid layer in which you wish to change the numbering direction.
2. To view your changes on-line, from the view menu select Show Block No.
3. From the Edit menu select Grid direction and the required Y direction.

### To change column numbering

1. Activate the grid layer in which you wish to change the numbering direction.
2. From the Edit menu select Grid direction and the required X direction.




*By default row and column numbers increase as X and Y increase.*

*Select Negative X to decrease column numbers as X increases.*

*Select Negative Y to decrease row numbers as Y increases.*

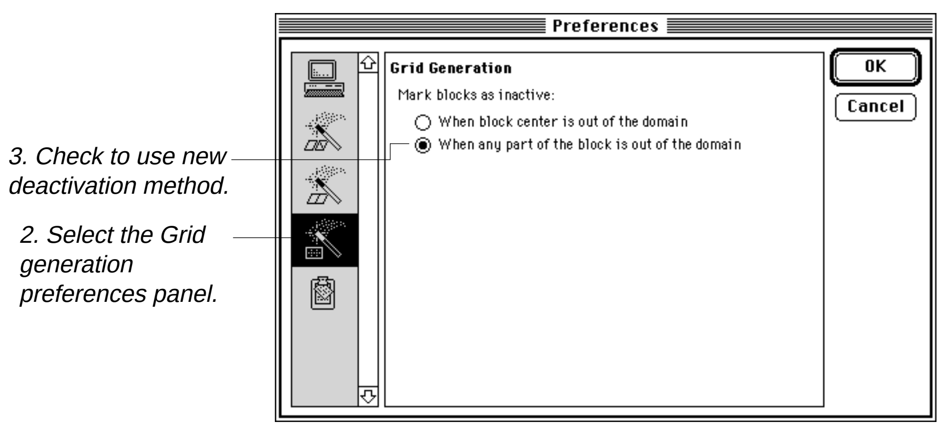
## New Method for Automatic Block Deactivation

The default automatic block deactivation method tests the location of the center of each block with the domain outline contours. If the center of the grid block lies outside the domain outline contour or inside a domain outline "island" it marks that block as inactive.

The new method tests whether a block intersects the domain outline. If is does, the block is considered inactive, and is deactivated.

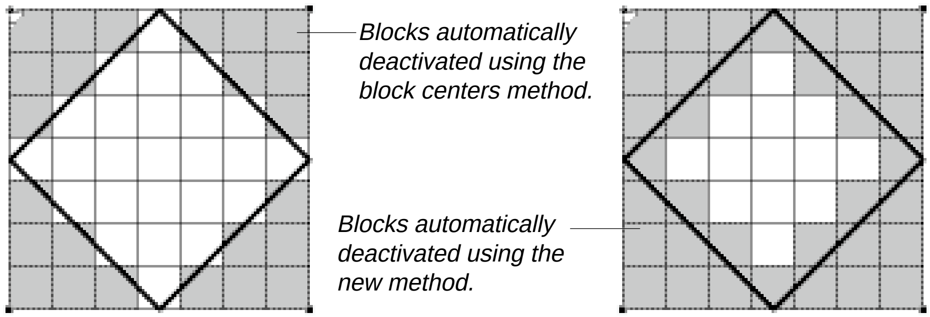
### To automatically deactivate blocks using the new method

1. From the Special menu select Preferences...
2. Select the Grid Generation Preferences panel.



3. Check the “When any part...” check box.

All grids automatically created or re-gridded will be deactivated according to the new method.

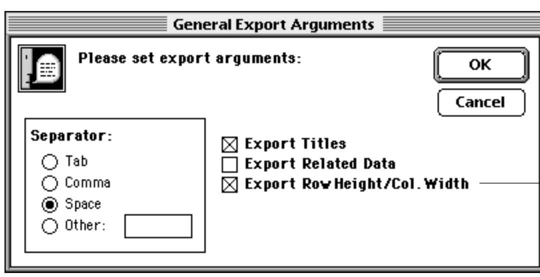



## Export Row Heights and Column Widths

Although you can export row heights and column widths by altering the default export template, you can now do so using the default grid export as well. The format of the exported file is the same as before except for the lines describing the row heights and column widths. For a full description of this file format refer to chapter 10.

### To export row heights and column widths

1. Activate a grid layer.
2. From the file menu select Export and from the sub menu select Export Grid... The General Export Arguments dialog opens.




3. Check the “Export Row Height/Col. Width” check box.
4. Click the OK button.
5. Specify the file name and save.

### Changes in file format

In case you use this option the file format differs from the standard one only in the lines describing the rows and columns positions.

### The first line of a grid export file

| | | |
|---|---|---|
| Number of Row | Number of Columns | Number of grid & block parameters +1 |

The number of grid and block parameters is incremented by one to account for the active/inactive block matrix.

### A line describing the absolute position of the first row

There is one line describing the first row absolute Y coordinate.

Row 1 Absolute Coordinate

After this line there is an empty line.

### A line describing row heights

There are “Number of Rows” lines describing the row heights. They are ordered from row 1to “Number of Rows”.

Row Height

After “Number of Rows” lines there is an empty line.

### A line describing the absolute position of the first column

There is one line describing the first column absolute X coordinate.

Column 1 Absolute Coordinate

After this line there is an empty line.

### A line describing column widths

There are “Number of Columns” lines describing the column widths. They are ordered from column 1 to “Number of Columns”.

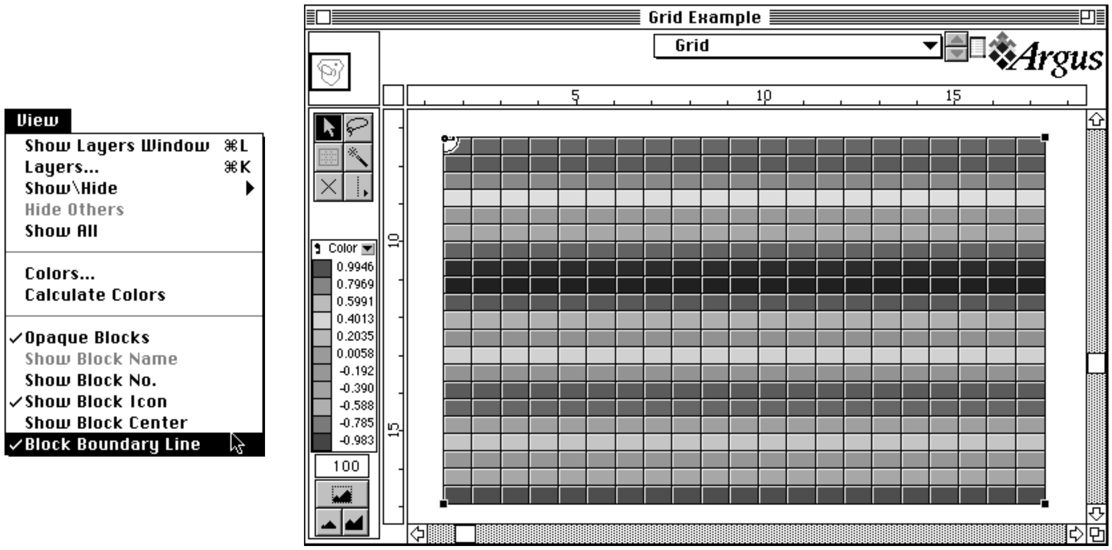
Column Width

After “Number of Columns” lines there is an empty line.

All other lines are identical to the original file format described in detail in chapter 10.

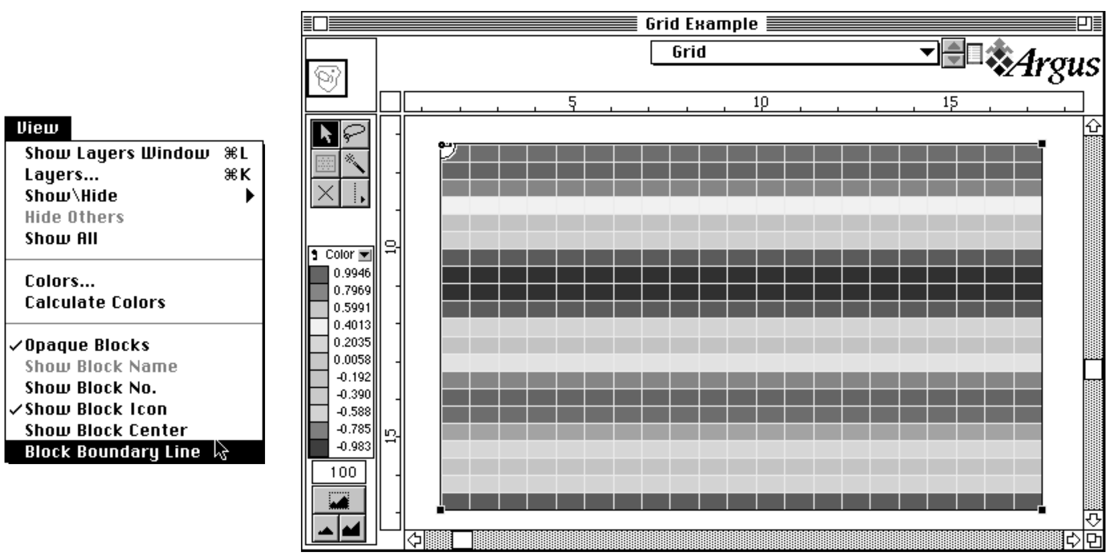
## Showing and Hiding Grid Lines

You can now turn off the drawing of grid lines. This might prove handy if you evaluate the grid by color using the color palette and popup menu.




### To show/hide grid lines

- From the View menu select Block Boundary Line.



# New Data Layer Capabilities

A number of new data layer capabilities have been added. These include:

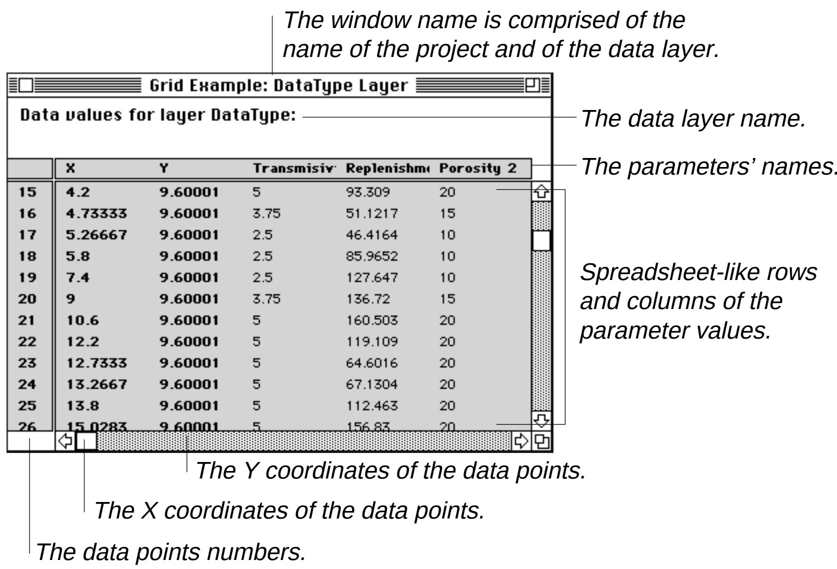
- Spreadsheet-like window for viewing the values and locations of data points.
- You can add, remove and rename data layer parameters.
- You can select the interpolation method which will be applied to the data when interpolated onto mesh or grid layers.

## Viewing Data Layer Values

You can now view the values assigned to data type layers' data points. The spreadsheet-like window allows you to scroll through the data points and view their X,Y positions and associated values.

### To view data points values

1. Activate the data type layer for which you wish to view the values at data points.
2. From the View menu select Show Values.
   The data is read from the data layer and presented in a new window type which opens above your main project window.


The larger the number of points and parameters, the longer it takes to open the window.

### To update the Show Values window

If you replace the data in a data layer by reading or importing a new data set, or if you remove, add or rename data layer parameters, the data presented in the Show Values window is not updated. If the Show Values window is open you need to close and open it again to update the information presented.

## Renaming, Adding and Removing Data Layer Parameters

You can now also rename, add and remove data layer parameters.

### To rename data layer parameters

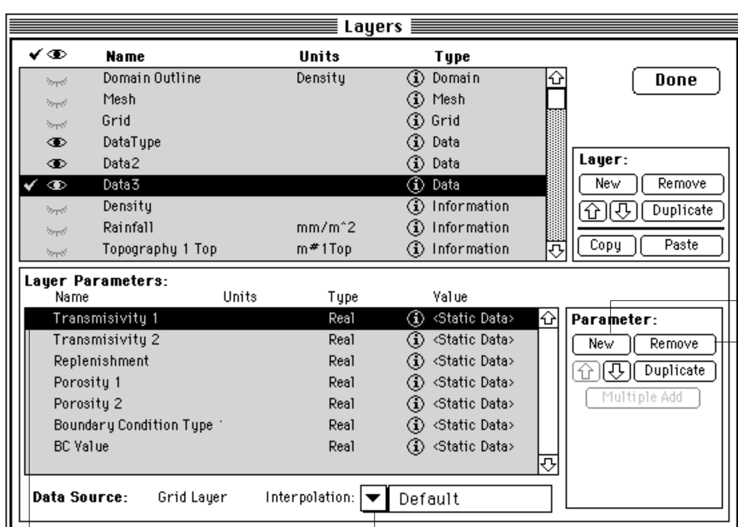
After you import data from a file all parameters are named “Imported Parameter 1” to “Import Parameter n”, where n is the number of parameters in the file. Naming the parameters allows you to more intuitively refer to them when you link them to other layers and when you view them in the Show Values window.

- To rename data layer parameters just type their new name in the Name field.

### To delete data layer parameters

When you import data from a file, all the parameters listed in the file are read to the data layer. If you do not wish to use some of them you can now delete them, thus also reducing the size of the Argus ONE project file.

1. In the layers dialog, select the line of the data layer from the layers list.
2. In the parameters list, select the line of the data parameter to be removed.
3. Click the Remove button in the parameter control panel.




*Click to add a new parameter.*

*Click to remove the selected parameter.*

*Type to change the parameter name.*

*Select the required interpolation method from the popup menu.*

### To add a data layer parameter

You can create empty data layer parameters. This allows you to relate to such parameters in other parameters, although the actual data is still unread. If when you read the actual data, the parameter names read are identical to the names of the parameters you manually added, the links to such parameters remain valid.

1. In the layers dialog, select the line of the data layer from the layers list.
2. In the parameters list, select the line after which you wish to add a parameter.
3. Click the New button in the parameter control panel.

## Selecting an Interpolation Method

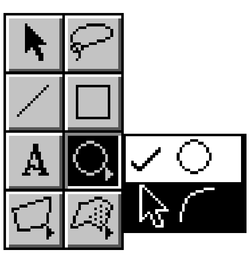
When you link information in data layers to grid or mesh layers, an interpolation procedure is invoked. With the introduction of Argus ONE PIE technology you may have additional (third-party) interpolation engines installed. If such engines are available to you they will appear in the Interpolation method popup menu.

# New Maps Layer Capabilities

A number of new maps layer capabilities have been added. These include:

- Text annotation tool.
- Text rotation.
- New color palette supporting the standard DXF colors.
- Coloring of graphic objects.
- Enhanced cross-section diagram.

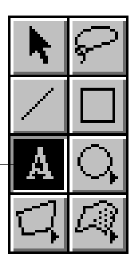
## Text Annotation



The text annotation tool allows you to annotate your maps layers. Combined with the text rotation and object coloring capabilities, your ability to prepare report quality documents with Argus ONE is greatly improved. It also allows you to edit text objects imported from DXF files.

To allow for the new tool, the Circle and Arc tools were combined into a group and appear under the circle popup menus.

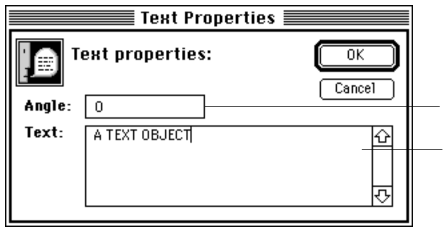
### To create text objects



*The Text tool.*

1. Activate a Maps type layer.
2. From the tool palette select the Text tool.
3. Click-drag the mouse where you want the text to start.
4. Drag the stretch rectangle to the size of the required font.
5. Release the mouse button to open the Text Properties dialog.

6. In the Text text edit box type the string.
7. To rotate the text, type the required angle in the Angle text edit box.
8. Click the OK button.

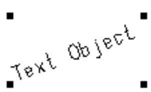




The text appears in the maps layer where you first clicked the text tool.


### To edit text objects

You can now edit all text objects, those imported from DXF files and those created using the text tool.

1. Double-click a text object to open the Text Properties dialog.
   Or -
   Select the object and from the Navigation menu, select Object info...
2. Edit the text and set the rotation angle.
3. Click the OK button.




### Other text editing capabilities

Text objects can be moved, deleted, scaled, and colored as all other graphic objects in maps type layers.

## DXF Color Palette for Maps Layers

The color palette in Maps layers now support the standard set of DXF colors. Using the white color is supported. However, a white object appears in black.

## Coloring Graphic Objects

All graphic objects can now be colored. The colors available are the ten standard DXF colors presented in the color palette.

### To color graphic objects

1. Select the graphic objects to be colored.
2. Click the required color in the tool palette color legend.
   All object are colored.

### To discolor graphic objects

1. Select the graphic objects to be discolored.
2. Click the black color in the tool palette color legend.

## Enhancements to the Cross-Section Diagram

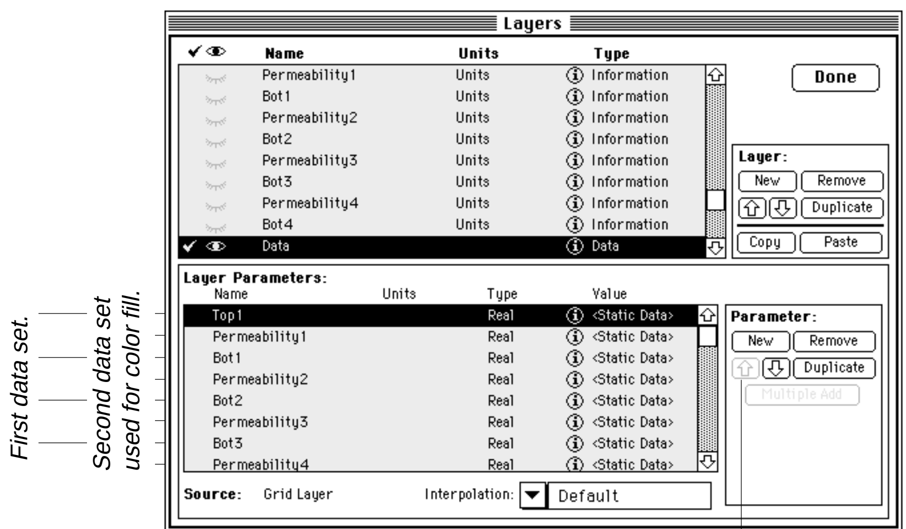
The cross-section diagram now also enables you to visualize two groups of parameters in a special way. Where one parameter type is distributed along the cross-section vertical dimension and presented as value lines, the distribution of the other type is used to fill the gap between values of the first, with colors.

This might become very handy if you need to plot the depth variation together with the distribution of an additional parameter.

### To create a Cross-Section Diagram for paired parameters

The two groups should be ordered in an alternating fashion.

**1.** Make sure the parameters are ordered in an alternating fashion in the data layer. If they are not, use the Layers dialog parameter list to order them so.



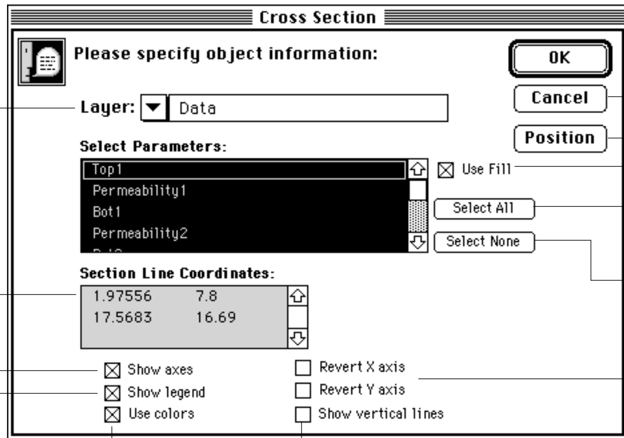

*To order the parameters in the required fashion use the promote/demote buttons.*

**2.** From the Layers menu select the maps layer in which you wish to create the post-processing object.

**3.** Select a graphic object to serve as the cross-section polygon.

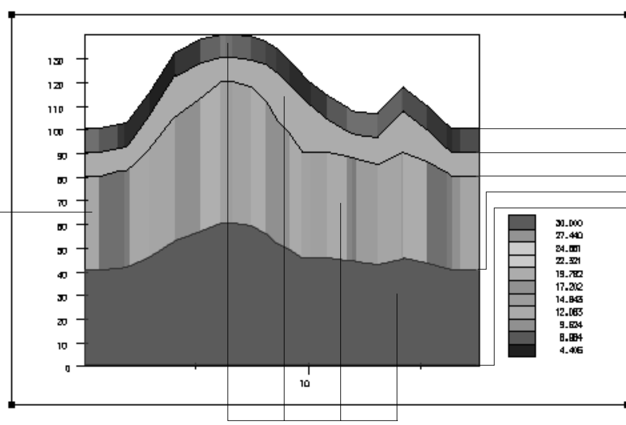
**4.** From the tool palette select the cross-section diagram tool.
If a drawing object is not selected then you can not choose the cross-section tool from the palette.

**5.** Click the cursor where you want the object to start and stretch a rectangle to where you want it to end.
The Cross-Section Diagram dialog opens to let you specify additional parameters.




6. From the Layer popup menu select the data layer containing the parameters you wish to plot. All data layers into which you have read or imported information are listed in the menu.
7. Select all the pairs of parameters you want to plot.
8. Click the OK button.

*A Cross-Section Diagram presenting the distribution of two parameters along the cross-section. If the parameter used for color fill is not constant with the cross-section, its distribution is colored with the projection of the grid lines or mesh nodes.*



# New Export Template Dialog and New Script Commands

Following is a list of enhanced or new script commands and format specifiers:

New format specifiers:

- `G format` - A general format specifier to support the 4 new parameter types.
- `Repeat factor` - Repeats the field it is placed before.

New and enhanced script commands:

- `Loop for Parameters` - In grid layer now exports only layer parameters.
- `Loop for Block Parameters` - A new loop which exports both layer and block parameters.
- `Else` - Enables you to simplify and eliminate unnecessary `If` blocks.
- `Define Variable` - Allows you to define a variable.
- `Set Variable` - Enables you to set the variable value.
- `Loop for Variable` - Enables you to write loops over variables.
- `Alert` - Enables you to pop up an alert during export.
- `Evaluate expression` - Enables you to compute a given expression.
- `Execute external code` - Enables you to call an external code when the export is performed.

## The G Format

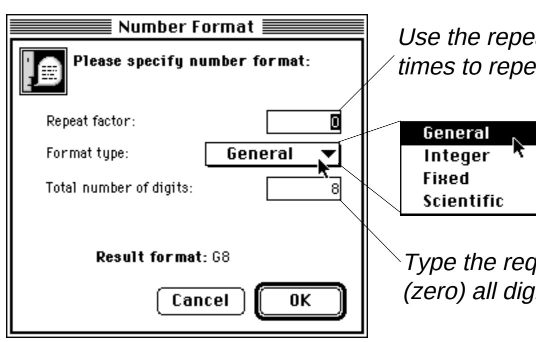
***Important Note****:*
*If you specify G format and zero (0) for the total number of digits, all digits or characters will be exported.*

With the introduction of the four new parameter types (i.e. real, integer, string and boolean) the available format specifiers (fixed, integer and scientific) could not support the export of strings and booleans. Also, when generic and default export templates are performed it is required to have an automatically switching format specifier which supports all parameter types, as the type of a parameter is unknown at the time the template is defined.

### To use the G format

**1.** In the number format dialog pull down the format menu to select General.

***Important Note****:*
*All default export templates are now in G format.*




*Use the repeat factor to specify how many times to repeat the field it appears after.*

*Select the G format from the format popup menu.*

*Type the required number of digits. If you specify 0 (zero) all digits or characters will be exported.*

**2.** In the total number of digits box type the required number.

### G format and Integers

Integers are exported under G format such that if they are shorter than the number of digits specified they are right aligned and are prefixed by spaces. If the integer number is longer than the specified number of digits the number is replaced by number of digits asterix characters to allow you to quickly locate the problem in your export files.

### G format and Reals

Reals are exported under G format such that if they are shorter than the number of digits specified, they are right aligned and are prefixed by spaces. If a real number is longer than the specified number of digits, the number is first rounded to try and fit it within the specified number of digits. If it can not be rounded in a meaningful way it is replaced by (*) characters to allow you to quickly locate the problem in your export files.

### G format and Strings

Strings are exported under G format such that if they are shorter than the number of digits specified they are left aligned and are suffixed by spaces. If the string is longer than the specified number of digits it is truncated to the specified number of digits.

### G format and Booleans

Booleans are exported under G format as the strings True or False. If you need to export them as zeros (0 for False) or one (1 for True), either assign them an integer type when you define the parameter, or assign them the Integer (I) format in the export template. (note that True is any number other than zero and not necessarily one (1))

## The Repeat factor

***Important Note:***
*Do not use the repeat factor in conjunction with a delimiter specifier.*

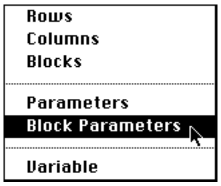
If you need to repeat a field n number of times set the repeat factor to n. It appears in the export template before the format specifier as in [2G8]. However, note that you should not use the repeat factor if you use the delimiter field after the format specifier.

### To insert a repeat factor

- In the number format dialog enter the repeat factor. The repeat factor appears before the format definition of the expression.

## Enhanced Loops for Grid Layer Parameters

When used within a grid export template the loop for parameters now exports only the grid layer parameters. Previously this loop exported both layer and block parameters.



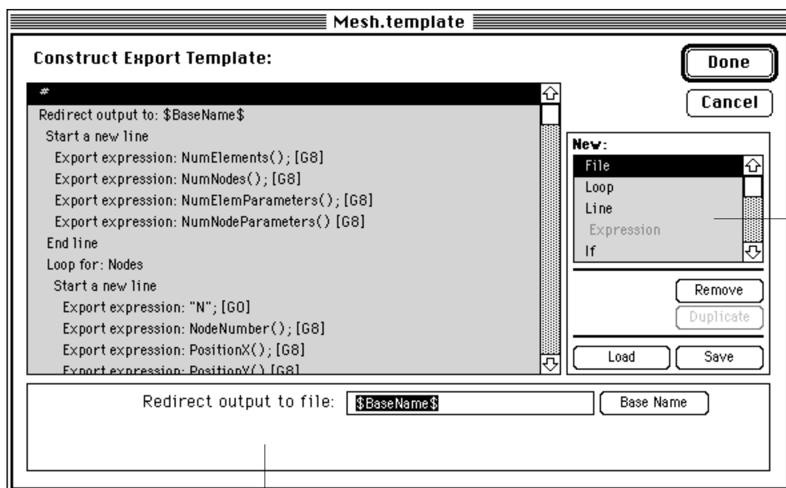

## New Loop for Block Parameters

The new loop for block parameters now exports both block and grid layer parameters. It replaces the previous loop for parameters when used in a grid export template. Both the loop for parameters and the loop for block parameters offer you more control of the parameters exported.

**Important Note**: Be sure to replace all loops for parameters to loop for block parameters if you designed and used custom grid export templates.

## Enhanced Template Editor Dialog

To allow access to the growing number of script commands and to the additional script commands variables, the Template Editor dialog was modified.




*The script commands list.*

*The argument editing control panel changes according to the script command being edited.*

## The Else Script Command

The Else script command enables you to simplify and shorten complex and nested If blocks. It also speeds the execution of the actual export operation.

All lines below an Else line and up to the matching Endif are nested within that Else command. You can implement an Elseif command by inserting an If statement following an Else statement.

### To insert an Else script command

- Locate the Else command in the commands list and double click it.

## Defining, Setting and Using Variables

Variables dramatically improve the export template functionality. Together with the `loop for variable` and `else` script commands, they take the Argus ONE export scripts a big step towards a full programming environment. Using variables enables you to temporarily store values for later use, to use a variable inside loops and as loop iterators, to test their value using an If statement, etc. The resulting export templates perform much faster and are easier to understand and debug.

### To define a variable

Any number of variables can be defined in an export template. Variable definitions must be placed at the top of the template, below the first comment line and above the first Redirect script command.

1. Scroll to locate the Define Variable command in the commands list.
2. Double-click the Define Variable line.
   The `Define Variable:` command is inserted in the template editor and the expression editor allows you to name the variable and set its type to one of the four available types.

### To set a variable

You might want to set a variable's value to: initialize it, assign it another variable or expression, set the upper limit of a loop, etc.

1. Scroll to locate the Set Variable command in the commands list.
2. Double-click the Set Variable line.
   The `Set Variable:` command is inserted in the template editor and the expression editor allows you to choose the variable and set its value.

## Using Loop for Variables

Loop for variables allow you to define any general loop. Using loops will shorten your export templates and will expedite the actual export procedure.

### To insert a loop for variable

1. From the commands list select and double-click the loop command to insert it into the template editor.
2. In the expression editor, from the Loop popup menu select Variable.
3. Enter the loop variable (make sure it is already defined), the loop starting value and the step. All three may be Argus ONE functions and or expressions.
4. Between the Loop and End loop command insert the commands to be executed within the loop.

## Advanced Script Commands

Three new advanced commands are available. They can be used to call external functions, to execute external programs or applications and to alert the user during the execution of a template.

### To insert an Alert

Alerts can be used for debugging purposes or to inform the user (yourself or your users) of the progress of the export or of its successful termination.

1. From the commands list select and double-click the Alert command to insert it into the template editor.
2. In the expression editor, enter the string to be presented to the user when the alert appears.

### To insert an Evaluate command

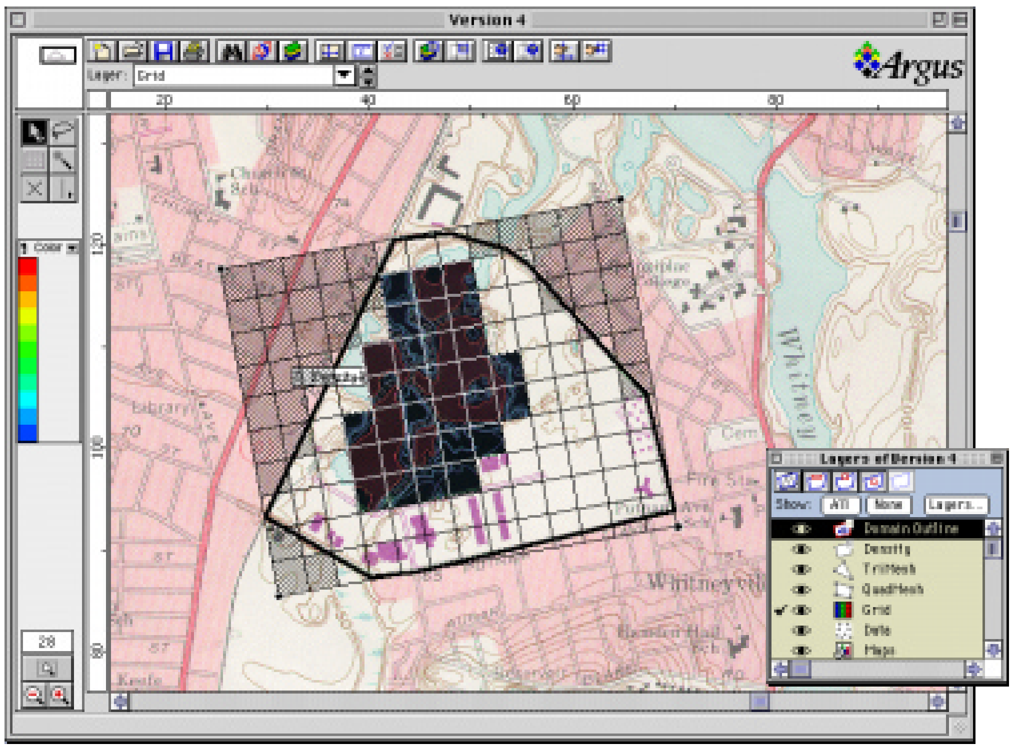
The Evaluate command enables you to evaluate an expression when the template executes without having to export the result. This may be used to call an external Argus ONE PIE function which has a side effect that you need to take place.

1. From the commands list select and double-click the Evaluate command to insert it into the template editor.
2. In the expression editor, enter the expression to be evaluated.

### To insert an Execute command

The Execute command enables you to run an external code or program. This can be useful if you need to call your code to execute using the exported data after the export is terminated.

1. From the commands list select and double-click the Execute command to insert it into the template editor.
2. In the expression editor, enter the name and path to the code to be executed.




# Argus ONE

## New Features Supplement

For version 4

This product was developed using Neuron Data Open Interface Elements.

Sales and Services offices at:

Argus Interware, Inc.

11 Merit Lane

Jericho, NY 11753

USA

# Argus Open Numerical Environments Version 4

The many new features and capabilities introduced in Argus Open Numerical Environments (Argus ONE) version 4 are described in detail in this supplement. We suggest that you go through the table of contents and locate items that may relate to your work.

## Contents

# Compatibility Issues

Full backwards compatibility exists between version 4and version 3. This means that you can open Argus ONE files created with older versions in Argus ONE version 4. However, you can not open files which will be saved under version 4 with earlier versions.

We suggest that until you are certain that all functions operate as you were accustomed to in earlier versions you save backup copies of your files created with the version you have been using prior to upgrading.

Also, if you have enough disk space, do not erase your previous copy until you are certain that all is fine with results you get from version 4.

# Revised Modules Structure

Argus ONE module structure has been revised in version 4. The two new modules which have been introduced are the GIS and the Programmable Export modules. The GIS module includes all non mesh and grid layers, such as Information, Data, Maps, and Domain type layers and all related functions and capabilities. The GIS module is required for the operation of all other modules.

Export template creation, editing and export by template have been grouped under the Programmable Export module to support future enhancements of programmable export capabilities for all GIS layers. Programmable export template capabilities for all mesh and grid modules are still distributed with those modules as before.

Currently the following modules are available:

- GIS
- Grid - Finite difference Grids
- Tri Mesh - Triangular Finite Element Meshes
- Quad Mesh - Quadrilateral Finite Element Meshes
- Programmable Export

# Improved Installer for MS Windows

A new installer for MS Windows 95 and NT has been adopted. The installer by InstallShield allows you to select the components you wish to install, to easily select the directory in which you want to install Argus ONE and also supports automatic uninstalling of the application.

The new installer does not erase files or folders you added yourself under the Argus ONE folder but only replaces identical files and folders. Thus, if you have installed PIE files or Export templates, or added folders under the Argus ONE folder they will remain intact when you upgrade your most current copy of version 4 to later revisions or versions.

## Support for MS Windows 3.x Discontinued

As of version 4, Argus ONE for PC will be supported only on MS Windows 95 and NT. Since all of the Microsoft development environments used by Argus are not supporting non 32 bit clean code development we can no longer support 3.x versions of MS Windows. Files created with Argus ONE on MS Windows 3.x versions are still supported and can be opened with Argus ONE version 4 running on MS Windows 95 and NT.

## Registration of Argus ONE Version 4

If you are upgrading from an earlier version to version 4 you need to renew your registration with Argus. If you purchased the upgrade, you will be sent new registration and activation codes automatically. You will then have to type them in the Registration Information Dialog which can be accessed from the About menu under the Help menu.

# User Interface Enhancements

Many components of the user interface have been enhanced and new ones have been added. The most notable ones are the toolbar, the tool tips, the information ruler which also shows help about tools, and the help system.

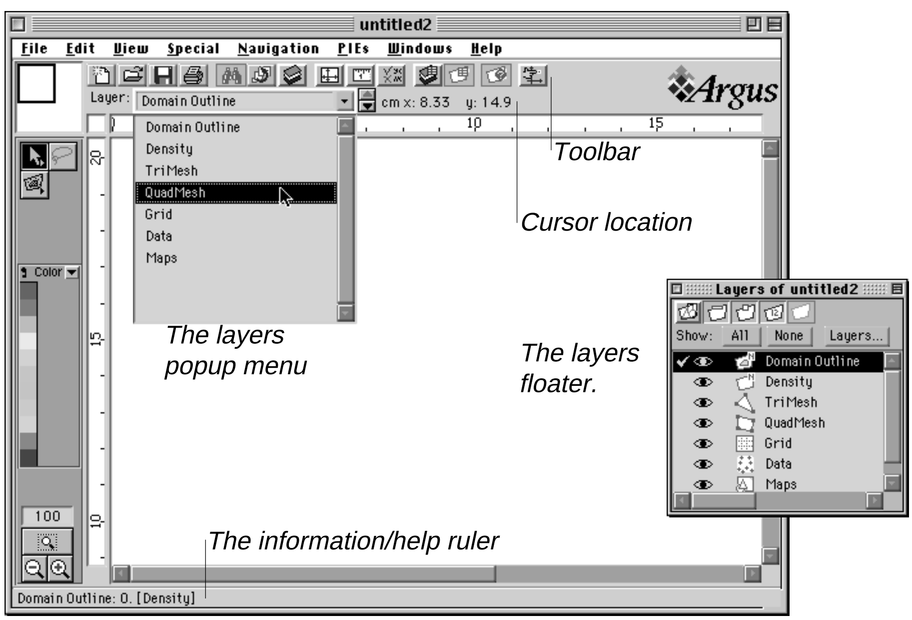
## Main Window's New Components and Changed Locations

The following new components have been added to the main window:

- **PIE menu -** Located to the right of the Windows menu
- **Help menu -** Located as the right most menu on the menu bar.
- **Toolbar -** Located below the menu bar. (On the Mac below the drag bar)
- **Information/Help ruler -** Located at the bottom of all windows and below the horizontal scroll bar in the main window.
- The layers floater automatically opens when a new project is created. It also features a View toolbar and layer-selection by typing the layer's name.

The following components have been relocated and changed:

- The cursor position info line is located below the toolbar.
- The layers popup menu is also located below the toolbar and shows only visible layers.
- The information ruler is located at the bottom of the window below the horizontal scroll bar.
- Some menu items have been moved to different menus.



## The Toolbar

*In some layer types you might need to resize the window to see all tools.*

The toolbar, located at the top of the window, contains shortcuts to the more often used menu commands. The number of toolbar icons is layer-dependent, and some icons might be dimmed (inactive) if they are not applicable.

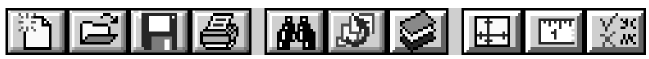
### Project related tools

1. File menu commands - New, Open, Save and Print
2. General commands - Search, Rotate and Scale, and Layers dialogs
3. Project settings - Drawing size, Scale and Units, and Preferences dialogs

### Layer related tools

1. Floating windows - Layers and Charting floaters
2. Info dialogs
3. Layer specific commands
4. Goto commands

### Project toolbar



*File General Project settings*

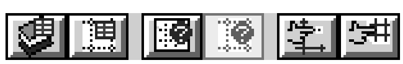
### Information layer toolbar

*Floaters Info Layer Goto*

### Mesh layers toolbar

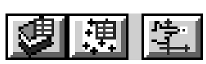
*Floaters Info Goto*

### Grid layer toolbar



*Floaters Info Goto*

### Data layer toolbar



*Floaters Goto*

### Map layer toolbar

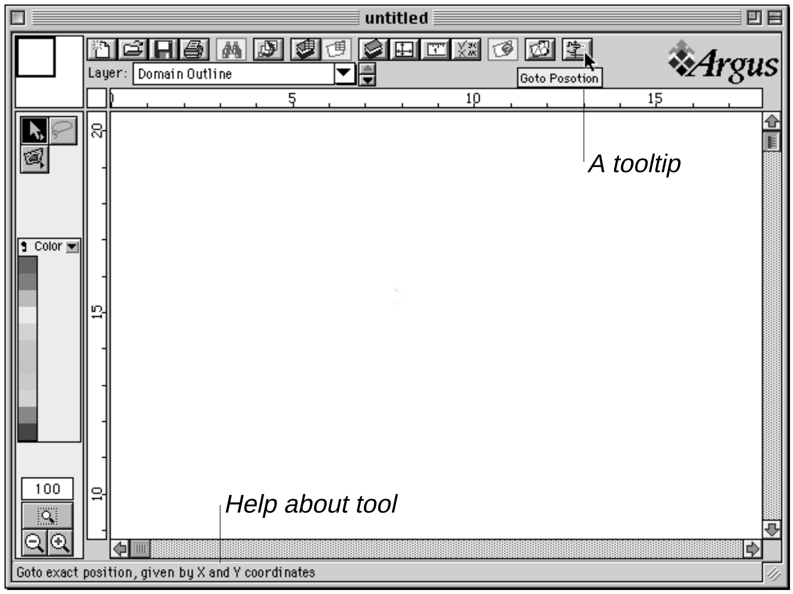
*Floaters Info Goto*

### Tooltips and tools help

*Toolltips and help in the information ruler are also available for tools in the tool palette and buttons in some dialogs.*

When you place the cursor above a tool's icon, a tooltip appears describing the action of that tool. The information ruler at the bottom of the screen shows a more detailed explanation of the tool.

*Macintosh: If Balloon Help is on a balloon will appear showing the text which appears in the information ruler.*

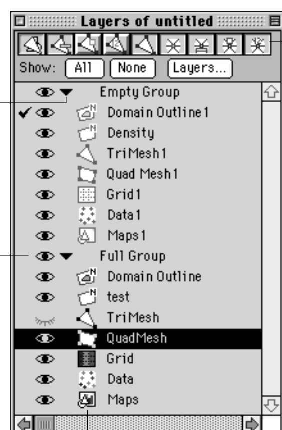




## Layers' Floater New Functionality

Two new columns of icons have been added to the layers' floater. The Collapse/Expand Group icons column and the Layer Indicator icons column. A View toolbar, at the floater's top enables quick access to view commands.

*Click to collapse all layers in the group.*

*Click the eye icon of a group layer to show/ hide all of its layers.*




*View toolbar - Shortcuts to View menu commands.*

*Each layer type has a different set of viewing commands.*

*You can change the viewing options of a layer even if it is not the active layer.*

*Layer indicator - Shows: layer type, empty/full layer and interpretation type.*

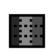
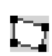
## Layer Indicators

The layer indicators are a set of dynamically changing icons indicating the layer type and some other characteristics of the layer.

A quick glance at the layers list will reveal to you the:

- Layer type
- If the layer contains objects
- The interpretation method (Information type layers)

The following table lists the icons and their meaning.

| Layer Type | Icon | Indicates[a] |
|---|---|---|
| Information |  | Interpretation method: Nearest |
| Information |  | Interpretation method: Exact |
| Information |  | Interpretation method: Interpolation |
| Information |  | Allow intersection is on[b] |
| Domain |  | Interpretation method: Nearest |
| Maps |  | |
| Data |  | |
| Grid |  | |
| Quadrilateral mesh |  | |
| Triangular mesh |  | |

a. When a layer's icon is dimmed the layer has no objects. When there are objects in the layer it's icon will be colored.

b. Allow intersection is independent of the layer interpretation method and thus may appear with any of the icon variations of information type layers.

## View Toolbar

The View toolbar at the top of the layers floater allows you to show and hide various attributes of objects in a layer. You can do so without first making the layer active.

### To change the viewing option of a layer

1. Click the line of the layer. You do not have to activate the layer.
   The toolbar will change to show the available view commands.
2. Click the appropriate tool.

## Collapsing and Expanding Groups of Layers

By entering Group type layers you can group your layers into subjects. Group layers allow you to:

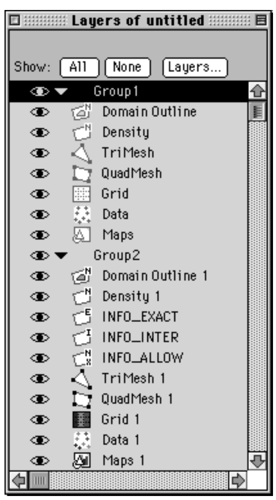
- Show and hide all layers belonging to a group with one mouse click.
- Collapse a group in the layers floater.

### To show and hide all layers belonging to a group

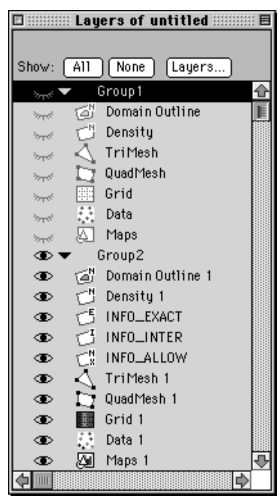
- Click the eye icon of the group layer they belong to.

### To Collapse a group layer

- Click the arrow to the left of the group layer's name.

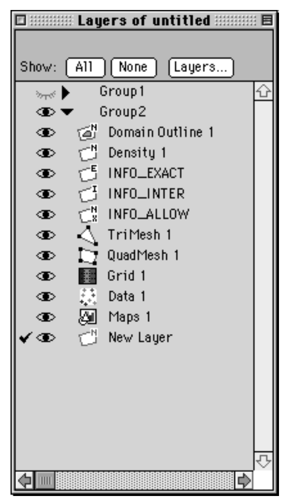



*To hide all layers belonging to a group, click the eye icon of that group...*




*...All layers of the group are hidden.*

*To collapse a group, click the arrow next to that group...*




*...All layers belonging to that group are collapsed.*

### Selecting a Layer by Typing its Name

You can type the first letters of a layer's name and it will be selected. This is especially convenient when you have many layers.

#### To select a layer by typing its name

1. Type the first (or as many as you can) letters of the layer's name.
2. The selection will move to the first layer whose first characters match the ones you typed.
3. To select the next layer, in alphabetical order, press the tab key.
4. To select the previous layer, in alphabetical order, press shift+tab keys.

If the list is longer than the floater, it will automatically scroll the selection into view.

## Greeting Window

The greeting window which opens whenever you launch Argus ONE disappears after a number of seconds. It can be accessed from within the Argus ONE workplace using the About Argus ONE... menu.

## About Window

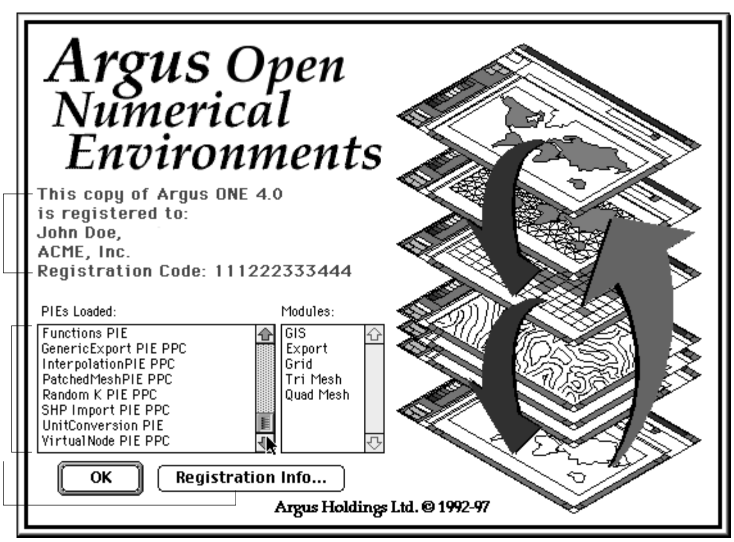
The About window allows you to change your registration information and also lists the following information:

- Version number
- Name and company of the registered user
- Registration code
- Lists of the available Argus ONE PIEs and modules


*Version and Registration information*

*Available PIEs and Argus ONE Modules*

*Click to change registration information*



## The PIEs Menu

The PIEs menu organizes menus which are created and installed by PIEs. Not all PIEs will have their menus installed under the PIEs menu. The PIEs menu also hosts the About Argus PIEs command which open a dialog listing the available PIEs and their description.

Their are 3 groups of menu items:

1. The New Project group - storing all commands to create a new PIE project.
2. The Run/Import/Export group - storing menus which invoke a model, export data or import data.
3. The Project Information group - storing the current project's PIE information menu.


*Select to open the PIE Information dialog*

*New PIE Projects commands*

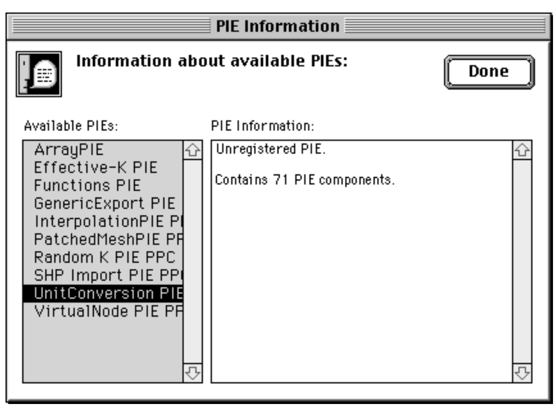
*Run/Import/Export commands*

*Edit Project Info command*

### The PIE Information Dialog

The PIE Information dialog provides you with valuable information regarding the PIEs that are installed in your Argus ONE workplace. Such information might include the PIE developer, PIE name and the PIE version. The information is provided by the PIE developer and Argus can't control it.

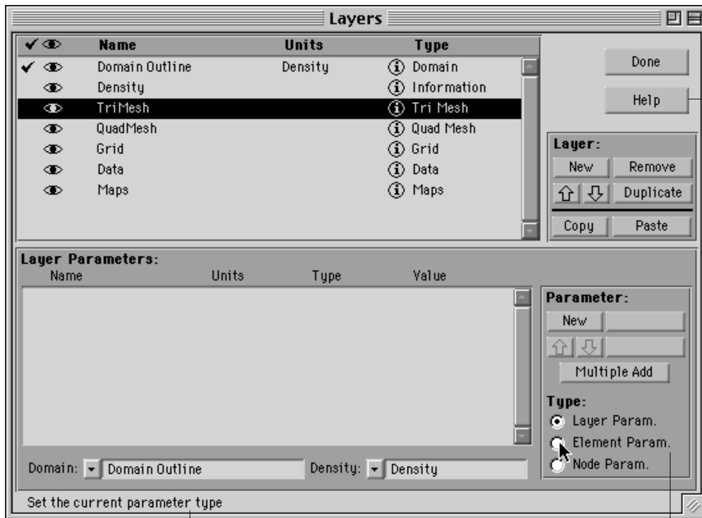
*List of available PIEs*




*To view a PIE's info select it from the PIEs list.*

## Dialog Enhancements

Most of Argus ONE dialogs are resizable. Some of the more complex dialogs feature tooltips for buttons, an information ruler and a help button.




Click to open the on-line help

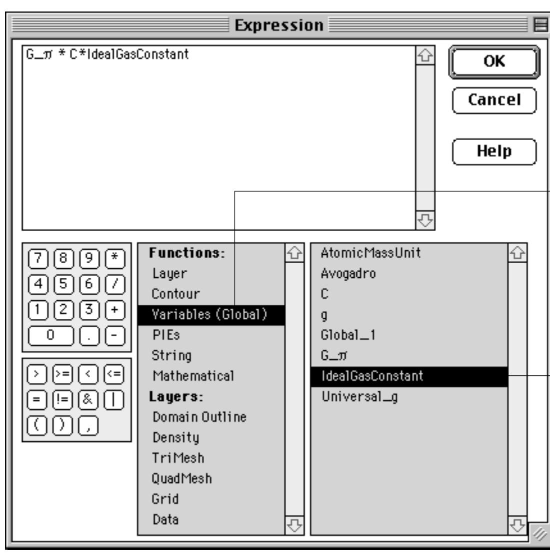
Click-drag to resize the dialog.

Help about the item the cursor lies above.

### Expression Dialog New Capabilities

The following two new features have been added to the expression dialog:

- You can select a function or a parameter by typing their first letters.
- Global and export template variables are accessible from the function list.




Selecting the Variables group presents all the available global variables in the function list.

To select an item from one of the lists by its name:

1. Click in the list from which you want to select an item.
2. Type the first letters of the item on your keyboard.

# New Data Layers Capabilities

Three new features have been added to Data type layers:

- Evaluation of data layer parameters using colors.
- Reading data from grids at block centers.
- A progress bar indicating the time remaining for reading data from layers.

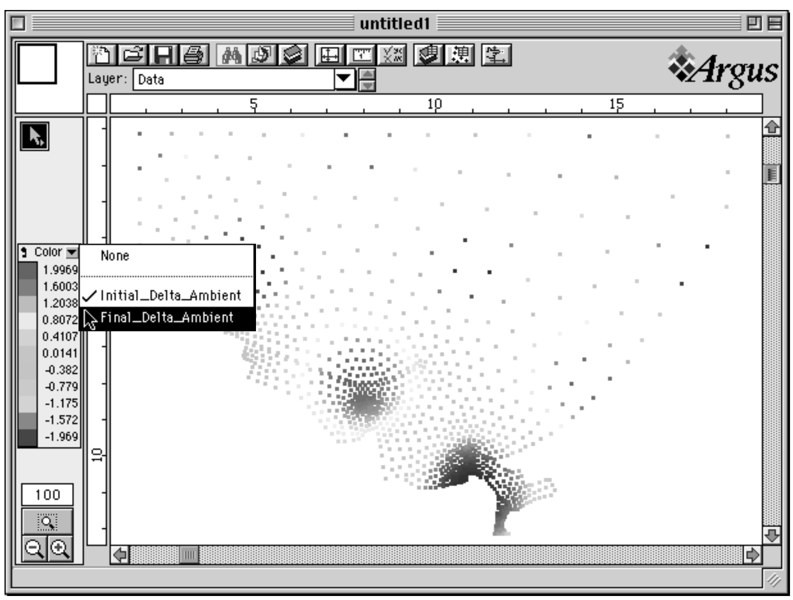
## Evaluation of Data Layer Parameters using Colors

As in Information, Grid, and Mesh type layers you can now visualize the values of data layer parameters using the Evaluate by Color method. This allows you to quickly view the distribution of data.

### To evaluate data by color

1. Make sure the active layer is a data type layer and that in contains data.
2. In the tool palette click and hold the Color popup menu to open the parameters list.
3. Select the parameter by which you wish to color the data points.

For a detailed discussion of how to control the color range see Chapter 1, ”The color tools and legend”.

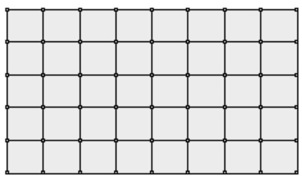


## Reading Data From Grids at Block Centers

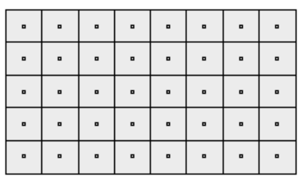
When using block centered grids the evaluation of block and grid parameters is performed at block centers. To conform with this evaluation locations, you can also read data from grids at block centers. This will also allow you to compare simulation results with inputted grids using the same locations for both. By default the data is read at block corners.

### To read data from grids at block centers

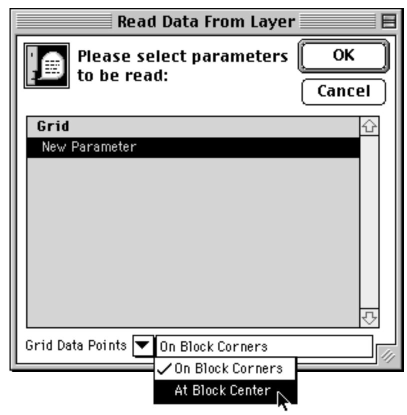
1. From the Edit menu select the Read Data from Layer... menu.
2. In the dialog select the layer and parameters their data you want to read.
3. Click-hold the Grid Data Points popup menu at the bottom of the dialog to open the menu.
4. Select “At Block Centers”.
5. Click the OK button.



*Data read at block corners.*

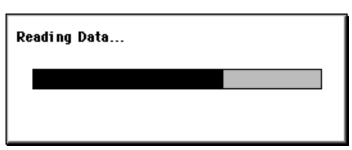


*Data read at block centers.*




## Read Data from Layer Progress Bar

The progress bar appears whenever the product of number of grid blocks and number of parameters being read exceeds one thousand (1,000).



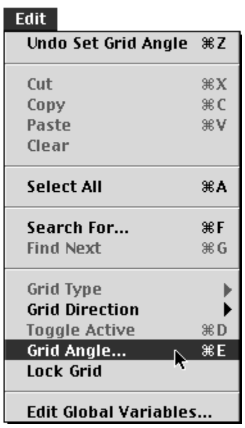
# Grid Rotation

Under certain conditions it is preferable to discretisize a domain using a rotated grid to minimize the number of grid blocks and thus to allow for a faster modeling cycle. Grids can be rotated using one of three commands.

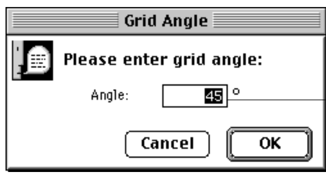
## Manual Grid Rotation

A grid can be manually rotated using the Rotate and Scale Dialog and by setting the grid angle.

### To rotate a grid using the Grid Angle command




1. From the Edit menu select Grid Angle..., The Grid Angle dialog opens.



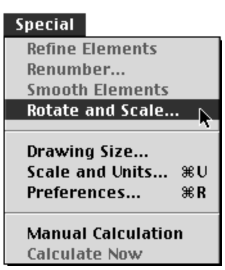

*Type the grid rotation angle in degrees*

2. Type in the grid rotation angle (positive and negative values are allowed).
3. Click the OK button.

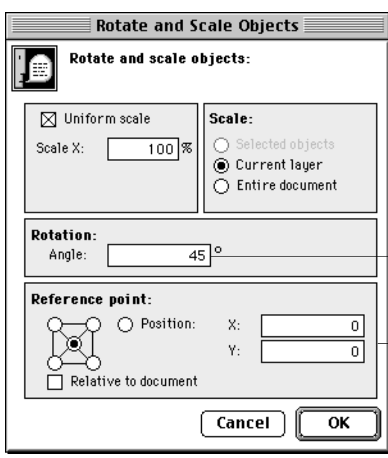
### To rotate a grid using the Rotate and Scale command

Rotating a grid using the Rotate and Scale command offers greater control by allowing you to set the point of reference and to combine rotation and scaling.

1. From the Special menu (or from the toolbar) select Rotate and Scale...
2. Enter the grid's rotation angle (both positive and negative values are allowed).
3. Select the reference point for the rotation.

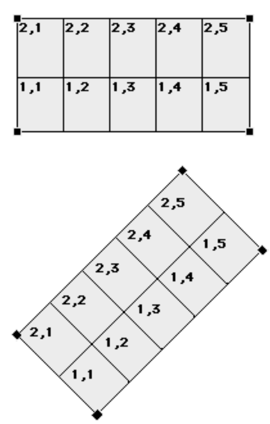



*If when rotated the grid extends beyond the document's borders the drawing size and origin are changed.*




*Type the grid rotation angle in degrees*

*Select a reference point*



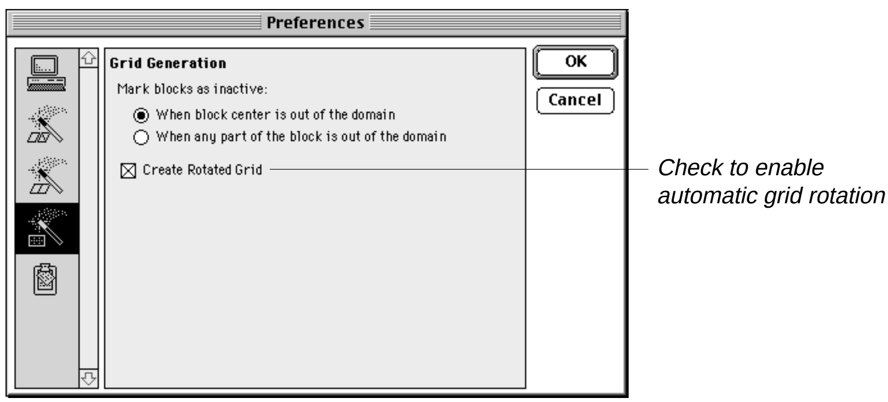
## Automatic Grid Rotation during Auto Grid Generation

When auto grid generation takes place Argus ONE can automatically calculate the grid angle that will minimize the number of grid blocks and thus also the number of inactive blocks.

In projects created by version 4 or later, automatic grid rotation is the default behavior.

### To enable or disable automatic grid rotation

1. From the Special menu select Preferences...
2. Click the Grid Magic Wand to select the auto grid generation preferences pane.
3. To enable automatic grid rotation check the Create Rotated Grid check box.
4. To disable automatic grid rotation uncheck the Create Rotated Grid check box.
5. Click the OK button.

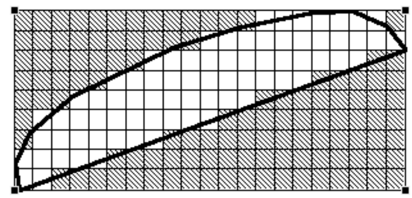


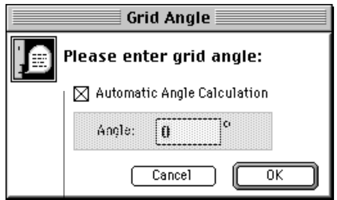

### Using automatic grid rotation

When this option is selected in the grid preferences the following dialog will appear when you click the magic wand above a domain contour to be grided.

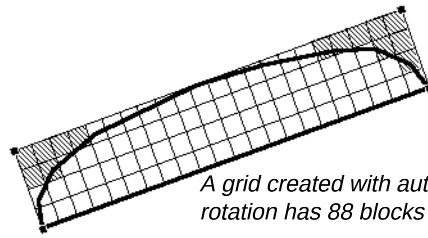
The dialog allows you to decide whether you wish to use auto-rotation, and if not what, at which angle to rotate the grid.

*A grid created without auto-rotation has 189 blocks*





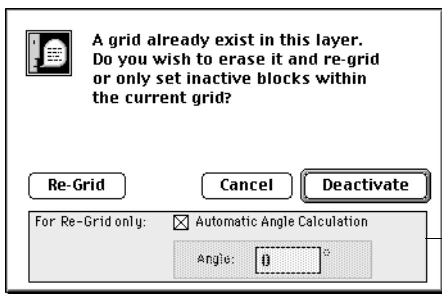

*Check to use auto rotation or uncheck and specify the angle of rotation.*



*A grid created with auto rotation has 88 blocks*

## Regridding with automatic grid rotation

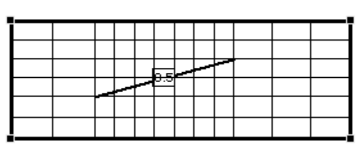
When a grid already exists and you choose to regrid the domain, the re-grid dialog allows you to select auto rotation, no rotation or manually specify the rotation angle.

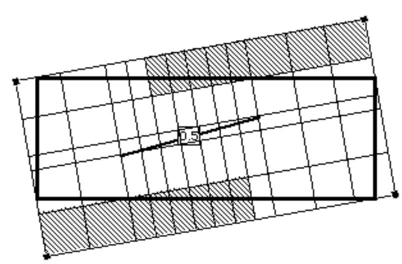



*Check to use auto rotation or uncheck and specify the angle of rotation.*

## The effect of density on auto grid rotation

Since, the auto rotation attempts to minimize the number of grid blocks, density defined in the density layer associated with the grid is also taken into consideration. Under certain geometries and grid densities it may happen that the minimum number of grid blocks will occur in a rotated grid.





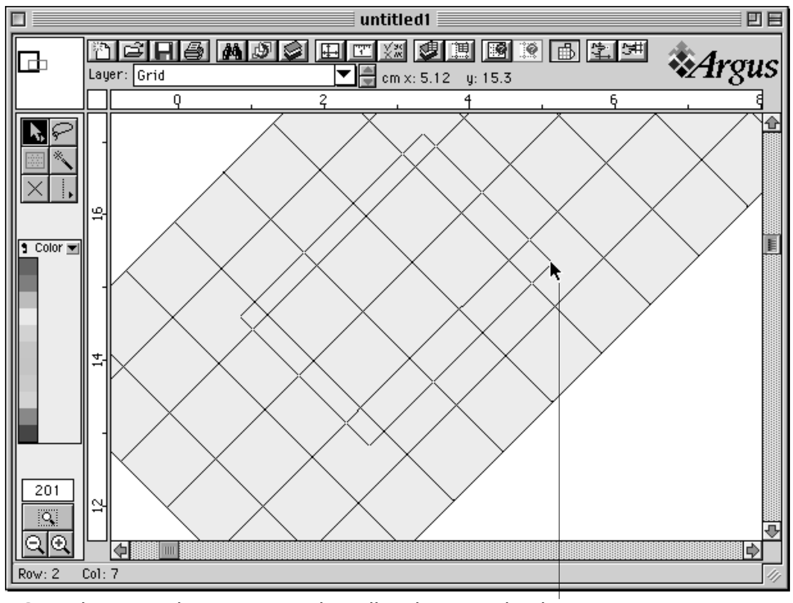
*The automatically rotated grid has fewer blocks than the unrotated one.*

## Operation of Tools in Rotated Grids

The operation of grid tools which involve stretching a rectangle reflect the rotation of the grid by creating the stretch rectangle along the grid axis.

The following grid tools create a rotated stretch rectangle

- Magnifying glass, zoom in and out
- Arrow, select with
- Arrow, move grid line
- Delete rows and columns
- Add rows and columns




*Stretch rectangles are rotated to allow better selection*

## Changes in Export and Import File Formats

The first line of a grid file also lists the grid rotation angle. If it is omitted Argus ONE assumes it is zero.

### The first line of a grid export/import file

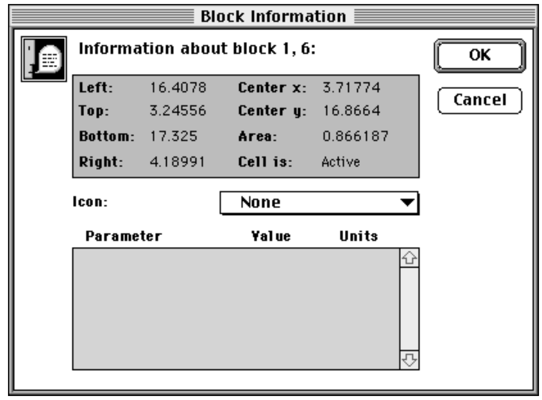
| | | | |
|---|---|---|---|
| Number of Row | Number of Columns | Number of grid & block parameters +1 | **Grid Rotation Angle** in radians (optional) |

## GridAngle() Function

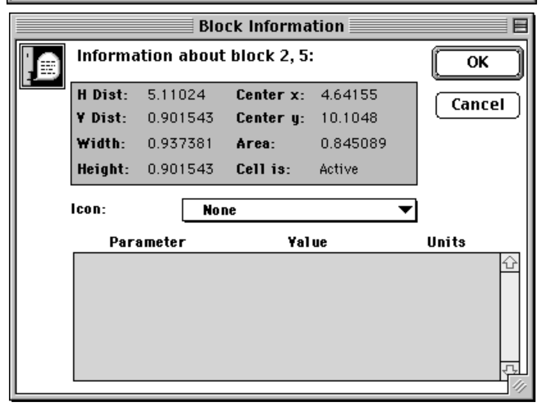
The `GridAngle()` function returns the grid angle of rotation in radians. See description under new functions section in this supplement.

## Block Information Dialog

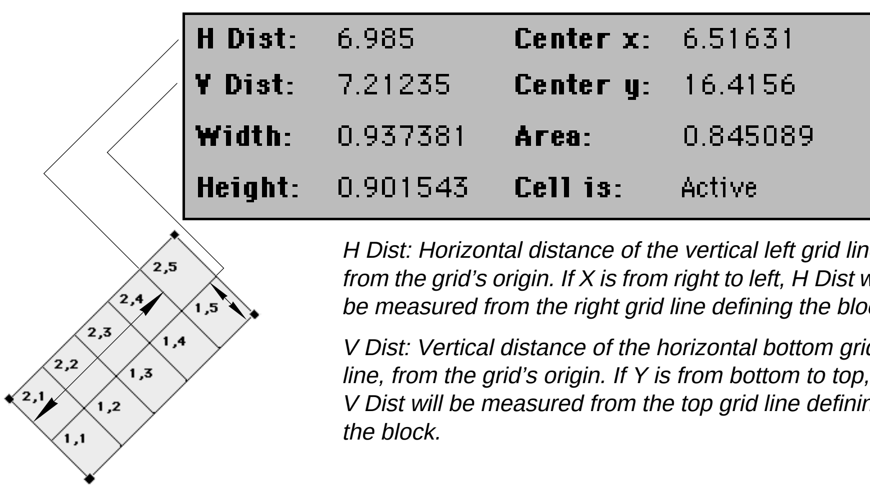
Some of the static information listed in the block information dialog has been changed to support rotated grids. Instead of listing the locations of the grid lines defining the block, the distances of the grid lines from the grid's origin, and the block height and width are specified.




*Block information dialog before version 4. Since the grid was always aligned with the coordinate system, all block information was related to the document's origin.*




*Block information dialog from version 4. Since the grid may be rotated with respect to the coordinate system, all block information is related to the grid's origin.*




*H Dist: Horizontal distance of the vertical left grid line from the grid's origin. If X is from right to left, H Dist will be measured from the right grid line defining the block.*

*V Dist: Vertical distance of the horizontal bottom grid line, from the grid's origin. If Y is from bottom to top, V Dist will be measured from the top grid line defining the block.*

# Miscellaneous New Features

## Global Variables

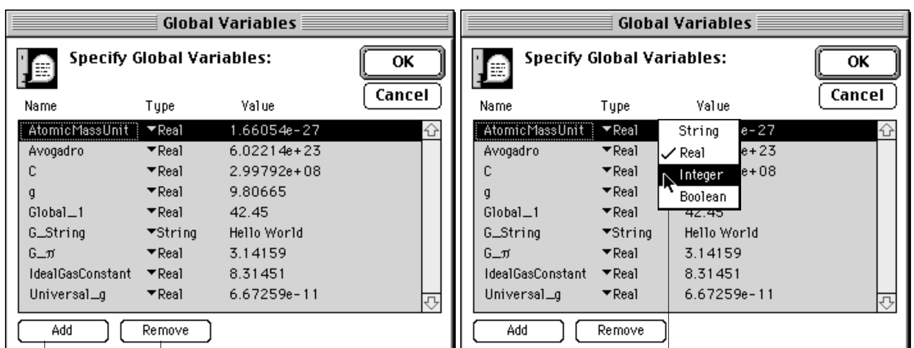
Global variables allow you to define constants which are independent of the Argus ONE layers in your project. Global variables can be of type real, integer, boolean and string. They can be accessed from the expression dialog and thus can be introduced into any Argus ONE expression, either into parameter expression or into export expression or variable.

### Defining and Editing Global Variables

You define a global variable by assigning it a name, a type and a value.

#### To define a global variable

1. From the Edit menu select Edit Global Variables...
   The Global Variable dialog opens
2. Click the Add button at the bottom of the dialog to add a new variable.
3. Click in the name text edit box and type its name.
4. Click the Type popup menu to set the variables type.
5. Click in the Value text edit box and enter the value.


*Click to remove a variable*

*Click to add a variable*

*Pull down to set the variable's type*

#### To remove a global variable

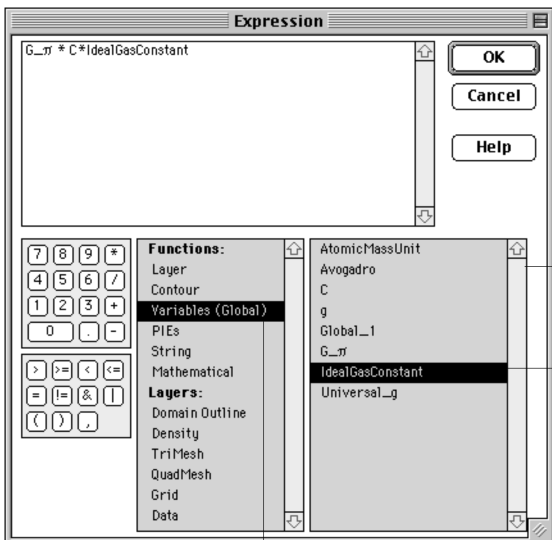
1. From the Edit menu select Edit Global Variables...
   The Global Variable dialog opens
2. Select the line of the variable you want to remove.
3. Click the Remove button at the bottom of the dialog.

## Using Global Variables

Global variables can be accessed from the expression dialog, both when it is invoked from the layers dialog and when it is invoked from the Edit template dialog. The expression dialog groups all global variables under a group named Variables (Global) to differentiate them from export template variables.

### To use a global variable

1. Open the expression dialog.
2. In the Functions and Layers list select the Variables (Global) group.
3. In the right list double click the variable you wish to enter to the expression or type its name.




*The Global Variables list*

*Select and double-click the variable to enter it into the expression text edit box above.*

*Select the Global Variables group to view all available variables in the list to the right.*

## Using Global Variables in Export Templates

If you are creating your own interface to a model, using global variables is a convenient way to set your model non-distributed parameters such as model control parameters.

# Group Type Layers

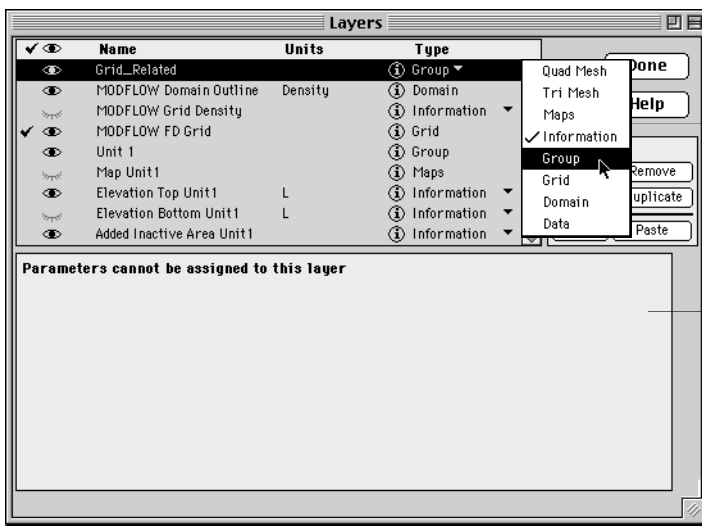
Group type layers enable you to organize any number and type of layers into groups. This allows you to hide, show and collapse all layers belonging to a group and thus better control your views. Group type layers have no parameters and can not store data or information of any type.

## Creating Group Type Layers

Group type layers, as layers of all other types, are created in the layers dialog. All layers below a group type layer and up to the next group type layer belong to that group.

### To create a group type layer

1. Open the layers dialog.
2. From the layers list, select the line of the layer below which you want to create a group type layer.
3. Click the New button in the layers control panel.
4. Click and hold the mouse button on the Type field to open the layers type popup menu and select Group.




*The layers type popup menu.*

*Select Group to define a group type layer.*

*Group type layers have no parameters and can't store data or information.*

For a detailed explanation on how to show and hide, and how to collapse and expand group layers in the layers floater refer to "Collapsing and Expanding Groups of Layers" on page 10 s4.

# The Hand Tool

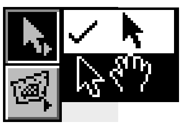
The hand tool enables you to scroll your view in the vertical and horizontal directions simultaneously.

## Using the Hand Tool

*Moving a document around with the hand tool is like moving a piece of paper around on a desk with your hand.*

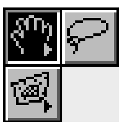
If you magnify your view to a size larger than the window, you can use the hand tool to move new areas into view.

The hand tool is interchangeable with the arrow tool and although it does allow you to select objects you can not move them with it.



### To select the hand tool

1. In the tool palette click and hold the arrow tool.
   A popup menu opens.
2. From the popup menu select the hand tool.

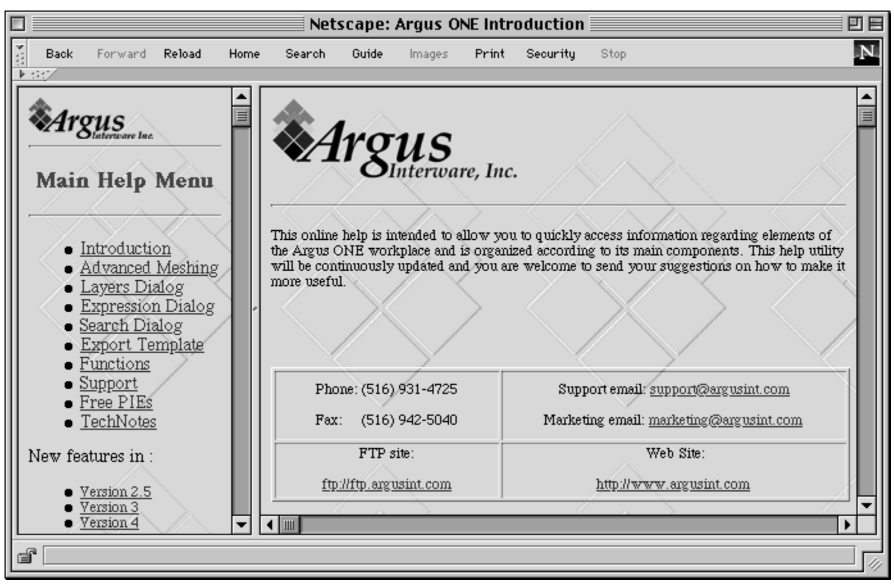


### To select the arrow tool

1. In the tool palette click and hold the hand tool.
   A popup menu opens.
2. From the popup menu select the arrow tool.

# On-line Help

On-line help is available through the browser of your choice. Whether you use the Help menu, or click the help button in a dialog, your browser will open to present the appropriate section in the on-line help.

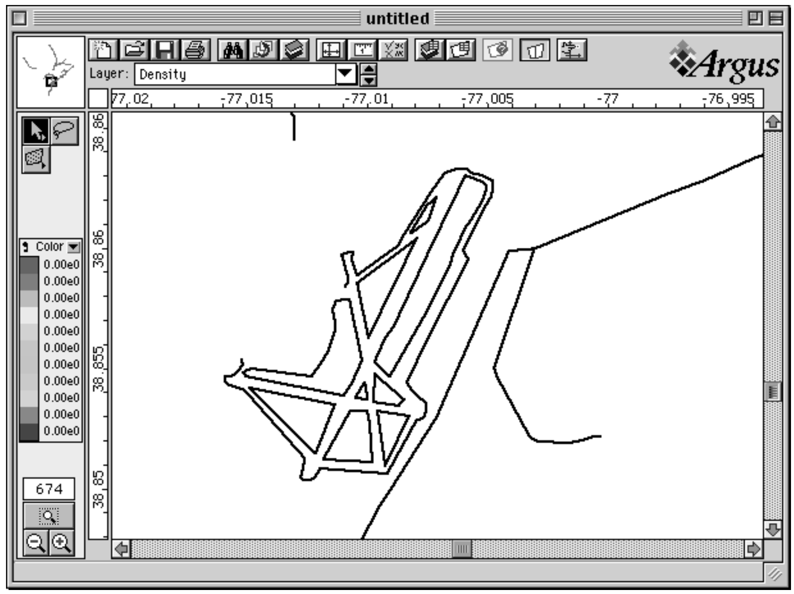
## Zoom to Fit

The zoom to fit command enables you to set your view so that all objects in your document fit in the current window size.

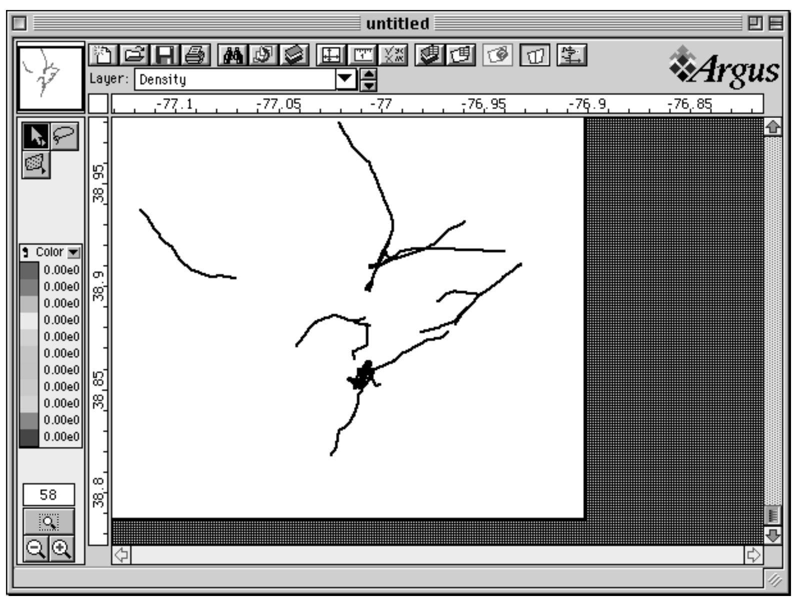
### To Zoom to Fit

- From the Navigation menu select Zoom to Fit.

*Not all objects fit in the current window size.*


*All objects fit in the current window size.*




*The Zoom to Fit command fits all objects in your document to the current window size, including those in* ***hidden layers****.*

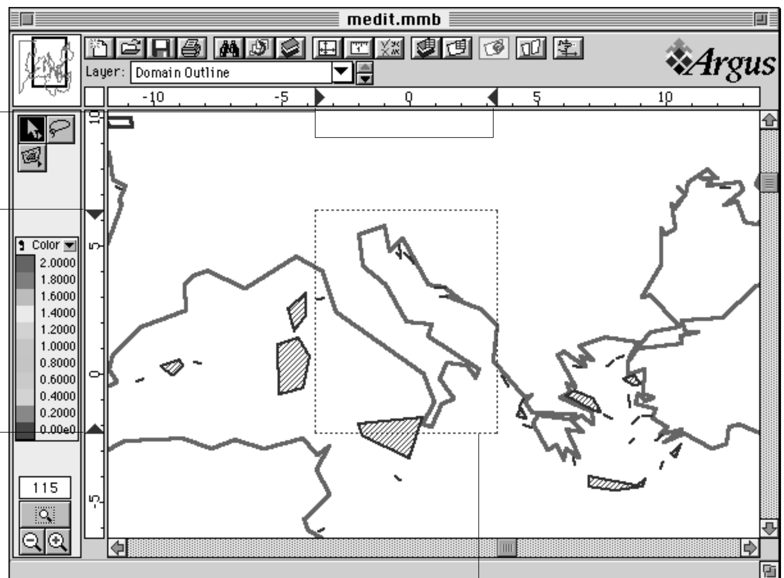
## Print Area

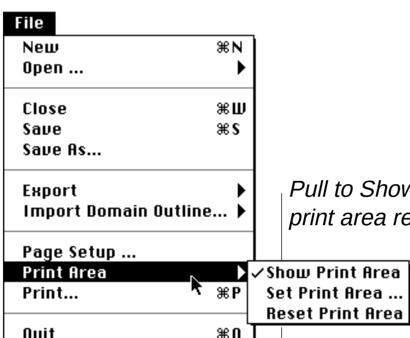
The print area option allows you to set an area within your project's drawing size that will print when you issue the print command. The default print area is set to be identical to the drawing size.

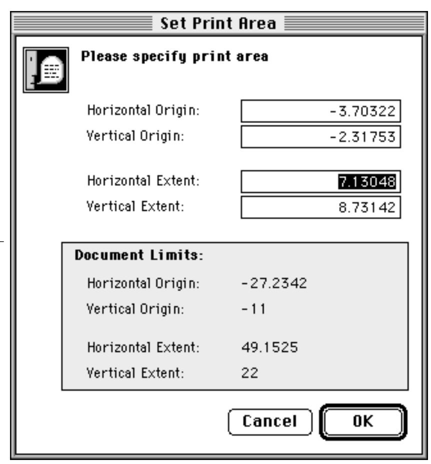
### Setting the Print Area

You can set the print area in one of two ways:

- Moving the Tab icons in the horizontal and vertical rulers.
- Using the Set Print Area dialog to define it numerically.



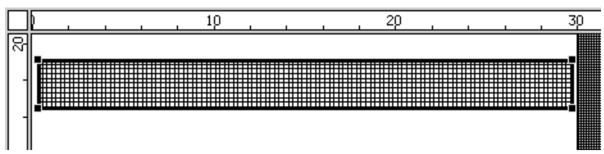



## Non Uniform Scaling

When you solve a cross-sectional problem, or a problem domain which lies along an elongated strip such as a river, one dimension of the domain would be much larger than the other. Such elongated problems present a visualization problem if the coordinate system is uniformly scaled, that is, when the scale of real world units to screen units is identical in both the X and Y axis.

Non uniform scaling is a very powerful option which enhances the Argus ONE workplace to allow one to design, work and visualize very elongated problem domains. The three Argus ONE meshing modules support this option and automatically create grids and meshes which take into account the non-uniformity of the problem domain while performing auto mesh and grid generation, thus creating elongated blocks and elements.

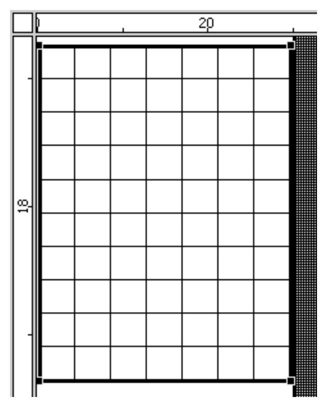


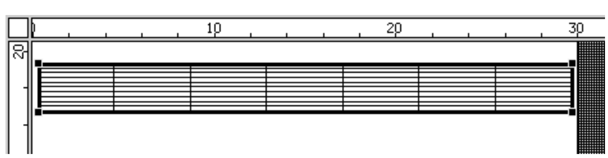

*Scale is set to,1 to 1 and the domain density is set to 0.25.*

*A domain of 30 units long by 2 units high is automatically grided using the above specified density and scale. The grid might be over discretisized in the X direction. Also viewing the grid using zoom-in is relatively limited because of the exaggerated length.*

*Scale is set to,15 (X) to 1 (Y) and the domain density is left at 0.25. When the domain is automatically grided at this scale, the X density (block side length) is reduced by the X scale (~0.25x15), while the Y density remains the same.*

*Two objectives are achieved:*
*1. The grid is less discretisized in the X direction, and*
*2. Viewing, zooming and editing is as convenient as working with non-elongated domains.*



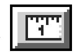

*You may at any time change the scale back to uniform (X=Y) and view your domain in the real world projection.*

*In the real world projection (X=Y) the blocks are elongated in the long axis of your problem domain.*

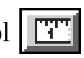
## The Scale and Units Dialog

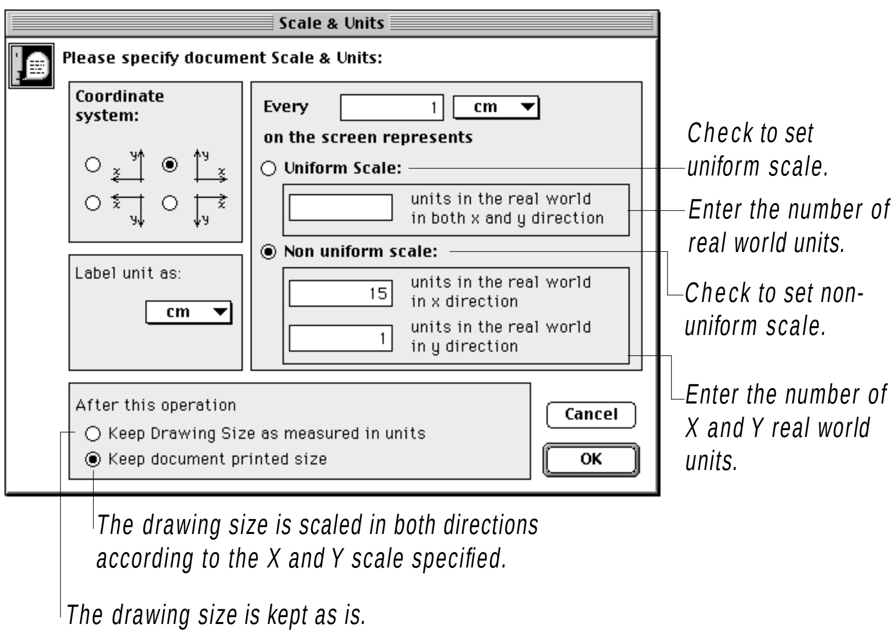
You switch between uniform and non-uniform scaling in the Scale and Units dialog.

### To set non-uniform scaling

1. In the toolbar click the Scale and Units tool .
2. Check the Non Uniform Scale radio button to set scale to non-uniform.
3. Enter the number of real world units in the X direction.
4. Enter the number of real world units in the Y direction.
5. Check the appropriate radio button to keep the drawing size or the printed size.

### To set uniform scaling

1. In the toolbar click the Scale and Units tool .
2. Check the Uniform Scale radio button to set scale to uniform.
3. Enter the number of real world units.
4. Check the appropriate radio button to keep the drawing size or the printed size.



## Effect of Non-Uniform Scale on Auto Mesh and Auto Grid Generation

Once you scale your project using a non-uniform scale, Argus ONE scales the domain and densities specified in the domain outline and density layers and sends this scaled information (as you see it on the screen) to the auto mesh and grid generation engines. After the auto mesh and grid generation finish their job, the mesh is rescaled and sent back to Argus ONE, where it is presented in the current scale. This process automatically creates elongated elements or blocks which might be required to decrease the number of equations solved.

### To use the non-uniform effect on meshes and grids

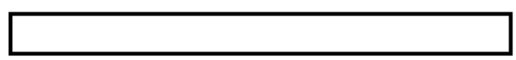
- Automatically mesh or grid a domain when non-uniform scale is on.

### To view your mesh or grid as they actually appear in the real world

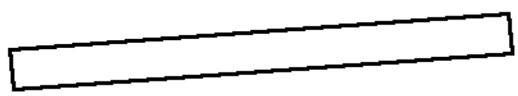
- Switch to uniform scale.

## Effect of rotation on objects presented in non-uniform scale

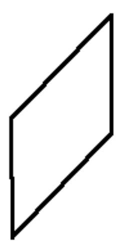
While rotated objects, when presented in uniform scale, retain angles, they do not conserve them when viewed in non-uniform scale. This is a mere visualization effect. If the scale is changed back to a uniform one, the real world is presented and it does conserve the angles.



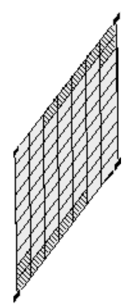
*1. A domain outline contour viewed in uniform scale. (lenght=30, height =2)*



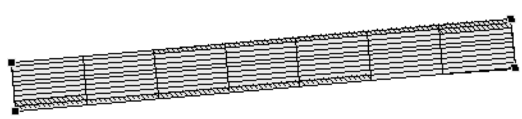
*2. The same domain, viewed in the same uniform scale, but rotated at 4 degrees from X.*



*3. The rotated domain viewed in non-uniform scale (15:1). Angles are not preserved.*



*4. The rotated domain overlaid with a grid automatically generated, viewed in non-uniform scale (15:1). Angles are not preserved.*



*5. The rotated grid viewed in uniform scale. Angles are preserved.*

# New Functions

The new functions added in version 4.0 are listed here by category.

## New Math Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Ceil*(number)*** | ***number*** | Rounds ***number*** (real) up to the nearest whole number. |

Example:
`Ceil (45.1)`— Returns `46.0`
`Ceil (45.9)`— Returns `46.0`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Floor*(number)*** | ***number*** | Rounds ***number*** (real) down to the nearest whole number. |

Example:
`Floor (45.1)`— Returns `45.0`
`Floor (45.9)`— Returns `45.0`

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **Rand*()*** | ***no arguments*** | Returns a random real number between 0 and 1. |

## New Contour Functions

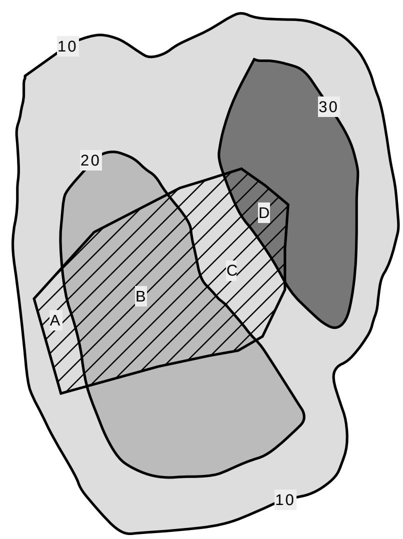
| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **EffectiveValue*( parameter_name)*** | ***parameter_name*** | The name of the parameter or layer with which contours in the current layer will be intersected.<br><br>Returns the sum of areas created by the intersection of a contour and the contours in the layer ***parameter_name*** belongs to, multiplied by the respective values of these contours. |

Example: In the following illustration the hatched contour belongs to a layer in which one of the parameters is assigned the expression `EffectiveValue(layer_name.parameter_name)`, where `layer_name` is the layer in which the other contours reside. The hatched contour is intersected with the contours in the referenced layer and returns the value: A*10+C*10+B*20+D*30, where A, B, C, and D are the areas of intersection between the hatched contour and the contours in the referenced layer, and 10, 20 and 30 are the values of the contours in the referenced layer.

If in the referenced layer, Allow Intersection is turned on, then the function will return: A*10+C*10+B*20+D*30**+B*10+D*10**, since the hatched contour is also fully encompassed by the most outer contour. This would also be the case if there were additional contours fully encompassing the hatched contour.

See also EffectiveValue() for mesh nodes and elements, and grid blocks.

*The hatched contour is tested vis-a-vis all contours in the referenced layer. Polygons A, B, C and D are created by the intersection of the hatched contour and the contours in the referenced layer. The area of each intersected polygon is multiplied by the value of the contour which intersected the hatched polygon, The sum of these products is returned.*



## New Grid Layer Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **GridAngle*()*** | | Returns the angle of rotation of the grid in radians. |

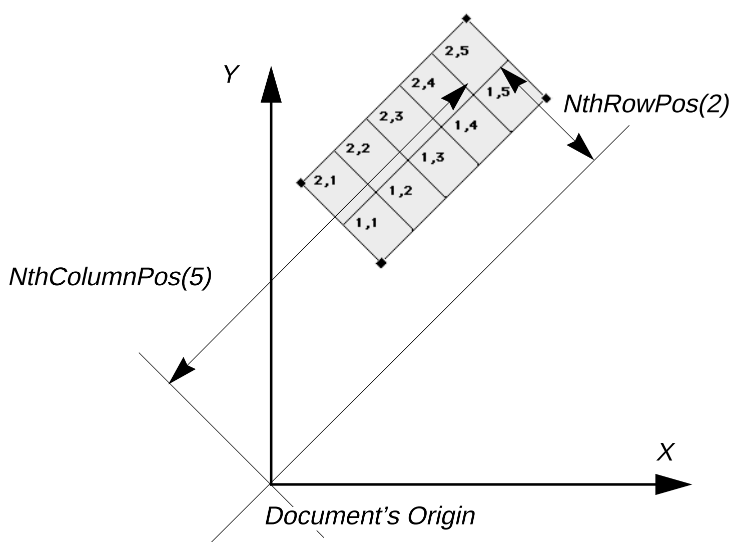
## New Block Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockGradX *(parameter_name)*** | ***parameter_name*** | Returns the X gradient calculated as central difference (real) between the value of ***parameter_name*** at the blocks to the left and right of the current block. |

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **BlockGradY *(parameter_name)*** | ***parameter_name*** | Returns the Y gradient calculated as central difference (real) between the value of ***parameter_name*** at the blocks to above and below the current block. |

**NthColumnPos *(col_num),* NthRowPos*(row_num)***

The two functions, described in Chapter 3, have been enhanced to return the appropriate values for rotated grids. If the grid is rotated these are measured as the distance from the document's origin along the grid rotated axes.



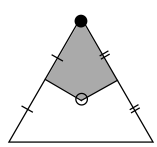
## New Node Functions

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **EffectiveValue *(parameter_name)*** | ***parameter_name*** | Intersects the nodes effective area with all contours in ***parameter_name*** to create a set of polygons, and returns the sum of polygons areas multiplied by their respective values. |

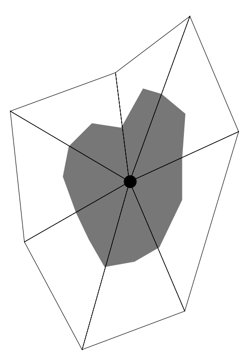
Example: In the following illustration the hatched polygon represents the node effective area. The hatched polygon is intersected with the contours in the referenced information layer and returns the value: A*20+B*10+C*30, where A, B and C are the areas of intersection of the node effective area (hatched) and the contours in the information layer, and 10, 20 and 30 are the values of the contours in the referenced information layer.

If in the referenced layer, Allow Intersection is turned on, then the function will return: A*20+B*10+C*30**+A*10+C*10**, this is because the hatched polygon is also fully encompassed by the most outer contour. This would also be the case if there were additional contours fully encompassing the node's effective area.

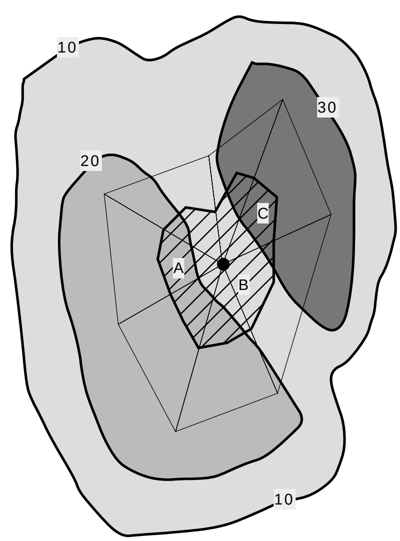
*Node EffectiveValue in Triangular Elements*



*Node effective area of a node in a single element.*

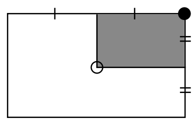


*Node effective area of a node common to several elements.*

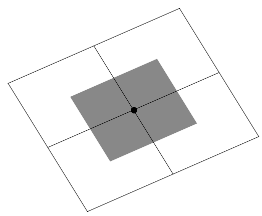



*The node's effective area is intersected with all contours in the referenced layer and returns the sum of polygon areas multiplied by their respective values.*

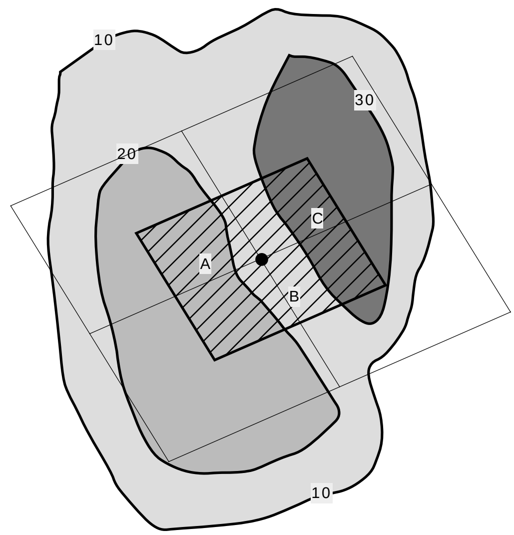
*Node EffectiveValue in Quadrilateral Elements*



*Node effective area of a node in a single element.*



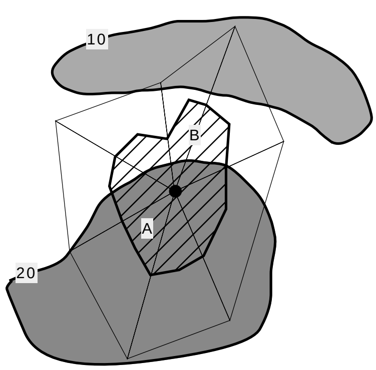
*Node effective area of a node common to several elements.*




*The node's effective area is intersected with all contours in the referenced layer and returns the sum of polygon areas multiplied by their respective values.*

Special cases: If part of the node's effective area lies outside all contours in the referenced layer, the function's return value varies with the referenced layer interpretation type. In the example below the area marked B is multiplied by:

- Nearest Contour method - The average of all adjacent contours. In this example B*(10+20)/2
- Exact Contour method - The default value of the referenced layer.
- Interpolation method - The interpolated value at the node's location.




*The area marked B, which is not intersected by a contour, is multiplied by a different value according to the referenced layer interpretation method.*

See also EffectiveValue() for contours, mesh elements and grid blocks.

# New Functionality in Information Type Layers

Information type layers now support Plug-In Extensions (PIEs) interpolators. Also, you may change an information type layer to a data type layer and vise-a-versa at any time.

## Using PIE Interpolators in Information Type Layers

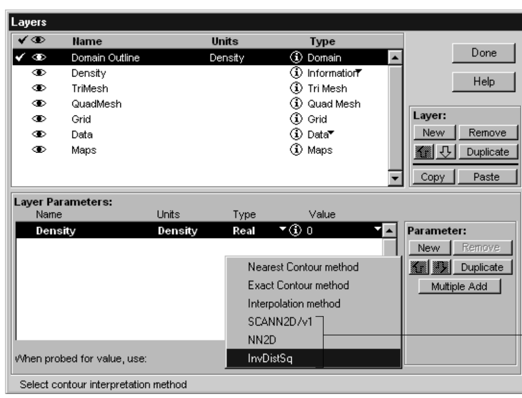
Information type layers allow for the use of an interpolator. With the introduction of Argus PIEs, you can use interpolators developed by others or yourself as the interpretation method for contours in information layers.

**Using an interpolator as the interpretation method for contours in information type layers is recommended only when the layer contains point objects.**

### To select a PIE interpolator

*PIE interpolators developed by third-parties are supported by their respective developers.*

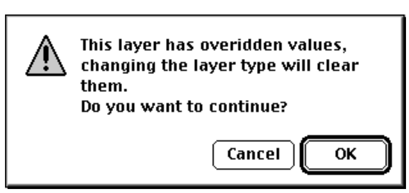
1. Open the layers dialog.
2. Select the layer its interpretation method you want to change.
3. From the interpretation method popup menu at the bottom of the dialog select the required interpolator.




*PIE Interpolators appear below the Argus generic interpolator (3rd on the menu).*

## Changing Information Type Layers to Data Type Layers

You may change information type layers to data type layers and vice a versa. If the layer contain objects an alert will come up warning you that you are about to delete all objects in that layer.



# New Maps Layer Capabilities

A number of new maps layer capabilities have been added. These include:

- Place Images
- Text annotation tool using system font
- Directly visualize data from mesh and grid layers
- Smooth contours in contour map post-processing objects
- New look of post-processing objects' dialogs
- Resize objects by click-drag

## Images

Placing images allow you to better orientate in your domain, and digitize information from images. Images can be placed in Maps type layers. Since Images are opaque it is recommended that you place images that cover the same area on the screen in different Maps type layers. This will allow you better control of image visibility by using the show/hide layer command and by using the promote demote layer option.

### Supported formats

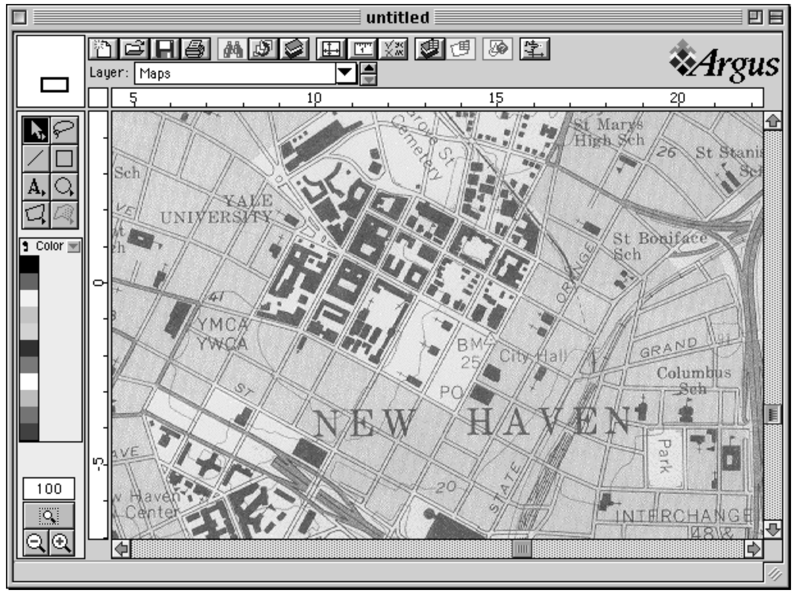
The following seven image formats are supported by Argus ONE.

| Format name | Format file extension |
|---|---|
| GIF | gif |
| TIFF | tif or tiff |
| XBitmap | bm |
| MacPaint | mcp |
| Windows DIB | BMP |
| Sun Raster | RIS |
| Nexpert Objects Images | NBI |

If you try to place a file which contains an image saved in a format other then those supported an alert will be presented.

### To Place an image

1. From the File menu select Place Image...
2. From the dialog box select the image file you wish to place.
3. The image appears on the screen.


If the image is large it can take a while for it to be placed. Because of technical limitations a progress bar is not available to indicate the time left to complete the process.

## Images and memory considerations

Contrary to vector graphics such as that of Argus ONE, DXF and Shape files, images consume large amounts of memory. Image size and the number of colors in the image are some of the factors effecting the amount of memory required by Argus ONE to present the image.

## Editing Images

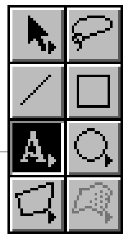
The following editing commands operate on images:

- Move
- Resize
- Scale
- Delete

The following editing commands are not applicable to images:

- Copy
- Rotate

## Text Annotation - System Font

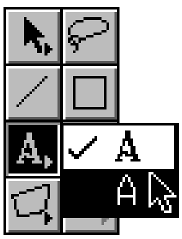


The Text tools.

The system-font text annotation tool, a variation of the vector font text annotation tool, uses a system font to present the text. Later versions of Argus ONE will allow you to select the system font from the available fonts installed on your system.

### To choose system-font or vector-font annotation text

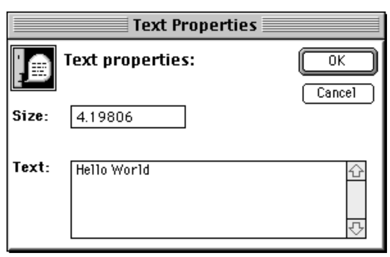
1. In the tool palette, click and hold the text tool to open the popup menu.
2. To select the system-font tool release the mouse button above the thick A icon.
3. To select the vector-font tool release the mouse button above the thin A icon.



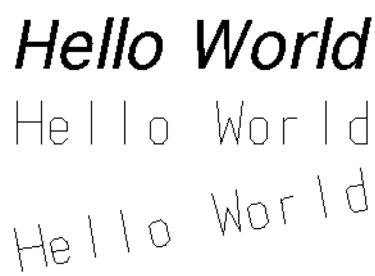
Click and hold to select text type.

### Creating and editing system-font annotation text

System-font annotation text is created and edited in the same manner as vector-font text. However, system-font annotation text can not be rotated. As a result, all imported text (which might be rotated) is represented using the vector font.




*System-font annotation text looks better but...*




*...Can't be rotated as vector text can.*

## New Look of Post-Processing Objects' Dialogs

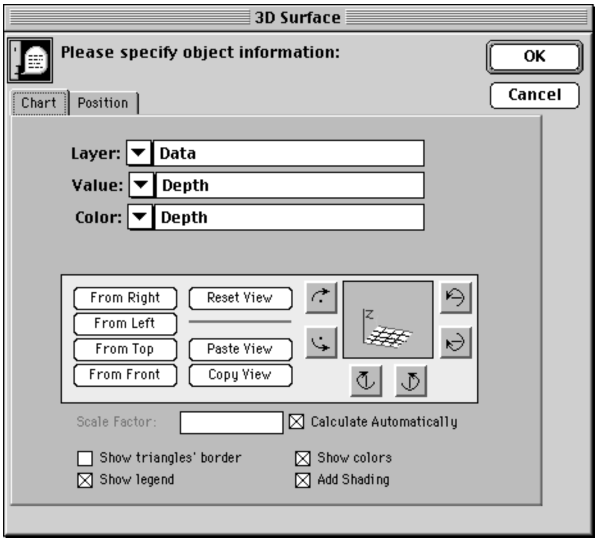
All post-processing objects' dialogs have been changed to tab dialogs. All dialogs have two panes. This change was introduced to allow for future panes offering more control of post-processing objects. The two panes available are the Chart pane and the Position pane.

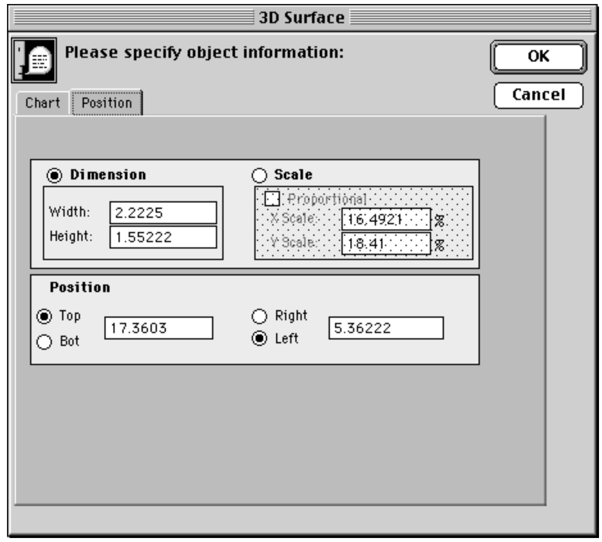
### To move between the panes

- Click the tab its pane you want to open.

*Click the Chart tab to bring its pane to the front*

*Click the Position tab to bring its pane to the front*

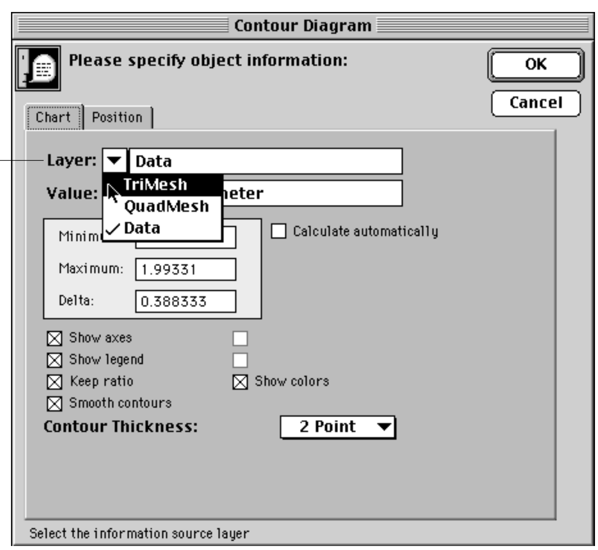



## Using Post-Processing Visualization Objects to View Mesh and Grid Data

Post processing objects can plot data directly read from mesh node parameters and grid and block parameters. Thus, to examine mesh and grid data you do not need to first read it into a data type layer. All grid, block, mesh layer and node parameters will be listed in the layer popup menu in the post-processing object dialog.

*All mesh layers having node parameters and all grid layers having block parameters will be listed as well as all data type layers.*

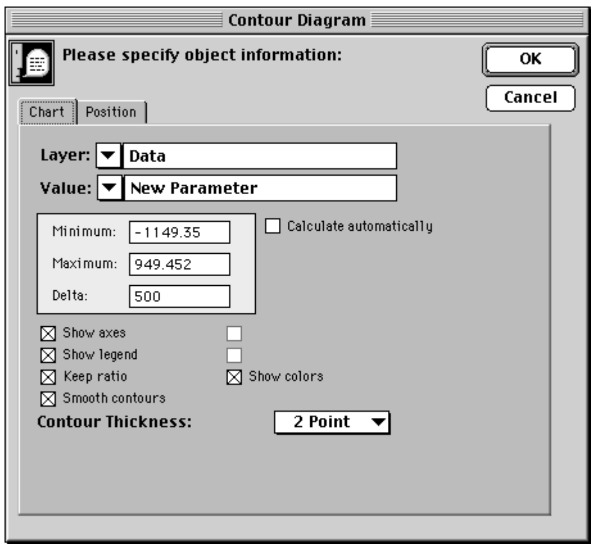


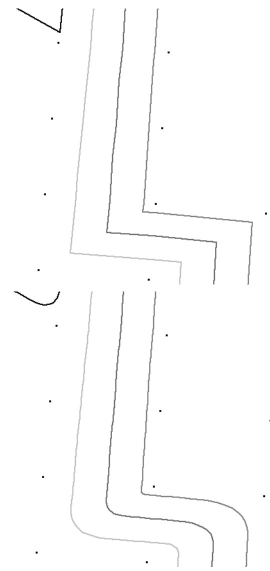
## Smooth Contours in Contour Diagram Objects

Contours created in Contour diagram post-processing objects may be smoothed.

### To smooth contours in contour maps

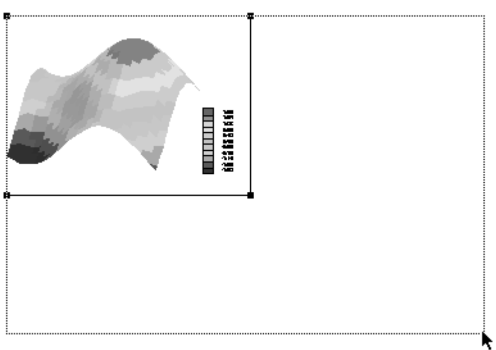
1. Open the Contour Diagram dialog box
2. Check the Smooth contours check box.
3. Click OK to close the dialog and accept the change.



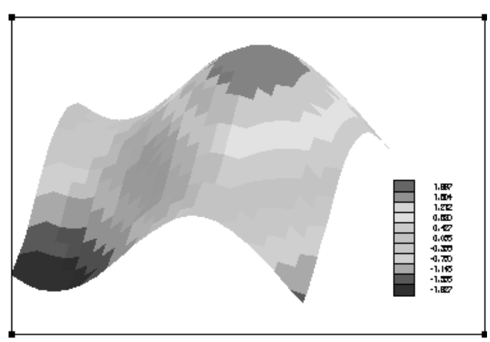



## Resizing Objects by Clicking and Dragging

All the objects available in Map type layers are resizable by click-drag.



*To resize an object select it and click drag one of its bullets. Drag the stretch rectangle to the required size and...*



*...release the mouse button. The object is resized.*

# Values Floaters

As Data type layers, Information, Mesh and Grid type layers have the ability to show values assigned to objects in them in a table type window. These windows, which are floating windows, are referred to as Values Floaters. They also include charting capabilities which enable you to plot values using various methods. These enhanced Values floaters allow you to resize columns, select the range of information for charting and they also automatically update when you add or delete objects to your layer, or add or delete parameters in the layers they belong too.

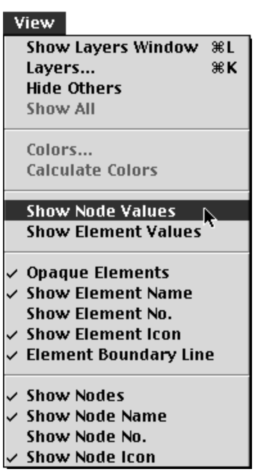
In future versions, you will be able to select an object in your layer and see its line highlighted in the values-floater, and vice-a-versa.

## Working with Values Floaters

You can have as many open values-floaters as you need at any time. You open a values-floater from within the active layer. However, once it is opened, it will stay open even if you switch the active layer. This allows you to view information in layers that are not currently active.

Values floaters belong to the project in which they were opened. If you have a number values-floaters opened in one project, and you switch to another project they disappear from sight. Once you bring to the front the project they belong to they reappear.

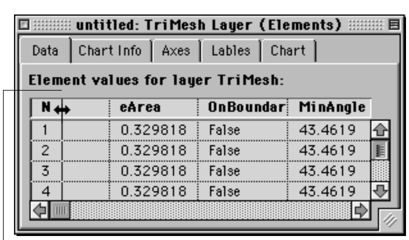
### Opening and Working with Values Floater



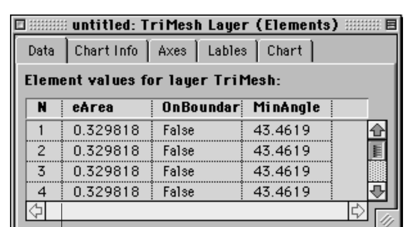

Once you have objects or data in Data, Information, Mesh and Grid type layers, a menu command named Show <object type> Values will be available under the View menu.

#### To resize the width of columns

1. Place the cursor above the dotted line between any two columns. The cursor changes into a two sided arrow.
2. Click-drag the cursor to the right or to the left to resize the column, a solid black line represents the new location of the column.
3. Release the mouse button.



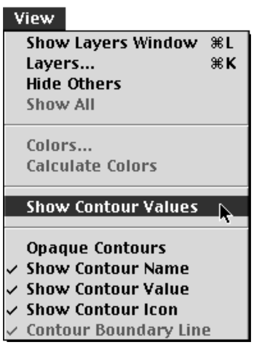

*Click-drag the cursor to resize the column width and...*

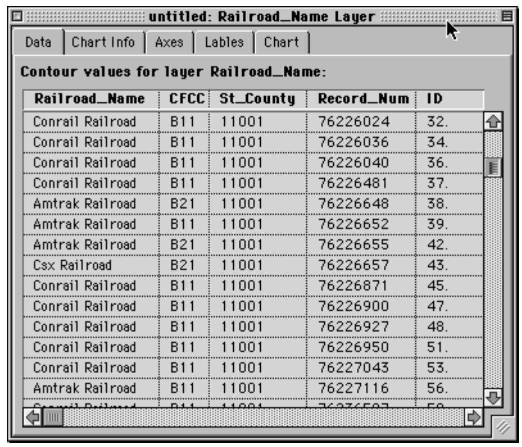



*...release the mouse button in the new location.*

## Values Floater - Information Type Layers

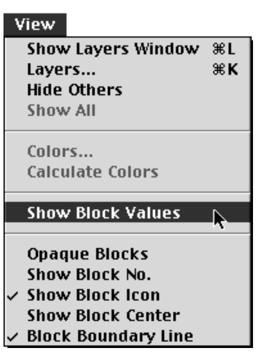
In information type layers the values-floater lists all contour objects and their values. If no parameters but the default one have been assigned to the layer, only the default's parameter values will be listed. The list updates as you add and delete objects or parameters. You may want to add parameters with contour functions such as ContourType() and ContourName() to display object specific information to assist you in identifying objects.

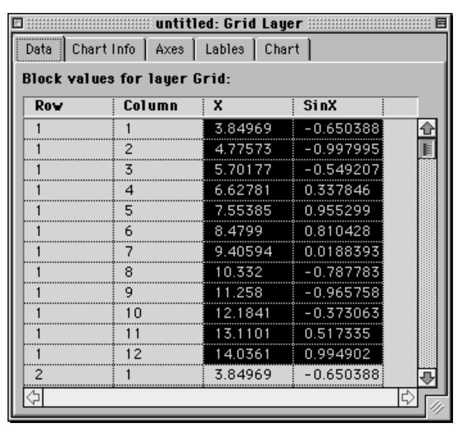



## Values Floater - Grid Type Layers

In grid type layers the values-floater lists all blocks and their values. This might become useful if you need to view numerically all block values as they will be exported to your model. If no parameters have been assigned to the layer, only row and column numbers are listed. The list updates as you add and delete blocks or parameters. You may want to add parameters with block functions such as BlockArea() and BlockName() to display block specific information to assist you in identifying blocks.



## Values Floater - Mesh Type Layers

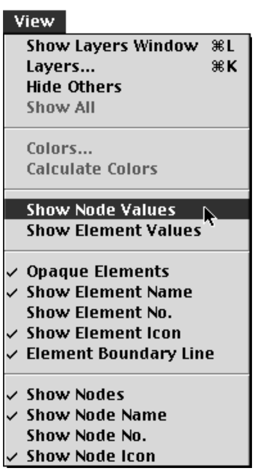
In mesh type layers two values-floaters are available, one for node values and the other for element values. This might become useful if you need to view numerically all node and element values as they will be exported to your model. You may want to add parameters with element or node functions such as ElementArea() and ElementName(), NodeName(), etc. to display element and node specific information to assist you in identifying elements or nodes.

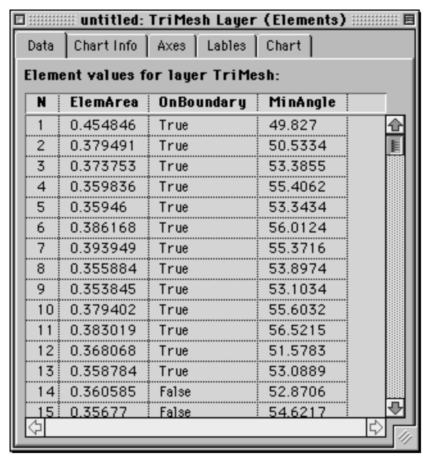
### The elements values-floater

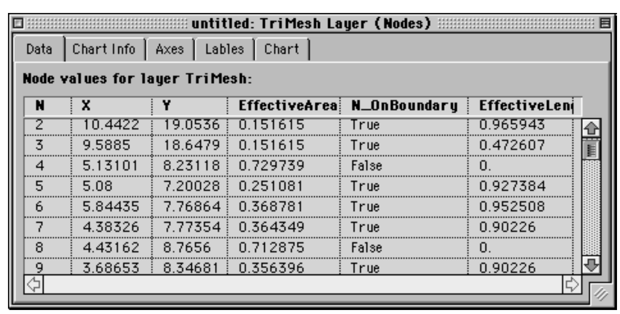
The elements values-floater lists all elements and their values. If no element or layer parameters have been assigned to the elements, only the element numbers are listed. The list updates as you add and delete elements or parameters.

### The nodes values-floater

The nodes values-floater lists all nodes and their values. If no node or layer parameters have been assigned to the nodes, only the node numbers and X and Y locations are listed. The list updates as you add and delete nodes or parameters.

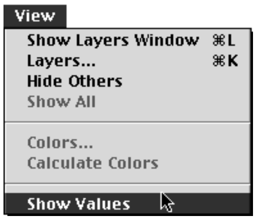


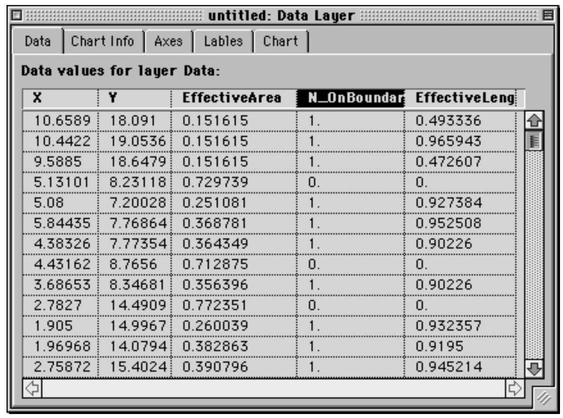



### Values Floater - Data Type Layers

In data type layers the values-floater lists all values assigned to data points. If there are no data points in the layer, the Show Values menu is dimmed.



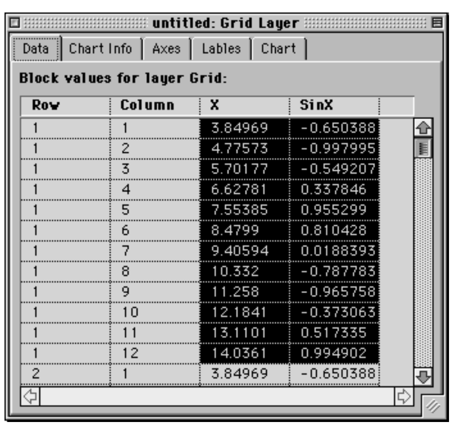

# Values Floaters and Charts

Value-floaters also feature charts. Using charts you can graphically investigate the distribution of values assigned to objects. There are eight chart types available and these can be controlled using the labels and axes control panes. These charts are volatile, they are not saved once you close the values-floater they appear in.

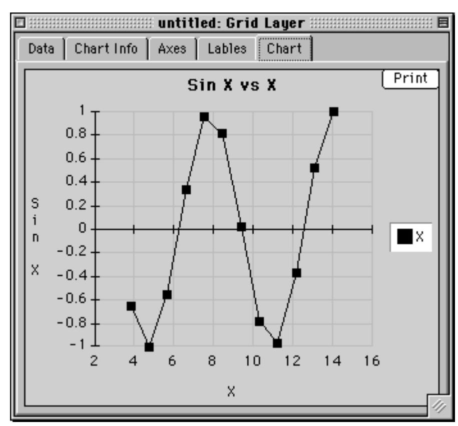
## Creating Charts

To create a chart you first need to select the cells to be charted.

*Once cells are selected you can click the Charts tab to view your chart. You may also click the Chart Info, Axes, and Labels tabs to adjust your chart.*




X and SinX are selected for row number 1 in the grid. Choosing XY Line chart type...




...and clicking the Chart tab produces a plot of the variation of SinX with X.

## Selecting Cells

Selection is performed by clicking the mouse in the cells. Selected cells are highlighted. You may use several techniques for cell selection.

### To select a single cell

- Click the cell. - The cell is highlighted.

### To select a number of cells using click-drag

- Click-drag from the upper-left cell to the bottom-right cell. If you stretch the selection outside the floater size the window will scroll automatically to allow you to add cells outside the current view. All cells within the rectangle are selected.

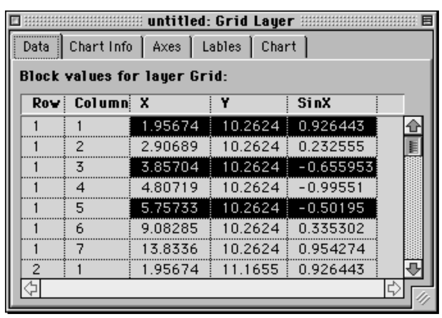
### To select a number of cells using click and shift-click

1. Click the first cell to be selected.
2. Shift-click the last cell to be selected.
   The group of cells (across rows and columns) is selected.

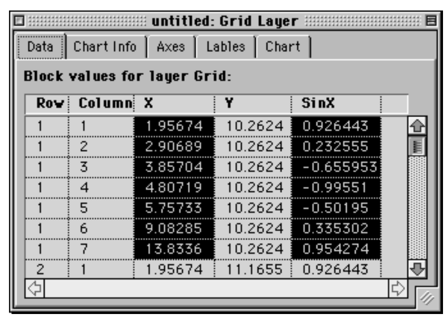
### To select discontinues cells or groups of cells

You may select discontinues groups of cells. This allows you to chart non-adjacent columns or rows.

1. Select the first group using one of the methods described above.
2. Hold the Control key and use any of the methods described above to add another non-adjacent group of cells to the selection.




Selecting discontinuous rows




Selecting discontinuous columns

### To select whole columns

You may select whole columns of cells.

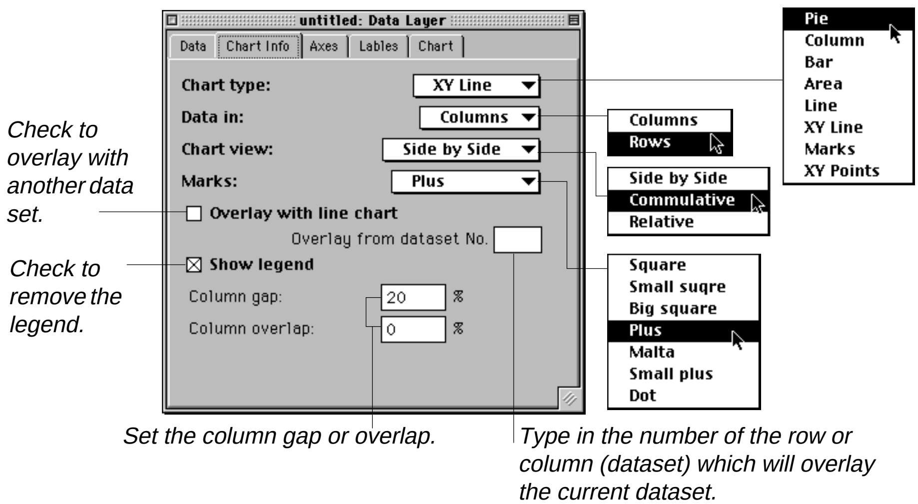
1. Click the column name cell.
2. To add more columns to the selection use the methods described above while clicking in the columns name cell.

# Controlling Charts

In addition to controlling the values to be charted and the chart size, you can control the chart type, the chart axes and labels. These are performed in the Chart Info, Axes and Labels panes.

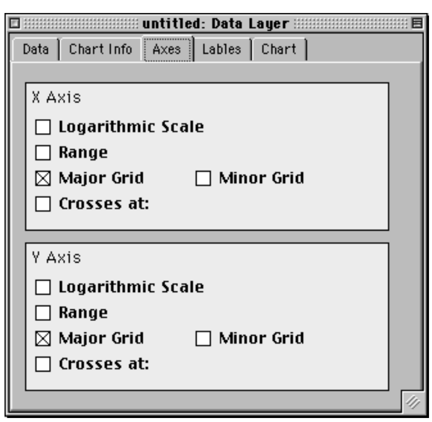
## The Chart Info Pane

The chart info pane allows you to set the chart type, to select whether data is in rows or columns, to set the chart view, to add marks at data points, to overlay the chart with another chart and to add a legend.



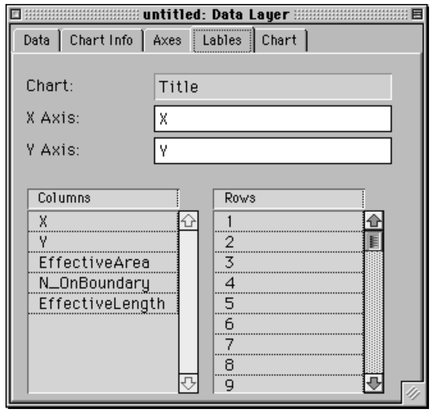

## The Chart Axes Pane

The chart axes pane allows you to control the X and Y axes of the chart including the scale, range, background grid, and where the axis will cross.



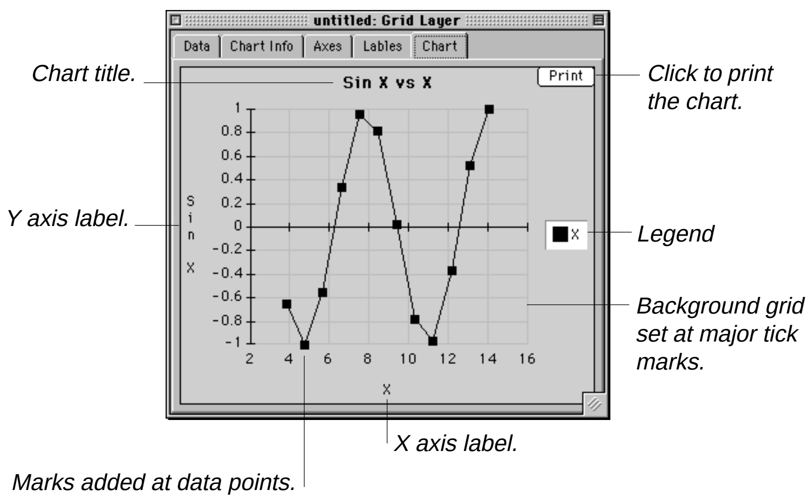
## The Chart Labels Pane

The chart labels pane allows you to set the names for all labels, including the title, the axes and the legend. The default values presented are taken from the parameters names appearing in the data pane.




## The Chart Pane

The chart pane is where your chart is presented. The chart automatically readjusts when you resize the window. You can print the chart by clicking the print button at the top-right corner of the window.



# New Export Script Commands

## Loop for Objects in Layer

Loop for objects in layer enables you to export objects stored in Information type layers from within a grid or a mesh export template. Future version which will feature export templates for information type layers, will support this loop type as well.

### Using Information Type Layer Functions

You can use all available information type layer functions from within this loop. The following example exports all contours in a layer called Domain Outline and for each object exports its name, area, length, type and value of the parameter named Density. It also exports for every contour the X and Y coordinates of its vertices.

```
Define Variable: i [integer]
Redirect output to: contours.exp
# The following loop will loop for every
# Object (contour) in the Domain Outline layer
# and for each object export some contour information
# and the value of the density parameter.
  Loop for: Objects in layer: Domain Outline
    Start a new line
      Export expression: ContourName(); [G0]
      Export expression: ContourArea(); [G0]
      Export expression: ContourLength(); [G0]
      Export expression: ContourType(); [G0]
      Export expression: Domain Outline.Density [G0]
    End line
# The following loop will export the X and Y coordinates
# of each vertex.
    Loop for: Variable i from: 1 to: NumVertices() step: 1
      Start a new line
        Export expression: NthVertexX(i); [G0]
        Export expression: NthVertexY(i) [G0]
      End line
    End loop
    Start a new line
    End line
  End loop
End file
```

### Nesting Loop for Objects in Layer within Loops for Blocks or Elements

You may nest a loop for Object in Layer within a loop over blocks and elements. This enables you to export all objects in a layer which intersect or are contained within a block or an element. Usually you will need to export information by block, however, if your code requires that you list all the objects in all layers per block or element you can nest a loop over objects in layer, within a loop over blocks or elements within a loop over layers.

The following example exports for each blocks the length of intersection of contours in each layer named rivers1, rivers2 and rivers3, as well as the area of intersection between each block and contours in each of the layers named lakes1, lakes2 and lakes3. The functions `ContourIntersectLength()` and `ContourIntersectArea()` which are unique to Loop for Object in Layer are described below. Also the use of square brackets as loop variable is also explained below.

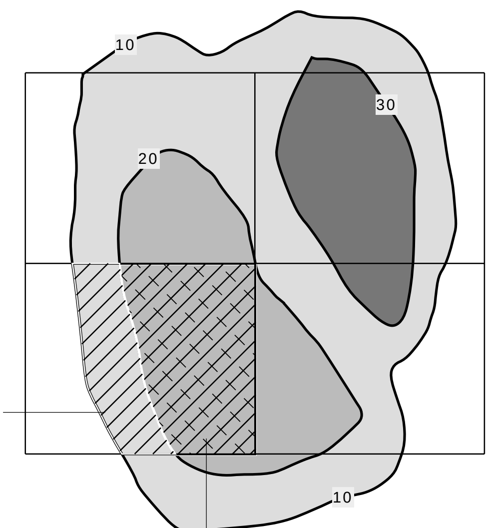
```
Define variable: i [integer]
Define variable: j [integer]
Define variable: NumLayers [integer]
Set variable: NumLayers:= 3
#
Redirect output to: rivers_lakes.exp
# Loop from 1 to 3
  Loop for: Variable i from: 1 to: NumLayers() step: 1
#   Loop for Blocks
    Loop for: Blocks
#     Loop for all objects in layers named rivers1,
#     rivers2, and rivers3 and export the length of
#     rivers which intersects with the current block
      Loop for: Objects in layer: Rivers[i]
       Export expression: ContourIntersectLength() [G0]
      End loop
    End loop
    Loop for: Blocks
#     Loop for all objects in layers named lakes1,
#     lakes2, and lakes3 and export the area of
#     lakes which intersects with the current block
      Loop for: Objects in layer: Lakes[i]
       Export expression: ContourIntersectArea() [G0]
      End loop
    End loop
  End loop
End file
```

### Functions unique to loop for objects in layer when nested within a loop for blocks or elements

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ContourIntersectArea*()*** | ***no arguments*** | Returns the area defined by the intersection of a closed contour and a block or element.<br>Operates on the information type layer the Loop for Objects in Layer is set to. |

Example: In the following illustration the bottom-left block is tested for ContourIntersectArea() and returns for each of the two contours intersecting it, the area of intersection (marked by a different hatch pattern). The areas returned may be exported, added or manipulated using other functions.

*Each block is tested in its turn (loop for blocks) with the 3 contours in the information layer (the layer the loop over objects in layer refers to), and returns the area of intersection. The bottom left block is intersected by the contour marked “10” and returns this hatched area.*

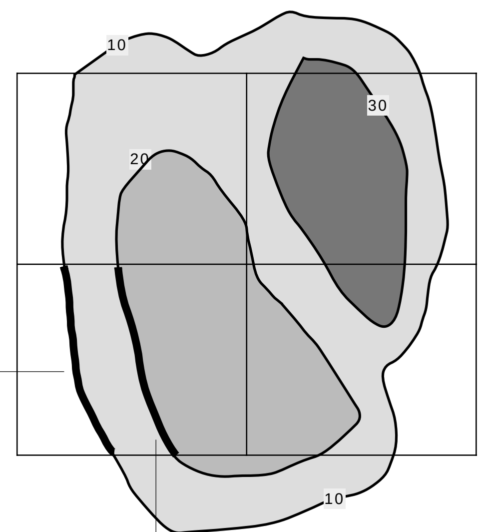



*The bottom left block is also intersected by the contour marked “20” and thus returns the crossed area as well.*

| *Name & Syntax* | *Arguments* | *Description & Return Value* |
|---|---|---|
| **ContourIntersectLength*()*** | ***no arguments*** | Returns the length defined by the intersection of a closed or open contour and a block or element. Operates on the information type layer the Loop for Objects in Layer is set to. |

Example: In the following illustration the bottom-left block is tested for ContourIntersectLength() and returns for each of the two contours intersecting it the length of intersection (marked by a thick line). The lengths returned may exported, added or manipulated using other functions.

*Each block is tested in its turn (loop for blocks) with the 3 contours in the information layer (the layer the loop over objects in layer refers to), and returns the length of intersection with each of the contours which intersect it. The bottom left block is intersected by the contour marked "10" and returns the length of this thick line.*




*The bottom left block is also intersected by the contour marked "20" and thus returns the length of intersection also marked by a thick line.*

## Looping Over Numbered Layers and Parameters

The square brackets [] notation allows one to automatically concatenate layers and parameter names with a loop variable. This allows you to repeat a sequence of an export template without prior knowledge of the numbers of layers to be exported.

In the following example `NumLayers` is either a global variable (see“Global Variables” on page 21s in this supplement) or a PIE variable which is set to the number of parameters sharing a name such as `Grid.Perm Unit1`, `Grid.Perm Unit2`, etc. As the loop progresses the loop variable `i`, the layer’s and parameter’s name, appearing before the square brackets, is concatenated with the value of `i` to produce the actual parameter name.

```
Loop for: Variable i from: 1 to: NumLayers step: 1
    Export matrix: Grid.Perm Unit[i] [G0]
End loop
```

## Export Arrays

Using arrays in exported templates is supported through an Array PIE. This PIE allows one to initialize, set and get values of a one dimensional array. Any number of such arrays can be used. For details regarding the use of Array PIEs refer to “Array PIEs” on page 54.

# New and Enhanced PIEs

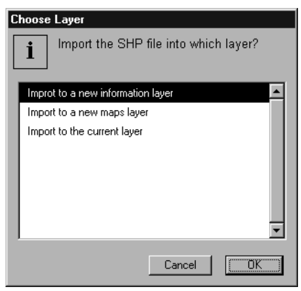
Although many PIEs are available, we describe here five PIEs which can be used by everyone. Other more specific PIEs are listed on the Argus web site.

## Enhanced Import Shape Files PIE

The shape file PIE now automatically filters .shp files, notifies you of the types of fields stored in a shape file, and automatically creates new layers and parameters into which shapes will be imported with their respective values.

### To import a shape file

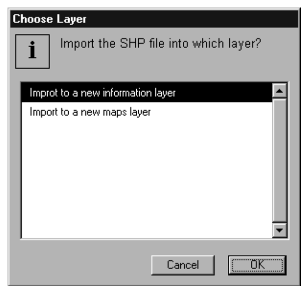
1. From the File menu select Import and Shape file.
2. Locate and open the file (.shp type)
3. Select the layer type into which the file will be read.
4. Select the fields to be imported. Fields' types are listed next to their names.




*Choose whether to import into the current layer or into a new one. New layers are named after the name of the shape file.*

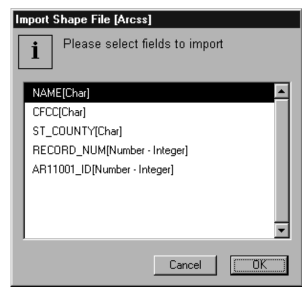
*If you import into an existing information type layer, and allow intersection is not on, shapes which intersect other shapes will be skipped. An alert will pop up and allow you to either accept or cancel the import.*

*Also, if you import into an existing information layer, the existing parameters will be used, and their type will not be changed to complement the actual type being imported.*




*If the current layer is a domain layer, in which allow intersection is not allowed, the PIE will only permit import into a new map or information layer.*

*Although allow intersection will be turned on when importing into a new information type layer, shapes which intersect themselves (allowed in ArcView, but not in Argus ONE information layers) will be skipped. To avoid skipping such shapes, import into a Maps type layer.*




*This dialog lists all the fields related to objects in the shape file. Click to select those you wish to import.*

*If a new information type layer is selected for import, Argus ONE creates a new layer, names it after the name of the shape file, creates layer parameters using the fields' names and types, and imports the data.*

*If the objects imported reside outside the current drawing size, Argus ONE will alert you and offer to resize it so that all new shapes will fit in the resized drawing area.*

## Array PIEs

A PIE named ArrayPIE is available for use within export templates. It offers a one dimensional array which can be initialized, assigned values and return them on call. You can use as many arrays PIEs as you need within an export template, by creating duplicates of the original ArrayPIE. The name by which the PIE will be called from within the export template is the name you give it when you duplicate it. If you want to use more than one such PIE you must have a unique name for each of the Array PIEs.

### Calling Functions for Array PIEs

*The members of an N size array start at 0 and end at N.-1*

There are four functions which are used to initialize the array, load its members, retrieve them, and dispose the array. The four functions are:

1. `ArrayInit(n)` - Initializes a one dimensional array of length `n`.
2. `ArraySet(i,v)`- Sets the member `i` to value `v`.
3. `ArrayGet(i)` - Retrieves the value stored in the `i` member and returns it.
4. `ArrayDispose()` - Disposes the array and the memory allocated to it.

If you wish to use two arrays, you may call the first Array1 (on PC Array1.dll) and the second Array2 (on PC Array2.dll). To use the above calls you should use the syntax (suffix the array name with the function): Array1Init(), Array1Set(), Array1Get(), Array1Dispose() for the first and Array2Init(), Array2Set(), Array2Get(), Array2Dispose() for the second.

The following example initializes a MyArray1 of length 10. While doing so it tests if the Array PIE exists and if the initialization terminated successfully. It then sets values to each of the array members and exports these values.

```
Set Variable: NumLayers:=10
Set Variable: ok:=1
Set Variable: Test:= MyArray1Init(NumLayers)
If: IsNAN(Test)
  Alert: “The PIE MyArray1.DLL could not be found!”
  Set Variable: ok:= 0
End if
If ok
  Loop for: Variable i from: 1 to: NumLayers step: 1
    Evaluate expression: MyArray1Set(i,i*2)
  End Loop
  Loop for: Variable i from: 1 to: NumLayers step: 1
    Start a new line
      Export expression: MyArray1Get(i) [G0]
    End line
  End Loop
  Evaluate expression: MyArray1Dispose()
End if
```

# Interpolator PIEs from SRI

Subterranean Research, Inc. has kindly allowed Argus to distribute its interpolators with the Argus ONE GIS module. If you find these PIEs useful or have any suggestions on how to enhance or improve them, please contact SRI (see address below).

Although the following documentation by Subterranean Research, Inc. (SRI) refers to use of their PIEs with data stored in data type layers, these interpolators are also available in information type layers. If an information layer contains close or open contours, those are sent to the interpolator PIEs as a list of their vertices assigned the contours' value.

## Using SRI PIEs

If you experience problems using these PIEs please contact SRI as well as Argus. Argus does not warrant the use of these PIEs in any way.

## Original Notes from SRI

NOTES FOR SRI_PIE2 Intended for Wintel Windows 95 or NT machines Original Release date: March 14, 1997 Revision Date:N.A.


Burlington, VT 05401 USA (802)-658-8878 infodesk@subterra.com

## What does it do

This PIE provides 3 different ways to interpolate data in Argus ONE data layers. The methods and the names by which they are referenced in the Interpolation Method popup menu are:

- InvDistSq - Inverse distance squared interpolator.
- NN2D - Nearest Neighbor interpolator.
- SCANN2D/v1 - Version 1 of 2-D Neural Network classifier/interpolator. (See technical notes below.)

### How to choose an interpolator from SRI_PIE2

In the Layers Dialog select the data layer whose data you want to interpolate using SRI_PIE2. (See Argus ONE User's Guide p. 56s in Supplement for version 3.)

- From the Interpolation Method popup menu, select an interpolation method (InvDistSq, NN2D, SCANN2D/v1).

## A simple example for testing the use of the PIE

In the following example a synthetic data set is created at mesh nodes, read into a data type layer, and then interpolated using NN2D. Simply substitute the reference name for any other interpolator desired.

### Creating a synthetic data set

1. In a mesh type layer create a mesh. (See Argus ONE Manual for help.)
2. Open the Layers Dialog.
3. Create in the mesh layer a new node parameter named OBSERVATIONS and for values assign it the expression X(). (This will assign the nodes the nodal X coordinate values at each node.)
4. Close the Layers Dialog.

### Reading the synthetic data set into a data layer

5. In the main window, select the DATA layer.
6. From the FILE menu select “Read Data from Layer”. A dialog box opens.
7. From the list, select OBSERVATIONS under the TriMesh layer. The data are read from all nodes to the data layer. The small dots indicate the position of data points.

### Setting the interpolation method to NN2D

8. Open the Layers Dialog.
9. In the layers list select the appropriate parameter of the Data layer.
10. From the Interpolation Method popup menu (at the bottom of the dialog), select NN2D.
11. Close the Layers dialog.

### Viewing the interpolated values at cursor position

12. As mouse is moved over the screen you will see, in the Information ruler, the X coordinate of the node nearest to the current location of the cursor.

## Some Important Technical Notes for Users of SCANN2D/v1

1.0 SCANN2D/v1 uses an artificial neural network that (1) matches input data exactly (like any other interpolator) and (2) estimates values at spatial locations between input data points. See the paper by Rizzo and Dougherty, “Characterization of aquifer properties using artificial neural, Neural Kriging”, Water Resources Research, 1994, for a thorough description.

2.0 SCANN2D/v1 is a classifier. That is, it assumes that the quantity being interpolated is described by its “class”, which is stored numerically as an integer. In SCANN2D/v1, class numbers 1 through 24 are allowed. As an example, say the hydraulic conductivity tests at a site are partitioned among 8 bins (with the class numbers 1 through 8 corresponding to ranges of hydraulic conductivity values for gravel, coarse sand, medium sand, fine sand, silty sand, silt, silty clay, and clay). Then SCANN2D/v1 will predict the class number for hydraulic conductivity at any specified point. To use this information in groundwater simulation, the user must then convert the hydraulic conductivity class number into an effective hydraulic conductivity value (say using Argus ONE's Expression facility in the Layers dialog).

3.0 SCANN2D/v1 uses "normalized coordinates", as described in the Rizzo and Dougherty paper cited above. Therefore, there are limitations on the amount of extrapolating that can be performed. The version released in SRI_PIE2 is limited as follows:

3.1 SCANN2D/v1 allows interpolation/extrapolation to points that lie within an "estimation rectangle". This estimation rectangle is defined by bottom-left and top-right coordinates of (xmin - xrange, ymin - yrange) and (xmax + xrange, ymax + yrange), respectively. The terms xmin, ymin, xmax, and ymax are the minimum and maximum x and y coordinates in the observational data that are being interpolated. The terms xrange and yrange are (xmax - xmin) and (ymax - ymin), respectively.

3.2 In SCANN2D/v1, any point outside of the allowed estimation rectangle is assigned the value 9999, which makes it easily detected during use.
Please contact Subterranean Research if your application requires an alternative estimation rectangle.

## For Mac or Unix users

The SRI_PIE2.DLL is not currently available for Macintosh or Unix computers. If this is a critical need for your organization, we would be happy to discuss it with you.

## Keeping Current with SRI's PIEs and Helper Applications for Argus ONE

This version of SRI's Plug-In Extensions for Argus ONE, SRI_PIE2.DLL, is available free for Argus Argus ONE users.

Subterranean Research, Inc. plans to update its PIE for Argus Argus ONE at least twice per year. Future versions, including updates, patches for bug corrections, and new features, will be available for an initial purchase and annual subscription fee. Pricing has not been set (as of March 14, 1997).

In addition, Subterranean Research, Inc. may be releasing one or more "Helper Applications" that can be used with Argus ONE, yet are not Plug-In Extensions. These will be available separately. Price and delivery are not yet available.

## Contacting Subterranean Research, Inc.

Please contact David Dougherty at Subterranean Research, Inc. with any questions or comments. Telephone (802)-658-8878, Fax (802)-658-8878, Email David.Dougherty@subterra.com.