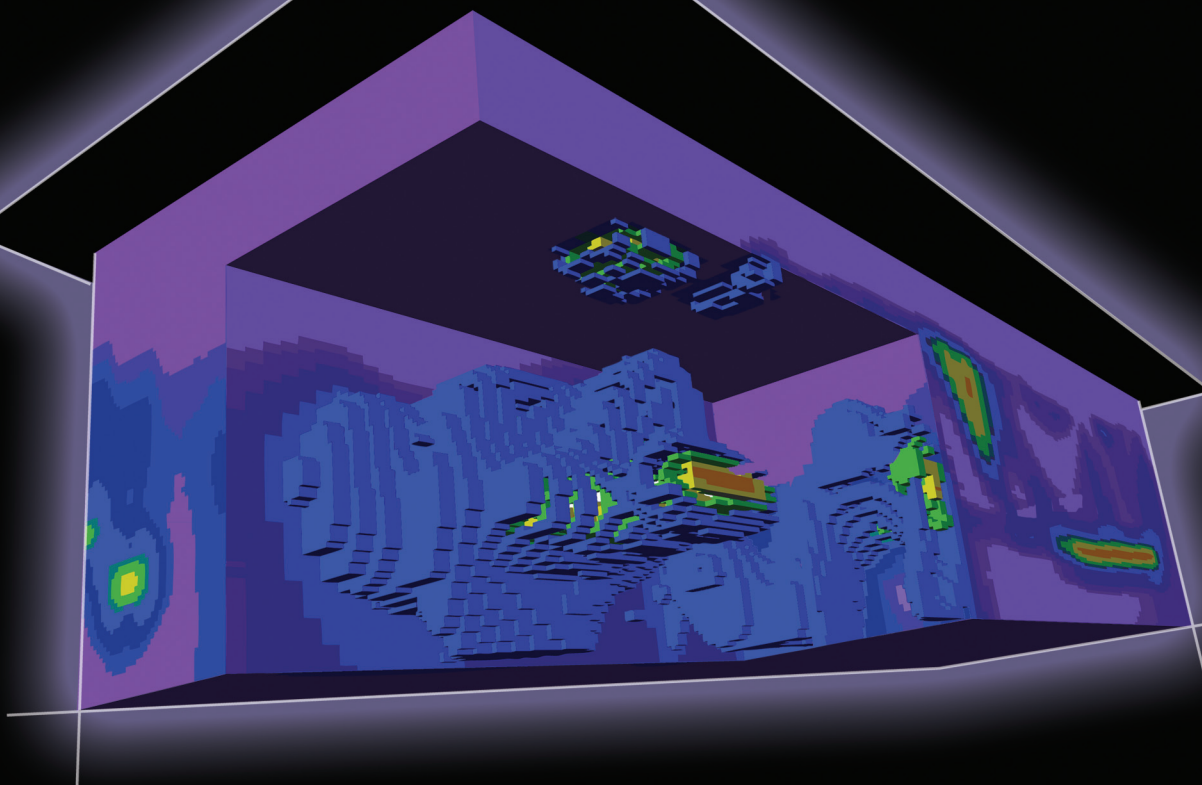


# voxler<sup>®</sup> 3

## Quick Start Guide



Golden Software, Inc.

## **Voxler<sup>®</sup> Registration Information**

Your **Voxler** serial number is located on the CD cover or in the email download instructions, depending on how you purchased **Voxler**.

Register your **Voxler** serial number online at [www.GoldenSoftware.com](http://www.GoldenSoftware.com). Or, complete the *Registration Form.PDF*, located in the main directory of the installation CD. Return the *Registration Form.PDF* by mail or fax. This information will not be redistributed.

Registration entitles you to free technical support, free minor updates, and upgrade pricing on future **Voxler** releases. The serial number is required when you run **Voxler** the first time, contact technical support, or purchase **Voxler** upgrades.

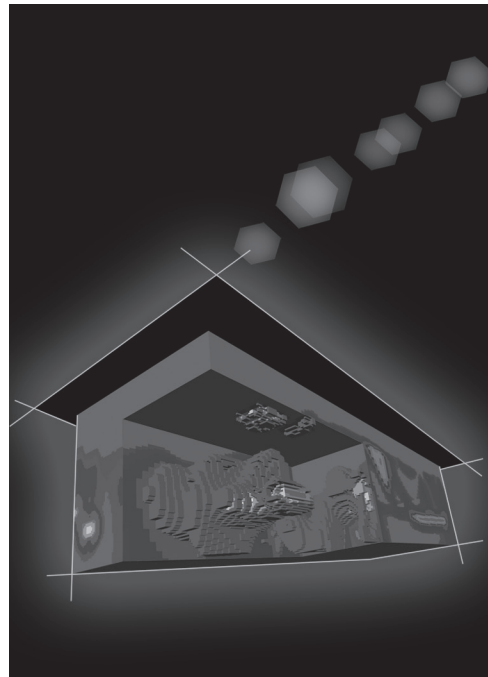
For future reference, write your serial number on the line below.

---

# **Voxler®**

## **Quick Start Guide**

3D Data Visualization



Golden Software, Inc.  
809 14<sup>th</sup> Street, Golden, Colorado 80401-1866, U.S.A.  
Phone: 303-279-1021 Fax: 303-279-0909  
[www.GoldenSoftware.com](http://www.GoldenSoftware.com)

## COPYRIGHT NOTICE

Copyright Golden Software, Inc. 2012

The **Voxler**® program is furnished under a license agreement. The **Voxler** software and quick start guide may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software or quick start guide on any medium except as specifically allowed in the license agreement. Contents are subject to change without notice.

**Voxler** is a registered trademark of Golden Software, Inc. All other trademarks are the property of their respective owners.

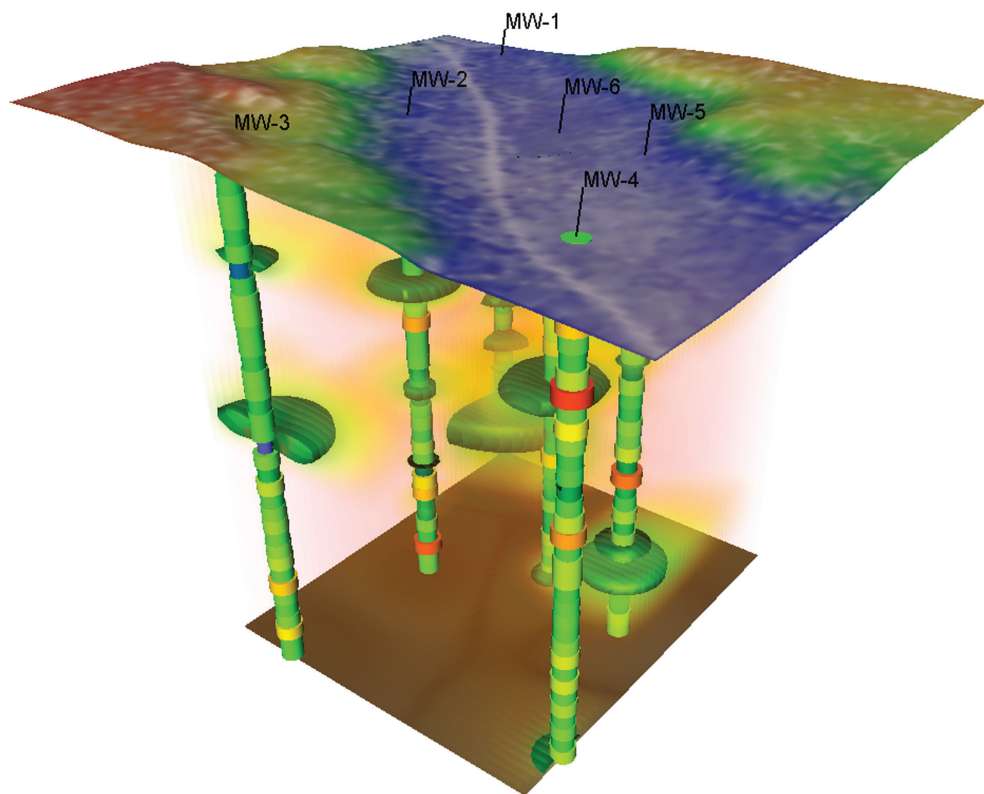
February 2012

## Table of Contents

Introduction to Voxler .....	1
Who Uses Voxler? .....	1
System Requirements .....	2
Installation Directions .....	2
Updating Voxler .....	2
Uninstalling Voxler .....	3
A Note about the Documentation .....	3
Three-Minute Tour .....	4
Example Voxler Files .....	4
Well Model (WellRender) .....	4
Iridium (FaceRender) .....	4
Using Voxler .....	5
Using Scripter .....	5
Example Script Files .....	5
Voxler User Interface .....	6
Menu Commands .....	8
Toolbars .....	8
Status Bar .....	8
Network Manager .....	8
Module Manager .....	10
Property Manager .....	11
Changing the Window Layout .....	12
Docking Managers .....	12
Viewer Window .....	12
Modules .....	13
Computational Modules .....	13
ChangeType .....	13
DuplicateFilter .....	14
ExclusionFilter .....	14
ExtractPoints .....	14
Filter .....	14
Gradient .....	14
Gridder .....	14
Math .....	15
Merge .....	15
Resample .....	15
Slice .....	15
Subset .....	15
Transform .....	15
Data Source Modules .....	15

Point Sets .....	16
Lattices.....	16
Geometry.....	16
WellData .....	16
FunctionLattice .....	17
TestLattice.....	17
General Modules.....	17
Info.....	17
Light .....	17
Viewer Window .....	17
Graphics Output.....	18
Annotation .....	18
Axes.....	18
BoundingBox .....	18
ClipPlane.....	18
Contours.....	18
FaceRender .....	19
HeightField .....	19
Isosurface .....	19
ObliqueImage .....	19
OrthoImage.....	20
ScatterPlot .....	20
StreamLines .....	20
Text.....	20
VolRender .....	21
WellRender .....	21
Automation.....	21
Scripter.....	22
Tutorial .....	22
Tutorial Lesson Overview .....	22
Using the Tutorial with the Demo Version.....	23
Starting Voxler.....	23
Lesson 1 - Loading Data .....	23
Lesson 2 - Creating Graphics Output Modules.....	26
Creating a ScatterPlot.....	26
Creating a Bounding Box .....	27
Lesson 3 - Changing Properties .....	28
Changing ScatterPlot Symbol Colors .....	28
Displaying ScatterPlot Labels .....	28
Changing the BoundingBox Properties .....	29
Rotating the View .....	30
Lesson 4 - Using Computational Modules .....	30
Gridding Data .....	31
Creating an Isosurface .....	31

Editing the Isosurface .....	32
A Note About Transparency.....	32
Filtering Data.....	32
Lesson 5 - Connecting Multiple Modules.....	34
Adding a Contours Module.....	34
Changing the Contours Properties.....	35
Changing the Transparency.....	36
Lesson 6 - Saving Information .....	36
Saving a Voxler File .....	36
Saving Data .....	37
Exporting Graphics .....	37
Copying Graphics .....	38
Capturing a Video.....	38
Lesson 7 - Importing and Displaying Wells .....	39
Importing Well Collar Information .....	39
Importing Trajectory Data .....	40
Displaying the Wells .....	41
Importing Log Data .....	41
Displaying Log Data on the Wells.....	43
Printing the Online Help.....	44
Printing One Topic .....	44
Printing One Book .....	44
Printing the Entire Help File .....	44
Getting Help .....	45
Online Help .....	45
Context-Sensitive Help.....	45
Internet Resources .....	45
Technical Support.....	46
Contact Information.....	46
Index.....	47





## Introduction to Voxler

Welcome to **Voxler**, a three-dimensional scientific visualization program, oriented primarily toward volumetric rendering and three-dimensional data display. While the emphasis is on three-dimensional volumes, **Voxler** can also utilize two-dimensional grids including DEM files, images, and scattered point data. **Voxler** can display streamlines, vector plots, contour maps, isosurfaces, image slices, three-dimensional scatter plots, direct volume rendering, three-dimensional block models, well traces, and more. Computational modules include three-dimensional gridding, resampling, numerous lattice operations, and image processing. **Voxler** is designed for displaying XYZC data, where C is a component variable at each X, Y, and Z location.

With **Voxler**, you can create stunning graphics output for your true three-dimensional models. Models can be sliced, displayed at any angle, and even animated with a simple mouse movement. Standard or custom colors can be applied to the models.

Creating three-dimensional models is as easy as importing the data file, applying any computational processes, such as gridding, to the data, and attaching the desired graphics output module. **Voxler** automatically selects reasonable default settings for each new graphics output. Customizing the display of the graphics is quick and easy for all module types. Calculating volumes directly from three-dimensional block models or within an isosurface is easy and provides a quick estimate of the volume of material displayed.

The **Scripter** program, included with **Voxler**, is useful for creating, editing, and running script files that automate **Voxler** procedures. By writing and running script files, simple mundane tasks or complex system integration tasks can be performed precisely and repetitively without direct interaction. **Voxler** also supports Active X automation using any compatible client, such as Visual BASIC. These two automation capabilities allow **Voxler** to be used for any three-dimensional modelling or display projects.

## Who Uses Voxler?

People from many different disciplines use **Voxler**. The geosciences generate large amounts of volumetric data from drill cores, seismic studies, ground penetrating radar, subsurface mapping, and remote sensing. Another source of data is from medical imaging generated by CT and MRI scans. Meteorological data, high-resolution microscopy, flow fields, and groundwater modeling are also sources for volumetric data. **Voxler** users include archeologists, climatologists, educators, engineers, doctors, hydrogeologists, geologists, geophysicists, medical researchers, students, and more. Anyone wanting to visualize the relationship of their three-dimensional data with stunning graphical output will benefit from **Voxler's** powerful features!

New features of **Voxler 3** are summarized:

- Online at: [www.GoldenSoftware.com/products/voxler/voxlernew.shtml](http://www.GoldenSoftware.com/products/voxler/voxlernew.shtml)
- In the program: click the **Help | Contents** command and click on the *New Features* page in the *Introduction* book

## System Requirements

The minimum system requirements for **Voxler** are:

- Windows XP SP2 or higher, Vista, 7, or higher
- 512 MB RAM minimum for simple data sets, 1 GB RAM recommended
- At least 100 MB of free disk space
- 1024 x 768 or higher monitor resolution with a minimum of 16-bit color depth
- Video card with OpenGL acceleration highly recommended

## Installation Directions

Installing **Voxler 3** requires logging onto the computer with an account that has Administrator rights. Golden Software does not recommend installing **Voxler 3** over any previous version of **Voxler**. **Voxler 3** can coexist with older versions (i.e. **Voxler 2**) as long as they are installed in different directories. By default, the program directories are different. For detailed installation directions, see the *Readme.rtf* file.

To install **Voxler** from a CD:

1. Insert the **Voxler** CD into the CD-ROM drive. The installation program automatically begins on most computers. If the installation does not begin automatically, double-click on the *Autorun.exe* file located on the **Voxler** CD.
2. Click *Install **Voxler*** from the **Voxler Auto Setup** dialog to begin the installation.

To install **Voxler** from a download:

1. Download **Voxler** according to the directions you received.
2. Double-click on the downloaded file to begin the installation process.

## Updating Voxler

To update **Voxler**, open the program and click the **Help | Check for Update** command. This will launch the Internet Update program which will check Golden Software's servers for any free updates. If there is an update for your version of **Voxler** (i.e. **Voxler** 3.0 to **Voxler** 3.1), you will be prompted to download the update.

## Uninstalling Voxler

**Windows XP:** To uninstall **Voxler**, go to the Windows Control Panel and double-click *Add/Remove Programs*. Select **Voxler 3** from the list of installed applications. Click the *Remove* button to uninstall **Voxler 3**.

**Windows Vista and 7:** To uninstall **Voxler** when using the *Regular Control Panel Home*, click the *Uninstall a program* link. Select **Voxler 3** from the list of installed applications. Click the *Uninstall* button to uninstall **Voxler 3**.

**Windows Vista:** To uninstall **Voxler** when using the *Classic View Control Panel*, double-click *Programs and Features*. Select **Voxler 3** from the list of installed applications. Click the *Uninstall* button to uninstall **Voxler 3**.

## A Note about the Documentation

The **Voxler** documentation includes this quick start guide and the online help. General information is included in the quick start guide. Detailed information about each command and feature of **Voxler** is included in the online help. Use the **Help | Contents** command in the program to open the online help. In the event the information you need cannot be located in the online help, other sources of **Voxler** help include our support forum, knowledge base, FAQs, newsletters, blog, and contacting our technical support engineers.

You can also purchase a full PDF user's guide that includes all of the documentation for the program. This PDF user's guide can be printed by the user, if desired. The guide can be purchased on the Golden Software website at [www.GoldenSoftware.com](http://www.GoldenSoftware.com).

Various font styles are used throughout the **Voxler** documentation. **Bold** text indicates menu commands, dialog names, window names, and page names. *Italic* text indicates items within a manager or dialog such as group names, options, and field names. For example, the **Save As** dialog contains a *Save as type* drop-down list. Bold and italic text also may be used occasionally for emphasis.

In addition, menu commands appear as **File | Open**. This means, "click on the **File** menu at the top of the **Voxler** window, then click on **Open** within the **File** menu list." The first word is always the menu name, followed by the commands within the menu list.

## Three-Minute Tour

We have included several example files with **Voxler** so that you can quickly see some of **Voxler's** capabilities. Only a few example files are discussed here, and these examples do not include all of **Voxler's** many graphics output types and features. The **Object Manager** is a good source of information as to what is included in each file.

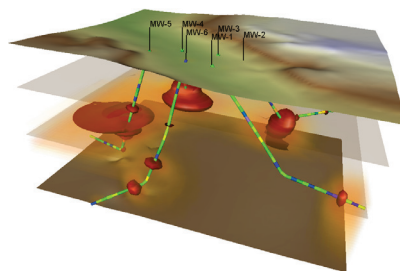
## Example Voxler Files

To view the example **Voxler** files:

1. Open **Voxler**.
2. In the **Module Manager**, open the *Samples* folder.
3. Double-click on any sample file name to open the sample file.

### Well Model (WellRender)

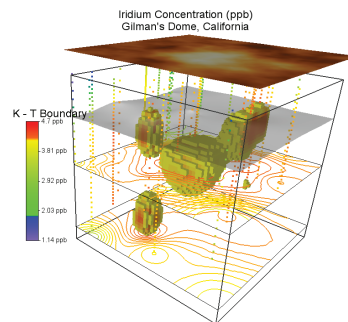
The well model sample file contains a *WellData* module connected to several graphics output modules, include a *WellRender* module. Wells are displayed as tubes in the **Viewer** window. The *WellData* information is extracted into a data file using the *ExtractPoints* module. The data is then gridded and multiple *HeightField* layers are displayed. A partially transparent *Isosurface* is also displayed in the **Viewer** window.



*The Well Model (WellRender) sample contains multiple deviated wells, HeightFields, and an Isosurface module.*

### Iridium (FaceRender)

The iridium sample file contains a three-dimensional *FaceRender* module. The *FaceRender* displays 3D blocks that can be used for volume computations. The iridium sample file also contains *ScatterPlot*, *Contours*, and *HeightField* modules to expand on the data presented. The sample also contains a legend, explaining the colors in the file and annotations.



*The Iridium (FaceRender) sample contains 3D blocks, displaying concentrations of iridium in at Gilman's Dome, California.*

## Using Voxler

The general steps to progress from an XYZC data file to a finished *ScatterPlot* graphics output are as follows.

1. Open **Voxler**.
2. Click the **File | Import** command.
3. In the **Import** dialog, select the data file and click the *Open* button.
4. In the **Data Import Options** dialog, set the file format options. You can select *Delimiters* and how to treat text. Click *OK*.
5. In the **Select Data Columns** dialog, set the *X*, *Y*, *Z*, and *Component* information. Set any additional *Options* and click *OK*. The data loads into **Voxler** and is displayed as a data module in the **Network Manager**.
6. Right-click on the data module and choose the **Graphics Output | ScatterPlot** command. A *ScatterPlot* module is displayed in the **Viewer** window.
7. Select the *ScatterPlot* module in the **Network Manager** and the properties are displayed in the **Property Manager**. Adjust the properties as desired.
8. Click the **File | Save As** command. Enter a *File name* in the **Save As** dialog and click the *Save* button to save your **Voxler** project.

## Using Scripter

Tasks can be automated in **Voxler** using Golden Software's **Scripter** program or any ActiveX Automation-compatible client, such as Visual BASIC. A script is a text file containing a series of instructions for execution when the script is run. You can do almost everything with a script that you can do manually with the mouse or from your keyboard in the program. Scripts are useful for automating repetitive tasks and consolidating a sequence of steps. **Scripter** is installed in the same location as **Voxler**. Refer to the *Voxler Automation* help book in the online help for more information about **Scripter**. Several example scripts are included in the online help and in the **Voxler** directory so you can quickly see **Scripter's** capabilities.

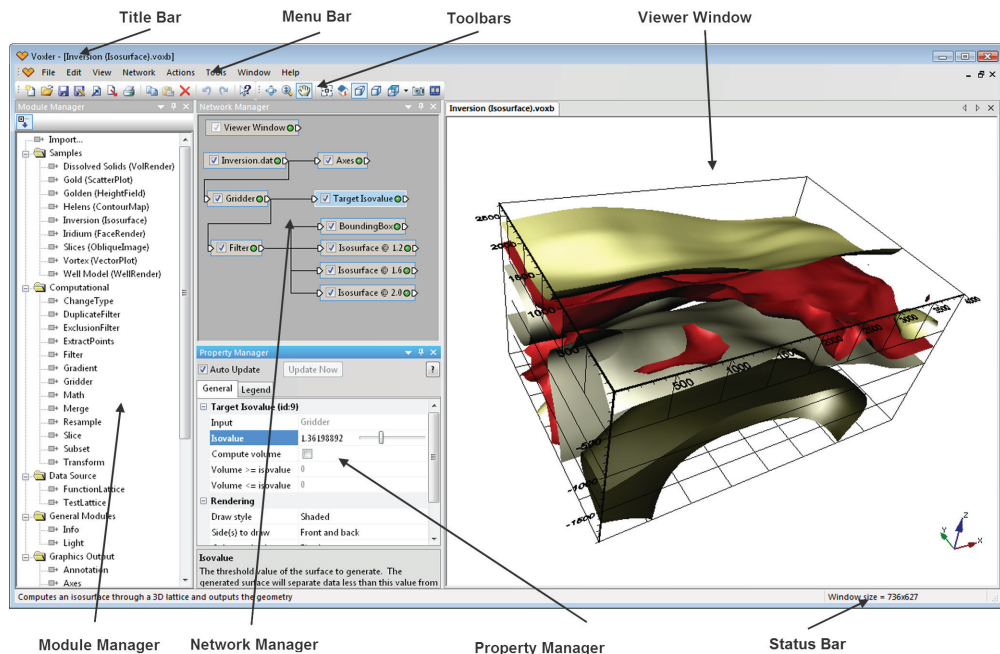
### Example Script Files

To run a sample script:

1. Open **Scripter** by navigating to the installation folder, C:\Program Files\Golden Software\Voxler 3\Scripter. Double-click on the Scripter.EXE application file.
2. Click the **File | Open** command and select a sample script .BAS file from the C:\Program Files\Golden Software\Voxler 3\Samples\Scripts folder.
3. Click the **Script | Run** command and the script is executed.
4. Most sample scripts will open **Voxler** and display and edit a module in the **Viewer** window.

## Voxler User Interface

The **Voxler** user interface consists of the title bar, menu bar, toolbars, **Viewer** window, managers, and status bar. **Voxler** uses multi-threading to keep the user interface responsive, even with computationally intensive background tasks. The user interface is based on a single-document, multi-view model. This allows a document to exist with any number of view windows open, enabling visualizing the model from multiple view points.



*This is the **Voxler** window with the **Module Manager** on the left side. The **Network Manager** and **Property Manager** are stacked in the center of the screen, and the **Viewer** window, where the graphics are displayed, is on the far right. The toolbars and menu are displayed at the top and the status bar is displayed at the bottom.*

The following table summarizes the function of the **Voxler** layout components.

Component Name	Component Function
Title Bar	The title bar lists the program name plus the saved <b>Voxler</b> file name, if any. An asterisk (*) after the file name indicates the file has been modified since it was last saved.
Menu Bar	The menu bar contains the commands used to run <b>Voxler</b> .
Toolbars	The toolbars contain <b>Voxler</b> tool buttons, which are shortcuts to menu commands. Move the cursor over each button to display a tool tip describing the command. Toolbars can be docked or floating.
Tabbed Windows	Multiple <b>Viewer</b> windows are displayed as tabs. Click on the tab to display a window.
Module Manager	The <b>Module Manager</b> contains a list of the modules that can be added to a <b>Voxler</b> project. These modules provide a quick way of adding modules to the <b>Network Manager</b> . The <b>Module Manager</b> is initially docked on the left side. The <b>Module Manager</b> can be dragged and placed at any location on the screen.
Network Manager	The <b>Network Manager</b> displays the modules, data, and connections in the existing <b>Voxler</b> project. The <b>Network Manager</b> is initially docked in the middle top position on the screen.
Property Manager	The <b>Property Manager</b> allows you to edit any of the properties of the selected module. Changes made in the <b>Property Manager</b> are immediately reflected in the <b>Viewer</b> window. The <b>Property Manager</b> is initially docked in the middle bottom position on the screen.
Viewer Window	The <b>Viewer</b> window contains the graphics output as directed by the modules in the <b>Network Manager</b> . The <b>Viewer</b> window is initially located to the far right side of the screen.
Status Bar	The status bar displays information about the activity in <b>Voxler</b> . The status bar is divided into two sections. The left section displays help messages and progress text. The right section usually displays the window size. It can also display a progress gauge or the estimated time remaining for long tasks.

## Menu Commands

The menus contain commands that allow you to add, edit, and control the modules in the **Viewer** window. See the *Menu Commands* help book in the online help for details about the various menu commands.

## Toolbars

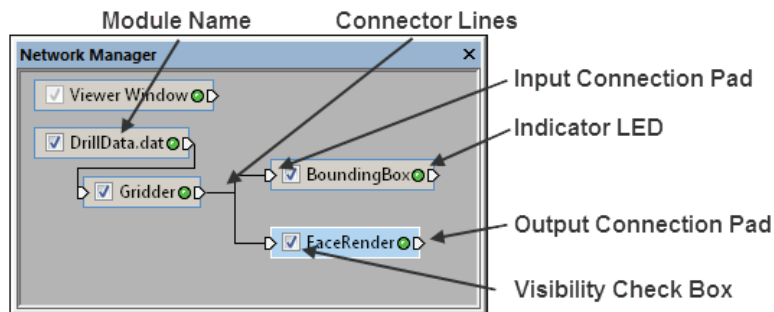
Toolbars display buttons that represent menu commands for easier access. Use the **View | Toolbars** command to show or hide a toolbar. A check mark is displayed next to visible toolbars. Hold the cursor over any tool button on the toolbar to display the function of the button as a screen tip. A more detailed description is displayed in the status bar at the bottom of the window.

## Status Bar

The status bar is located at the bottom of the window. Use the **View | Status Bar** command to show or hide the status bar. The status bar displays information about the current command or activity in **Voxler**. The status bar is divided into two sections. The left section displays help messages and progress text. The right section normally displays the window size. It can also display a progress gauge and the estimated time remaining for long tasks.

## Network Manager





**Voxler** uses a **Network Manager** to show a graphical representation of data and processes performed in the project. All data, modules, and processing paths for the current project are visible in the **Network Manager**.



*Most modules contain a name, visibility check box, input connection pad, output connection pad, indicator LED light, and a connection line.*



Modules are connected to perform a desired task. A module is a data set or process to be applied to a data set. Modules accept data on their input connection pads, modify the data, and pass it along through the output connection pads. The final output from the pipeline is usually a graphical representation of data, such as a *BoundingBox* or *FaceRender*. The module is displayed as a small rectangle. The rectangle can be selected and dragged with the mouse.

Visibility Check Box		The visibility check box indicates whether a module's output is visible in the <b>Viewer</b> window. Check the box to display a module and all "downstream" (connected) modules. Uncheck the box to hide a module and all downstream modules. A gray check mark indicates that a module is disabled because of a hidden upstream module.
Module Name	(Name)	Each module is named with the loaded data file name or by the function performed by the module. You can change the name with the <b>Edit   Rename</b> command. Alternatively, right-click the module and select <i>Rename</i> .
Indicator LED		The indicator LED is a small round "light" showing module status. <ul style="list-style-type: none"> <li>• Green: the module is up to date</li> <li>• Yellow: the module has been modified and needs to be updated</li> <li>• Red: the module is in an error state</li> </ul>
Connection Pad		An input connection pad is located on the left side of the module. An output connection pad is located on the right side of the module. The presence of connection pads indicates that a module may be connected to other modules. Only modules with the appropriate type of data may be connected.
Connector Lines		Connector lines are drawn between connected modules. Lines or pipes may be displayed. See the <b>Tools   Options</b> dialog to change the display of connector lines.





There are several ways to connect or disconnect two modules. After you have selected a module:


- Right-click on a module in the **Network Manager** and select *Connect* from the context menu
- Click on the connection pad of a module in the **Network Manager**
- Click the **Network | Connect** command

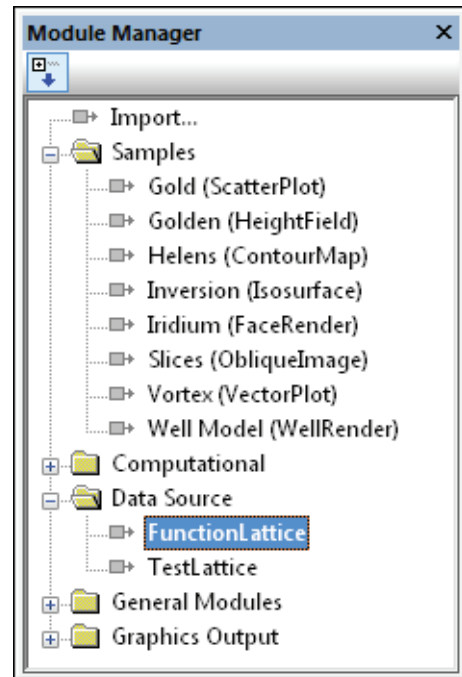
The procedure is the same whether you are connecting or disconnecting two modules.

Once a *Connect* command is initiated, the cursor snaps to the **Network Manager** and **Voxler** enters graphical connect mode. Move the mouse until the blue connection line touches the compatible module you want to connect to or disconnect from and the connection line turns yellow. Click the mouse on the module to make or break the connection. The blue connection line turns yellow if the modules are compatible.

## Module Manager

The **Module Manager** displays a list of available modules and other commands. Several simple sample files are displayed at the top of the **Module Manager** window. Double-click any file in the *Samples* folder to load it into **Voxler**. The  and  buttons indicate the folder can be expanded or collapsed to show or hide additional information. To expand a folder, click on the  control, select the item and press the plus key (+) on the numeric keypad, or press the right ARROW key on your keyboard. To collapse a folder, click on the  control, select the item and press the minus key (-) on the numeric keypad, or press the left ARROW key. You can also double-click on a folder name to expand or collapse the folder.

The toolbar at the top of the **Module Manager** contains a *Show All Modules*  button. If the button is depressed, all modules are listed. If *Show All Modules* is not depressed, only those modules that are compatible with the currently selected module are displayed.








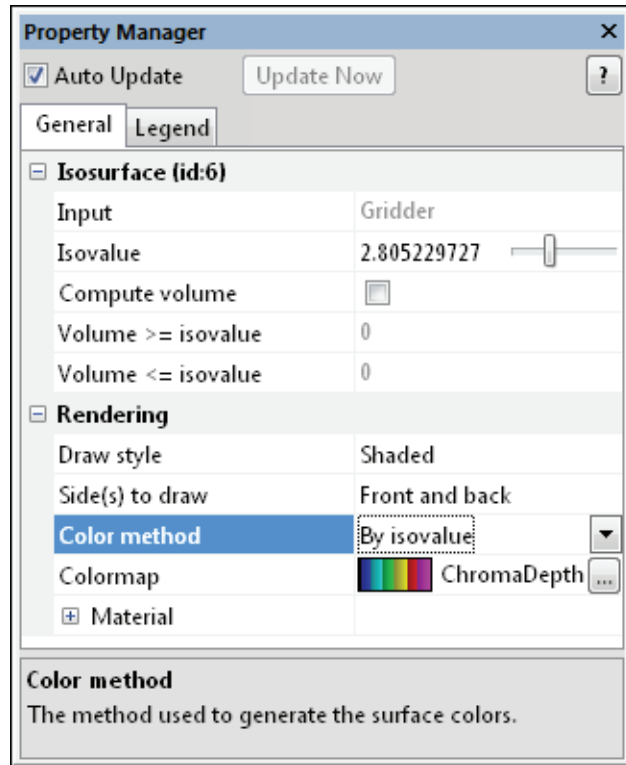
*Double-click on a module to add the module to the **Network Manager**.*

You can add modules to the **Network Manager** by double-clicking on the module in the *Graphics Output*, *Computational*, *Data Source*, or *General Modules* folder in the **Module Manager**. The item is added to the **Network Manager**. If a module is selected in the **Network Manager** and the module in the **Module Manager** is compatible, the two are connected. Otherwise, the module is loaded in the **Network Manager** without connections to any existing modules. You can also drag and drop a module from the **Module Manager** to the **Network Manager**.





## Property Manager

The **Property Manager** allows you to edit the properties of the currently selected module. The **Property Manager** contains a list of all properties for a selected object. The **Property Manager** can be left open so the properties of selected objects are always visible. Information about the object properties is located in the online help.

Features with multiple options appear with a  or  button to the left of the name. To expand a section, click on the  button. To collapse a section, click on the  button. For example, click on a *Isosurface* module to select it. In the **Property Manager**, click on the **General** tab. Click the  next to *Rendering* and you see several options, *Draw style*, *Side(s) to draw*, *Color method*, *Colormap*, and *Material*.



The **Property Manager** is used to change properties of the selected module.

To change a property, click on the property's value next to the property name. Select a new property from the list, scroll to a new number using the  buttons, select a new value by scrolling using the , open a dialog with the  button, or type a new value and press ENTER on your keyboard. How a property is changed depends on the property type. For example, an *Isosurface* has a *Isovalue* option that is changed by typing a value or dragging the slider. The *Colormap* option can be changed by clicking the existing color and selecting a new color from the list or by clicking the  button and selecting new options in the dialog.

Occasionally, some properties are dependent on other selections. For example, with the *Isosurface* module, the *Colormap* option is not available unless the *Color method* is set to *By isovalue*.

Module properties automatically update after changes are made in the **Property**

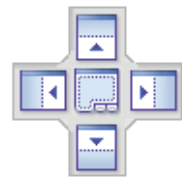
**Manager.** If you want to disable the automatic update of properties, uncheck the *Auto Update* box at the top of the **Property Manager**. This allows multiple changes to be made without updating the **Viewer** window after each change. This can be convenient with large data sets because the redraw time is reduced. After making all changes, click the *Update Now* button to update all module properties in the **Viewer** window.

## Changing the Window Layout

The windows, toolbars, managers, and menu bar display in a docked view by default; however, they can also be displayed as floating windows. The visibility, size, and position of each item may also be changed. Refer to the *Changing the Window Layout* topic in the online help for more information on layout options.

## Docking Managers

**Voxler** has a docking mechanism feature that allows for easy docking of managers. Left-click the title bar of a manager and drag it to a new location while holding down the left mouse button. The docking mechanism displays arrow indicators as you move the manager around the screen. When the cursor touches one of the docking indicators in the docking mechanism, a blue rectangle shows the window docking position. Release the left mouse button to allow the manager to be docked in the specified location.



*The docking mechanism has docking indicators.*

## Viewer Window

The **Viewer** window displays the three-dimensional graphical output produced by the modules in the **Network Manager**. All visible items in the **Network Manager** are displayed in the **Viewer** window.

The **Viewer** window behavior is controlled by several settings from the toolbars or the **View** menu. A virtual camera is used to view, rotate, zoom, and pan the view. The camera position and target are controlled with the **View | Camera Properties** command. The default position and zoom is called the *Home* position. By default, *Home* includes all of the visible graphics and resets when a large change in the extents occurs. To set the home position, click the **View | Set Home** command. To return to the home position, click the **View | Home** command.

A world axis triad, located in the lower right corner, displays the global coordinate system orientation.

Choose the method that overlapping objects are drawn in the **Viewer** window with the **View | Still Draw Style** options. These commands are useful if a particular display

mode takes too long to render and you want to temporarily display graphics in a faster drawing format such as *Wireframe*, *Low Resolution*, or *Bounding Box*. By choosing a faster drawing format, the scene will redraw quicker, but some elements may appear incorrect. A similar command, **View | Animating Draw Style**, applies to the graphics when the camera position is being changed, such as when the graphic is spinning or while changing the zoom level.

There are several transparency modes available with the **View | Transparency Type** command. Different modes work better for various types of geometry. Some modes provide faster rendering while others give better quality. Occasionally, you will need to experiment with transparency modes to find the best display. If you find that your transparent graphics are partially opaque at certain orientations, try selecting **View | Transparency Type | Sorted object, sorted triangle blend** or **View | Transparency Type | Sorted Object Blend**. These methods provide a more accurate transparency mode, but are also slower and more memory-intensive than others. If you find that you are running out of memory or rendering is taking a very long time, choose the **View | Transparency Type | Blend** command. This method usually provides a reasonable tradeoff between accuracy and speed. See the *Transparency Type* section in the online help for advantages and disadvantages of each transparency type.

## Modules

A module is a data set or process to be applied to a data set. Modules are the building blocks from which the final output is constructed. Modules accept data on their input connection pads, modify the data, and pass it along through the output connection pads. There are four types of modules: computational, data source, general, and graphics output. Detailed information about each module is located in the online help. Basic information about each module is listed below.

### Computational Modules

Computational modules alter the data by changing their type, filtering, creating a gradient, gridding, performing mathematical transformations, merging, resampling, slicing, creating a subset, or transforming coordinates.

#### ChangeType

The *ChangeType* module changes the lattice or point set data type from one primitive type (e.g. integer, float, etc.) to another. Smaller types save memory at the expense of reduced numeric precision. All components of the input data set are converted. The *ChangeType* module changes the data components type only, not the coordinates. Use the *Transform* module to change the coordinates.

### DuplicateFilter

The *DuplicateFilter* module removes duplicate data points in a point set. Duplicate data are two or more data points having nearly identical X, Y, and Z coordinates. The *DuplicateFilter* properties control the definition of a duplicate point. Several options are available for determining which point, if any, to keep when points are considered duplicates.

### ExclusionFilter

The *ExclusionFilter* module excludes data points in a point set according to a user-specified Boolean function. Some functions available are IF, AND, OR, NOT, and several comparison operations ( $=$ ,  $<$ ,  $>$ , etc). See the complete list of functions and operators in the online help on the *Mathematical Functions* page.

### ExtractPoints

The *ExtractPoints* module converts points on well paths into points to use for gridding. It will also convert a lattice to a point data set. The number of output components, based on the number of input components or log items in the original data, can be set.

### Filter

The *Filter* module applies a digital filter to a uniform lattice. The lattice may be two-dimensional (images) or three-dimensional (volumes). Each filter reads the input lattice, performs a particular filtering operation on the data values in the lattice nodes, and sends the results to the output lattice. The input and output lattices are always the same size and type. *Filter* module computations include data statistics such as local minimum, maximum, median, average, standard deviation; and image modification such as brightness and contrast.

### Gradient

The *Gradient* module computes a gradient field from a single component of a two- or three-dimensional lattice. A gradient is a three-dimensional vector pointing in the direction of greatest slope. The output lattice contains three-component data at each lattice node. A centered difference algorithm is used to calculate the gradient. The output lattice geometry is identical to the input lattice geometry.

### Gridder

The *Gridder* module interpolates scattered point data onto a uniform lattice. The output lattice range, resolution, interpolation method, and associated parameters are set. Since gridding can take quite a while to execute, it is necessary to click the *Begin Gridding* button in the **Property Manager** to start the process.

### Math

The *Math* module creates a new output lattice by applying a numeric expression to one or more input lattices. The output lattice is calculated one node at a time by applying the numeric expression to the input lattice nodes.

### Merge

The *Merge* module combines two or more input lattices into a single uniform output lattice. You can specify the output lattice range and resolution.

### Resample

The *Resample* module allows the resolution of a lattice to be changed. This is performed by computing new data values at each output lattice node by interpolating the data values from the input lattice. The *Resample* module does not perform extrapolation.

### Slice

The *Slice* module creates a two-dimensional slice through a three-dimensional input lattice. The plane orientation may be preset to one of the local axis planes or in an arbitrary direction.

### Subset

The *Subset* module extracts a particular region of interest for further analysis. You can specify the geometric range, sampling frequency, and data components of the subset.

### Transform

The *Transform* module transforms the X, Y, and Z coordinates of an input point set or lattice using a standard 4x4 transformation matrix. The order of transformations is: scaling, rotation, and translation. Rotation and scaling are performed around the object's *Origin*. The *Origin* can be the lower left corner, the upper right corner, the center, or a custom position.

## Data Source Modules

Data source modules serve as the source of raw data. The data may be imported from a file or created from mathematical functions. **Voxler** supports several different file types. See the *File Format Chart* in the online help (**Help | Contents**) for a detailed list of supported file formats. Data are passed from one module to another to accomplish tasks such as gridding, slicing, or displaying graphics.

The four main types of imported data include: point sets, lattices, geometry, and well data.

### Point Sets

Point sets contain one or more three-dimensional point locations. Each location has an X, Y, and Z coordinate along with optional data components. Occasionally, this is called "XYZC data" where XYZ represent the three-dimensional position and C represents one or more data component values at that position. Any ASCII data file, Excel files, and many database files can be imported with the **File | Import** command to create a point set. Data are normally in columns in points sets, with each column containing a separate variable.

### Lattices

A lattice consists of a one-, two-, or three-dimensional data array. An array is a regular, structured matrix of points. A one-dimensional lattice is a line of data. Examples of two-dimensional lattices include bitmaps or **Surfer** grid files. A three-dimensional lattice defines a three-dimensional volume. Each node (or point) in the lattice can contain one or more components or data values. Lattices are further categorized by the node geometry: uniform, rectilinear, and curvilinear. Detailed information about the lattice geometry and components are found in the online help. Some types of lattices that can be imported into **Voxler** using the **File | Import** command include **Surfer** grid files, DEM files, images, P3D files, LAT files, and many other formats.

### Geometry

Geometry consists of triangles, texture maps, line segments, and other objects. Geometry is collected at the end of the pipeline and displayed in the **Viewer** window. Geometry is usually represented internally using integer or single precision floating point. Some geometry data types that **Voxler** can import using the **File | Import** command include DXF, SHP, E00, BLN, and many other formats.

### WellData

The *WellData* module is a container for well data imported into the project. Each well is imported into the *WellData* module using the **File | Import** command. A *WellData* module can have any number of wells, with each well containing information specific to that well. Each well can contain X, Y, Z, MD (Measured Depth), Azimuth, Inclination, and any number of Logs. The log is the data variable associated with the downhole location, and is usually the variable to be modeled. Data for wells can be imported from ASCII text files, Excel files, database files, and LAS files.



### FunctionLattice

The *FunctionLattice* module creates a new uniform lattice from a user-defined function. You can specify the output lattice range, resolution, number of components, and mathematical equations for defining each component.

### TestLattice

The *TestLattice* module generates a variety of lattices for testing and experimenting with various modules. You can specify the output lattice range, resolution, and data type.

## General Modules

General modules display module information and provide custom lighting in the **Viewer** window.

### Info

The *Info* module displays information about the connected module, such as data limits, number of components, and component type.

### Light

The *Light* module creates a new directional, point, or spot light and adds it to the scene. Lights are cumulative. Every time a new light is added, it makes the scene a little brighter. You can add approximately eight lights to the scene. To view a scene with only light modules, uncheck the **View | Headlight** command to turn off the global light.

### Viewer Window

The *Viewer Window* module contains various properties that affect the entire scene, such as background color. The *Viewer Window* module is automatically created. The *Viewer Window* module is displayed in the **Network Manager**; it is not listed in the **Module Manager** since it always exists and cannot be deleted. The *Viewer Window* module only controls the options for the current **Viewer** window. To change the default **Viewer** window settings, click the **Tools | Options** command and click on the **Colors** tab. The *New viewer window background* controls the color of future **Viewer** windows.

## Graphics Output

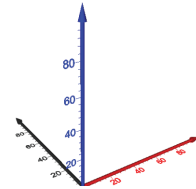
Graphics output modules create graphics in the **Viewer** window. Typically, these modules require data input.

### Annotation

The *Annotation* module creates a text string that is always parallel to the screen. By default, the current date and time is used as the text string. You can enter your own text in the **Property Manager**. Use the *Text* module to anchor the text to the scene.

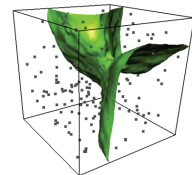
### Axes

The *Axes* module creates a set of axes. The axes are attached to an input point set or lattice. The axis labels are planar, although the plane orientation can be changed in the **Property Manager**. A grid can also be displayed between any two axes. By default, the X axis is red, the Y axis is green, and the Z axis is blue. These colors can be changed in the **Property Manager** for the existing axes or in the **Tools | Options** dialog for default conditions for future axes.



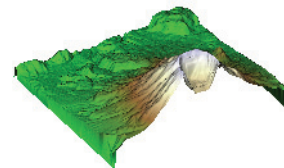
### BoundingBox

The *BoundingBox* module draws a bounding box around the input module extents. Additionally, labels can be displayed for the minimum and maximum corners. The labels are displayed as screen-aligned text centered on the minimum and maximum corners.



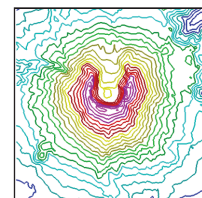
### ClipPlane

The *ClipPlane* module clips input geometry according to a user-defined clipping plane. All geometry on one side of the plane is drawn. The geometry on the other side of the plane is removed (clipped). The side that is clipped and the location of clipping can be altered in the **Property Manager**. Multiple modules can be attached to the same clipping plane.



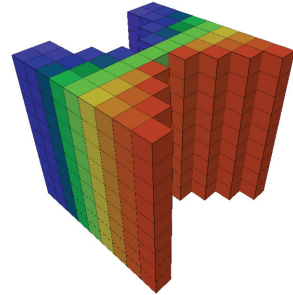
### Contours

The *Contours* module generates contour lines for a two-dimensional data set or for slices of a three-dimensional data set. Contour lines represent the boundary between data less than a given level (threshold) and data greater than the level. For three-dimensional data sets, the *Contours* module creates a planar slice through the lattice and contours the two-dimensional slice. Contour lines are colored by mapping data values to colors through a *Colormap*.



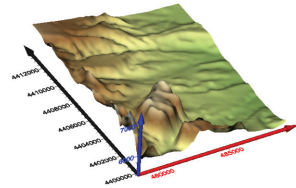
## FaceRender

The *FaceRender* module displays uninterpolated cubes of an input lattice. A *FaceRender* cube represents one unit in each of the X, Y, and Z directions. Component values are represented by different colors in the *FaceRender*. To determine the component value and color for each cube, **Voxler** calculates the average component value by summing the values at each of the eight corner points and dividing by eight. If one or more of the corner points has a null (blank) value, that cube is not displayed. Additionally, the cube is not displayed if the color map value for the average data value for that cube is partially or fully transparent.



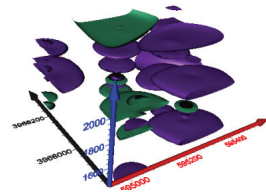
## HeightField

The *HeightField* module displays a lattice slice in three-dimensions. The slice is scaled in the perpendicular direction by a user-specified data component and scale factor. The surface is colored by mapping the data values through a *Colormap*.



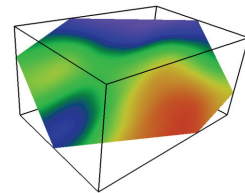
## Isosurface

The *Isosurface* module creates an isosurface through an input lattice. An isosurface is a surface of constant value in a three-dimensional volume. The isosurface value is set in the *Isovalue* property in the **Property Manager**. The isosurface separates regions less than the selected *Isovalue* from regions greater than the selected *Isovalue*. All points on the isosurface have the same value (the isovalue).



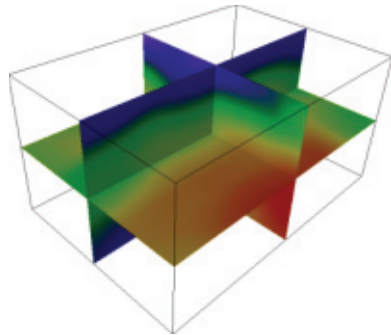
## ObliqueImage

The *ObliqueImage* module displays a color image on a two-dimensional cutting plane through a lattice. In medical terminology, this is known as a multi-planar reconstruction (MPR). The slice is represented using colors mapped through a *Colormap* for scalar data, or as direct RGBA colors for lattices containing color data.



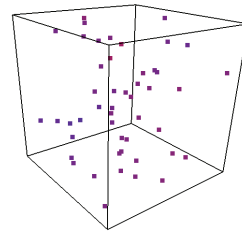
### OrthoImage

The *OrthoImage* module displays an orthogonal slice through a lattice parallel to one of the three axial planes (XY, XZ, or YZ). Orthogonal indicates elements are perpendicular or at right angles. The slice is represented by mapping data to a *Colormap* for scalar data, or as direct RGBA colors if the lattice already contains color data. The scalar to color mapping may be specified with a linear gray mapping function with contrast enhancement or with a *Colormap*.



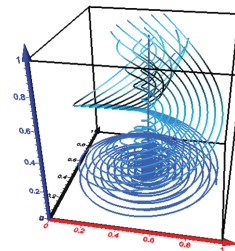
### ScatterPlot

The *ScatterPlot* module displays a set of symbols at each point of a point set or each node of a lattice. The symbols are screen-aligned and do not scale or “tilt” as the camera is changed. The symbol positions, however, are maintained in three dimensions. Labels can be added to points in a *ScatterPlot*.



### StreamLines

The *StreamLines* module computes streamlines through a velocity field. Streamlines are lines within a volume of space that indicate flow direction and magnitude. The technique injects massless particles at specified seed points and traces their paths through the field. The particles stop when the new velocity is zero, the maximum stream length is exceeded, or when the stream intersects the bounds of the field.

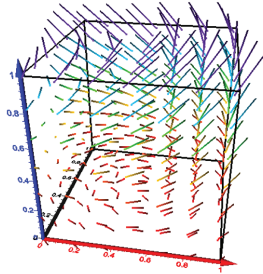


### Text

The *Text* module creates a two-dimensional text string aligned with the camera plane. The text has a three-dimensional anchor point that is transformed with the scene. The text is not scaled according to the distance from the camera, nor is it influenced by rotation or scaling. It is, however, still obscured by graphics lying in front of it. The text is positioned according to the current transformation: the X origin is the first pixel of the leftmost character of text and the Y origin is the baseline of the first line of text with the baseline being the imaginary line on which all upper case characters are standing. Use the *Annotation* module to create text that is not anchored to the scene.

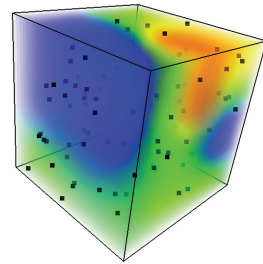
### VectorPlot

The *VectorPlot* module displays vectors on a three-dimensional lattice or point set. It is often useful to combine the *VectorPlot* module with another module, such as *StreamLines*.



### VolRender

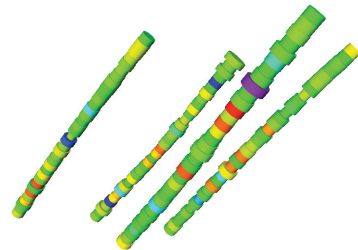
Most visualization techniques convert volume data to surfaces. This module uses an alternative technique called *direct volume rendering* to render voxels directly. A voxel is short for **volume pixel**, the smallest distinguishable box-shaped part of a three-dimensional image.



Volume rendering is a three-dimensional display of data that simulates the transmission and absorption of light through the points in the volume. Light rays are cast through the volume, where particles within the volume simultaneously emit and absorb light. The color of an individual pixel on the screen is computed by compositing the contributions from each particle that intersects the ray. This allows visualization of inhomogeneity inside objects with appropriate opacity adjustment.

### WellRender

The *WellRender* module displays well traces from well data. Wells are displayed as tubes. Thickness and color can vary down the well. In addition, direction data and interval data can be displayed on the well.



## Automation

**Voxler** operations can be controlled through automation scripts. You can do almost everything with a script that you can do manually with the mouse or from the keyboard. Scripts are used to automate repetitive tasks, consolidate a complicated sequence of steps, or act as a "front end" to help novice users access **Voxler's** capabilities without having to become familiar with **Voxler**. Since **Voxler** exposes its services through automation, you can use any programming tool that accesses automation objects. Such tools include Visual BASIC, Windows Scripting Host, and many of the Microsoft Office applications, among others.

### Scripter

Golden Software's **Scripter** is a program for developing and running scripts. A script is a text file containing a series of instructions carried out when the script is run. Instructions are written in a Visual BASIC-like programming language. **Scripter** offers many features to help you write, edit, and debug scripts. Its features include language syntax coloring, a list of the procedures defined in the script, an object browser for examining procedures available in external objects, a visual dialog editor, break points, single-step execution (including options to step over and to step out of procedures), a watch window for displaying the values of script variables, and more.

To start the **Scripter** program, select it from the Windows Start menu. **Scripter** is installed in the same program group as **Voxler 3**. To open **Scripter**, click the Windows Start button and locate **Golden Software Voxler 3 | Scripter**. If **Scripter** is not present, the installation of **Scripter** may have been skipped when **Voxler** was installed. See the Readme.rtf file for information about the installation process.

Once **Scripter** is open, open a script file with the **File | Open** command. In the **Open** dialog, select any .BAS file. Execute the script by clicking the **Script | Run** command.

### Tutorial

The tutorial is designed to introduce you to some of **Voxler's** basic features. After you have completed the tutorial, you should be able to begin to use **Voxler** with your own data. We strongly encourage completion of the tutorial before proceeding with **Voxler**. The lessons should be completed in order; however, they do not need to be completed in one session. The tutorial should take approximately one hour to complete.

Now that you have an overview of **Voxler**, let's create some graphics. We will import data, link the data to modules, change properties, and save information.

### Tutorial Lesson Overview

The following is an overview of lessons included in the tutorial.

- *Lesson 1 - Loading Data* shows how to import data.
- *Lesson 2 - Creating Graphics Output Modules* shows how to create a *ScatterPlot* and a *BoundingBox*.
- *Lesson 3 - Changing Properties* shows how to change module properties and rotate the view.
- *Lesson 4 - Using Computational Modules* shows how to create a uniform lattice from the data, display the lattice as an *Isosurface*, and filter the data.
- *Lesson 5 - Connecting Multiple Modules* shows how to connect multiple output

modules to a single input module.

- *Lesson 6 - Saving Information* shows how to save graphics, data, and a .VOXB file.
- *Lesson 7 - Importing and Displaying Wells* shows how to import well data, display the well data as a *WellRender* module, and modify the *WellRender* module properties.

## Using the Tutorial with the Demo Version

Some **Voxler** features are disabled in the demo version, which means that some steps in the tutorial cannot be completed by users running the demo version. This is noted in the tutorial and users are prompted to proceed to the next step.

## Starting Voxler

To begin a **Voxler** session:

1. Navigate to the installation folder, which is C:\Program Files\Golden Software\Voxler 3 by default.
2. Double-click on the Voxler.exe application file.
3. A new empty **Viewer** window opens in **Voxler**. This is the work area for producing graphics. If this is the first time that you have opened **Voxler**, you will be prompted for your serial number. Your serial number is located on the CD cover, or in the email download instructions, depending on how you purchased **Voxler**.

If **Voxler** is already open, click the **File | New** command or the  button to open a new empty project before continuing with the tutorial.

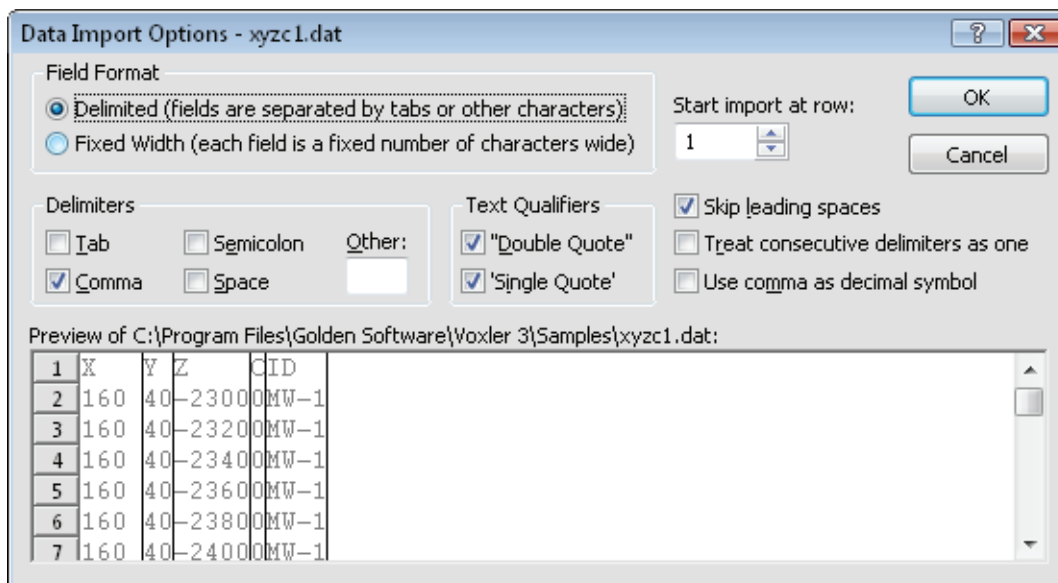
## Lesson 1 - Loading Data

Many data types are used in **Voxler**. For a detailed list of supported formats, refer to the online help. The four main types of data are point sets, lattices, geometry, and well data. Point sets are data files that contains columns of data values. Point sets contain X, Y, and Z values in addition to optional components and label columns. To load data into **Voxler**,

- Click the **File | Import** command, or
- Right-click in the **Network Manager** and select *Import*, or
- Double-click the *Import* option in the **Module Manager**.

To load a data file:

1. Click the **File | Import** command. The **Import** dialog opens.
2. In the **Import** dialog, change the *Look in* field to the *Samples* directory, located inside the main **Voxler** folder. By default, this folder is located at C:\Program Files\Golden Software\Voxler 3\Samples. Click on the *XYZC1.dat* file and click *Open*.
3. In the **Data Import Options** dialog, check the box next to *Comma* to parse the data into separate columns as shown by the vertical lines in the *Preview* section. Click *OK*.



The **Data Import Options** dialog allows you to parse data.

4. The **Select Data Columns** dialog offers options for changing the columns for the *X*, *Y*, *Z*, *Component*, and *Labels*. The *X*, *Y*, and *Z* coordinates locate the points in space. The component data are the variables associated with each point that are to be modeled. The labels are the text or numbers associated with any of the points that explain the point. This dialog also allows you to specify the import starting row and a value to replace non-numerical and empty cells. In addition, you can choose to import the data as either points or wells in this dialog.
5. Check the box next to *Use header row* and ensure that the value is set to 1. The *Start import at row* is automatically changed to 2 so that the text in row 1 is not imported.
6. Select the *Import as Points* option so that points are created from the data.



7. Set the *X*, *Y*, and *Z* columns to Column A: *X*, Column B: *Y*, and Column C: *Z*, respectively.
8. Set the *Single Component* to Column D: *C*.
9. Check the box next to *Import labels* and set the column to Column E: *ID*.

Select Data Columns - xyzc1.dat

	A	B	C	D	E
1	X	Y	Z	C	ID
2	160	40	-2300	0	MW-1
3	160	40	-2320	0	MW-1
4	160	40	-2340	0	MW-1
5	160	40	-2360	0	MW-1

Options

☒ Import as Points ☒ Use header row: 1 Start import at row: 2 End import at row: 370

☐ Import as Wells

Points

X: Column A: X ☐ No data

Y: Column B: Y ☒ Single component: Column D: C

Z: Column C: Z ☐ Multiple components:

☒ Import labels

Column E: ID

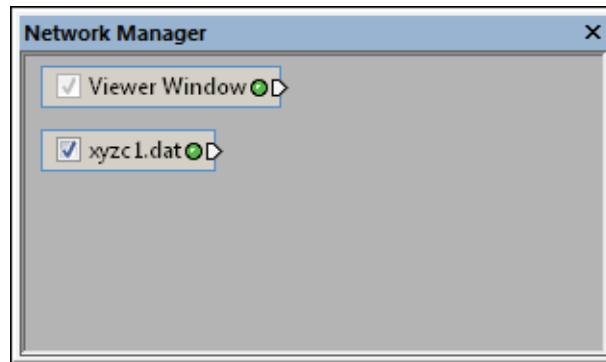
☐ Replace non-numerical and empty cells with: 0

OK Cancel

Select data for *X*, *Y*, *Z*, component, and label data in the **Select Data Columns** dialog.

10. Click *OK* to finish importing the data.

Once the **Select Data Columns** dialog closes, the **Network Manager** displays a module with the data file name. This module represents the raw data. To display graphics, you will add modules to the data set.



*After a data file is loaded, it appears as a module in the **Network Manager**.*

## Lesson 2 - Creating Graphics Output Modules

Once the data are loaded, a data module appears in the **Network Manager**. The **Viewer** window is blank because no graphical modules have been added to the data module. In this lesson, we add a graphics output module.



### Creating a ScatterPlot

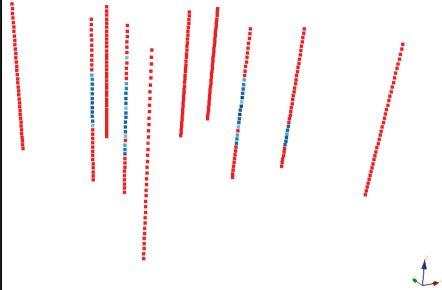
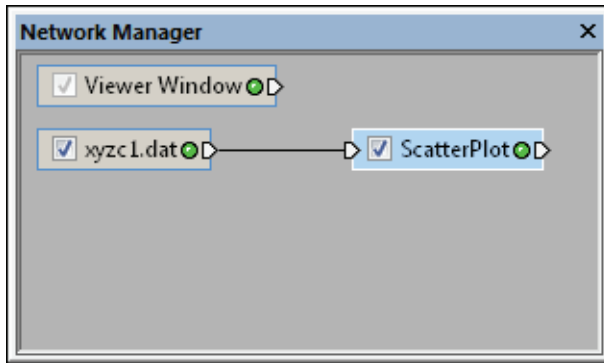
A *ScatterPlot* is a model of point data within a volume of space, optionally with colors representing data values. To create a *ScatterPlot*:

1. Click on the *xyzc1.dat* module in the **Network Manager** to select it. The selected module is highlighted.
2. In the **Module Manager**, double-click on the *ScatterPlot* module name in the *Graphics Output* section of the list. Alternatively, right-click on *xyzc1.dat* module in the **Network Manager** and select *Graphics Output | ScatterPlot* from the context menu.

In the **Module Manager**, only the modules that can be used with the data type are listed if the *Show All Modules* button is not selected. The button is selected if it looks



like  and not selected when it looks like . In this example, we used point data, so the main graphics outputs are *Axes*, *BoundingBox*, *ScatterPlot*, and *VectorPlot*. Other graphics, such as *Isosurfaces*, need lattices as inputs so they are not listed when the button is not selected.



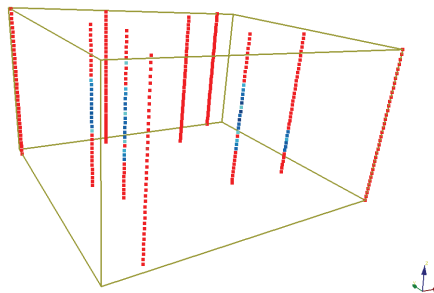
A ScatterPlot module appears connected to the xyzc1.dat module in the **Network Manager** and the scatter plot is displayed in the **Viewer** window.

### Creating a Bounding Box

Next we will add a bounding box around the input data. A bounding box is a three-dimensional box that extends around the input data extents.

To draw a bounding box:

1. In the **Network Manager**, click the xyzc1.dat module.
2. In the **Module Manager**, double-click *BoundingBox* under *Graphics Output*. Alternatively, right-click the xyzc1.dat module in the **Network Manager** and select *Graphics Output | BoundingBox* from the context menu.




The ScatterPlot should appear similar to this, after the BoundingBox is added.

## Lesson 3 - Changing Properties

Once modules have been created, their properties can be changed in the **Property Manager**. Click on a module in the **Network Manager** to select it and display the module's properties in the **Property Manager**. Note that some items in a module's property list are informational only and cannot be changed. These items appear gray in the list.

### Changing ScatterPlot Symbol Colors

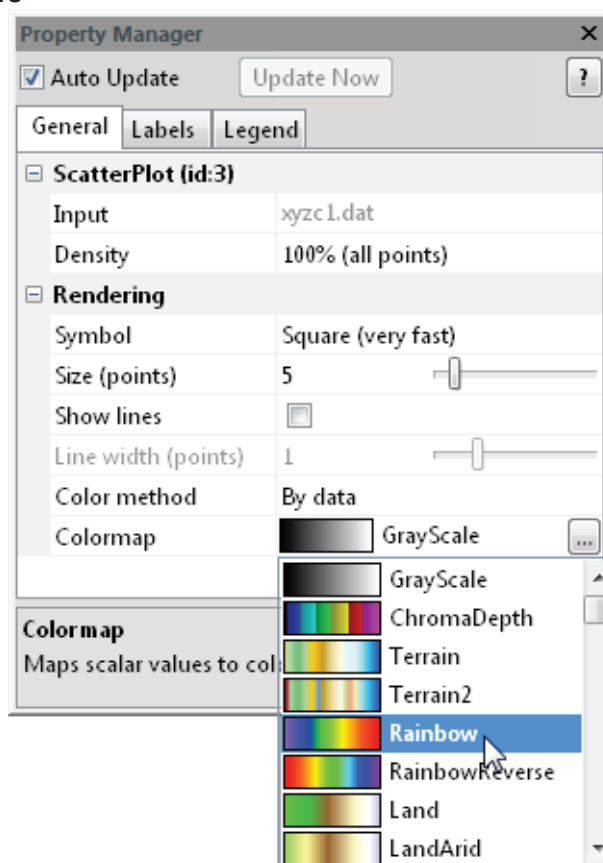
One property that can be changed for a *ScatterPlot* is the symbol color. To change the symbol colors:

1. Click on the *ScatterPlot* module in the **Network Manager**. The *ScatterPlot* module properties open in the **Property Manager**.
2. In the **Property Manager**, click on the **General** tab.
3. Scroll down to the *Colormap* option. Click the sample color spectrum (*GrayScale*) to the left of the  button to open the option list. Click *Rainbow* to change the colors. The colors are mapped to the data variable C, as selected when the data were loaded in lesson 1.

The *ScatterPlot* is updated to show the selected colors.


### Displaying ScatterPlot Labels

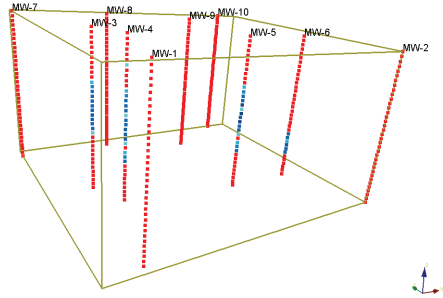
Labels can be displayed on the *ScatterPlot* from the X, Y, Z, component, or label column. Let's add the label from the label column so that the data can be identified.



Select the desired color.

To add labels:

1. Click on the *ScatterPlot* module in the **Network Manager**.
2. In the **Property Manager**, click on the **Labels** tab.
3. Check the box next to *Show labels* to turn on the display of labels for the module.
4. Change the *Label field* to the desired column. In this case, let's select *Labels*. The labels are added next to the points that contain information in the label column.
5. Click the  $\oplus$  next to *Font* to open the font properties for the labels.
6. To increase the size of the labels, click and drag the  next to *Size (points)* until the desired size is shown in the **Viewer** window. The labels automatically change size as the slider is moved.



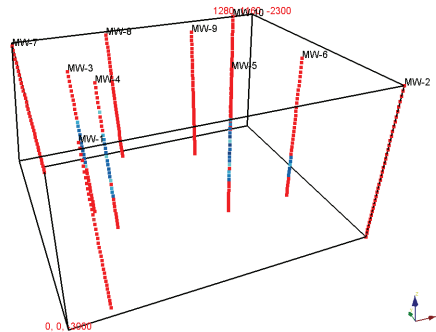
*Labels are displayed next to all points that contain text in the label column.*

## Changing the BoundingBox Properties

Changing the bounding box properties is similar to changing the *ScatterPlot* properties.

To change the bounding box line thickness and color:

1. Click on *BoundingBox* in the **Network Manager**.
2. In the **Property Manager**, click on the **General** tab.
3. Click *Yellow* next to the *Color* property. The color palette opens.
4. Click on the color black to change the bounding box color to black.
5. Highlight the value 1 next to *Line width (points)* and type 1.5. Press ENTER on the keyboard and the line thickness updates.



*Alter any properties of the module in the **Property Manager**.*

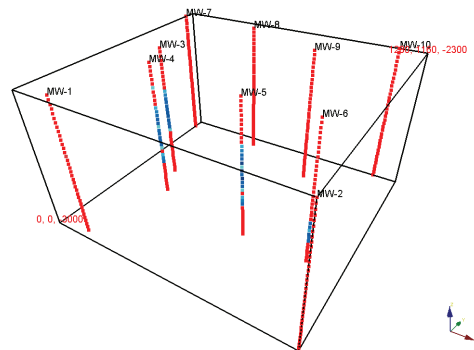
To add labels:

1. Click on the **Labels** tab.
2. Check the box next to *Show labels* to display labels at the maximum and minimum values for the bounding box.
3. To change the label color, click the black color box next to *Color* and select another color, such as red.

### Rotating the View

The **Viewer** window contents can be rotated and animated (spinning). Currently, we are viewing the *ScatterPlot* from the lower left side. We can rotate the *ScatterPlot* to see the symbols and labels more clearly.

The view is rotated by clicking on the **Viewer** window, holding down the left mouse button, and dragging the mouse. If you release the mouse button while the mouse is still moving, the **Viewer** window will enter spin mode. Experiment with different rotations. If you spin the graphic, you can stop the spin by clicking anywhere in the **Viewer** window.



*The same ScatterPlot and BoundingBox are shown after rotation.*

A world axis triad is displayed in the lower right corner of the **Viewer** window. The axis triad is informational only. The triad is a depiction of the X, Y, and Z directions that shows the **Viewer** window camera orientation. This is useful when rotating graphics to see how the graphics have been rotated in space.



The rotation of the **Viewer** window can be recorded with the **Actions | Capture Video** command.

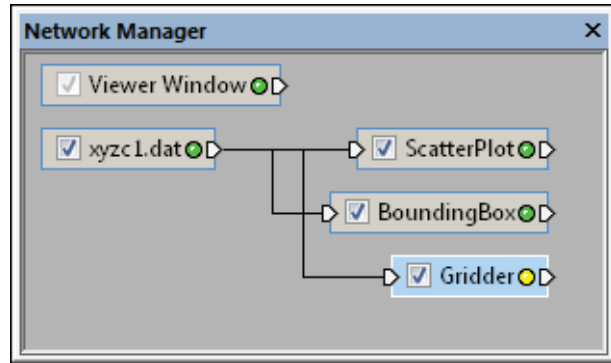
## Lesson 4 - Using Computational Modules

Computational modules use data inputs to grid point sets, filter data, merge data, and perform other tasks. Several visually interesting graphics output modules require lattices, including *Contours*, *HeightFields*, *Isosurfaces*, *ObliqueImages*, *OrthoImages*, *StreamLines*, and *VolRenders*. The xyzc1.dat file contains scattered point data. You can use a *Gridder* module to create a lattice from the scattered point data. After the point set is converted to a lattice, we can display it as an *Isosurface*, *VolRender*, or *Contours* module. Filtering can be applied to see the effect on the network and graphics output modules.

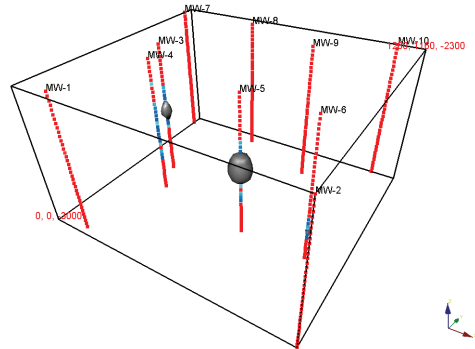
## Gridding Data

To create a lattice from xyzc1.dat:

1. In the **Network Manager**, click the *xyzc1.dat* module.
2. In the **Module Manager**, double-click the *Gridder* module in the *Computational* section. The *Gridder* module is loaded into the network. Alternatively, you can right-click on the *xyzc1.dat* module in the **Network Manager** and select *Computational | Gridder*.
3. In the **Network Manager**, the *Gridder* module displays with a yellow indicator LED , indicating that additional input is required. In this case, we need to initiate gridding in the **Property Manager**. With the *Gridder* module selected in the **Network Manager**, click on the **General** tab in the **Property Manager**.
4. Click the *Begin Gridding* button in the **Property Manager** to begin the gridding process. The *Gridder* module indicator LED changes to  when the gridding is complete.



The indicator LED is a small circle to the right of the module name. The *Gridder* module indicator LED initially appears yellow, indicating you must make a properties change.



An Isosurface with the default properties is displayed in the **Viewer** window.

## Creating an Isosurface




The *Gridder* module interpolated the scattered point data onto a uniform lattice. To display the lattice in the **Viewer** window, the *Gridder* module needs to be connected to a graphics output module. An *Isosurface*, a surface of constant value in three dimensions, is one way a lattice can be displayed.

To create an *Isosurface*:

1. Click the *Gridder* module in the **Network Manager** to select it.
2. Double-click the *Isosurface* module in the **Module Manager**. Alternatively, right-click on the *Gridder* module and select *Graphics Output | Isosurface*. The *Isosurface* module appears connected to the *Gridder* module in the **Network Manager** and an *Isosurface* displays in the **Viewer** window.

## Editing the Isosurface

We can experiment with the isovalue (constant value) to change the *Isosurface* appearance. To change the *Isosurface* properties:

1. Click on the *Isosurface* module in the **Network Manager** to select it.
2. In the **Property Manager**, click on the **General** tab.
3. Change the value next to *Isovalue* by double-clicking on the default value, typing 20, and pressing ENTER on your keyboard. Alternatively, the  next to the number can be moved to change the *Isovalue*. A new *Isosurface* is calculated and immediately displayed in the **Viewer** window.
4. In the **Property Manager**, click the *GrayScale* color spectrum next to *Colormap*. Select *Rainbow* and the colors automatically update.
5. Open the *Material* section by clicking the  next to *Material*.
6. Change the *Opacity* value by highlighting the existing value, typing 0.5, and pressing ENTER on the keyboard or moving the  until the value is 0.5. Changing the *Opacity* to a lower value allows the isosurface to be partially transparent. The lower the value, the more transparent the isosurface.

## A Note About Transparency

An *Opacity* value of 0.0 is fully transparent. An *Opacity* value of 1.0 is fully opaque. Transparency can be very time consuming to get absolutely correct. As such, **Voxler** contains several different algorithms that trade off speed against correctness. See the *Transparency Type* page in the help for suggestions on options to try if the transparency does not look correct for your particular data. To change the transparency, click the **View | Transparency Type** command and select the desired transparency type. Usually the **Sorted object, sorted triangle add** and **Sorted object, sorted triangle blend** methods result in good output but these methods are significantly slower than the other methods. A quick method that produces good results in many cases is **Blend** or **Delayed Blend**.

## Filtering Data

You can add computational modules between the data file module and the *Gridder* module to change the *Isosurface*. There are many data filtering options in **Voxler**. Filtering modifies the data stream, which affects all downstream modules. Typically, the “downstream” modules are automatically changed when “upstream” modules are altered. The *Gridder* module is one exception due to the time required to grid the data.

As an example of filtering data, assume the data contains points that are very close together and we would like to combine these duplicate points into a single representative value.




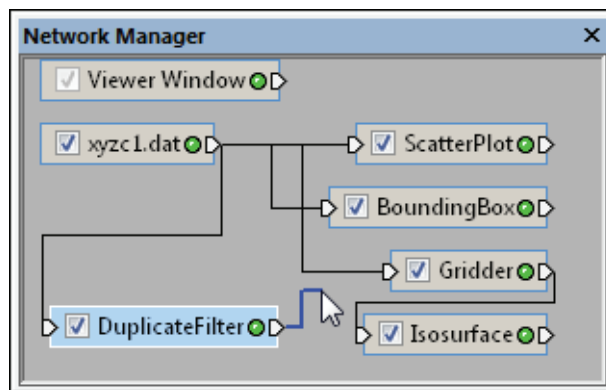
To average these duplicate points:

1. Click the *xyzc1.dat* module in the **Network Manager**.
2. In the **Module Manager Computational** section, double-click the *DuplicateFilter* module to add it to the **Network Manager**. Alternatively, right-click on the *xyzc1.dat* module and select *Computational | DuplicateFilter*.
3. Click on the *DuplicateFilter* module in the **Network Manager** to select it.
4. In the **Property Manager**, change the *Keep* option to *Median Z*.
5. In the **Property Manager**, enter 20 for the *Z Tolerance*.


Since there are no output modules currently connected to the *DuplicateFilter* module, there are no visible changes in the **Viewer** window. We can make changes by connecting the *DuplicateFilter* module to the *Gridder* module.

To connect the *DuplicateFilter* module:


6. First, move the *DuplicateFilter* module to the left side of the **Network Manager** so the connections are easily seen. Click on the *DuplicateFilter* module and drag it to the left side of the **Network Manager**.
7. Click on the output connection pad  on the right side of the *DuplicateFilter* module in the **Network Manager**.
8. In the **Network Manager**, hold the mouse button down and drag the cursor to the left side of the *Gridder* module to connect the two modules. The connection line changes from blue to yellow when the cursor is over a module to which it can be connected. Release the mouse button when the line turns yellow. The connection line color changes to black when the connection is completed.

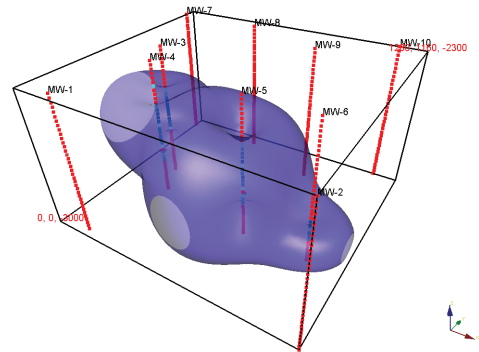


Click on the *DuplicateFilter* output connection pad and drag the blue line to the *Gridder* module to connect the modules.

Since the *Gridder* module accepts only one input, connecting the *DuplicateFilter* module causes the *Gridder* module to automatically disconnect from the *xyzc1.dat* module. In addition, the *Gridder* module indicator LED turns yellow  indicating that additional attention is required. Once the gridding is complete, the *Isosurface* module automatically updates and the new graphical output is sent to the **Viewer** window.

To update the *Gridder* and *Isosurface* modules:

1. In the **Network Manager**, click the *Gridder* module to view its properties in the **Property Manager**.
2. In the **Property Manager**, click on the **General** tab.
3. Click the *Recalculate* button next to *Data dependent parameters*. This recalculates the lattice limits and other parameters to use the new input coming in from the *DuplicateFilter* module.
4. Click the *Begin Gridding* button in the **Property Manager**. The progress gauge displays the gridding progress and the *Gridder* module indicator LED turns green  when the gridding is complete. The *Isosurface* module automatically updates with the new information and the results display in the **Viewer** window.



*After the data are regridded, the Isosurface automatically updates to reflect the changes since it is "downstream" from the Gridder module.*

## Lesson 5 - Connecting Multiple Modules

Modules can have multiple connections. For example, the output from a *Gridder* module can be connected to several graphics output modules to show multiple aspects of the data in one graphic.


### Adding a Contours Module

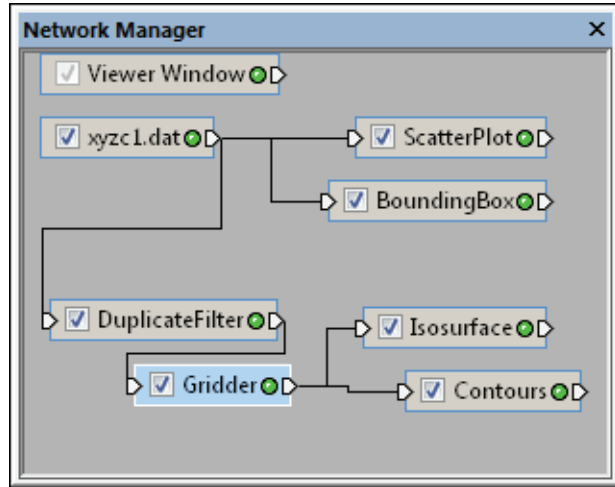
To add another graphics output module to the *Gridder* module:

1. Click the *Gridder* module in the **Network Manager**.
2. In the **Module Manager**, double-click the *Contours* module in the *Graphics Output* section to connect it to the *Gridder* module. Alternatively, right-click on the *Gridder* module and select *Graphics Output | Contours*. The **Network Manager** now contains a connected *Contours* module and the **Viewer** window displays contours with the default settings.
3. To visualize the connections better, click on the *Gridder* module and drag it under the *DuplicateFilter* module.

## Changing the Contours Properties

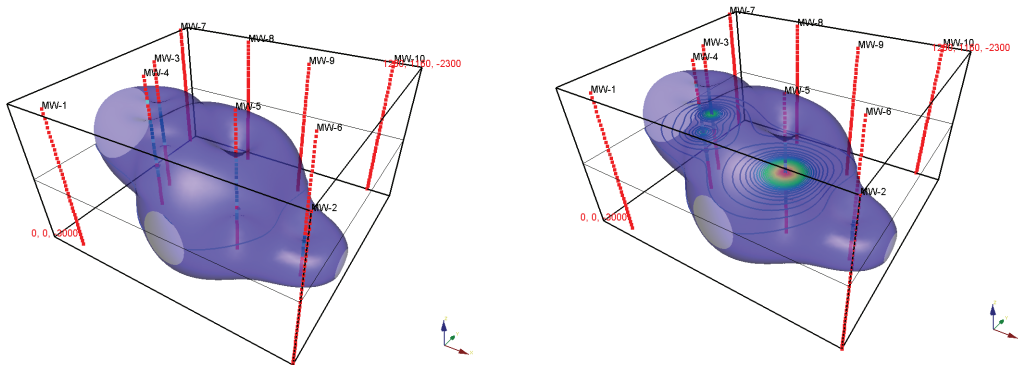
Sometimes, the default settings are not exactly what we want to show in the **Viewer** window. To change the contour properties:

1. Click the *Contours* module in the **Network Manager**.
2. In the **Property Manager**, click on the **General** tab.
3. Click *Automatic* next to *Level method* and choose *Min, max, interval*. This property section allows you to set the minimum and maximum contour values, and the contour interval (number of units between contour lines).
4. In the **Property Manager**, double-click on the default *Level interval* value, type 5, and press ENTER.
5. In the **Property Manager**, locate the property named *Colormap*. Click the sample color spectrum (*GrayScale*) to the left of the  button to open the drop down list. Click *Rainbow* to change the colors of the contour lines.



Click on the Gridder module and drag it to a new location to visualize the connections better.

With each contour property change, the *Contours* update automatically in the **Viewer** window.



The image on the left shows the default Blend transparency type. The image on the right uses the Delayed Blend and is a better choice for this project.

### Changing the Transparency

As discussed in *A Note About Transparency* on page 32, the transparency settings may need adjusting to fit the needs of your project. All of the contours may not be visible in the **Viewer** window depending on the transparency settings.

To change the transparency mode to delayed blend choose the **View | Transparency Type | Delayed Blend** command. Alternatively, right-click in the **Viewer** window and select **Transparency Type | Delayed Blend** from the context menu. If the desired level of transparency is still not shown, experiment with the transparency options to see how it affects the scene.

## Lesson 6 - Saving Information

There are several ways to save **Voxler** information:

- Select **File | Save** to save the data set and all of its associated modules as a **Voxler** network file.
- Select **File | Save Data** to save a selected module's data.
- Select **File | Export** to export graphic files such as bitmaps.
- Select **Actions | Copy Snapshot** to copy the Viewer window view to the clipboard for pasting in other programs.
- Select **Actions | Capture Video** to capture the screen rotation and save to an .AVI movie format.

If you are using the demo version of **Voxler** you will not be able to use the save or export commands.

### Saving a Voxler File

The **File | Save As** command saves the project as a **Voxler** .VOXB file. **Voxler** files contain all of the source data, modules, connections, and graphics.

To save a **Voxler** file:

1. Click the **File | Save As** command. The **Save As** dialog opens.
2. Type *tutorial* into the *File name* field. Note there is only one option in the *Save as type* list, *Voxler Network Files (\*.voxb)*.
3. Click the *Save* button and the dialog closes. The network is saved so that it can be reused in **Voxler**. The network file format includes all data, including the raw source modules, and everything else needed to reload the network in the future.

If you are using the demo version of **Voxler** you will not be able to use the **File | Save** or **File | Save As** commands.

### Saving Data

The **File | Save Data** command saves the output data from the currently selected module. You may also select the module whose output you want to save, right-click the module and select *Save Data*. Data can be saved to many different formats, depending on the type of module selected.

To save the selected module's output data:

1. Click the *DuplicateFilter* module in the **Network Manager**.
2. Click the **File | Save Data** command. Alternatively, right-click on the *DuplicateFilter* module and select *Save Data*. The **Export** dialog appears.
3. Type *tutorial duplicate data* into the *File name* box.
4. Select *DAT Golden Software Data (\*.dat)* in the *Save as type* box.
5. Click the *Save* button.
6. In the **Data Export Options** dialog, accept the defaults and click the *OK* button.

The data are saved as a point set in the specified location. If you are using the demo version of **Voxler** you will not be able to use the **File | Save Data** command.

### Exporting Graphics

The **File | Export** command exports the entire **Viewer** window. This is a graphics-only export, so no data are saved for this operation. **Voxler** is designed to export the visible portion of the plot when using the **File | Export** command. It uses the monitor to define the limits of the exported image. To export a larger or smaller portion of the display, zoom in or out prior to export.

To save a graphic:

1. Click the **File | Export** command. The **Export** dialog opens.
2. In the **Export** dialog, type *tutorial graphic* into the *File name* field.
3. Keep *BMP Windows Bitmap (\*.bmp)* in the *Save as type* field.
4. Click the *Save* button. The **Export Options** dialog opens.
5. In the **Export Options** dialog, leave the default selections and click *OK*.

The image is saved as a bitmap .BMP file in the specified location. If you are using the demo version of **Voxler** you will not be able to use the **File | Export** command.

### Copying Graphics

Click the **Actions | Copy Snapshot** command to copy a raster version of the current view of objects in the **Viewer** window to the clipboard. The size of the copied image is the same as it appears in the current **Viewer** window.

To paste the raster graphics to another program, switch to the other program and choose the **Edit | Paste** command or press CTRL+V.

If you are using the demo version of **Voxler** you will not be able to use the **Actions | Copy Snapshot** command.

### Capturing a Video

Videos can be sent to co-workers or other colleagues that may have an interest in the output of your **Voxler** model. A video allows you to send another person the full model with rotations, so they can get a full idea of the project.

Saving the file to a **Voxler** .VOXB file using the **File | Save** command is another good way of sharing models. Any other **Voxler** user can open the .VOXB file and rotate the graphic to the desired orientation. If a user does not have **Voxler**, they can download the demo to experiment with the model themselves.

To capture a video:

1. Adjust the **Viewer** window to the desired size for the video. The video capture will use the size of the current **Viewer** window. The size of the window will affect the file size of the final .AVI file.
2. Click the **Actions | Capture Video** command. The **Capture Video** dialog opens.
3. Change the *Path* to specify a location where you want to save the captured video. The default path location is C:\Users\<user name>\Documents\VoxlerVideo.avi.
4. Adjust the *Frame rate* to 15, which is a value that produces good results.
5. Adjust the *Quality* to 80% with the slider.
6. Click the *Start Capture* button to begin the video capture. The *Estimated time (sec)*, *Estimated file size (MB)*, and *Estimated frames* information is dynamically displayed during capture.
7. Click the *Stop Capture* button to end the video capture. The *Estimated time (sec)*, *Estimated file size (MB)*, and *Estimated frames* information is statically displayed when the video has successfully completed.

If you are using the demo version of **Voxler** you will not be able to use the **Actions | Capture Video** command.

## Lesson 7 - Importing and Displaying Wells

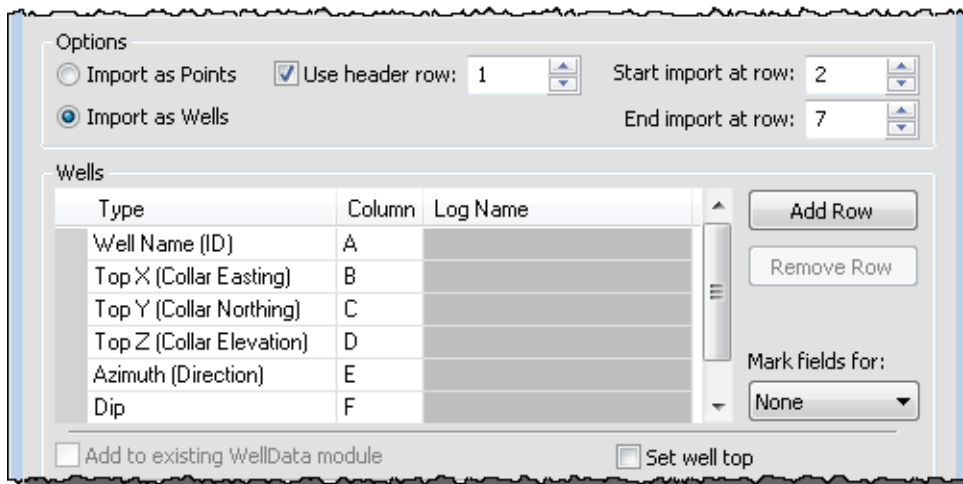
**Voxler** can import well data from numerous sources, including from LAS files. Typically the physical location of the well (collars) will be imported from one file and the well data (logs) will be imported from another file. Often, trajectory data indicating the direction of the well trace will also be imported, when the well is not assumed to be vertical.

To start this project in a new empty window, click the **File | New** command. If you have not already done so, you can save the previous tutorial information with the **File | Save** command, unless you are using the demo version.


### Importing Well Collar Information

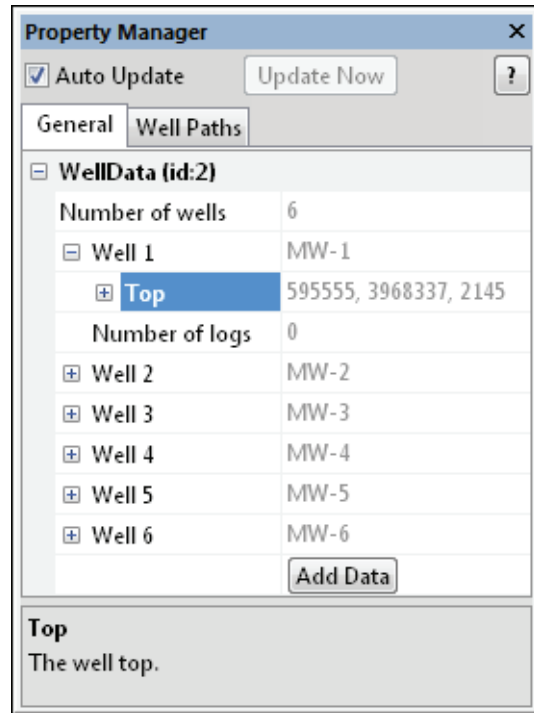
For this example, well collars, directional survey trajectory data, and log data are on three tabs of an Excel spreadsheet. It is important in **Voxler** to always import the collar information first, before the log or trajectory data. Each tab is imported separately, with the following steps.

1. Click the **File | Import** command.
2. In the **Import** dialog, select the well collar file. For this example, select the *SampleWellData.xlsx* file from the Samples directory. By default, the Samples directory is located at c:\Program Files\Golden Software\Voxler 3\Samples.
3. Click *Open*.



*Set each column to the correct Type so that the collar information imports correctly.*

4. In the **XLSX Import Options** dialog, select the *Collars* table and click *OK*.
5. In the **Select Data Columns** dialog,
  - a. Select *Import as Wells* to display the import options for well data.
  - b. Check the box next to *Use header row* and make sure the row is set to 1.
  - c. The *Start import at row* is automatically changed to 2, so that the header row is not imported.
  - d. In the *Wells* section, confirm that the *Type* is set to *Well Name (ID)* for Column A, *Top X (Collar Easting)* for B, *Top Y (Collar Northing)* for C, *Top Z (Collar Elevation)* for D, *Azimuth (Direction)* for E, *Dip* for F, and *Total Depth* for G.
  - e. Leave all other options as is and click *OK*.
6. In the **Property Manager**, you can see the six wells that were imported. If you click on the  next to any well, you can see the *Top* information that was imported for that well.



*The collar information is displayed as the well Top.*

## Importing Trajectory Data

At this point, all that has been imported is the collar, or top location, of the six wells. The well trajectory is the trace of the well. The trace defines the way the location of the well moves as it gets deeper. To import the well trajectories, follow the steps in this section.

1. Click the **File | Import** command.
2. In the **Import** dialog, select the well file. Select the *SampleWellData.xlsx* again and click *Open*.
3. In the **XLSX Import Options** dialog, select the *Trajectories* table and click *OK*.
4. In the **Select Data Columns** dialog,
  - a. First, take a look at the format of the columns. The first column ties the data in this worksheet to the previous worksheet because the well IDs are the same for both worksheets.



	A	B	C	D
1	ID	MD	Azimuth	Inclination
2	MW-1	0	340	44.62
3	MW-1	20	340	44.89
4	MW-1	40	340.29	45
5	MW-1	60	340.62	44.9

*The trajectory data contains the well name, and data that indicates the direction the well travels.*

- b. Select *Import as Wells* to display the import options for well data.
- c. Check the box next to *Use header row* and set the value to 1.
- d. The *Start import at row* value will automatically change to 2, so that the header row is not imported.
- e. In the *Wells* section, confirm that the *Type* is set to *Well Name (ID)* for Column A, *MD (Measured Depth)* for B, *Azimuth (Direction)* for C, and *Inclination (Deviation)* for D.
- f. Make sure that the box is checked next to the *Add to existing WellData module* option. This adds the trajectory information to the existing *WellData* container module that has the collar information.
- g. Leave all other options as is and click *OK*.

The information is imported, but the actual X, Y, and Z values for the path of the well need to be calculated by the program.

To have the program calculate the actual X, Y, and Z values,

1. Click on the *WellData* module in the **Network Manager**.
2. Click on the **Well Paths** tab in the **Property Manager**.
3. Click the *Compute Paths* button and the paths are calculated.

### Displaying the Wells

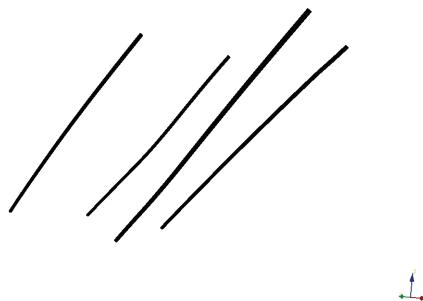
At this point, we can display the wells by clicking the **Network | Graphic Output | WellRender** command. The well traces are then displayed in the **Viewer** window.

### Importing Log Data

The well appearance can be altered by importing additional data.

1. Click on the *WellData* module in the **Network Manager**.
2. In the **Property Manager**, click on the **General** tab.

3. Scroll to the bottom of the list and click the *Add Data* button.
4. In the **Import** dialog, select the well file. Select the SampleWellData.xlsx again and click *Open*.
5. In the **XLSX Import Options** dialog, select the *Samples* table and click *OK*.
6. In the **Select Data Columns** dialog,
  - a. Select *Import as Wells* to display the import options for well data.
  - b. Check the box next to *Use header row* and make sure that the row is set to 1.
  - c. The *Start import at row* option is automatically set to 2, so that the header row is not imported.
  - d. In the *Wells* section, confirm that the *Type* is set to *Well Name (ID)* for Column A, *From Depth* for B, and *To Depth* for C.
  - e. Click the *Add Row* button.
  - f. The new row should show *Log* as the *Type*, column D as the *Column*, and *TiO2* as the *Log Name*.
  - g. Click the *Add Row* button again.
  - h. The new row should show *Log* as the *Type*, column E as the *Column*, and *MnO*



The well traces are displayed in the **Viewer** window.

**Options**

☐ Import as Points
 ☒ Use header row: 1
 Start import at row: 2
 End import at row: 211

☒ Import as Wells

**Wells**

Type	Column	Log Name
Well Name (ID)	A	
From Depth	B	
To Depth	C	
Log	D	TiO2
Log	E	MnO

Mark fields for: None

☒ Add to existing WellData module
 ☐ Set well top

WellData
 X: 0

Import the log data using the settings above.

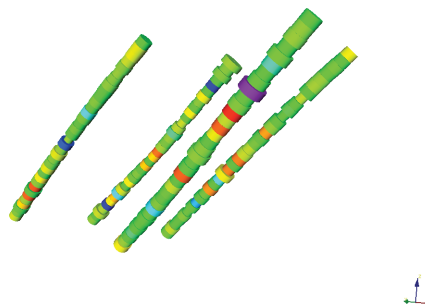
as the *Log Name*.

- i. Make sure that the box is checked next to the *Add to existing WellData module* option. This adds the log data to the existing *WellData* container module that has the collar and well trajectory information already imported.
- j. Leave all other options as is and click *OK*.

## Displaying Log Data on the Wells

Once the log data is imported, the well appearance can be modified to display the logs using these steps:

1. Click on the *WellRender* module in the **Network Manager**.
2. Click on the **Interval Data** tab in the **Property Manager**.
3. Check the box next to *Show intervals* to add variable width log information.
4. Change the *Interval log* to MnO to use the MnO log *To Depth* and *From Depth* as the interval definition.
5. Change the *Color method* to *By log* so the colors of the intervals vary with log data.
6. Set the *Color log* to MnO.
7. Change the *Colormap* to *Rainbow* to display different colors along the length. The colors are determined by mapping the values in the MnO log to the colors in the *Rainbow* colormap.
8. Set the *Size method* to *By log*.
9. Set the *Size log* to TiO2. The width of the log is now determined by the data in the TiO2 log.



*The wells are now displayed as tubes of variable width and color, based on the data from the two log columns.*

Congratulations, you have completed the **Voxler** tutorial! Advanced tutorial lessons and detailed information for each feature are available in the help file. Training videos for additional help are available on the Golden Software website.

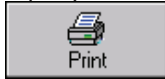
## Printing the Online Help

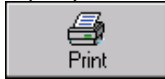
The online help topics may be printed. You can print a single topic, a section of the table of contents, or all topics in the table of contents. Open the online help by selecting the **Help | Contents** command in the **Grapher** window.

### Printing One Topic

To print one topic:

1. Click the topic you want to print.




2. Click the  button.
3. If the **Contents** page is open in the help navigation pane, the **Print Topics** dialog appears. Select *Print the selected topic* and click **OK**.

### Printing One Book

To print one book, the tutorial for example:

1. Click the **Contents** tab on the left side of the help window.
2. Expand the **Voxler 3** book and click on the *Tutorial* book.




3. Click the  button.
4. The **Print Topics** dialog appears. Select *Print the selected heading and all subtopics* and click **OK**. All the topics included in the *Tutorial* book are printed.

### Printing the Entire Help File

To print all of the topics in the help file table of contents:

1. Select the top-level book in the help book, **Voxler 3**.



2. Click the  button within the help window.
3. The **Print Topics** dialog appears. Select *Print the selected heading and all subtopics* and click **OK**. All the topics included in the online help table of contents are printed. **WARNING:** Printing the entire help file takes hundreds of letter-sized sheets of paper and is very time consuming to print. There is no table of contents or index printed with the file.


## Getting Help

The quick start guide is a quick way to learn about the basics in **Voxler**. There are also other sources of help with **Voxler**.

### Online Help

Extensive information about **Voxler** is located in the online help. Click the **Help | Contents** command to access the online help. You can navigate the online help using the **Contents**, **Index**, **Search**, and **Favorites** tabs on the navigation pane to the left of the topic page.

### Context-Sensitive Help

**Voxler** also contains context-sensitive help. Highlight a menu command, window region, or dialog and press the F1 key to display help for the highlighted item. You may also access context-sensitive help by pressing SHIFT+F1 or clicking on the  button. Then, click on a menu command, toolbar button, or screen region to view information specific to that item. The help window appears with additional information. In addition, most dialogs and the **Property Manager** contain a help button. Click the  button in a dialog title bar or at the top of the **Property Manager** to open the help topic for the displayed properties.

### Internet Resources

There are several internet help resources. Direct links to the Golden Software home page ([www.GoldenSoftware.com](http://www.GoldenSoftware.com)), the **Voxler** product page, frequently asked questions, and the knowledge base are available by clicking **Help | Golden Software on the Web**.

- Click the *Forums* button in the online help (**Help | Contents** command) to post a question to our public support forums.
- Click the *Knowledge Base* button in the online help to search for an answer in our frequently updated knowledge base.
- Use the **Help | Feedback** commands to send a problem report, suggestion, or information request by email directly to **Voxler** technical support.
- Search the FAQs on our website at [www.GoldenSoftware.com](http://www.GoldenSoftware.com).
- Watch the training videos on our website at <http://www.GoldenSoftware.com/supportvideo.shtml>.
- Read through our blog items at <http://www.GoldenSoftware.com/blog>.

### Technical Support

Golden Software's technical support is free to registered users of Golden Software products. Our technical support staff is trained to help you find answers to your questions quickly and accurately. We are happy to answer all of your questions about any of our products, both before and after your purchase. We also welcome suggestions for improvements to our software and encourage you to contact us with any ideas you may have for adding new features and capabilities to our programs.

Technical support is available Monday through Friday 8:00 AM to 5:00 PM Mountain Time, excluding major United States holidays. We respond to email and fax technical questions within one business day. When contacting us with your question, have the following information available:

- Your **Voxler** serial number (located on the CD shipping cover or in the **Help | About Voxler** dialog)
- Your **Voxler** version number, found in **Help | About Voxler**, including whether it is the 32-bit or 64-bit version of **Voxler**
- The operating system you are using (Windows XP, Vista, 7, or higher), including whether it is a 32-bit or 64-bit operating system

If you encounter problems with **Voxler**, you are welcome to send an email message to Golden Software using the **Help | Feedback | Problem Report** command. This message is delivered directly to [voxlersupport@goldensoftware.com](mailto:voxlersupport@goldensoftware.com). Report the steps you perform when the problem occurs and include the full text of any error messages that are displayed. You are welcome to attach a .ZIP file (10 MB maximum) containing the .VOXB file that illustrates the problem. Contact technical support for other arrangements if you have very large zipped attachments to send.

### Contact Information

Telephone: 303-279-1021

Fax: 303-279-0909

Email: [VoxlerSupport@GoldenSoftware.com](mailto:VoxlerSupport@GoldenSoftware.com)

Web: [www.GoldenSoftware.com](http://www.GoldenSoftware.com) (includes FAQs, knowledge base, support forum, training videos, newsletters, blog, downloads, and more!)

Mail: Golden Software, Inc., 809 14<sup>th</sup> Street, Golden, Colorado 80401-1866, USA

## Index

3 minute tour 4

---

### **A**

active x 5  
Annotation 18  
automation 5, 21, 22  
Axes 18

---

### **B**

BASIC language 21  
blog 45  
bold text 3  
BoundingBox 18, 27, 29

---

### **C**

capture video 36, 38  
ChangeType 13  
check for update 2  
ClipPlane 18  
collars 39  
commands 3  
computational modules 13, 30  
connecting modules 9, 34  
contact information 46  
contents 45  
context-sensitive help 45  
Contours 4, 18, 34, 35  
copy snapshot 36, 38

---

### **D**

data 15, 23  
    geometry 16  
    lattice 16  
    point sets 16  
    save 37  
data import options 24  
data sources 15  
demo 23  
disk space 2  
docking 12  
documentation 3  
DuplicateFilter 14, 33

---

### **E**

email 46  
example files 4  
ExclusionFilter 14  
export 36, 37  
ExtractPoints 4, 14

---

### **F**

F1 key 45  
FaceRender 4, 19  
FAQs 3, 45  
fax number 46  
feedback 45  
Filter 14  
filtering data 32  
font 3  
forums 45  
frequently asked questions 45

full user's guide 3

FunctionLattice 17

---

### **G**

general modules 17

general steps 5

geometry 16

Golden Software on the web 45

Gradient 14

graphic examples 4

graphics output modules 18

Gridder 14, 30, 31

---

### **H**

HeightField 4, 19

help 3, 45, 46

- contents 3, 44

- FAQs 45

- forums 46

- online 44

- video 46

help button 45

help file 45

help menu

- contents 45

- feedback 45

- Golden Software on the web 45

---

### **I**

import 5, 15, 23, 24

Info 17

information request 45

install 2

Internet help 45

Isosurface 19, 31

italic text 3

---

### **K**

knowledge base 3, 45

---

### **L**

labels 28, 29, 30

lattice 16

Light 17

log 42, 43

logs 39

---

### **M**

mailing address 46

managers 6, 12

- docking 12

- module manager 6, 7, 26

- network manager 6, 7, 26

- property manager 6, 7, 28

Math 15

menu 3, 7, 8

menu bar 6, 7

menu commands 8

Merge 15

module manager 6, 7, 10, 26

modules 13, 26, 34

- computational 13

- connecting 9, 34

- general 17



graphics output 18  
module types 1  
monitor resolution 2

---

## **N**

network manager 6, 7, 8, 26  
new features 2  
newsletters 46

---

## **O**

ObliqueImage 19  
online help 44, 45  
opening Voxler 23  
OrthoImage 20

---

## **P**

phone number 46  
point sets 16  
print help 44  
print tutorial 44  
problem report 45  
property manager 6, 7, 11, 28  
purchase guide 3

---

## **Q**

quick tutorial 4

---

## **R**

RAM 2  
rearranging managers 12  
Resample 15

rotate 30  
run script 5

---

## **S**

sample 4, 5  
save 36, 38  
save data 37  
ScatterPlot 4, 20, 26  
    labels 28  
script 5, 21  
script definition 22  
scripter 1, 5, 22  
scripts 21  
select data columns 24, 25, 40  
serial number 23, 46  
Slice 15  
starting Voxler 23  
status bar 6, 7, 8  
StreamLines 20  
Subset 15  
suggestions 45  
support forums 3, 45  
system requirements 2

---

## **T**

tabs 7  
technical support 46  
telephone number 46  
TestLattice 17  
Text 20  
three minute tour 4  
title bar 6, 7  
toolbars 6, 7, 8

tour 4  
trajectory 40  
Transform 15  
transparency 13, 32, 36  
tutorial 22  
    print 44

---

### **U**

uninstall 3  
update 2  
user interface 6  
users 1  
user's guide 3  
using Voxler 5

---

### **V**

VectorPlot 21  
version number 46  
videos 45  
viewer 12  
viewer window 6, 7, 12, 17, 30  
vista 3  
VolRender 21  
volume 1

---

### **W**

web 46  
web address 46  
WellData 4, 16, 40, 41  
WellRender 21, 41  
wells 39, 40, 41, 43  
well trace 40  
window layout 12

Before calling, please check the following available resources as your question may already be answered.

<b>Registration:</b> Register online at <a href="http://www.GoldenSoftware.com">www.GoldenSoftware.com</a> or fax or mail the <i>Registration Form.PDF</i> , located in the main directory on the CD
<b>Knowledge Base:</b> <a href="http://www.GoldenSoftware.com/activekb">www.GoldenSoftware.com/activekb</a> or in the <b>Voxler</b> program using the <b>Help   Golden Software on the Web   Knowledge Base</b> command
<b>Forums:</b> <a href="http://www.GoldenSoftware.com/forum">www.GoldenSoftware.com/forum</a> or in the <b>Voxler</b> program using the <b>Help   Contents</b> command and click on <i>Forums</i>
<b>Frequently Asked Questions:</b> In the <b>Voxler</b> program using the <b>Help   Golden Software on the Web   Frequently Asked Questions</b> command
<b>Tutorial:</b> Complete the tutorial section in this quick start guide or in the <b>Voxler</b> program using the <b>Help   Tutorial</b> command
<b>Online Help:</b> In the <b>Voxler</b> program using the <b>Help   Contents</b> command
<b>Support Videos:</b> <a href="http://www.GoldenSoftware.com">www.GoldenSoftware.com</a>

### Business Hours

Technical Support:

Monday through Friday, 8:00 AM - 5:00 PM, Mountain Time

Product Sales:

Online orders available 24 hours, 7 days a week with 2 business hour delivery

### Golden Software Contact Information

[www.GoldenSoftware.com](http://www.GoldenSoftware.com)

[VoxlerSupport@GoldenSoftware.com](mailto:VoxlerSupport@GoldenSoftware.com)

phone: 303-279-1021

fax: 303-279-0909





[www.GoldenSoftware.com](http://www.GoldenSoftware.com)