GOCAD in a Hurry

Created and modified by Amy Cheng
amy@ensg.u-nancy.fr

for

ASGA
Nancy, France

Created May, 1996
4th Edition July, 1997

Disclaimer
Although an attempt has been made to ensure the veracity of the following material, no responsibility is assumed by the author for any use, or for any consequences resulting from any use, of the information contained herein. No guarantee of suitability for any purpose is offered or implied.

PREFACE

This version of **GOCAD in a Hurry** is created for GOCAD release 1.4. It provides a quick and very brief overview for people who want to start using GOCAD in a hurry.

If you need more detailed information on any of the subjects in the **GOCAD in a Hurry**, please see the `GOCAD Quick Reference`. ASGA also offers GOCAD TUTORIAL series that assists you to build a solid GOCAD foundation.

This and other GOCAD documents and associated data can be downloaded from ASGA; please contact Jean-Claude Dulac (jcdulac@t-surf.com) or Fabien Bosquet (bosquet@ensg.u-nancy.fr).

Please send your comments or questions about this document to:

Amy Cheng (amy@ensg.u-nancy.fr).

CONTENTS

CONTENTS

1. MINIMUM

Many users have expressed their interest in using GOCAD, while bemoaning their lack of time to really learn GOCAD from scratch. This document, **GOCAD in a Hurry**, offers a one-hour overview of GOCAD.

This chapter covers the minimum knowledge you need to start using GOCAD.

Here are the contents of this chapter:

However, do keep in mind that you may learn GOCAD in a hurry, but you cannot master GOCAD in a hurry.

1.1 FREQUENTLY ASKED QUESTIONS

1.1.1 What is GOCAD?

GOCAD is a computer application that allows you to import, create, modify, integrate, view, and export Geological Objects, all in 3D. GOCAD also offers a variety of geological, geophysical and geostatistical analyses that you can perform on these Objects.

1.1.2 Who has GOCAD?

GOCAD is the product of an academic consortium based at the National School of Geology in Nancy, France. It is sponsored by major oil companies, software and hardware vendors, and Universities in Europe, North America, South America and the Middle East. Any sponsor of GOCAD "has" GOCAD.

1.1.3 Who uses GOCAD and why?

Geophysicists use GOCAD to build velocity models for seismic processing or uncertainty analyses. Geologists use GOCAD to build structural models to better understand their fields. Reservoir engineers use GOCAD to build reservoir models for flow simulations. Geostatisticians use GOCAD to create multiple property realizations. And everyone uses GOCAD as a visualization tool to give effective and informative presentations.

1.1.4 How do I access GOCAD?

That, you have to find out from your local system administrator; or ask the person who passed you this document.

1.1.5 Can I run GOCAD on my PC?

Not yet. Currently (and current may be very short), you need a work station with a high-end graphic card. SGI, IBM, HP and SUN all provide one or more models that support GOCAD.

1.1.6 Can I really learn GOCAD in a hurry?

Yes, you can learn the basic stuff in less than one hour, provided that you already know how to use other workstation-based applications, such as Seisworks or Earth Vision. It will also help if you already have some GOCAD data files.

1.1.7 Can you bring non-GOCAD files into GOCAD?

There are several built-in converters that allow you to bring in non-GOCAD files, such as Zmap ASCII files. If you have a file format that you use a lot, you can contact Jean-Claude Dulac (jcdulac@t-surf.com) to contract GOCAD to have a converter written for you.

1.1.8 Are there a lot of funny names I have to learn?

Not really. GOCAD uses layman's terms when it talks to you (in the user's interface). If you see a word that you don't understand, check it out in Appendix C: GLOSSARY *(page C-1)*.

1.1.9 What kind of Objects do you have in GOCAD and what do you use them for?

You can get a better picture in 1.2.2 What Are the GOCAD Objects *(page 1-10)*, and a more detailed story in A.3 GOCAD Objects *(page A-10)*. Basically, there are the usual ones:

1. **Surface**
   for modeling geological horizons, faults, etc. (A.3.4 Surface (TSurf) *(page A-12)*);

2. **PointsSet** *(discrete points)*
   for modeling scattered measurements, etc., such as input from Zmap files (A.3.2 PointsSet (VSet) *(page A-11)*);

3. **Curve**
   for modeling fault cuts, culture data, etc. (A.3.3 Curve (PLine) *(page A-12)*);

4. **GShape** *(geological shape)*
   for modeling channels, etc. (A.3.6 GShape *(page A-14)*); and

5. **Well**
   for displaying well path and well log measurements (A.3.7 Well *(page A-15)*).

And there are the volumetric ones:

1. **Voxet** *(regular 3D grid)*
   for carrying seismic data, satellite images, etc. (A.3.8 Voxet *(page A-16)*);

2. **SGrid** *(stratigraphic 3D grid)*
   for modeling reservoir layers, etc. (A.3.9 SGrid *(page A-19)*); and

3. **Solid**
   for modeling closed tetrahedralized volume, such as salt bodies (A.3.5 Solid (TSolid) *(page A-13)*).

There are more, but you cannot learn everything in a hurry.

1.1.10 What does GOCAD look like?

See 1.2.1 The GOCAD Window *(page 1-8)*.

1.1.11 How do you get things done in GOCAD?

You click with the left mouse button. This includes selecting commands, selecting Objects, and selecting parameter values. Sometimes you need to enter a name from the keyboard. On two or three occasions you get to click with the middle and the right mouse buttons; this happens when you are digitizing.

1.1.12 If I already know how to use C-Gocad very well, does that help?

Sure, you'll just have to learn where to find things in GOCAD. And Appendix D: From C to ++ *(page D-1)* is there to assist you.

1.1.13 Is it easy to change things in GOCAD?

Changing things has two meanings in GOCAD: changing the physical appearance (display) of an Object, and actually changing the shape (geometry and connectivity) of an Object. The former is very very easy; the latter can be easy, but it takes more than 5 minutes to master.

1.1.14 So how do I change the display of an Object?

Anything concerning the "look" is done in the Attribute Manager, located in the lower left portion of the GOCAD window. Once you have selected an Object there, you can decide which component to display, in what color, what size, etc. See A.1.2 Attribute Manager *(page A-3)*.

1.1.15 How do I use the mouse buttons, you said?
Unless otherwise specified, **click** always means click with the left mouse button.

1. To select an item, be that a Menu, a command, a color, an Object, or a button, click with the left mouse button.

2. To move things in the Camera, press down the Shift-key and the left mouse button to turn things in the Camera; press down the Shift-key and the middle mouse button to shift things; press down the Shift-key and the right mouse button to zoom.

3. To select multiple items from the Object display list or from the Camera, press down the Ctrl-key and click on the item names in the Display List with the left mouse button. Release the Ctrl-key when you are done picking.

4. To digitize, start and continue with the left mouse button, delete the previous point with the right mouse button, and end the digitization with the middle mouse button.

5. To use a slider bar, move the cursor onto the slider bar, press down the left mouse button and drag the slider bar in the desired direction.

1.1.16 Does GOCAD give me any hints about how to complete a command?

Yes. in the following three ways:

1. First is the shape of the Cursor. If the cursor has a cross-hair shape, GOCAD is waiting for you to pick something from the Camera. If you are not sure why or what, there is always a message in the **Info Area** *(page A-7)* telling you what you are supposed to be picking.

2. Second is a Dialog window. If the command you have selected requires more than picking an Object, GOCAD will pop up a Dialog window and list all the things you need to specify. How do you specify them? You click with the left mouse button in the Dialog window. Occasionally, you also need to enter something from the keyboard, usually a name for a new Object. See **Operation Menu Bar** *(page A-4)*.

3. Third, well, is an error message when you have done something that makes it impossible for GOCAD to execute the command. An error message is printed in the Message Area in the **Main Control Area** *(page A-2)*.

1.1.17 Anything else I should know before I start?

A few things:

1. **NEVER** put a space or a special character (expect for underscore and dash) in the name of an Object.

2. **NEVER** use a GOCAD-reserved term, such as tsurf, vset, destroy, apply, etc. as the name of an Object.

3. Always toggle off unnecessary display components. Not only it is less distracting, it is faster for GOCAD to draw the Objects in the Camera.

4. When you move the cursor like mad and as a result lose sight of the Objects in the Camera, press the **Globe** button near the center of the GOCAD window. The Globe is the quick access button to the command **Autosetup** in the Camera **Settings** Menu.

1.1.18 Am I ready to start now?

Do you know the different areas in the GOCAD window? Where is the **Main Control Area** *(page A-2)*? Where is the **Attribute Manager** *(page A-3)*? Where is the **Operation Menu Bar** *(page A-4)*? Where is the **3D Camera** *(page A-6)*? Where is the **Info Area** *(page A-7)*? Where are the **Quick Buttons** *(page A-6)*? Where is the **Object-display List** *(page A-7)*? Or maybe you want to go to **The GOCAD Window** *(page 1-8)* or A.1 GOCAD Window *(page A-1)* to refresh your memory before you start.

1.1.19 Can I start now?

Yes, you can go to 2. Getting Started *(page 2-1)*, although it wouldn't hurt for you to finish the rest of this chapter first.

1.1.20 Don't you want to know how to get out of GOCAD when you are done?

The first Menu in the Menu Bar in the **Main Control Area** *(page A-2)* is **Project**. The last command in that Menu is **Exit.** When you are done, select this **Exit** command and GOCAD will politely ask for confirmation before it goes away.

1.2 GOCAD PICTORIAL

In this section, we introduce you to some common terms and common tasks in GOCAD, using pictures. More detailed stories (words) are given in Chapter 2. Getting Started *(page 2-1)*.

1.2.1 The GOCAD Window

1. **Main Control Area**
    There is a Menu Bar, a History Display Area, a Command Entry Area, and a Message Area. Commands you find here deal with the "big picture" stuff, such as copy Objects, delete Objects, record history, etc.

2. **Attribute Manager**
    Here you modify the display options of an Object (the "Style" of an Object), such as the color of a Surface, whether to display triangulation, the width (thickness) of a Curve, etc.

3. **Operation Menu Bar**
    Commands you find here are Object-specific, such as Cut a Surface, Close a Curve, Deform an SGrid, Create a Voxet, etc. The Operation Menu Bar is flexible; the Menus and the contents of the Menus change as the user changes the **Mode** of the Menu Bar.

4. **3D Camera**
    This is where you view the Objects. There is a Camera Menu Bar, the Camera Movement Control Buttons, Quick Buttons and an Object-Display List. All currently existing Objects are listed in the Object-Display List, from which you select Objects to be displayed in the Camera.

5. **Info Area**
    Here GOCAD prints out one-line command instructions, Object information and other messages.

If you want to know more about the GOCAD window, see A.1 <u>GOCAD Window</u> *(page A-1)*.

1. Main Control Area   **3. Operation Menu Bar**

gocad – Main Window – **untitled**

File   Edit   History   Applications   Windows

gocad on Sty e Bubble update
gocad update all cameras

Enter Command

2. Attribute Manager

Object Bubb e   Type TSurf

Attributes Categories   Graphic

Name

☐ Visible

Bounding Box

☐ Visible

Parts

☐ Visible

Atom markers

☐ Visible

Size   1

Symbol   point

Atoms Link

☑ Visible

Control nodes

☐ Visible

☑ Apply changes immediately

Apply   Discard   Close Editor

View   Display   Settings

Camera
Movement
Control

Orbit increment

4. 3D Camera

Quick
Buttons

All

aultPlane
NewDrill
NewDrillAmpl
NewDrillAmpl_n
OP
opHorizon
opSurfData

☐ Immediate

Object
Display
List

Info: Terminal menubar gives access to the terminal menus   **5. Info Area**

Ac

1.2.2 What Are the GOCAD Objects

Atomic Objects are made of Atoms, A.K.A. Vertices, Points or Nodes.
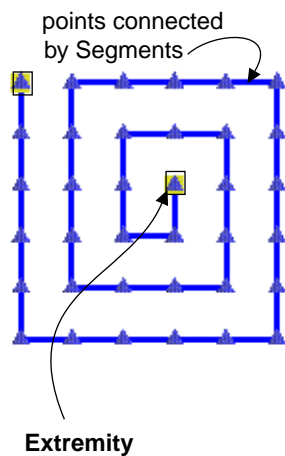
PointsSet

discrete points

Curve

points connected by Segments

Extremity

Surface

points connected by Triangles

Border

Solid

points connected by Tetrahedra

GShape, a Backbone (Curve) with a set of Sections

Same Object
Different display options
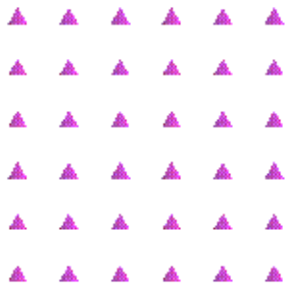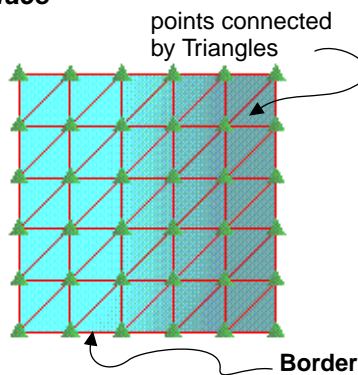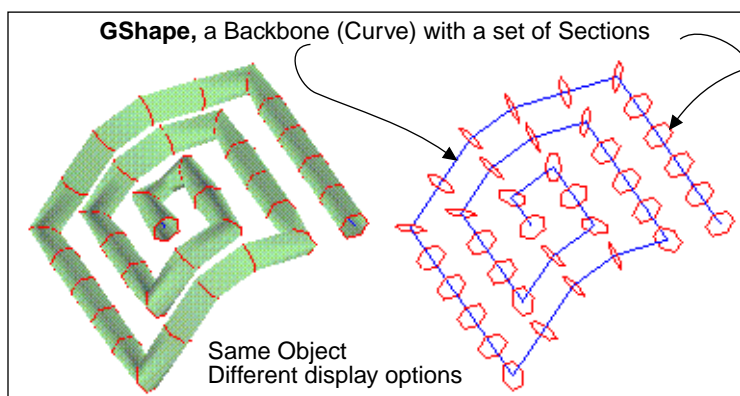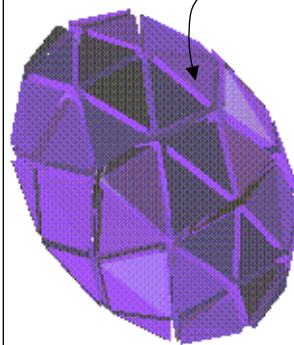
Ac

What Are the GOCAD Objects continued

Voxet, 3d regular Grid

SGrid, 3d Stratigraphic Grid

Section

Cage

a Well Object

Well Curves displayed against a Horizon

Ac

1.2.3 Properties in GOCAD

In **PointsSets**, **Curves** and **Solids**, Properties are displayed in color

In **SGrids** and **Voxets**, Properties are displayed in color

In **Wells**, Properties are displayed as Well Curves, which are in displacement and/or color

In **Surfaces**, Properties can be displayed in color or by contour lines

GOCAD also supports 3dProperties, also known as **Vector Properties**

Ac

1.2.4 Load Objects

gocad – Main Window – ** untitled **

File **Edit** **History** **Applications** **Windows** [G CAD]

1. Select the command Load Objects...;

2. A pop-up window appears;

3. Select the file(s);

Gocad – Load Objects

Filter

CLASS/GocadInAHurry/*

Directories   Files

naich/CLASS/GocadInAHurry/.   data.mx
naich/CLASS/GocadInAHurry/..   dataNewDri
                                                        dataNewDri

File types   **All**   Filters   **none**

Select All   **Deselect All**

OK   **Filter**   **Cancel**   **Apply**

4. Press OK;

All

A1
B1
BOTTOM
BaseHorizon
BotData
BottomSurf
Bubble
C1
D1
FaultPlane
NewDrill
NewDrillAmpl
NewDrillAmpl_n

■ **Immediate**

Apply

5. Objects in the files will appear on the Object Display List.

Ac

1.2.5 Display Objects

Single Selection Display, One Object at a time

1. In the Object Display List, move the cursor onto the name of the Object that you want to display, and click with the left mouse button.

2. The selected Object is highlighted;
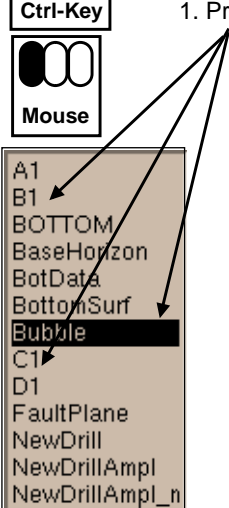
3. And displayed in the Camera.
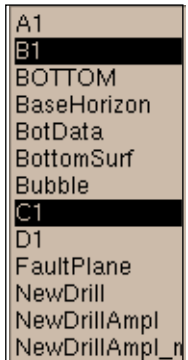
Ac

Toggle Selection: Select or De-select when you click

Ctrl-Key

Mouse
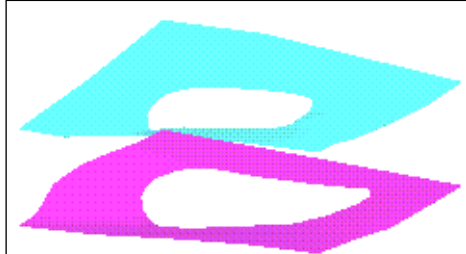
1. Press down the Ctrl-key while you click on the Objects' names.

2. Previously selected Objects will be de-selected and previously un-selected Objects will be selected;

3. And the display in the Camera will change accordingly.

Ac

Display Objects continued

Sequential Selection: Select a series of sequentially listed Objects

1. Move the cursor onto the name of the first Object.

2. Press down the left mouse button while you drag the cursor down the list. Release the mouse button after the last Object.

3. The swept area will be highlighted and the display will change accordingly.

A1
B1
BOTTOM
BaseHorizon
BotData
BottomSurf
Bubble
C1
D1
FaultPlane
NewDrill
NewDrillAmpl
NewDrillAmpl_n

Mouse

A1
B1
BOTTOM
BaseHorizon
BotData
BottomSurf
Bubble
C1
D1
FaultPlane
NewDrill
NewDrillAmpl
NewDrillAmpl_n

Ac

1.2.6 Manipulate Objects in the Camera

Shift-key + **Mouse** =

Shift-key + **Mouse** =

Shift-key + **Mouse** =

Zoom out

Zoom in

Ac
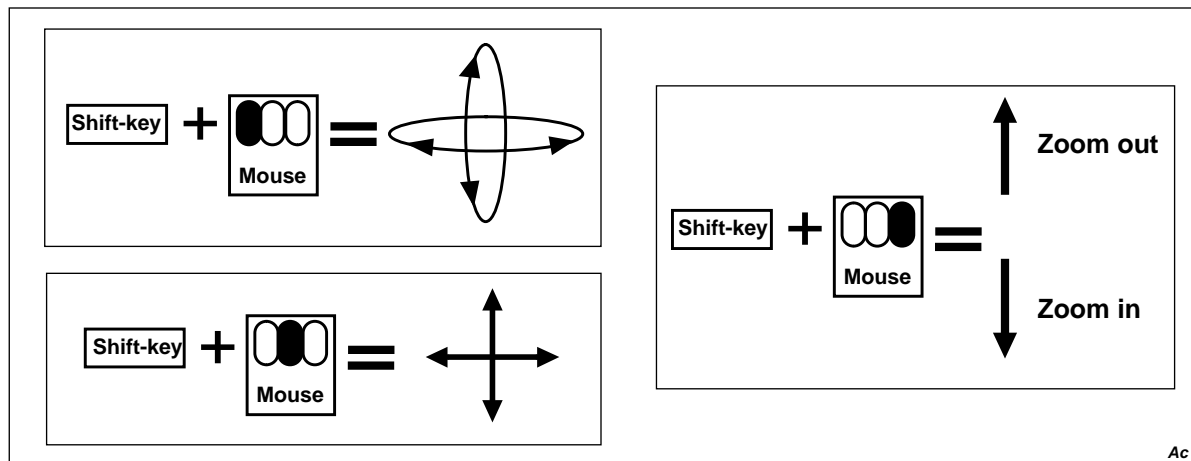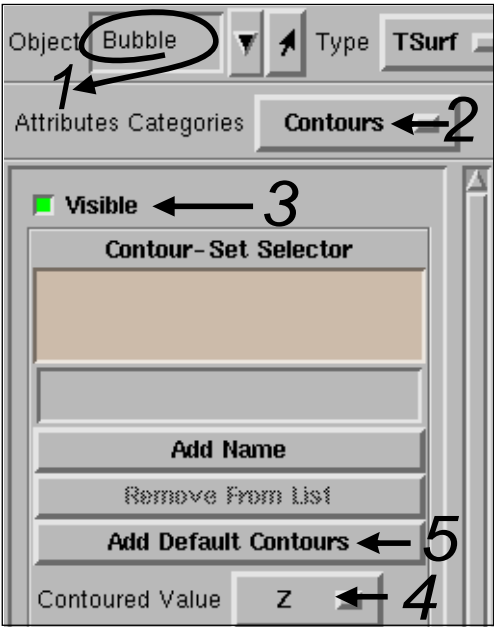
1.2.7 Display Property
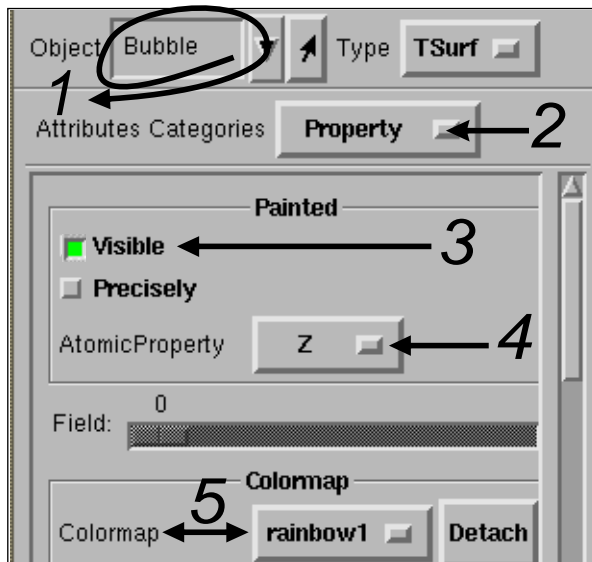
1. *Select the Object in the Attribute Manager.*

2. *Select the Property Attribute Category.*

3. *Toggle on the display.*

4. *Select the Property you want to display.*

5. *Select the Colormap you like.*

6. *View the Object in the Camera.*

Without Property   *With Property Z in Rainbow1*

1.2.8 Display Contour Lines

Object [Bubble] ▼ ◢ Type [TSurf]

1

Attributes Categories   [Contours] ←— 2
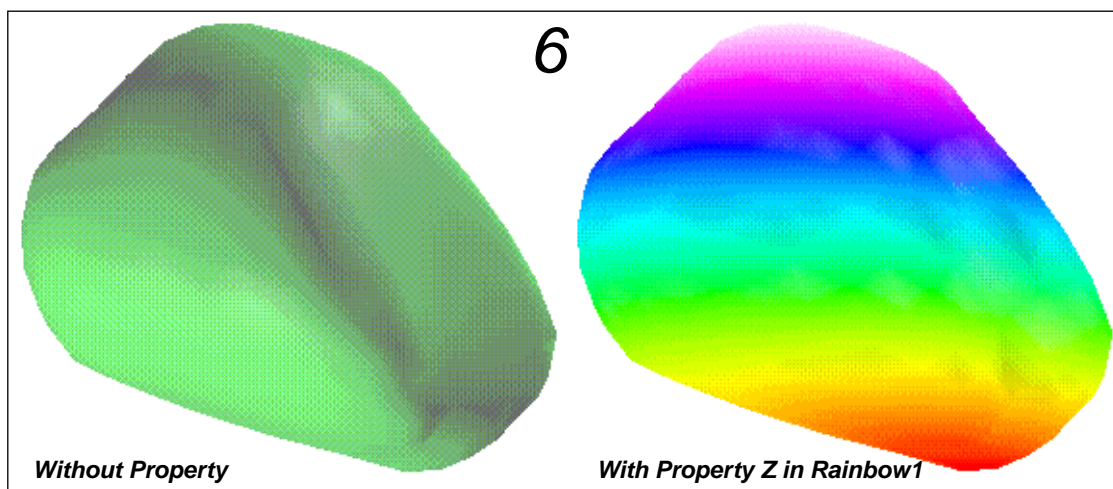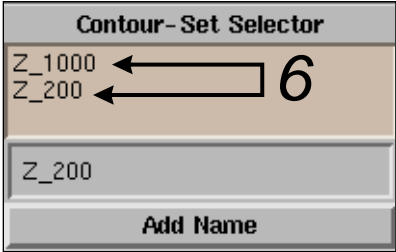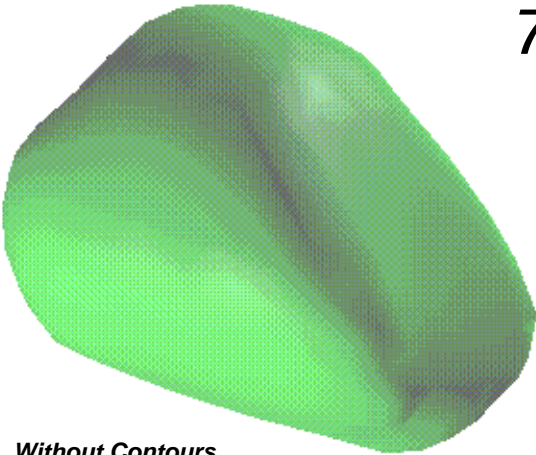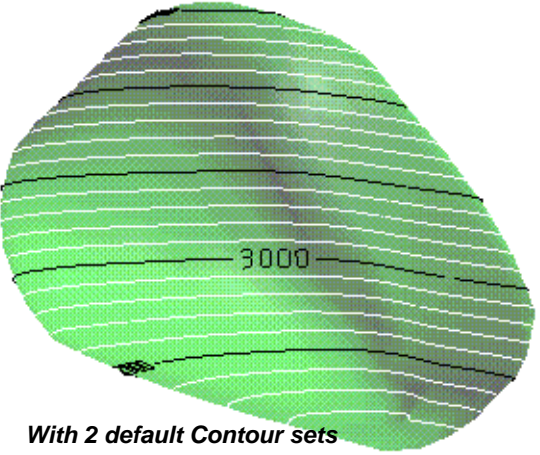
1. **Select the Object in the Attribute Manager.**
2. **Select the Contours Attribute Category.**
3. **Toggle on the display.**
4. **Select the Property to be contoured.**
5. **Press the Add Default Contours button.**
6. **Default Contour Sets are added.**
7. **View the Object in the Camera.**

■ **Visible** ←— 3

Contour–Set Selector

Add Name

Remove From List

Add Default Contours ←— 5

Contoured Value   Z ←— 4

Contour–Set Selector

Z_1000 ←—
Z_200 ←—   6

Z_200

Add Name

7
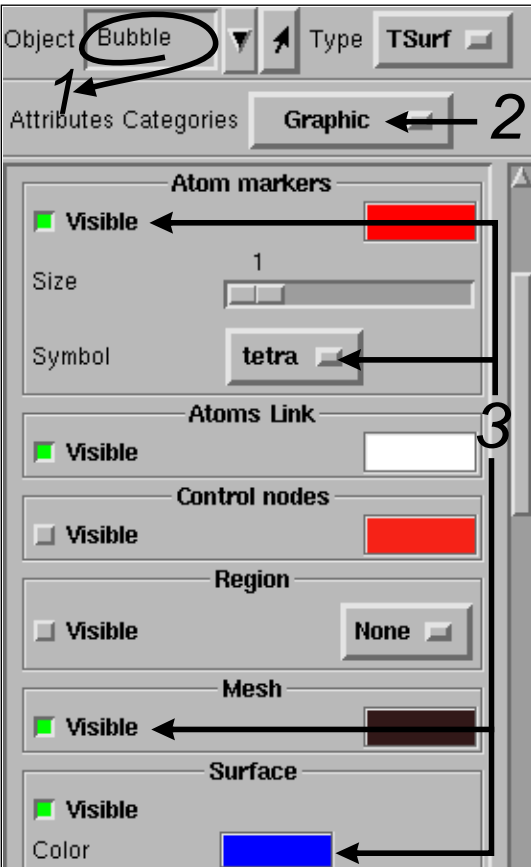
Without Contours   **With 2 default Contour sets**

Ac

1.2.9 Change Object color and other Attributes
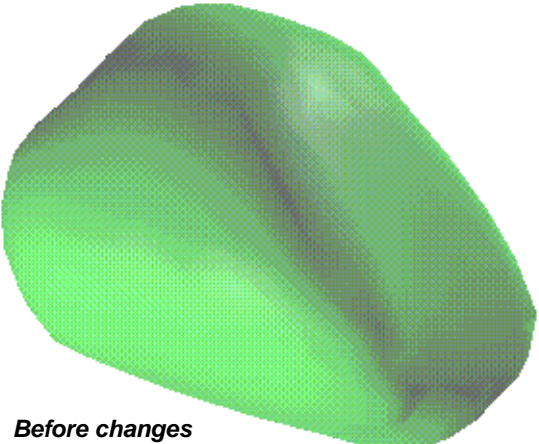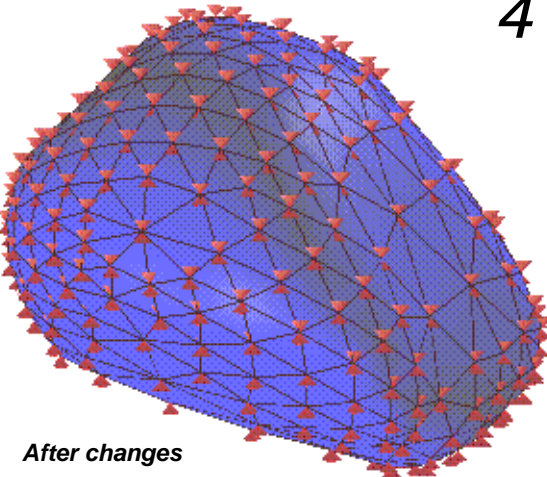
1. *Select the Object in the Attribute Manager.*
2. *Select the Graphic Attribute Category.*
3. *Change the Attributes to your liking.*
4. *The display of the Object in the camera changes accordingly.*

Before changes

After changes

1.2.10 Texture Mapping in GOCAD

Surface and the Voxet that carries the information

1. *Select the Surface in the Attribute Manager.*

2. *Change the Attribute Category to Texture.*

3. *In the Draping section, identify the Voxet.*

4. *Select the Voxet Property to be mapped onto the Surface.*

5. *Specify which particular Voxet Section is to be mapped onto the Surface.*

6. *Toggle on Visible, Smoothed and Precise.*

Voxet Property is texture-mapped onto the Surface.
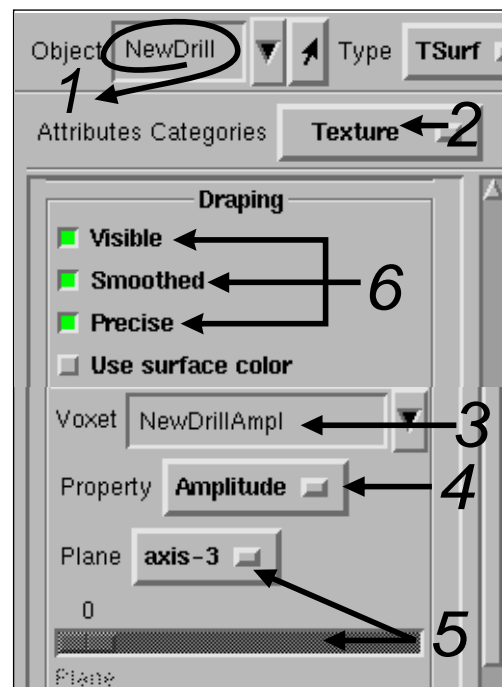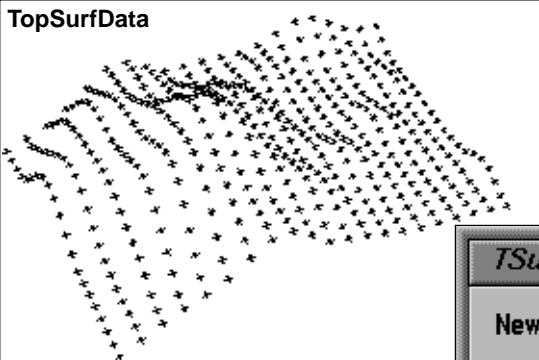
Ac

1.2.11 Create a Surface from Points

Input Data: a *PointsSet*

TopSurfData

1. Select the command *From PointsSet*, in *Surface Mode /New* Menu.

2. Give a name for the new Surface.
3. Identify the input data.
4. Press *OK*.

TSurfCreateFromAtomicGroup Dialog

New From Set of Points

Warning: points should be mappable to a plane.
Otherwise, surface topology can be inconsistent.

name TopSurf   *2*

AtomsSet points   TopSurfData   *3* ▼ ↗

☐ **use normal**

normal
X 0.   Y 0.   Z 1.   ↗

■ **dissociate vertices**

4
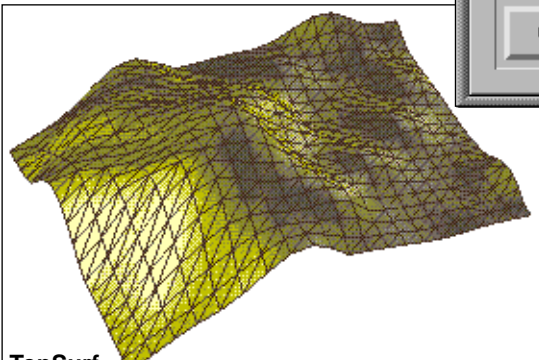
OK   Apply   Cancel   Help

A new Surface is created.

TopSurf

Ac

1.2.12 Modify an Existing Surface

Existing Surface and New Data to Fit

2. Identify the Surface.
3. Identify the Control Points data.
4. Make sure that **Optimize....** is off
5. Press **OK**.

TSurfSetFcp Dialog

Set Control Points

Surface   BottomSurf   *← 2*

AtomsSet control points   BotData   *← 3*

dir_shoot

X 0.   Y 0.   Z 1.

☐ optimize shooting direction   *← 4*

OK *5*   Apply   Cancel   Help
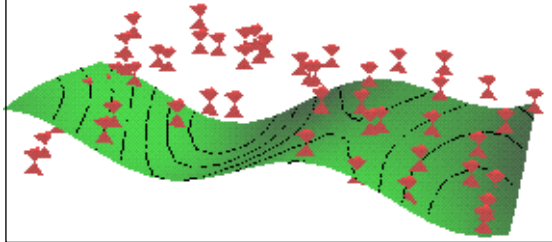
1. Select the command **Set Control Points**, in **Surface Mode/ Constraints/ Control Points** Menu.
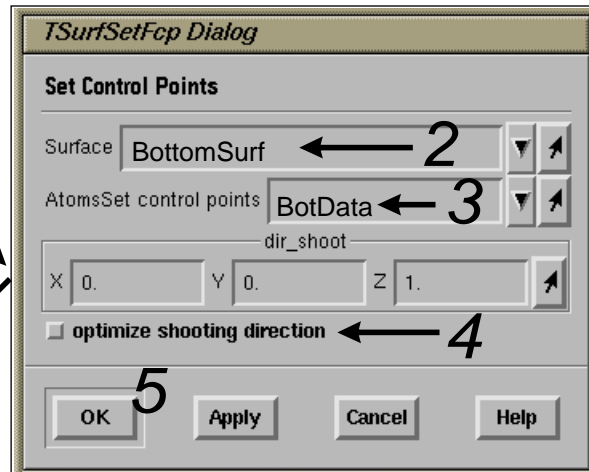
New Shape is defined.
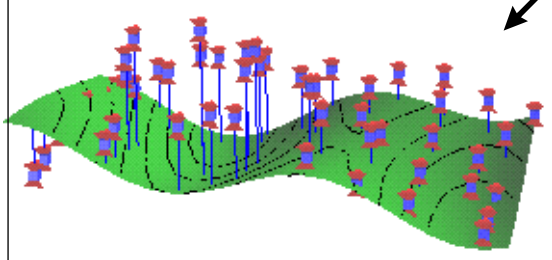
6. Select the command **Fit Geometry-On Entire Surface**, in **Surface Mode / Interpolate** Menu.

7. Identify the Surface you want to modify.
8. Set **nbiter** (number of iterations) based on how complicated the desired geometry is.
9. Press **OK**.

AtomicGroupRunDSICommand Dialog

Fit Geometry

AtomsSet   BottomSurf   *← 7*

nbiter 10   *← 8*

☐ smooth

☐ conjugate

OK *9* Apply   Cancel   Help
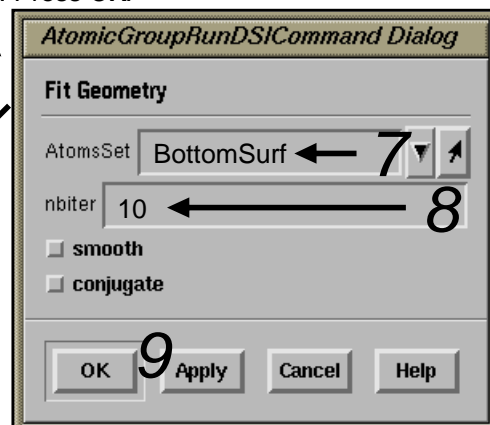
Surface is modified to fit new data.
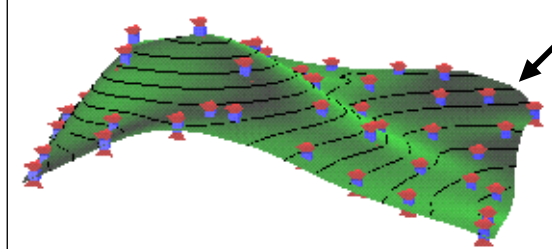
Ac

1.2.13 Build a 3DModel
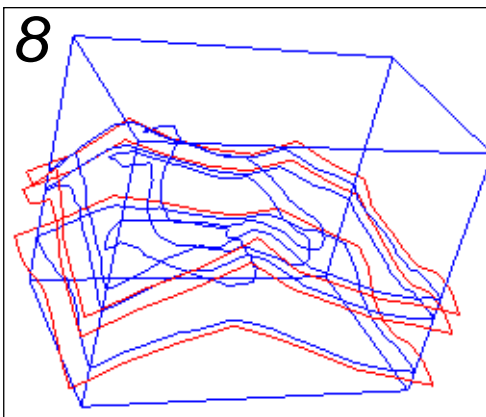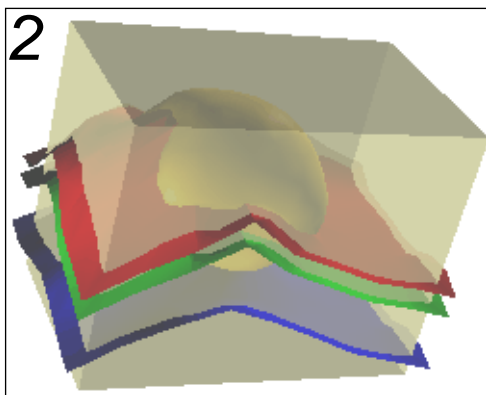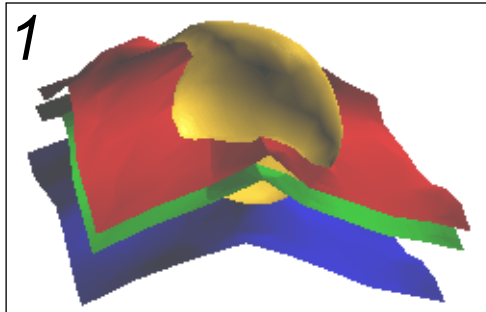
1. Display the Surfaces you want to use to construct the Model.

2. Add a bounding-box Surface.

3. Select the command **Surface/ Model / New/ From Surfaces**.

4. Give the new Model a name.

5. Identify the Surfaces to be included in the Model.

6. Toggle off **self-intersection**.

7. Press **OK**.

8. The Weiler Model display of the new Model.

Model3dCreateFromSurface Dialog

Create Model3d From Surfaces

name | SaltModel   *4*

Surface surfaces | Surf SaltAboveSurf SaltBelowSurf *5*

☑ **copy surfaces** ←———— *6*

Perform a shallow copy of the input surfaces (recommended)

☐ **self intersection**

Assumes that the surfaces self-intersect (much slower)

☐ **Z = Depth**

Z is Depth or Altitude (Altitude by default)

☑ **build**

7

OK   Apply   Cancel   Help

AC

Build a 3DModel continued

9. Different Regions in the new Model, listed in the Attribute Manager.

10. Display of the different Regions in the new Model.
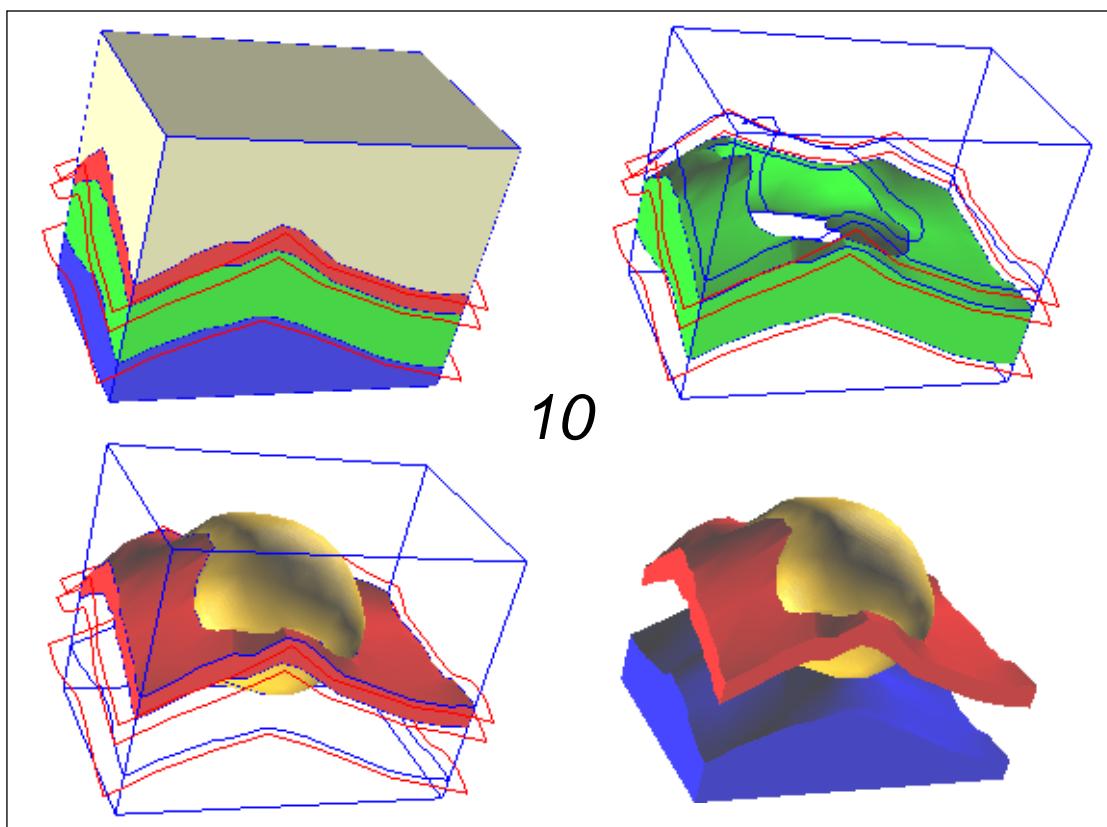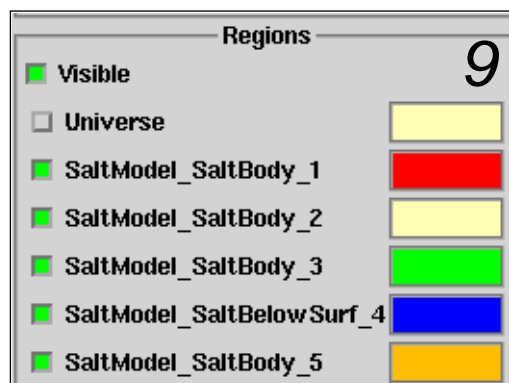
1.2.14 Deform an SGrid

1. Create an SGrid.

2. Display the Surfaces to be fit.

3. Select the command **Init Geometry - Proportional Between Top and Bottom**, in **SGrid Mode/ Edit** Menu.

4. Specify the SGrid you want to deform.

5. Specify the Surface that defines the Top shape.

6. Specify the Surface that defines the Bottom shape.

7. Toggle off **constrain**.   8. Press **OK**.

9. The SGrid is deformed to honor the Geology.

SGridInitProportional Dialog

SGrid Init Proportional

Stratigraphic Grid | ReservoirSG | *4*

Surface top | TOP | *5*

Surface bottom | BOTTOM | *6*

☑ **interpolate**

☐ **constrain** ← *7*

☐ **check thickness**

minimum thickness | 0.

8

OK   Apply   Cancel   Help

1

2

9a

9b

Ac
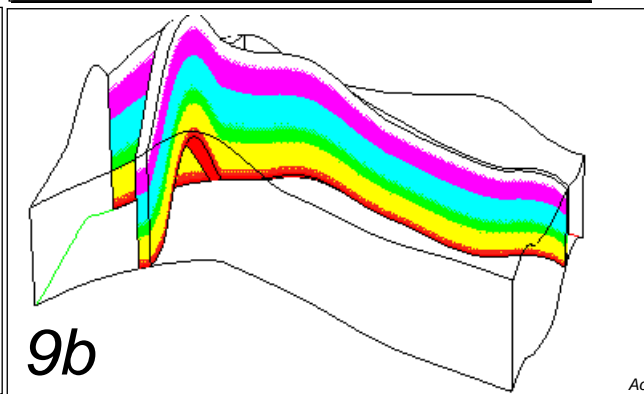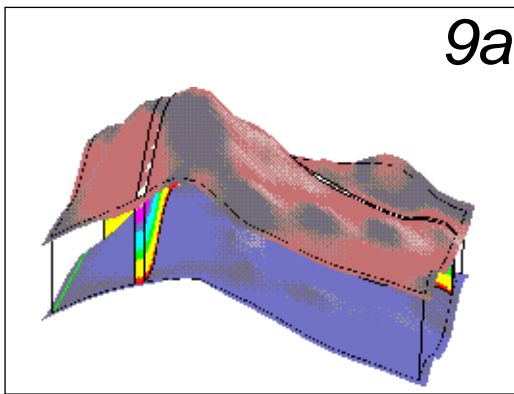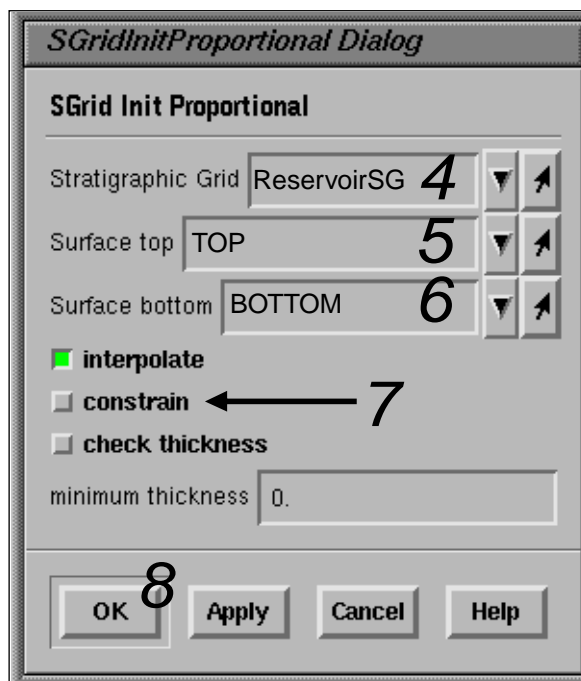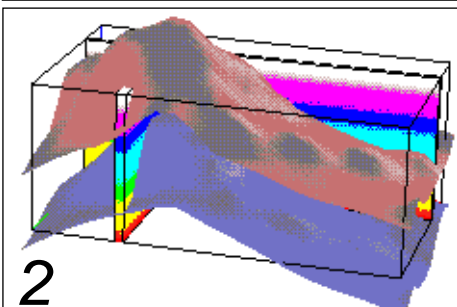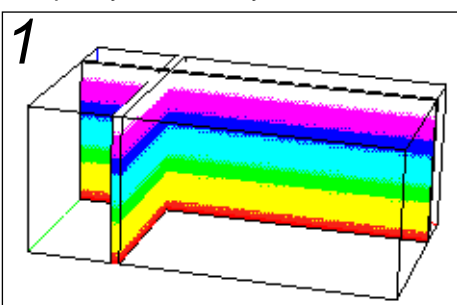
1.2.15 GOCAD Applications

Geophysical Applications: Velocity Modeling, Seismic Analysis, etc.
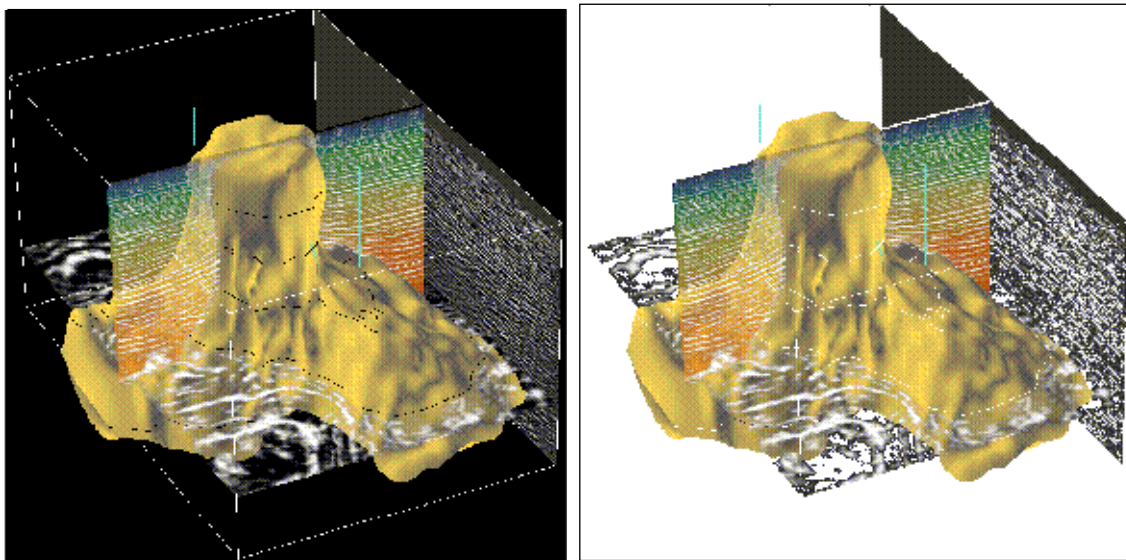
Courtesy of UNOCAL
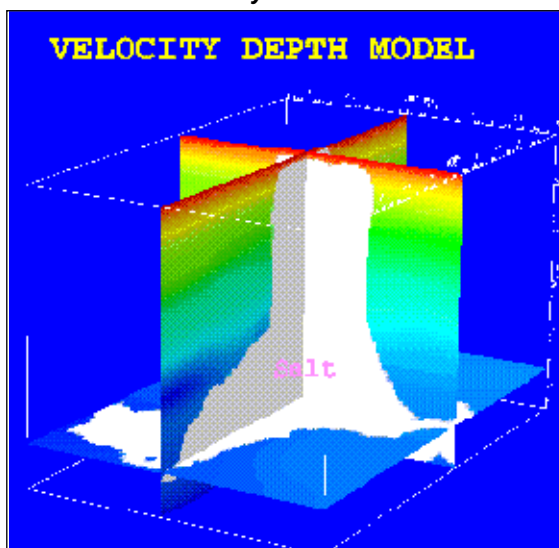
GOCAD Applications continued

Structural Applications: Structural Modeling, Curvature Analysis, etc.
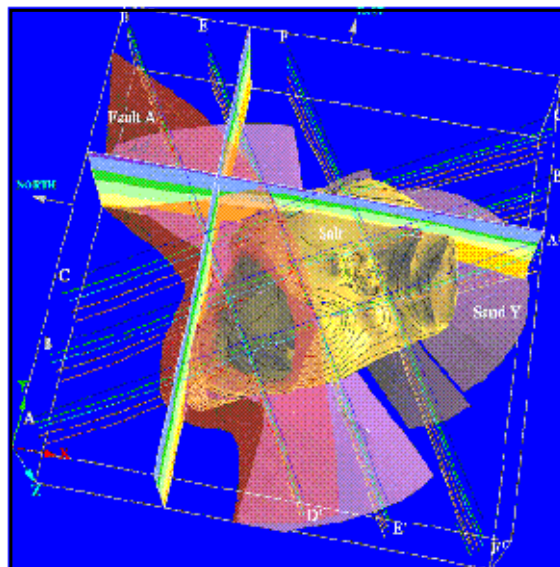
ASGA Model
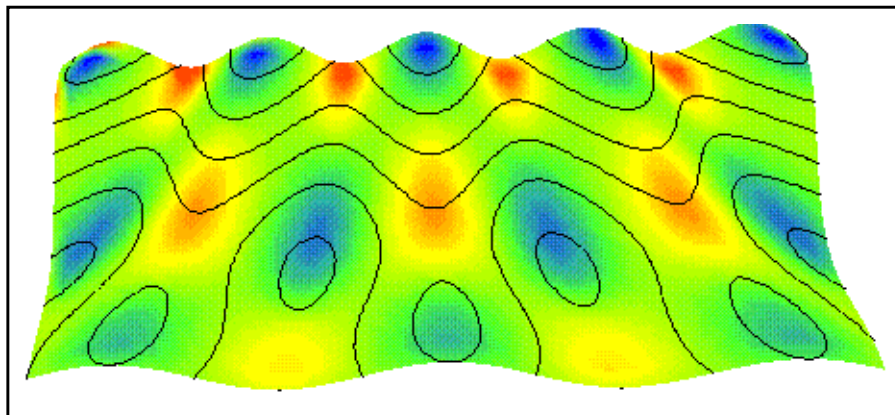
Courtesy of Chevron

Ac

GOCAD Applications continued

Reservoir Applications: Reservoir Modeling, Geostatistical Simulations, etc.
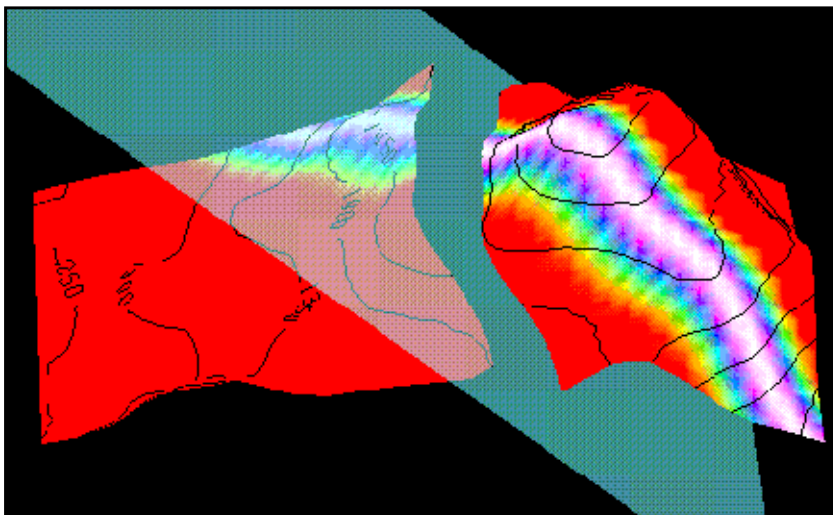
Courtesy of XC Science
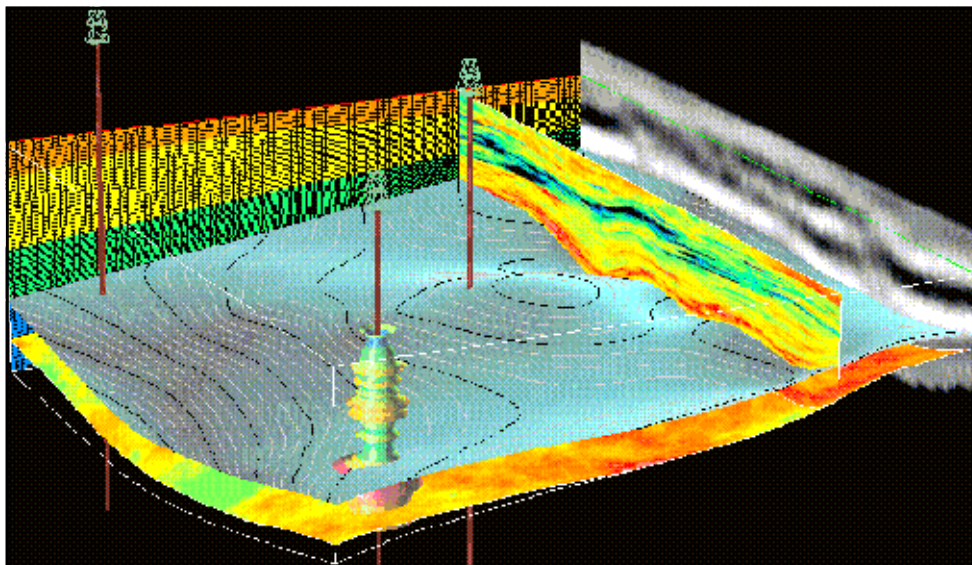
Courtesy of Elf

2.   GETTING STARTED

In this chapter we will provide you with one or more examples for each of the following tasks:

2.1 LAUNCH GOCAD

2.1.1 How?

Find out from your system administrator or your friend what the local command is to launch GOCAD. Let's suppose it is "***gocad***".

2.1.2 Where?

You can launch GOCAD from any directory; that directory becomes the default directory throughout your GOCAD session. However, if you want GOCAD to create a record of the executed commands in your session, you should pick a directory in which you have write privilege. Furthermore, if you are going to run command scripts that involve loading files using relative file paths, you should make sure that you start in the proper directory relative to the script and the data files.

2.1.3 Steps

1. Once you are in the desired directory, say "***GocadInAHurry***", type "***gocad***" (or whatever it should be), and hit Return or Enter.

2. After 3 to 5 minutes, you will see the famous GOCAD window in front of you.

3. We recommend that you enlarge the window to full-screen size to get the maximum viewing area.

2.2 LOAD DATA

Assume that you have a GOCAD file (copied from your friend?) named "***starter.mx***". Of course we are introducing a bad habit here, because "***.mx***" is not a standard GOCAD file suffix; but we use it here to remind ourselves that there is more than one Type of Object in that file.

2.2.1 Which Command?

Main Control Area/File Menu/Load Objects... or Main Control Area/Objects Menu/Load...
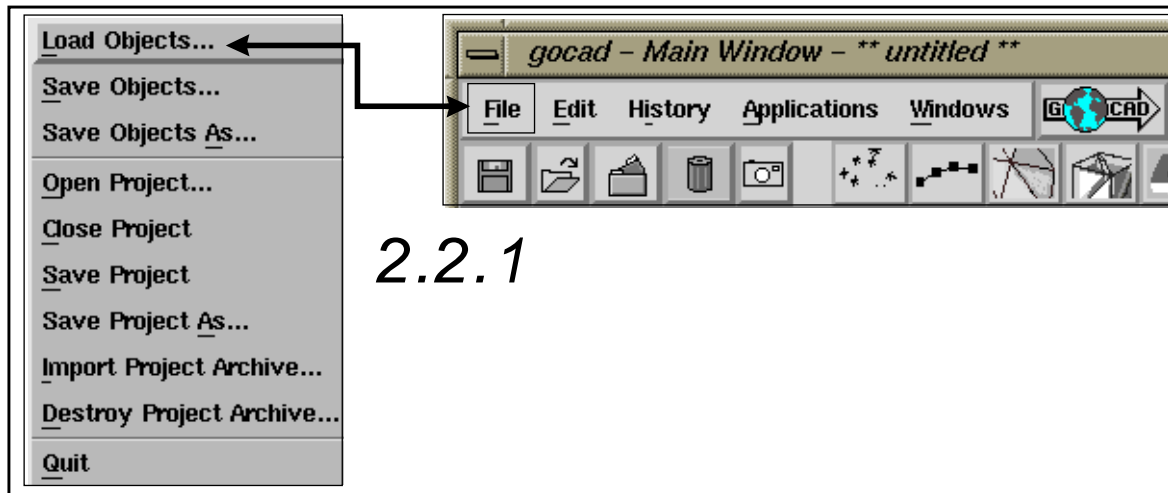
2.2.2 Steps

1. In the ***Main Control Area***, click on the "***File***" Menu to unfold it.

2. In the unfolded Menu, locate the command "***Load Objects...***" and click on it to select it.

3. You will get a pop-up window named ***Gocad - Load Objects*** in front of you. This is a standard

input window used by many workstation applications:

3.a.   If you are far from the directory where the file is, set the proper path in the *Filter* box and hit return or press Filter to get you there fast.

3.b.   Skip this step if the file you want is listed in the *Files* box already. In the *Directories* box, double click on the directory that has the file you want; this will open it.

3.c.   In the *Files* box, click on all the files you want to load at this time. Selected files are highlighted. You can click on selected files to de-select them.

3.d.   Press *OK* if these are all the files that you need. Press *Apply* if you are going to load some more later from another directory.
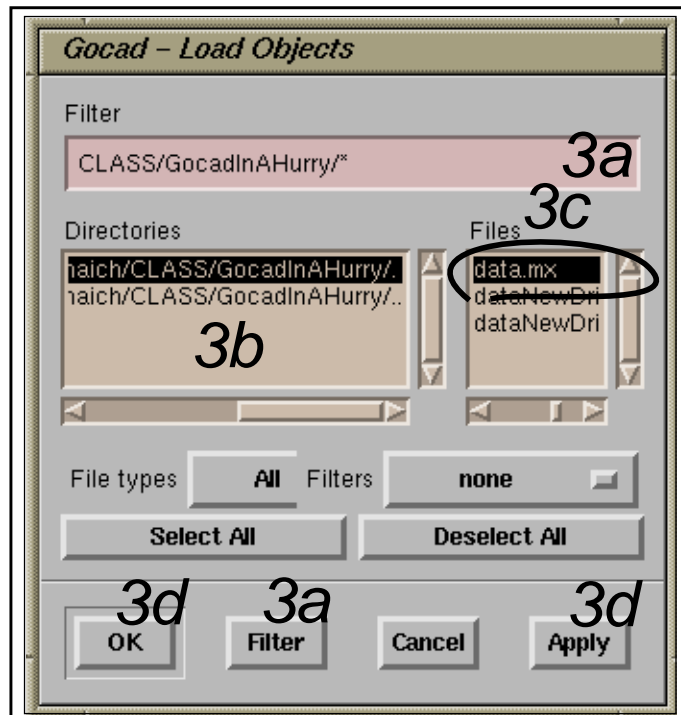
2.2.3 What if I have C-Gocad files?

In theory, the steps in 2.2.2 Steps *(page 2-2)* will read C-Gocad files as well without any extra efforts. However, if you have problems reading a C-GOCAD file, in the *Gocad - Load Objects* window, below the *Files* box, there is a *Filters* (*Converters*) list. Its default is *none*. Click on it to unfold the list and select *old2new*. Then try again.

2.2.4 What if I have non-GOCAD files?

You need to convert them to GOCAD formats. See Appendix B: DATA Formats *(page B-1)*.

2.2.5 What about that "All" button?

That is another filter list. Click on it and you will see all GOCAD Object Types listed. If you select, say *Surface*, then GOOCAD adds "*.ts*" to the *Filter* path and only those files that end with "*.ts*" will be listed in the *Files* box.

2.3 DISPLAY OBJECTS IN THE CAMERA

The file that we have just loaded, "***data.mx***" contains data for 2 **PointsSets**, 16 **Surfaces** and 1 **Voxet**; they are now all listed in the ***Object Display List***.

Displaying Objects in the Camera is very easy, you just select from the ***Object Display List*** what you want to show.

Displayed Objects' names are always highlighted in the ***Object Display List***.

2.3.1 How do I display the Object Bubble?

1. Move the cursor onto the name "**Bubble**" in the ***Object Display List***, and click with the left mouse button.

2. The name **Bubble** is highlighted and the Object **Bubble** is displayed in the Camera.

2.3.2 What if I don't want to display Bubble any more?

You can hide **Bubble** in 4 ways:

1. You can select something else to display; that will remove **Bubble** from the Camera. Or,

2. You can move the cursor onto the highlighted name, **Bubble**, press down the ***Ctrl-key*** and click with the left mouse button; that will also remove **Bubble** from the Camera. Or,

3. You can select the command ***Hide Object*** from the ***Display*** Menu in the ***Camera Menu Area***, then click on **Bubble** <u>IN THE CAMERA</u>.

4. You can select ***Hide All*** from the ***Display*** Menu, which will hide all displayed Objects.
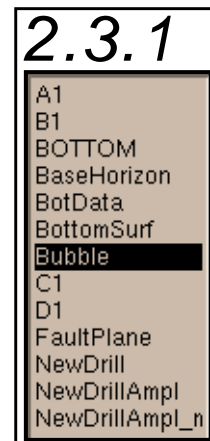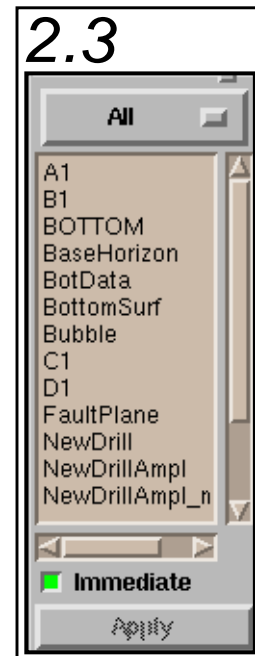
2.3.3 What if I want to display all the Objects?

1. Move the cursor onto the first item on the **List**, **A1** in this case.

2. Press down the left mouse button and drag down the **List** till you have the last item covered.

3. Every item along the way, all 19 of them in this case, will be high-lighted and displayed.

2.3.4 What if I only want to display B1 and C1?

1. Click on the first item, **B1** or **C1**, with the left mouse button.

2. The selected item is highlighted and displayed.

3. Then press down the **Ctrl-key** and click on the other desired item with the left mouse button.

4. The second item is now also highlighted and displayed.

2.3.5 Is there a pattern here?

There sure is.

1. **Click**
    To display a single Object in the Camera, in the **Object Display List**, click with the left mouse button on the name of the Object that you want to display. This not only displays the selected Object, but also removes any currently displayed Objects from the Camera.

2. **Click with the Ctrl-key pressed down**
    This turns a click into a toggle; which means if you click on some-thing that is selected, the click de-selects it; if you click on something that is not selected, the click selects it.

3. **drag**
    If you press down the left mouse button while you move cursor down the **Object Display List**, everything that you come across will be selected and anything that is not covered by your path is not selected or it is de-selected. If you don't want to affect the Objects that are currently dis-played in the Camera, press down the Ctrl-key while you do the dragging.
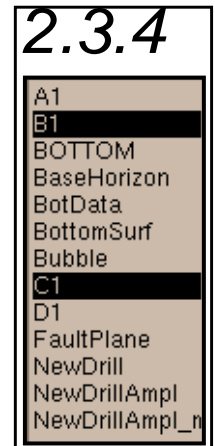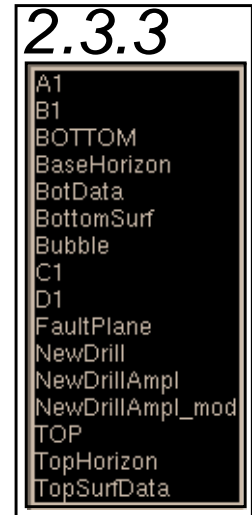
2.3.6 Why is there an All button again?

This is another filter list. Click on it and you will see all GOCAD Object Types are listed. If you select, say, **Surface**, then only Surface Objects will be listed in the Object Display List. This feature is convenient when you have a whole bunch of stuff. However, users often forget that they have selected a specific Object Type and get very mad when they can't find a certain Object on the **List**.

2.3.7 Can you tell which Objects are being displayed?

Yes. The names of displayed Objects are always highlighted in the **Object Display List**.
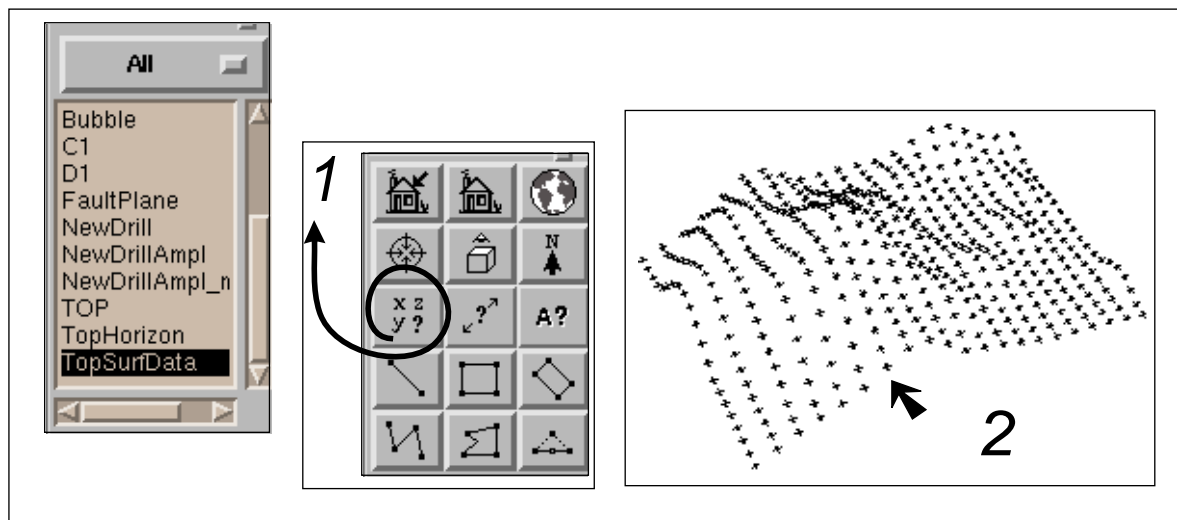
If you want to know the name of a particular Object displayed in the Camera:

1. Click on the **XYZ?** Query Button.

2. The cursor will turn into an up-left black arrow. Click on the Object of interest in the Camera.

3. The name and other information about the clicked Object will be printed in the **Info Area**.

Info: Pick [2000, 900, 1372.22913] target : TopSurfData type = VSet
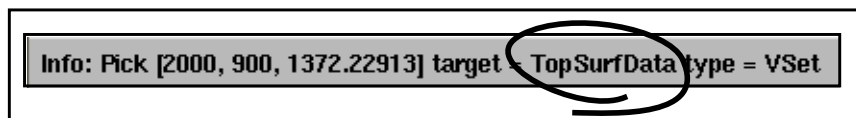
2.4 VIEW OBJECTS IN THE CAMERA

Looking at Objects from different angles in the Camera will be described as **Manipulating Objects in the Camera**. Although the phrase "move Objects" is often loosely used to describe the same function, you must not confuse changing your viewing position (which is what manipulating Objects in the Camera really does) with "moving an Object to a different location in space".

2.4.1 Are there any pre-defined views?
There are three pre-defined views in the 3D Camera.

1.   *Top View*
     This gives you a bird's-eye view of the Objects, like a map view.
     You can access this command (which is in the *View* Menu in the Camera) by pressing the *TopView* Quick Button.

2.   *North View*
     This gives you a "looking into the North" view of the Objects.
     If you have not applied a negative Z-scaling factor to the Camera (see 2.4.3 Can I apply a vertical stretch to see things better? *(page 2-9)*), you will be looking at the XZ plane with Z+ pointing upward and X+ pointing towards the right.
     You can access this command (which is in the *View* Menu in the Camera) by pressing the *NorthView* Quick Button.

3.   *Autosetup*
     This brings the Camera to a GOCAD-computed view that is supposed to give you the best coverage of everything that is currently displayed in the Camera.
     You can access this command (which is in the *Settings* Menu in the Camera) by pressing the *Globe* Quick Button.
     When you are displaying Objects that have very different coordinates, this is a life-saver.
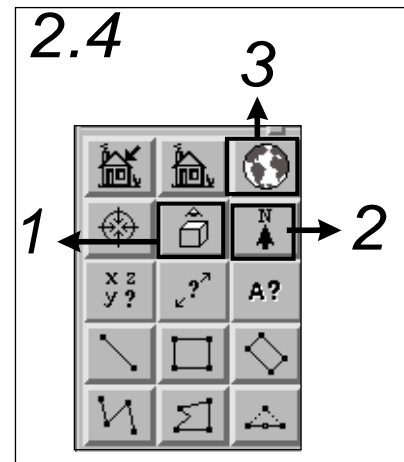
2.4.2 How do I manipulate the Objects?

Three ways:

1. Use one of the Pre-defined view Buttons described in 2.4.1 <u>Are there any pre-defined views?</u> *(page 2-7)*; or

2. Use the ***Camera Movement*** Buttons.

  2.a. Select a ***Movement*** button.

  2.b. Set the ***Increment***.

  2.c. Select the ***Direction***. This is when the movement is carried out.

3. Or use the Shift-key plus Mouse button options:
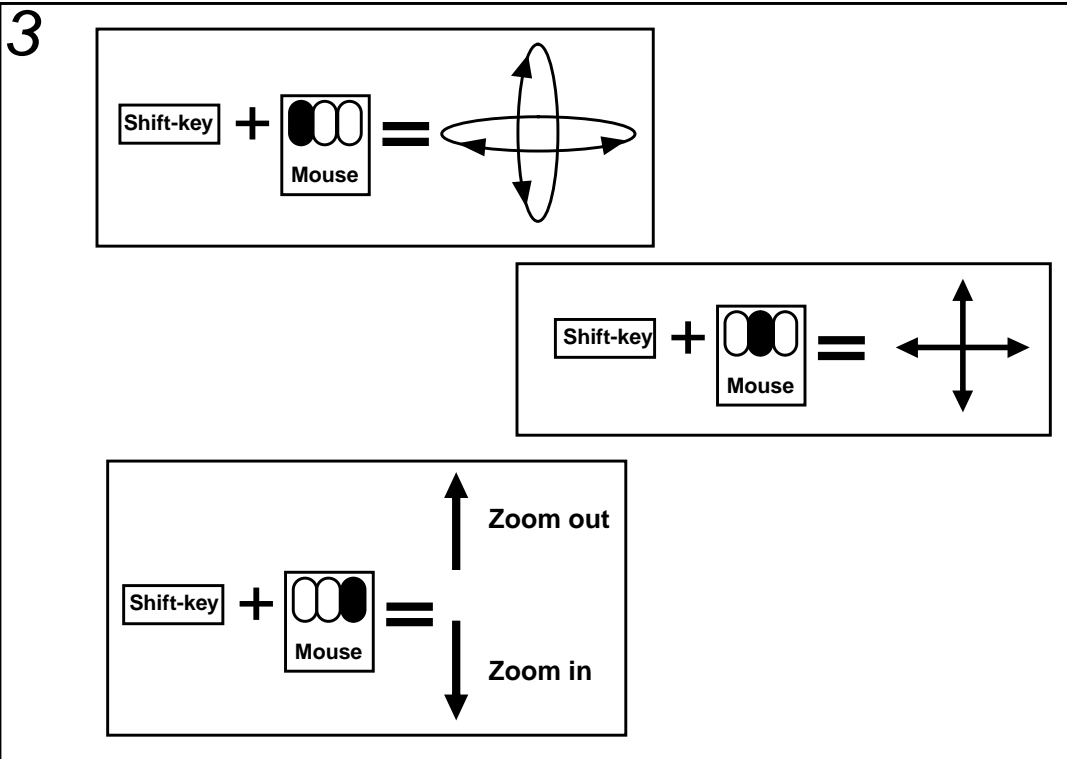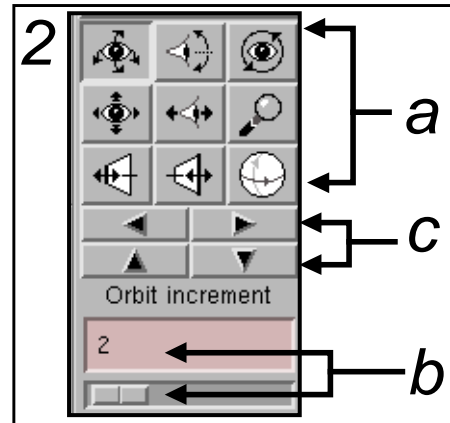
2.4.3 Can I apply a vertical stretch to see things better?

You sure can.

1. In the Camera Menu Area, click on the **Settings** Menu to unfold it and select **Scaling...**

2. The **CameraScale Dialog** window will pop up.

3. The name of the Camera from which you selected this command is automatically entered.

4. To apply a 10X vertical exaggeration, change **sz** to 10; to apply a 20X vertical exaggeration, change **sz** to 20...

5. If you want to stretch the X axis instead, change the **sx** to a larger number, etc.

6. Press **OK** to execute the **Scaling** command.
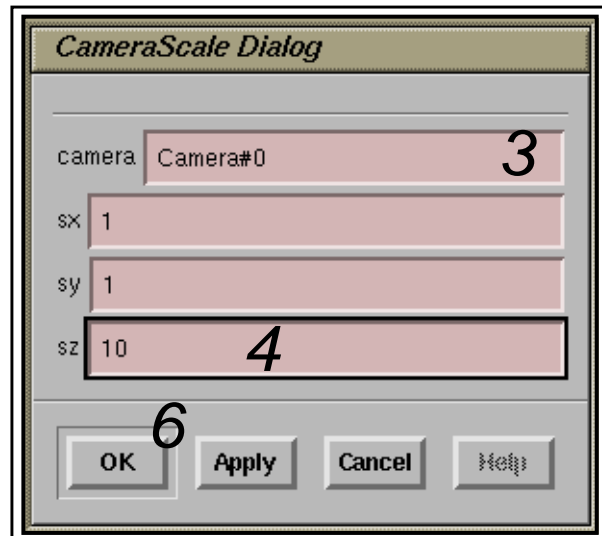
2.4.4 How can I zoom in to a particular spot?

First you need to move the focus point of the Camera to that particular spot (the spot needs to be on an Object): click on the **Focus** Quick Button, then click on the spot that you plan to zoom in on. Now you can zoom in (using the **shift key** + **right mouse button** or the Camera Zoom button) without missing the point of interest.
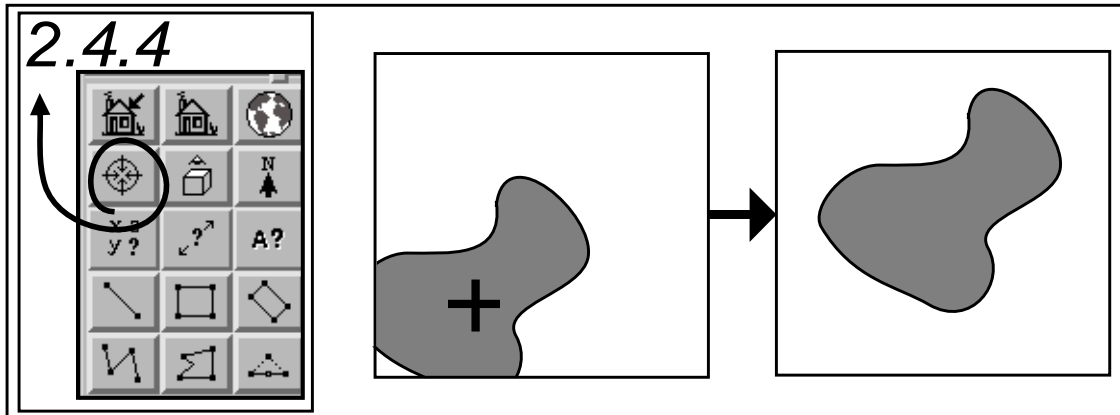
2.5 CREATE AN OBJECT FROM ANOTHER OBJECT

The example we are giving here is to create a Surface from a PointsSet. This is a very common task, since the easiest way to bring in a non-GOCAD file is to modify the file into a PointsSet file.

The PointsSet we are using is **TopSurfData**. The Surface we are creating will be named **TopSurf**.

1. Display only one Object in the Camera, **TopSurfData**.

2. Change the *Operation Menu Bar* to *Surface Mode* by clicking on the *General Mode* to unfold it and selecting *Surface*.
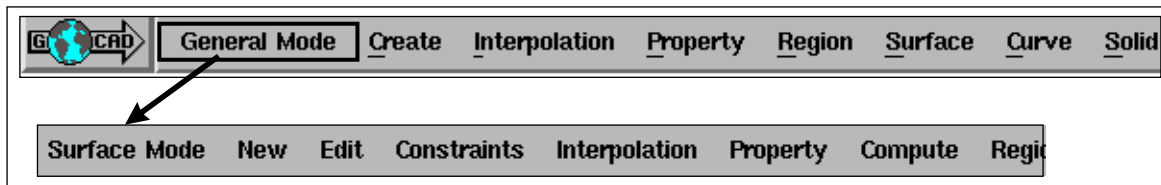
3. Once the *Operation Menu Bar* is updated, click on the *New* Menu to unfold it and select *From PointsSet-PointsSet*.

4. The *TSurfCreateFrom-AtomicGroup Dialog* window will pop up.

5. We will enter **TopSurf** to be the *name* for the new Surface.

6. The PointsSet Object **TopSurfData** is auto-selected as the *AtomsSet points*. (If you don't know why, you should read 2.a.1. Default selection *(page A-5)* in Appendix A: BASICS *(page A-1)*).

7. Leave the other entries in the *Dialog* window at their default values and press *OK*.
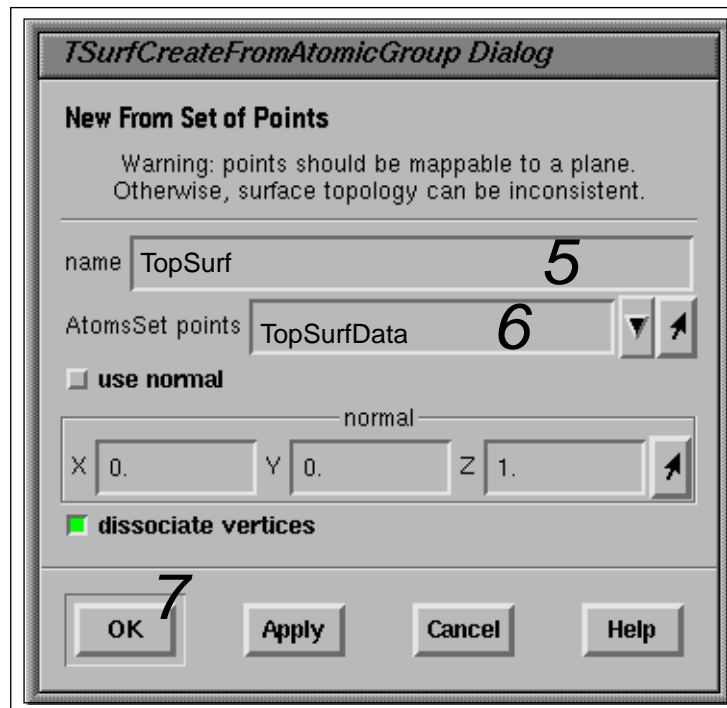
8. The *Dialog* window will go away, and there will be a new Object named **TopSurf** in the *Object Display List*.

9. Add **TopSurf** to the Camera.

10. This command creates a new Surface honoring every input data point, without adding any extra points.

TopSurfData   **TopSurf**
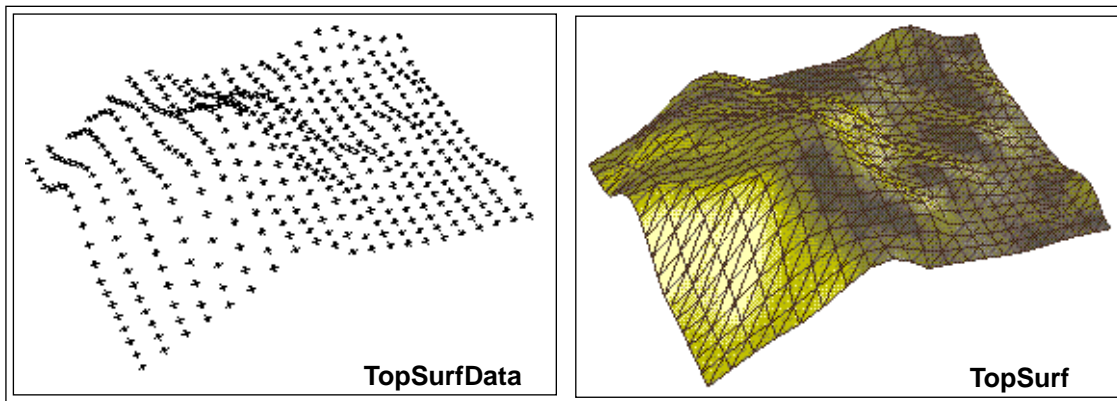
2.6 CREATE AN OBJECT FROM NOTHING

The example we are giving here is to create an SGrid using one of the Quick Create Buttons. This is an easy way to create an SGrid or a Voxet, if you do not require its corners to have precise given coordinates.

The SGrid we are creating will be named **ReservoirSG**.

1. Display only one Object in the Camera, **TOP**.

2. Click on the ***TopView*** Quick Button.

3. Click on the ***Rectangle Create*** Button to unfold the Menu and select ***SGrid***.

4. Move the cursor into the Camera; it is now in a cross-hair shape.

5. Move the cursor towards the lower left corner of the displayed **TOP**, but keep it within **TOP**.
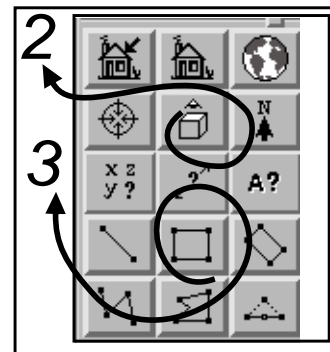
6. Press down the left mouse button and do not release until told to do so. This point you click on will become the **origin** of the SGrid.

7. You will see a yellow rubber band as you move the cursor (with the left mouse button still pressed down). This rubber band depicts the outline of the SGrid you are creating.

8. Move the cursor towards the opposite corner of the displayed Surface until it is near the corner.

9. If you like the outline, release the left mouse button.
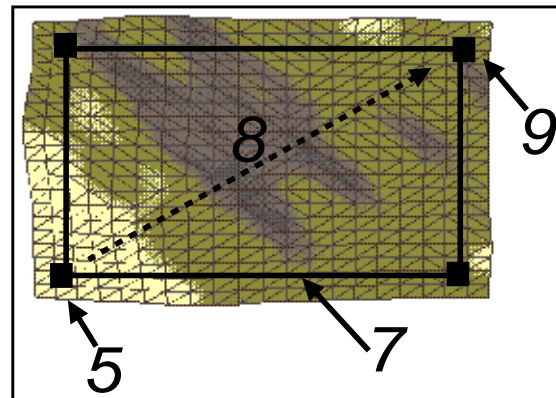
10. The ***SGridCreateFromPoints Dialog*** window will pop.

11. We will enter **ReservoirSG** to be the ***name*** for the new SGrid.

12. The **XYZ** values of the ***origin*** and of the two ***end points*** are all entered for you by GOCAD based on the rubber band that you drew.

13. Enter **10** for the ***cell thickness***.

14. Leave the other entries in the *Dialog* window at their default values and press *OK*.

SGridCreateFromPoints Dialog

Create SGrid From Points

name ReservoirSG ← *10*

origin
X 500   Y 300   Z 1194.80383

point_u
X 9500   Y 300   Z 1194.80383

point_v
X 500   Y 5700   Z 1194.80383

cell thickness 10. ← *13*

nu 10

nv 10

nw 10

☐ **cell centered**

14
OK   Apply   Cancel   Help

15
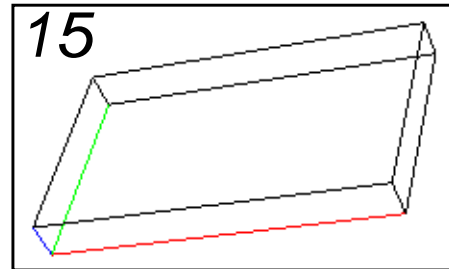
15. Display **ReservoirSG** in the Camera.

16. This command creates a new SGrid using nothing but numerical values entered by the user. In this case, we drew a rubber band to define many of the numerical values.

17. We can't really say that we did not use any other Object to create this SGrid, because in a sense, we used the Surface to help us orient ourselves in the 3D Camera.

2.7 SAVE OBJECTS

GOCAD offers two Save-Object functions.They are both in the ***Main Control Area***, since saving an Object is a generic task:

2.7.1 Default Save

This option allows you to save multiple Objects, but each in its own default file.

1. Display only the Objects you want to save in the Camera.

2. In the ***Main Control Area***, click on ***File*** to unfold the Menu and select ***Save Objects...***.

3. The ***GObjSave Dialog*** window will pop up with all the displayed Objects auto-selected.

4. Press ***OK***.

5. Although you will only get one confirmation message in the ***Info Area***, every selected Object is saved in its own file in the directory where you launched GOCAD. The name of the file is the name of the Object plus the proper suffix. For example, **TopSurf** is saved in a file named "***TopSurf.ts***".
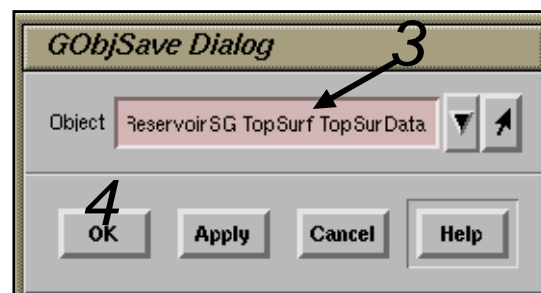
6. If the default file name already exists, GOCAD will ask for confirmation to overwrite the existing file. If you do not confirm, the operation for that Object will be aborted.

2.7.2 Save Objects As...

This option allows you to save one or multiple Objects in a file with a user-specified name.

1. Display only the Objects you want to save in the Camera. It is easier this way.

2. In the *Main Control Area*, click on *File* to unfold the Menu and select *Save Objects As...*

3. The *Gocad - Save Objects As...* window will pop up.

4. In the *Objects to save* box, all displayed Objects are auto-selected.

5. There are also some *Output Filters* and *Output Options* that you can specify.

6. In the *Selection* box, enter the full path or the relative (to the default directory) path of the file in which you want to save the Objects. You can use the *Filter* and the *Directory* box to help you get to the desired output directory.

7. Press *OK*.

8. You will get a confirmation message in the *Info Area* telling you what Objects have been saved and the full file path.

9. If you selected (entered) an existing file, GOCAD will ask for confirmation to overwrite the existing file. If you do not confirm, the command will be aborted.
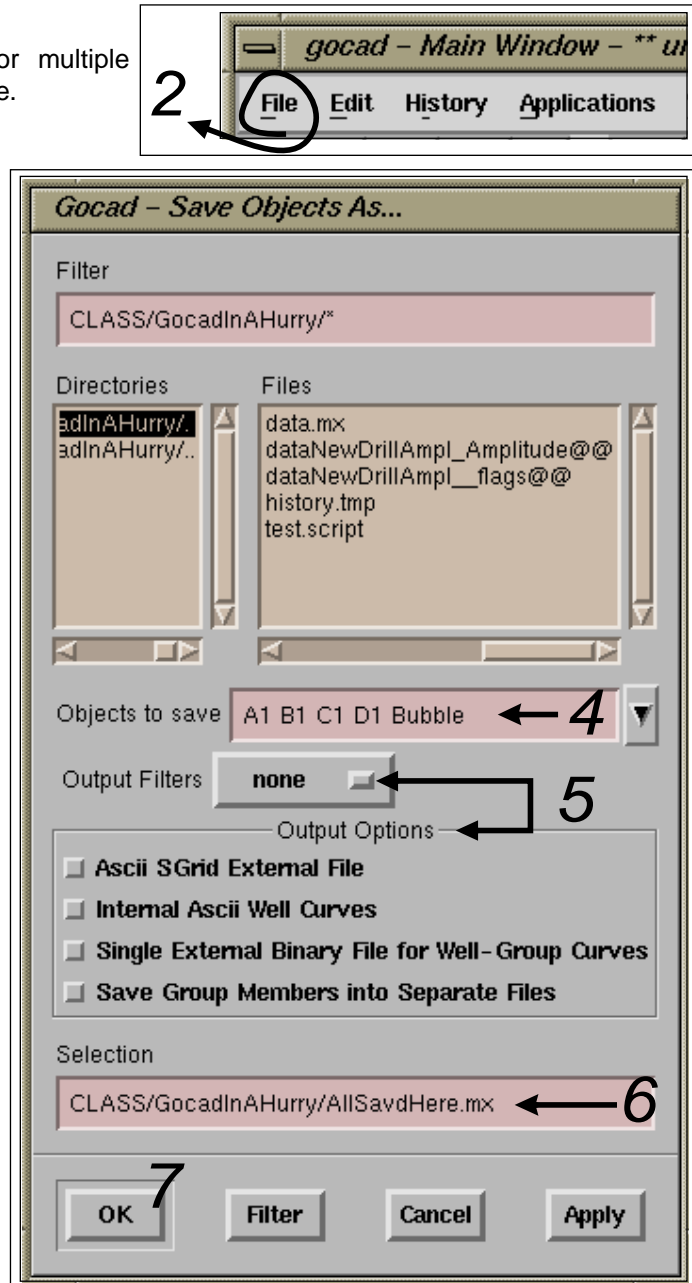
2.8 COPY AN OBJECT

When you copy an Object in GOCAD, you don't have to copy the entire Object:

1. Display in the Camera only the Object you want to copy.

2. In the *Main Control Area*, click on *Edit* to unfold it, and select *Copy.*

3. The *GObjCopy Dialog* window will pop up.
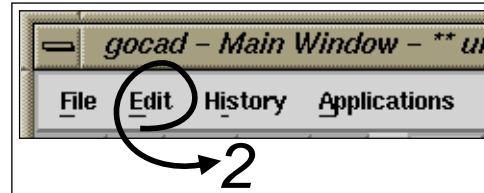
4. In the *Object* box, the displayed Object is auto-selected. Or you can select an Object from the pull-down list or use the pick-arrow to pick an Object in the Camera.

5. *name*
   Enter a name for the new Object (the duplicated one).

6. *copy_style*
   Do you want to copy the original Object's Style?

7. *copy_points*
   Do you want to create a set of Atoms for the new Object (toggle on) or let the new (duplicated) Object share the Atoms of the original Object (toggle off)?

8. *copy_properties*
   Do you want to copy the original Object's Properties (names and values)?

9. *copy_regions*
   Do you want to copy the original Object's Region definitions? Relevant only if the original Object has Regions defined.

10. *copy_constraints*
    Do you want to copy the original Object's constraints? Relevant only if the original Object is constrained.

11. Press *OK*, or *Apply* if you have more to copy or you want multiple copies.

2.9 DELETE OBJECTS

2.9.1 Delete selected Objects

Deleting an Object is a generic (non-Object-specific) task, so this command is found in the Main *Control Area*, under the *Edit* Menu:

1. Display in the Camera only the Objects you want to delete.

2. In the *Main Control Area*, click on the *Edit* Menu to unfold it and select *Delete.*

3. The *GObjDestroy Dialog* window will pop up.

4. In the *Object objects* box, the displayed Object is auto-selected. Or you can select an Object from the pull-down list, or use the pick-arrow to pick an Object from the Camera.

5. Press *OK*, or *Apply* if you have more to delete.

6. All selected Objects will be deleted; their names will disappear from the *Object-Display List*.
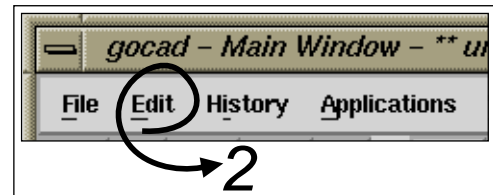
2.9.2 Delete All Objects

To delete all existing Objects in your current GOCAD session:

1. In the *Main Control Area*, click on the *Edit* Menu to unfold it and select *Delete All.*

2. The *Gocad - Warning* window will pop up for your confirmation (since once you delete the Objects, they are gone!)

3. Press *OK* to delete all Objects, or press *Cancel* if you realize that it is not what you want.
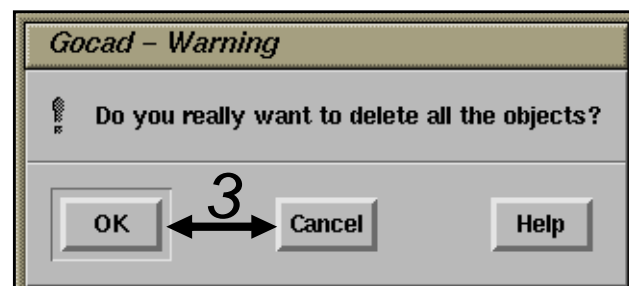
2.10 MODIFY AN OBJECT

Commands that allow you to modify an Object are Object-specific, which means the command to trim a Surface is different from the command that trims a Curve. Therefore, we usually change the *Operation Menu Bar* to a specific *Mode* for convenience. Here, we are going to use a Surface as an example, so we will change the *Operation Menu Bar* to *Surface Mode* to find the commands we need.

2.10.1 How do I change the shape?

Most of the time, you want to change the shape of an Object because you want it to "fit" something else.

In GOCAD, you do this by setting Constraints on an Object. Constraints describe the "modification goal" for that Object. Then you run **DSI**, the *GOCAD Discrete Smooth Interpolator*, to gradually (iteratively) change the Object to honor (fit) the Constraints.
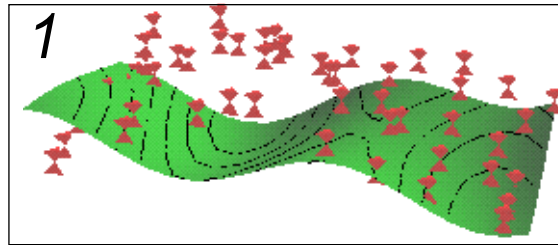
The example we are giving here is to modify an existing Surface, **BottomSurf**, to fit new Well data. The new Well data points are imported as a PointsSet, **BotData**, discrete points in space.

1. Display the Surface **BottomSurf** and the PointsSet **BotData** in the Camera. It is always a good idea to observe the relationship between the two.

2. In the *Operation Menu Bar* (in *Surface Mode*), click on the *Constraints* Menu to select the command *Control Points*/ *Set Control Points*.

3. The *TSurfSetFcp Dialog* window will pop up.

4. For *Surface*, **BottomSurf** is auto-selected because it is the only displayed Surface.

5. *AtomsSet control points* is the Atomic Object that will serve as the controller; its Atoms will specify the desired geometry for the Surface BottomSurf. We will select **BotData** in this case.

6. The *dir_shoot* specifies the "pull" direction: which part of the Surface will each Control point affect. Given the same Surface and the same set of Control Points, different *dir_shoot* values can create vastly different results. In this case, leave it as **(0,0,1)**.

7. Make sure that *optimize shooting direction* is off. if it is on; this option searches for the nearest point on the Surface to "pull".

8. Press *OK*. The Dialog window will go away.

9. You will see a set of blue cubes where the **BotData** points are, and some of them have tails extending onto the Surface. The tails are along the *dir_shoot*. You can tell by the impact point of each tail how each Control Point is going to affect the Surface.
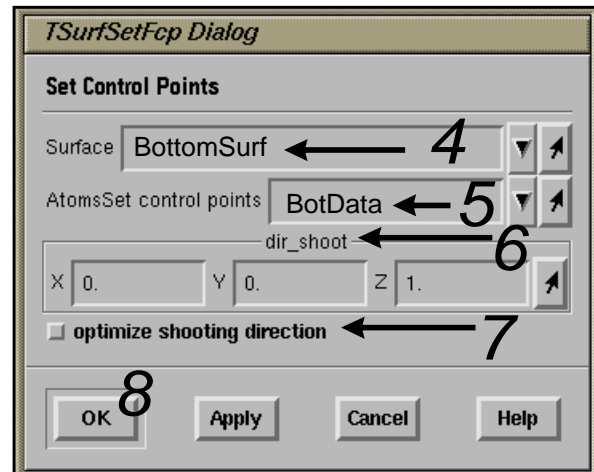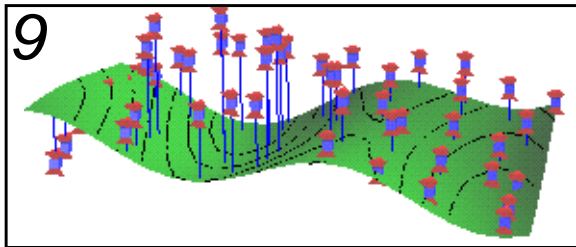
10. In the *Operation Menu Bar*, click on *Interpolation* Menu to select *Fit Geometry/ On Entire Surface*.

11. The *AtomicGroupRunDSICommand Dialog* window will pop up.

12. *AtomsSet* is the Object you want to modify. In our case, select **BottomSurf**.

13. *nbiter* is short for *number of iteration*, since *DSI* is an iterative process. Leave it at **10**. 10 iterations will be run in this execution.

14. *smooth* is a toggle that specifies that smoothing the Surface instead of honoring Constraints should be the goal of this *DSI* execution. We will leave it off, since we *WANT* to honor the Constraints.

15. *conjugate* calls a more complicated routine. Leave this off also.

16. Press *OK*. The window will go away.
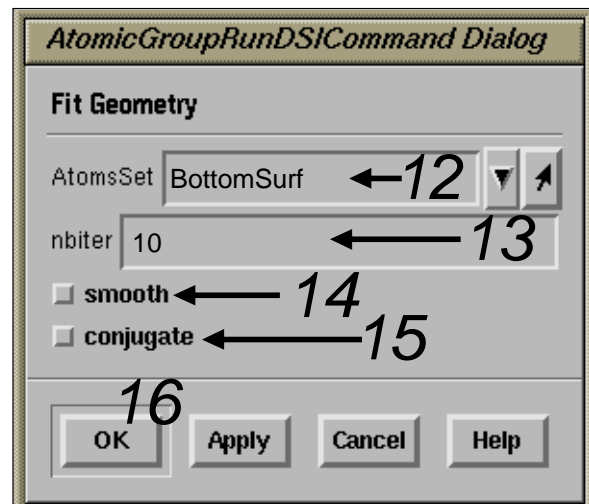
17. After a couple of seconds, you will see that the Surface **BottomSurf** is modified to fit its Control Points, the Atoms in the PointsSet **BotData**.

Note that the outline of the Surface has also changed, because there was no constraint saying that it couldn't.
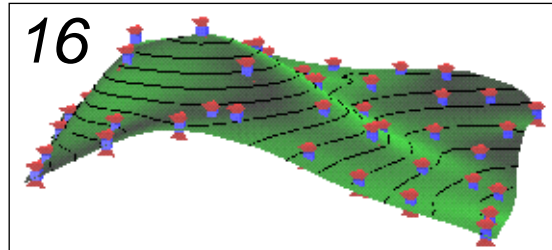
18. You have now seen how to modify an Object to fit another Object. There are more than 30 commands in the Constraints Menu that allow you to set or edit Constraints on a Surface. You can have more than one set of Constraints on a given Surface; for example, we could have added a constraint to maintain the outline of the Surface. These options combined allow you to specify almost any desired shape.

Original Surface and Well Data   **Modified Surface**
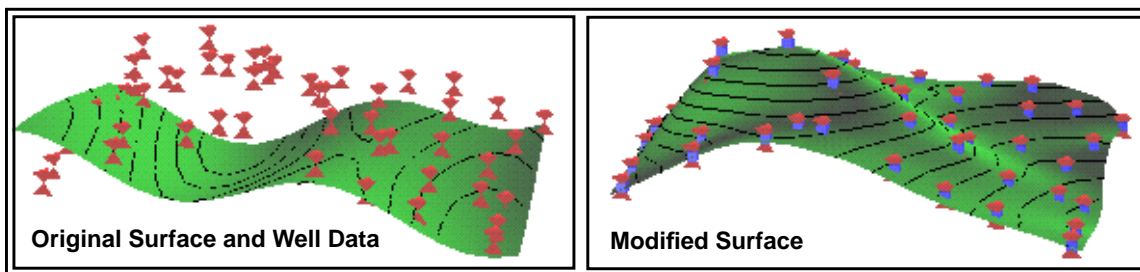
19. Once you are done with modifying an Object, you probably want to delete the Constraints that you have set on that Object.

20. In the **Operation Menu Bar**, click on **Constraints** to unfold the Menu and select **Edit - Remove**.

21. This command does not have a Dialog window. Once you have selected this command, you will notice that the cursor has turned into a cross-hair and there is an instruction in the **Info** area telling you to pick the Constraint you want to delete.

Info: For command: delete_cnstr_link, please select a list (using Extended Selection Mode) of PrjCnstr

This display is chopped off due to paper-width limit.

22. Go to the Camera and click on the Constraint that you want to delete.

23. If your pick was valid, you will see the selected Constraint disappear from the Object.

24. Another way to change the shape of an Object is described in 2.10.5 <u>How do I move it?</u> *(page 2-26)*.

2.10.2 How do I change its color?

Ooo! Wrong place!

Changing the display color of an Object, or anything concerning the display of an Object, is handled by the **Attribute Manager**. See 2.11.1 How do I change the color of a Surface? *(page 2-27).*

In GOCAD, "**Modify an Object**" means changing the substance of an Object, such as geometry, connectivity, Properties, etc.; which is different from "**change Attributes of an Object**", which means changing the "look" of an Object, such as display color, symbols, whether to display triangles, etc.

2.10.3 How do I delete some parts?

To delete part of an Object, you look for the word "**Remove**" or "**Clip**" in the **Edit** Menu in the **Operation Menu Bar**, in the particular **Object Mode**. Be flexible, the word "**Remove**" sometimes appears as "**Kill**" or "**Delete**".

You cannot delete just a portion of a Voxet, an SGrid, or a Well.

In Atomic Objects, you can delete an Atom (a **Node**), an element (a **Triangle** in Surfaces, a **Segment** in Curves and a **Tetrahedron** in Solids), or a piece of connected elements called a **Part** (a **TFace** in Surfaces, an **ILIne** in Curves, and a **TVolume** in Solids). **Remove** deletes the selected **Part** and **Clip** preserves the selected **Part** (deletes everything else).

Deleting elements from the Surface BaseHorizon

| Original Surface | Deleted a Triangle | Deleted a Node |
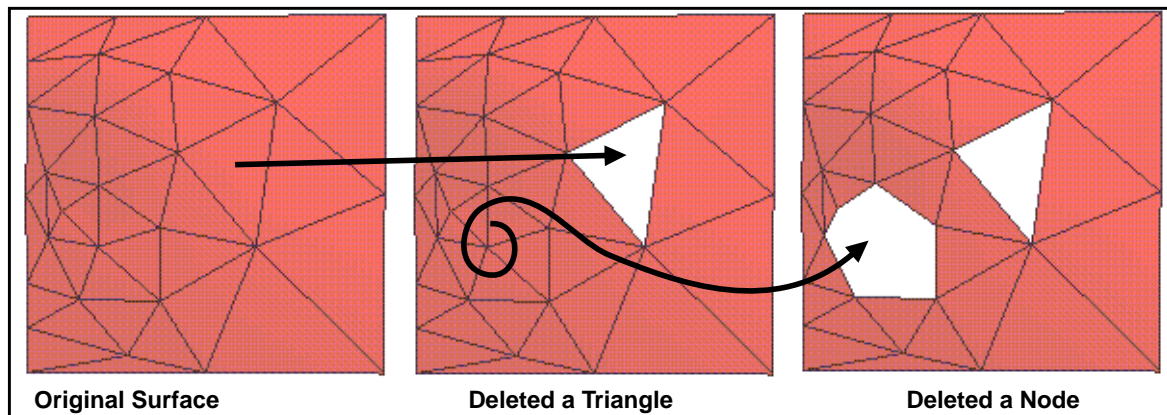
2.10.4 What if the Triangles that I want to delete do not from an isolated piece (TFace)?

To delete an Atom, or a Triangle (Segment, Tetrahedron), all you have to do is select the right command (**Remove Node**, **Remove Triangle**) then click on the element that you want to delete.

To delete a whole bunch of stuff, e.g. a chunk of a Surface, you need first to isolate that portion. There are two ways to separate the portions.

If you have a Horizon, say **TopHorizon**, that is supposed to be truncated by a fault, say **FaultPlane**, you would want to remove any "extra pieces" beyond the fault plane:

1. Display the Horizon Surface and the Fault Surface in the Camera. For the following operation to work, the Fault Surface must not end within the Horizon Surface (the fault must cut the Horizon from edge to edge).

2. Toggle on the **Mesh** display in the **Attribute Manager** for **TopHorizon** so you can see its Triangulation.

3. In the **Operation Menu Bar** (in **Surface Mode**), click on the **Edit** Menu and select **Cut By Surfaces**.

4. The **TSurfCutBySurfaces Dialog** window will pop up.

5. The first **Surface** is the Surface you want to cut, the Horizon Surface. In our example, **TopHorizon**.

6. The **Surface by** is the Fault that you want to cut the Horizon with; in our example, **FaultPlane**.

7. **build_constraints** tells GOCAD to set specific types of Constraint (e.g. **Set Border On Surface**) on the new Border(s) once the cut
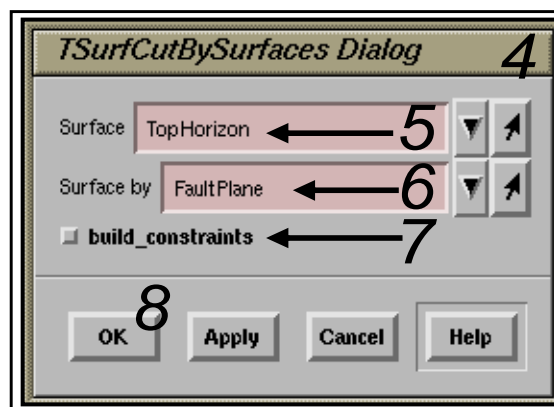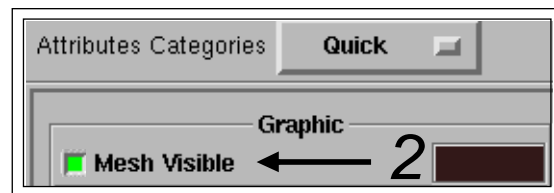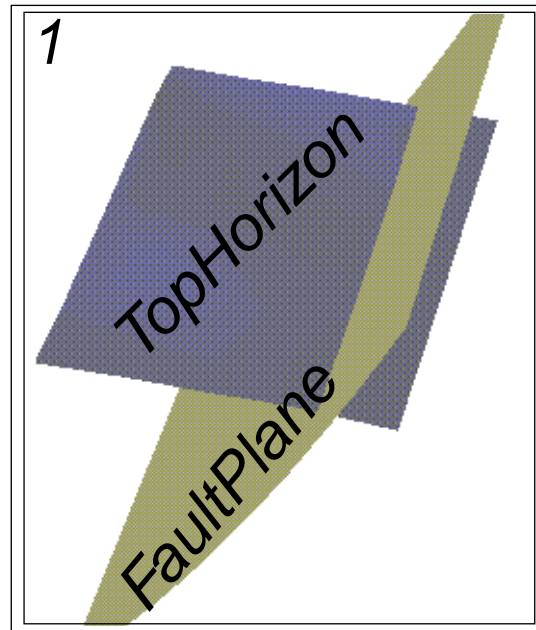
is performed. If you are going to further modify the Surface, you should leave this toggle on. If all you want to do is to get rid of that extra piece beyond the Fault, toggle this off, which we will.

8. Press **OK**. The Dialog window will go away.

9. After a couple of seconds, you will see that the Mesh of the Surface **TopHorizon** has changed. There is now a cut line along the intersection with **FaultPlane**.
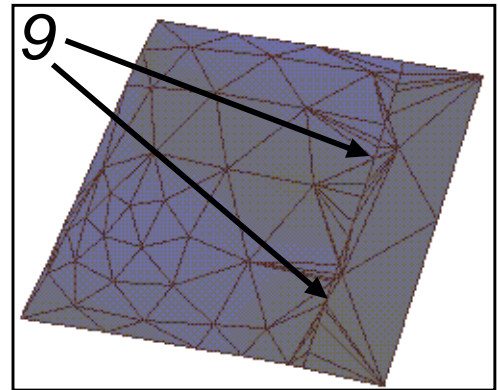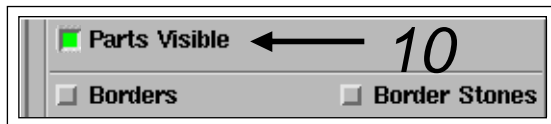
10. In the **Attribute Manager** (select **TopHorizon**), in the **Quick Category**, toggle on **Parts Visible**. If the **cut** operation was performed properly, you will see the single Surface now appear in 2 different colors, indicating the cut operation has cut the continuous Surface into two disconnected pieces (TFaces).

11. You can now go to the **Operation Menu Bar** and click on the **Edit** Menu to unfold it and select "**Part/ Remove One Part**" or "**Part/ Clip**".
What is the difference between these two commands? "**Remove**" _DELETES_ the piece you subsequently click on, while "**Clip**" _KEEPS_ the piece you subsequently click on (and deletes everything else in that Surface).

Before   After
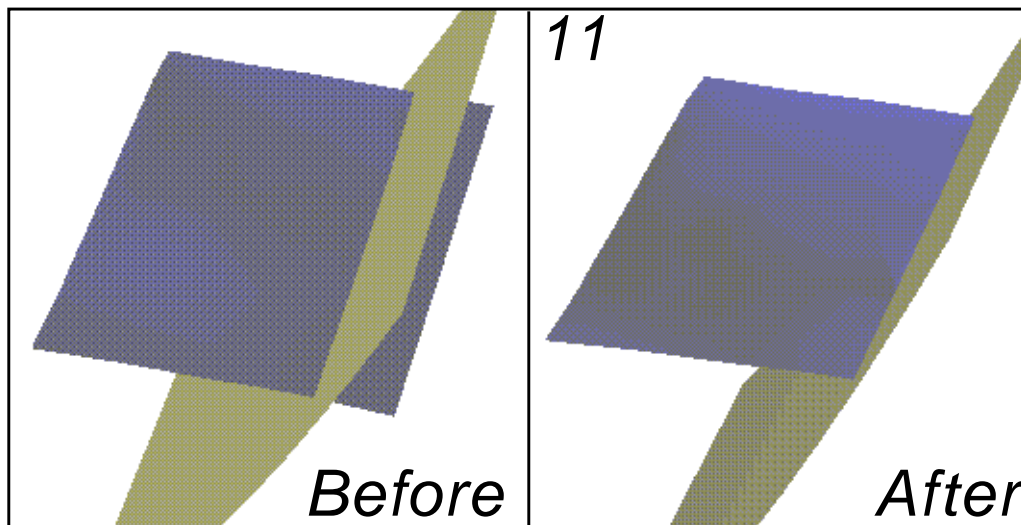
If you just want to manually cut off some parts that you don't want, you can use the **Scissors** option in two of the **Create** Buttons. Remember however, this is a cookie-cutter cut, which means it penetrates the Object all the way through, not just the part of the Object that is visible.

1. Display the Object you want to trim in the Camera.

2. You probably want to change the Camera View to **Parallel** (**Camera** Menu Bar/**View** Menu/**Parallel** toggle) to avoid distortion.

3. Move the Camera (or the Object) until you have an unobstructed view of the part you want to cut.

4. There are two **Scissors** options, **Open Cut** and **Closed Cut**. Currently, the **Closed Cut** is working a lot better than the **Open Cut**.

5. Click on the **Open Cut** or the **Closed-Line** Button to unfold the Menu and select **Scissors**.

6. Move the cursor into the Camera and click at the beginning of the desired cut line. Currently, the first click must be on the Object you want to delete, in order to register the cut. See the figure below.
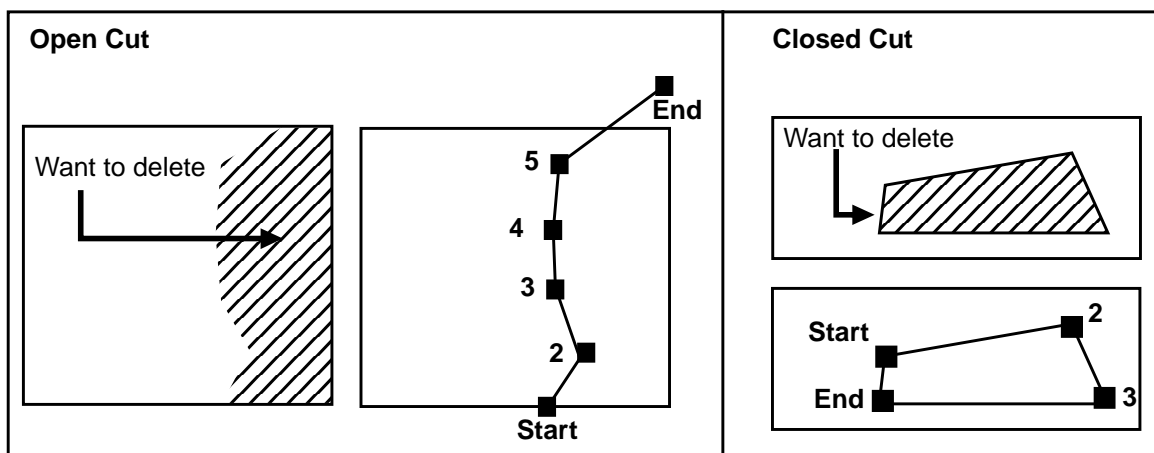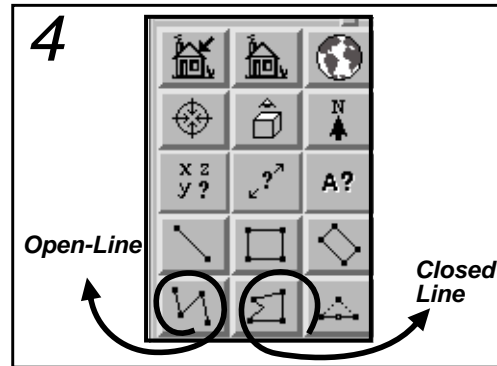
7. Continue to click along the desired cut line till the rubber band depicts the desired cookie cutter (you can use the right mouse button to delete the previous click), then click with the middle mouse button to end. Don't click on the start point to end.

8. Once you click with the middle mouse button, the *Scissors confirm* window will pop up for your confirmation to cut.
You can press *OK* to execute the Cut, or you can press *Cancel* to abort the operation.

9. When the cut operation is completed, in the *Attribute Manager* (select the Surface), in the *Quick Category*, toggle on *Parts Visible*.

If the cut operation was performed properly, you will see that the single Surface now appears in 2 different colors, indicating the cut operation has cut the continuous Surface into two disconnected pieces (TFaces).
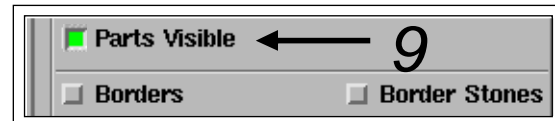
10. You can now go to the *Operation Menu Bar* and click on the *Edit* Menu to unfold it and select "*Part/Remove One Part*" or "*Part/ Clip*". What is the difference between these two commands? "*Remove*" <u>DELETES</u> the piece you subsequently click on, while "*Clip*" <u>KEEPS</u> the piece you subsequently click on (and deletes everything else in that Surface).
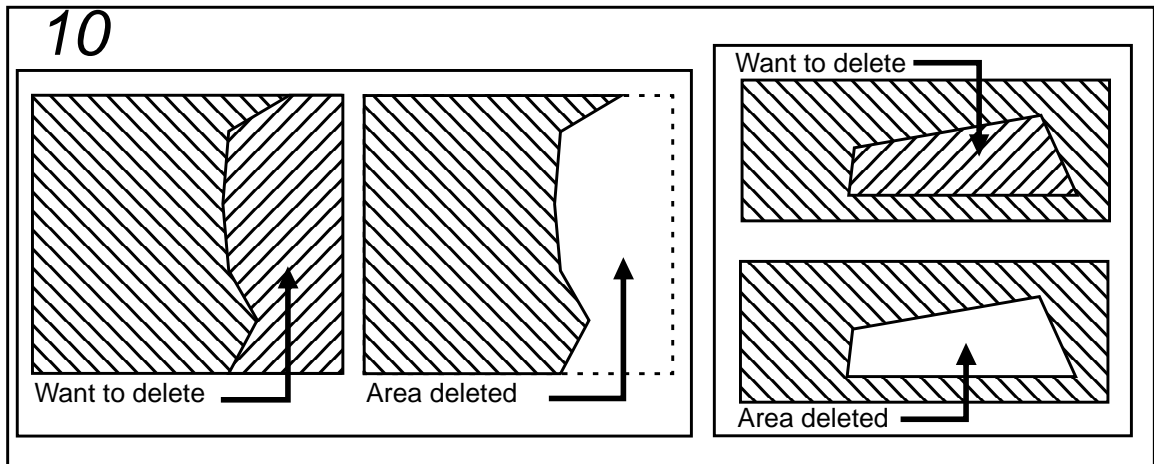
2.10.5 How do I move it?

To move an Atom or a Part of an Object, you can either specify the new position of that Atom, specify the vector by which you want to move that Atom, or simply drag the Atom with the left mouse button to its new location.
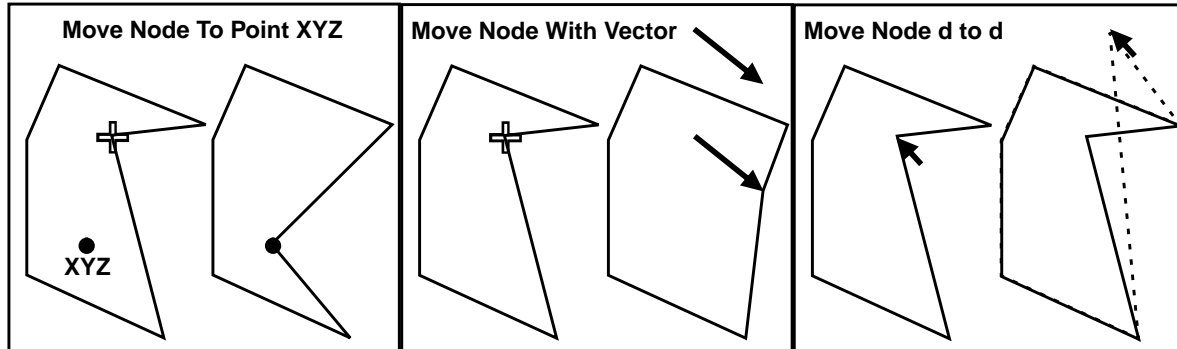
To move an entire Object, say **BottomSurf**, by a certain amount, say 2000 in the X direction, 500 in the Y direction and 1000 in the Z direction:

1. Display the Object you want to move in the Camera.

2. In the **Operation Menu Bar** (any **Mode**), click on **Property** to unfold it and select **Apply Script/ Everywhere...**.

3. The **GObjApplyScriptOnProperty Dialog** window will pop up.

4. **Object** is the Object you want to move, **BottomSurf** in this case.

5. In the slate area, type: **{X=X+2000; Y=Y+500; Z=Z+1000;}**

6. Press **OK**.

7. You will see the Object **BottomSurf** appear at its new location.

8. Had you typed in **{Z=1000;}** instead, you would have flattened the Surface **BottomSurf**.
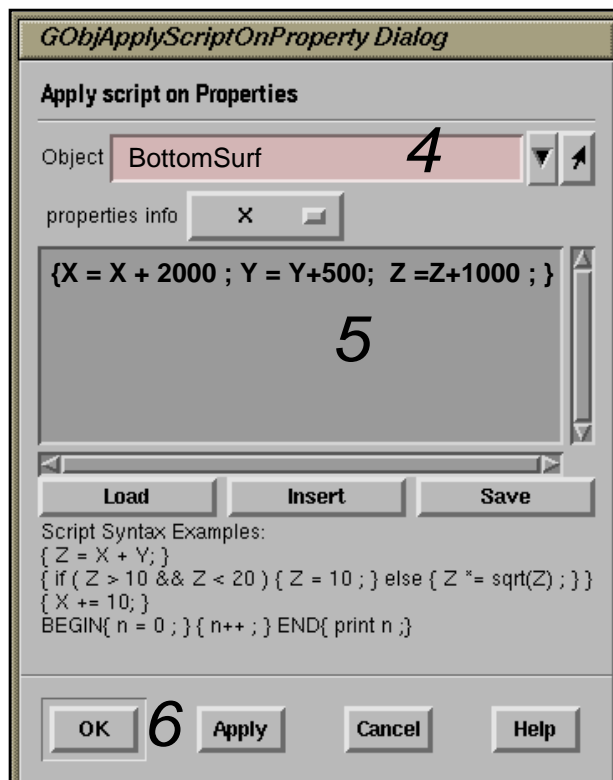
2.11 CHANGE THE LOOK OF AN OBJECT

In GOCAD++, everything concerning the "look" of an Object (we call it the **Style** of an Object) is controlled by the **Attribute Manager**. An individual item in a **Style** is called an **Attribute**.

When you use the **Attribute Manager**, these are the steps:

1. Select the Object in the **Attribute Manager**.

2. Determine which **Attribute Category** contains the Attribute(s) you want to change. This is usually very intuitive.

3. Find the corresponding Attribute(s) and change the value(s).

2.11.1 How do I change the color of a Surface?

We will use the Surface **BottomSurf** as an example.

1. Select **BottomSurf** in the Attribute Manager. The default **Attribute Category** is **Quick**, which contains the most frequently changed Attributes, including **Surface Color**.

2. Change the **Attribute Category** to **Graphic**.

In the **Graphic** Category, you will find all the basic components of an Object (Surface in this case).

You can toggle on or off the display of each component, change its display color, symbol, or size, etc.
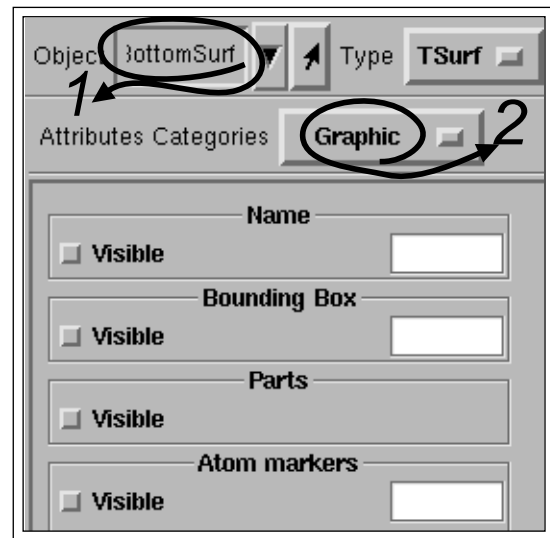
2.11.2 How do I display a Property on a Surface?

We will again use the Surface, **BottomSurf**, as an example.

1. Select **BottomSurf** in the Attribute Manager.

2. Click on the **Attribute Categories** List (default is **Graphic**) and, well, there is a **Property Category**. Select that.

3. Toggle on the first item, **Visible**.

4. The default **Atomic Property, Z,** will immediately be displayed on **BottomSurf** in the default **Colormap**, **grey**.

5. To display the Property your want:

   5.a. Click on the **Atomic Property** list to select the Property; **OldDepth** in this case.

   5.b. Click on the **Colormap** list to select the Colormap you want; **rainbow1** in this case.

   5.c. The **Low clip** value is automatically calculated to be the minimum value of the selected Property. If you prefer, you can enter the value you want.

   5.d. The **High clip** value is automatically calculated to be the maximum value of the selected Property. If you prefer, you can enter the value you want.

   5.e. There are a lot more stuff you can play with in this Category, such as **transparency**, **No-Data points**, etc. Can't learn everything in a hurry.
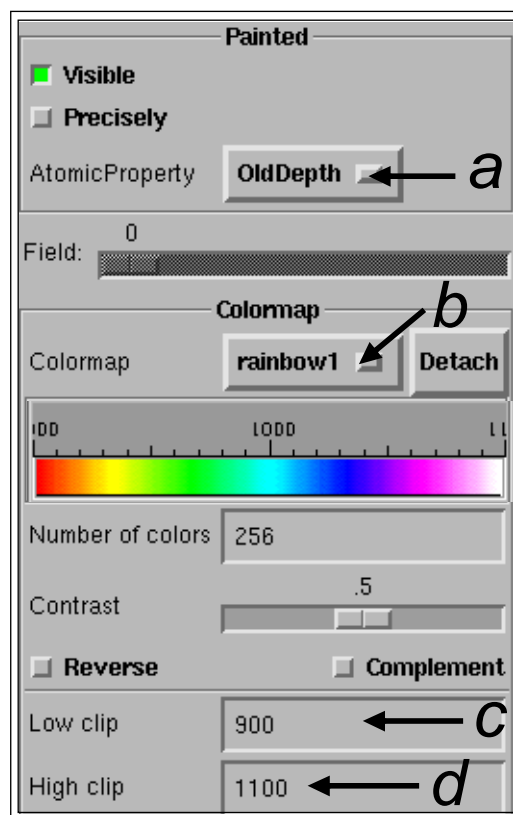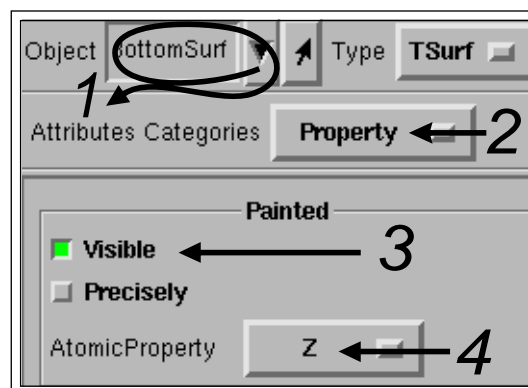
6. The display of **BottomSurf** is updated accordingly, as soon as you make your changes.

6

Without Property

With the Property OldDepth displayed in rainbow1
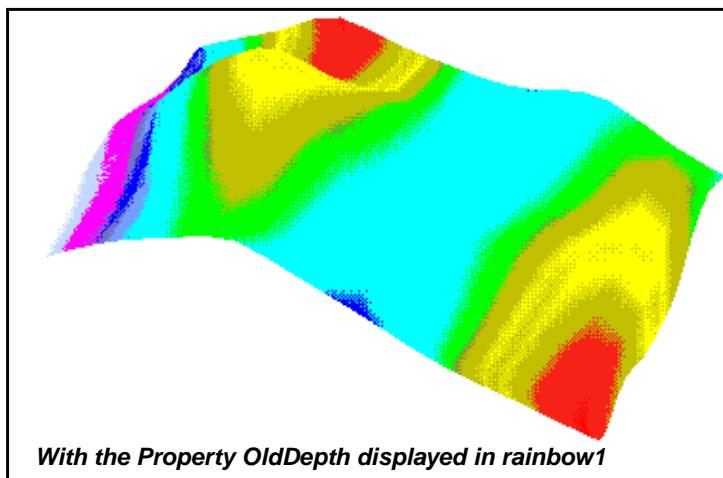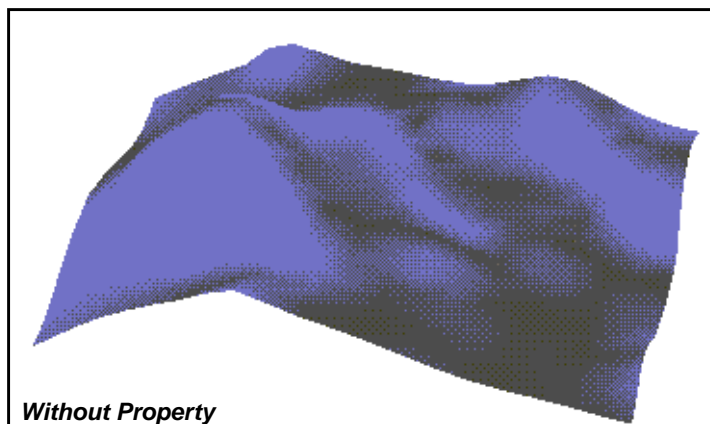
2.11.3 How do I display contour lines on a Surface?

We will use the Surface **Bubble** as an example.

1. Select **Bubble** in the Attribute Manager.

2. Click on the **Attribute Categories** List (default is **Quick**) and, well, there is a **Contours Category**. Select that.

3. Toggle on the first item, **Visible**.

4. Select the Property that you want to contour. Z, for example.

5. The easiest way is to display contours press the **Add Default Contours** button. GOCAD will calculate and display two sets of Contour lines (in Z) that best fit your Object.

Without Contours

With 2 default Contour sets
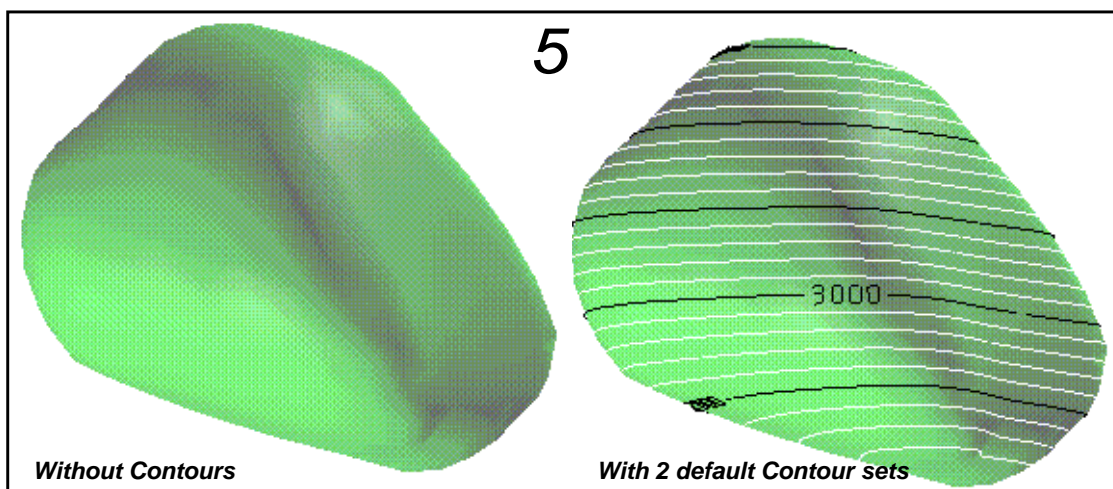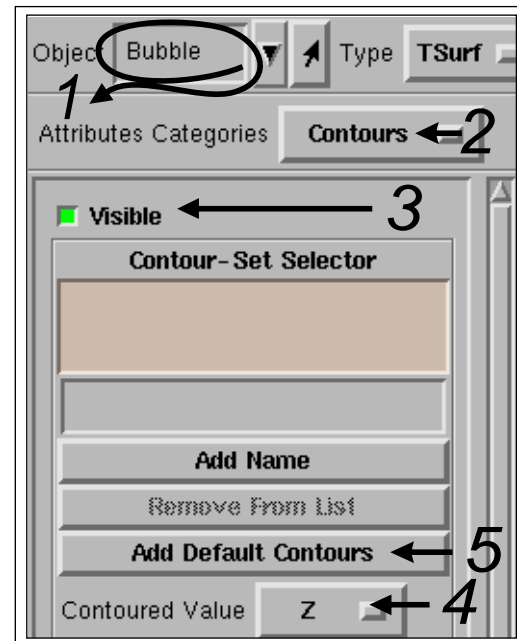
6. To create your own:

6.a. Enter a name, something descriptive, for the new Contour Set, in the Entry area.

6.b. Press *Add Name*. The new name will appear on the **Contour-Set Selector** and will be highlighted.

6.c. Select the Property that you want to contour from the *Contoured Value* list (default is **Z**).

6.d. Specify the *Contour interval* you want (keyboard-entry).

6.e. Specify the value of the first contour line in *Contour Shift* (default is **0**). If you enter 157 and your contour interval is 500, you will be getting contour lines at 157, 657, 1157, 1657,etc.

6.f. The default color for contour lines is black. If you want to change that, click on the Color square to call up the *Gocad - Color Chooser* window and select the color you want.

6.g. The default contour line thickness is 1. If you want to change that, use the slider bar.

6.h. By default the Contour lines are labeled. If you don't want the lines labeled, toggle off *Labeled*.

6.i. To toggle off individual sets of Contours, toggle off the lower *Visible*.
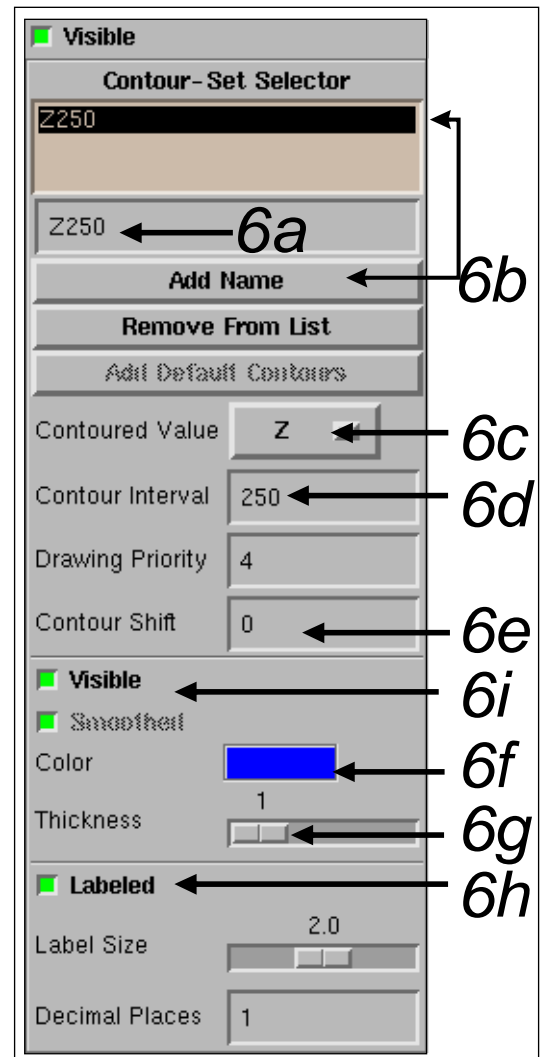
2.11.4 How do I display seismic data on a Surface?

Texture mapping!

You do need to have the Seismic Data stored as a Property in a Voxet (usually a single-layered Voxet for this purpose).

You could always import the seismic amplitude as a Property of the interpreted Surface, but then the resolution of the amplitude values would be controlled by the triangulation density of the Surface. You need a lot fewer points to keep an accurate picture of the shape of the Surface than you would need to keep a reasonable resolution of the seismic amplitude. Therefore, we will use texture mapping, which allows high-resolution property data to be draped onto low-resolution triangles.

To texture-map (drape) a Voxet (**NewDrillAmpl**) Property on a Surface (**NewDrill**):

1. Display the Surface and the Voxet in the Camera.

2. Take a *Top View* at the two Objects. Only the portions of the Surface that are covered by the Voxet will have values draped onto them.

Bottom View (left) and west View (right) of the Voxet and the Surface
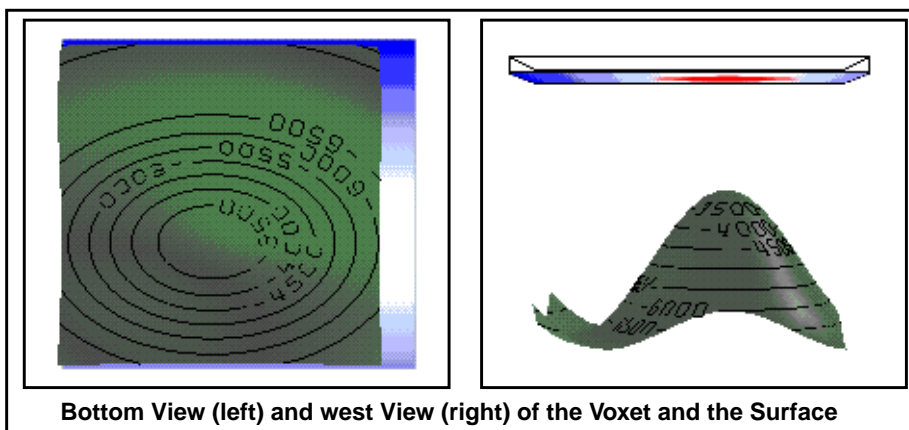
3. Select **NewDrill** in the Attribute Manager.

4. Click on the *Attribute Categories* List (default is *Graphic*) and well, there is a *Texture Category.* Select that.

5. Go to the lower section, *Draping*.

6. In the *Voxet* box, click on the downward triangle to select the Voxet that has the seismic data; in this case, **NewDrillAmpl**.

7. When you move the cursor away from the *Voxet* box, the *Property* list will be updated. Select the Property that is the seismic amplitude; **Amplitude** in our case.

8. You then need to specify which Section of the selected Voxet is to be draped onto the Surface. In our example, the Section is along *axis-3*.

9. Since our Voxet is a single-layered Voxet, once we have selected *axis-3*, the *Index slider bar* is frozen to **0**, because there are no other choices.

10. Toggle on *Visible*.

11. You will see **Amplitude** from **NewDrillAmpl**, **Section (3,0)** is now draped onto the Surface **NewDrill** (the contour display is turned off in the figure below).
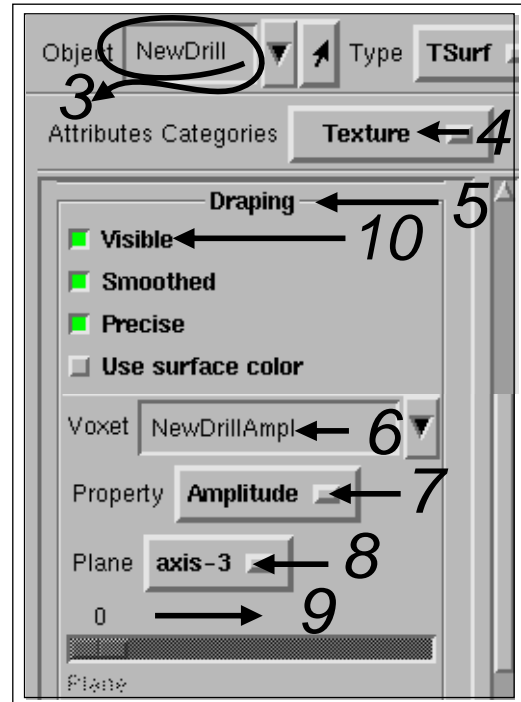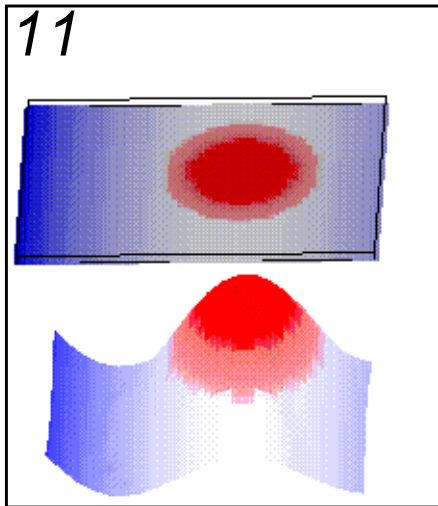
2.12 EXIT GOCAD

But why would you want to?

1. In the **Main Control Area**, click on **Project** to unfold the Menu and select **Quit**.

2. GOCAD will politely ask you to confirm your decision.

3. If you press **OK**, you are out.

4. If you decide to stay, press **Cancel**.
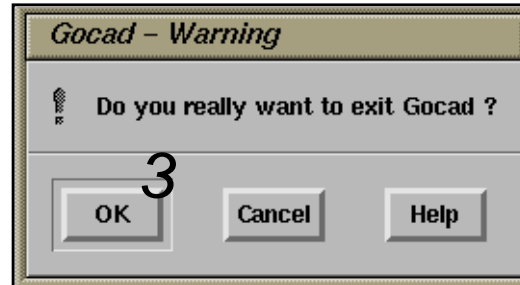
Appendix A: BASICS

This chapter covers the basic knowledge you need to be comfortable using GOCAD. Here are the contents of this chapter:

A.1 GOCAD WINDOW
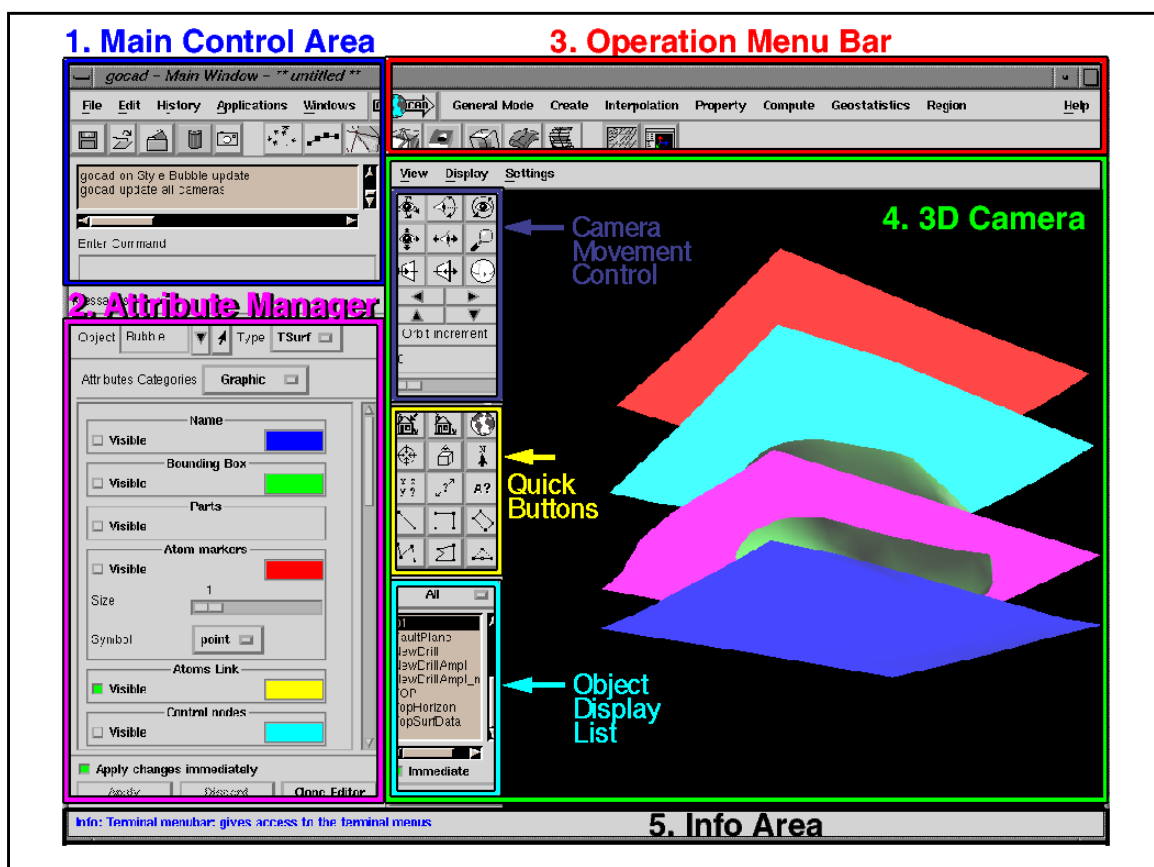
The is what the GOCAD composite window looks like. We roughly divide it into the following areas for clearer communication:

A.1.1 Main Control Area

In the upper left hand corner of the composite window is the **Main Control Area**.

This is where you communicate with GOCAD. The full command that you send to GOCAD when you push a button will appear in the display area (this area has no special label). Error messages from GOCAD are printed in the area labeled "Messages:" (and in the *Info Area* *(page A-7)*).

The Menus in the Menu Bar here contain commands that deal with the "big-picture" stuff: loading data, outputting files, deleting Objects, opening new Windows, and recording the History of the current GOCAD session.

Commands that you find here are generic; they can be applied to any Type of Object. Commands that deal with a specific Type of Object are called from the *Operation Menu Bar* *(page A-4)* area.

Also found here are the
1. *History Display Area,*

2. *Quick Access Icons,*

3. *Command Entry Area* and

4. *Message Indicator.*
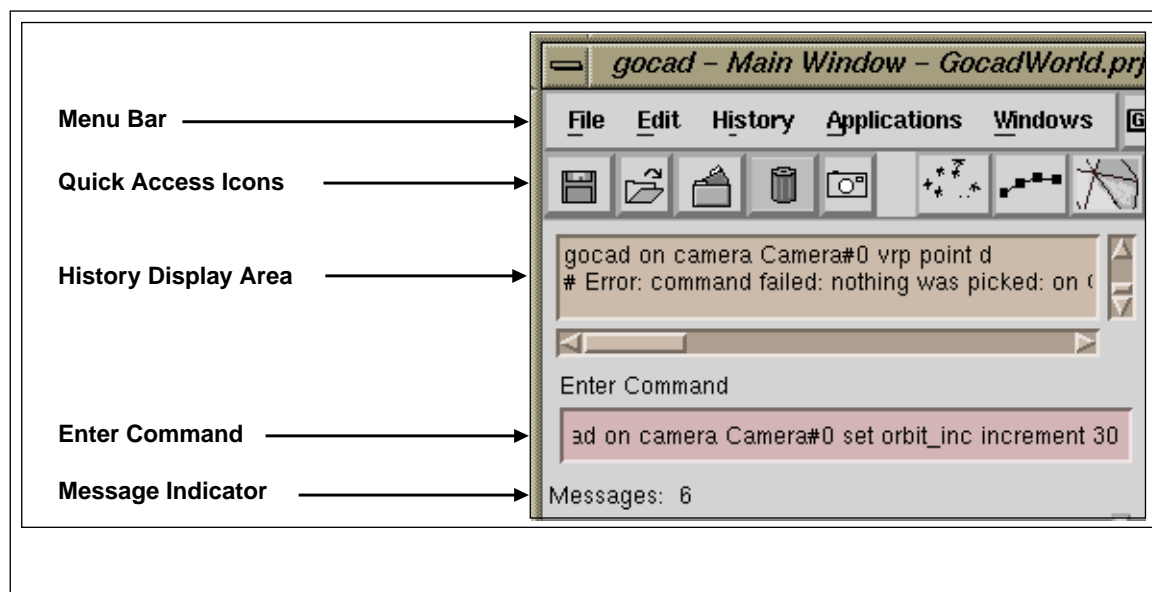
A.1.2 Attribute Manager

In the lower left hand corner is the Attribute Manager.

The way an Object behaves is called the Style of the Object. Detailed items in the Style are called Attributes. In the Attribute Manager, you can modify or assign Attribute values, and thus change the appearance of an Object.

Attributes of an Object are categorized for easy access. For example, a Surface Object has 5 Attribute Categories: Graphic, Property, Constraints, Contours, and Texture. Attributes that deal with contour line displays, for instance, are kept in the Contours Category.

In the Attribute Manager, individual Attribute items are displayed and edited in one of the following 5 ways:

1. **Toggle**
    The parameter has only two possible values, on of off.

2. **Color Selection**
    A color square, which when clicked on will bring up the color selection window, allowing the user to select a new display color for that component.

3. **Slider Bar**
    Allows the user to select a value within a fixed range.

4. **Selection List**
    A pull down list containing choices for the user to select.

5. **Keyboard Entry**
    The user enters the specific value from the keyboard.
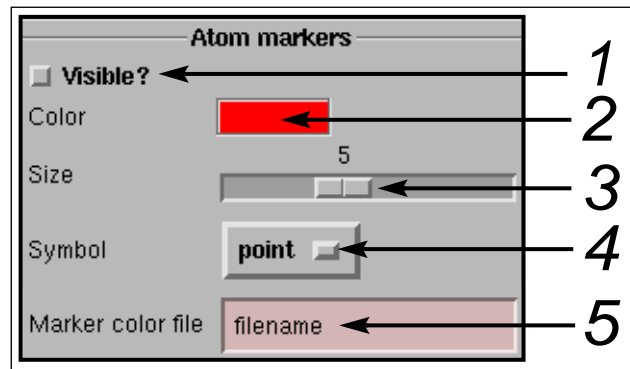
If you need more information on this subject, see the GOCAD Quick Reference.

A.1.3 Operation Menu Bar

On top of the right hand side of the composite window is the Operation Menu Bar. The Menus here contain commands that allow you to modify Objects of various Types.

| **G⊕CAD** | **General Mode** | **Create** | **Interpolation** | **Property** | **Region** | **Surface** | **Curve** | **Solid** |

The Menus and the contents of the Menus change as the user changes the Mode of the Menu Bar. For example, the default Mode is General. In General Mode, there are 6 Menus that contain some of the most commonly used commands, covering assorted Types of Objects. If the user changes the Menu Bar to, say, Surface Mode, the Menu Bar will then have 7 Menus containing only commands that deal with Surface Objects.

When you select a command from a Menu, GOCAD may ask you to select an Object in the Camera, or pop up a *Dialog* window for you to enter parameters, or both. A *Dialog* window typically consists of three parts:

PropertyRename Dialog — *1a*

Rename Property — *1b*

Object Bubble ▼ ↗ — *2a*

old name X — *2b*

new name — *2c*

☐ optimize_dir — *2d*

.20
height — *2e*

— *3*

OK   Apply   Cancel   Help

1. ***Name:***

    1.a. Name of the *Dialog* window;

    1.b. Name of the command

2. ***Object selections and parameter specifications***
        which may include:
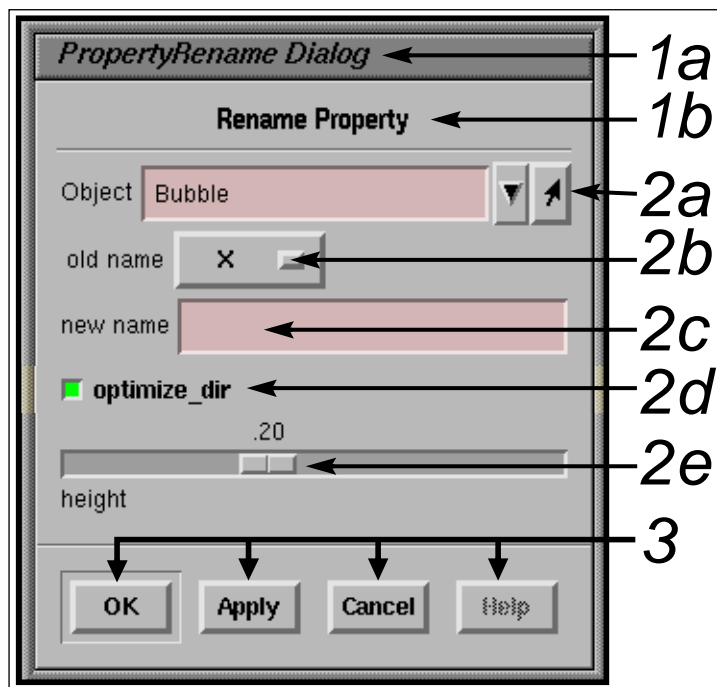
2.a.   Selecting one or multiple Objects, which can be done by

2.a.1.   *Default selection*

The displayed Object(s) that fits the requirements for this particular selection will be selected automatically by GOCAD.

2.a.2.   *Pull-down list*

The downward triangular arrow contains a list of Objects that fit the requirement for this particular selection.

2.a.3.   *Pick arrow*

The north-east-pointing arrow will invoke the pick mechanism, allowing the user to select an Object by clicking on it in the Camera.

2.a.4.   *Keyboard entry*

The user can always enter the Object name(s) from the keyboard.

2.b.   Specifying a parameter value by selecting an item from a list.

2.c.   Specifying a parameter value by entering a name or a number from the keyboard.

2.d.   Specifying a parameter value by toggling on or off a toggle.

2.e.   Specifying a parameter value by setting a slider bar.

3.   *Execution Options*

3.a.   *OK*

closes the Dialog window and carries out the operation.

3.b.   *Apply*

carries out the operation without closing the Dialog window.

3.c.   *Cancel*

closes the Dialog window without executing the command.

3.d.   *Help*

brings up the GOCAD on-line help for this Dialog window; not implemented yet.

If you need more information on this subject, see the *GOCAD Quick Reference*.

A.1.4 3D Camera

Below the Operation Menu Bar is the 3D Camera. There are three parts to the 3D Camera.

Camera

The dark background area is the viewing area, where you can display Objects in three dimensions.

Camera Menus

In the top portion of the Camera, there are three Menus, containing commands regarding the Camera, such as changing the Camera background color, the Camera scale, etc.

Camera Panel

Below the Camera Menus is the Camera Panel, which can itself be divided into three areas:

1. ***Movement Control Buttons***

   In the top portion, there are 9 **Movement Control Buttons**. These buttons allow you to move the Camera (or the Objects in the Camera). For more details on how these buttons work and how to use them, please see the *GOCAD quick Reference*.
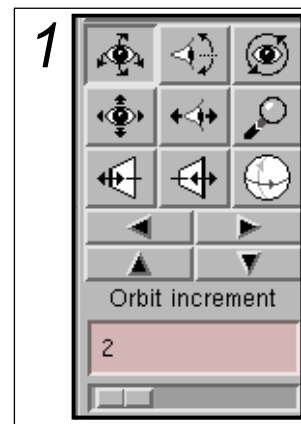
2. ***Quick Buttons***

   Near the center of the Camera Panel, there are 15 Quick Buttons, **QB**s.

   The middle three **QBs** are query buttons; you can use them to ask GOCAD the XYZ location of a point on an Object in the Camera, to ask the distance between two points on Objects in the Camera, or to review Attributes of an Object in the Camera.

   The upper 6 **Quick Buttons** offer you quick access to certain commands in the Camera Menus.

   The lower 6 **Quick Buttons** are **Create Buttons** that allow you to create Objects via clicking in the Camera.
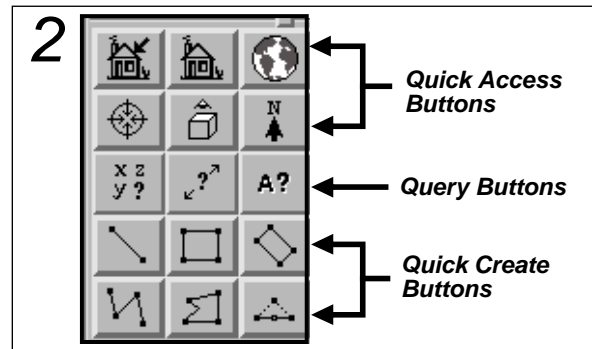
3. *Object-display List*
   At the bottom of the Camera Panel is the **Object-display List**. Every Object you have loaded or created in GOCAD will be listed here. From the list, you select Objects to be displayed in the Camera. The selected (displayed) Objects are highlighted.

If you need more information on this subject, see the *GOCAD Quick Reference*.

A.1.5 Info Area

At the bottom of the GOCAD composite window is the **Info Area**.

Info: Pick [2000, 900, 1372.22913] target = TopSurfData type = VSet
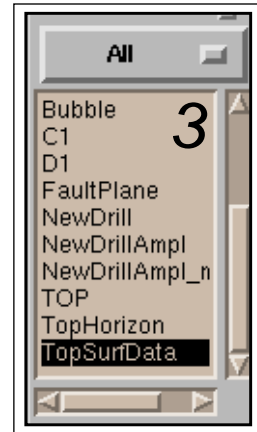
This is where GOCAD prints out one-line instructions for certain commands. This is also where GOCAD answers your queries about location of or distance between points.

When you try to pick an Object in the Camera for an Operation but fail to make a valid pick, GOCAD will print out an error message here.

A.2 BASIC GOCAD OPERATIONS

A.2.1 Cursor Mode

The GOCAD cursor has five shapes; each indicates a different function that will be executed when the user click with the left mouse button. Unless explicitly specified, **clicking** always means clicking with **Left mouse button**.

Square Spiral
Normal Camera Mode; observed in the Camera viewing area only.
This indicates that GOCAD will ignore any click in the Camera. If you click on an Object while the cursor is in this Mode, nothing happens.

Red Up-left Arrow
Normal Mode; observed outside of the Camera viewing area.
This indicates that GOCAD is in its Menu Selection Mode. If the user clicks on any Menu, Attribute, or Button, in the GOCAD window that item will be activated.

Black Up-right Arrow
Command-selection Mode; observed outside of the Camera viewing area only.
This indicates that the user has selected a Menu. The command that the user then clicks on will be invoked.

Cross-hair
Object-selection Mode; observed in the Camera viewing area only.
This indicates that GOCAD is waiting for the user to select an Object or an Object component from the Camera by clicking on the Object.

Black Up-left Arrow
Query Mode; observed in the Camera viewing area only.
This indicates that in the ***Info Area*** *(page A-7)* or in the ***Attribute Manager*** *(page A-3)*, GOCAD will print out information about any point on an Object that user clicks on. This Mode is invoked by any of the three Query Buttons in the Camera.

A.2.2 Object Selection in the 3D Camera

Single Selection
Click on the desired Object or Object component with the Left mouse button.

Multiple Selection
Click on the desired Objects or Object components with the Left mouse button while holding down the

Ctrl-Key on the keyboard. Release the Ctrl-key to end the selection.

Sequential Selection

Press down the left mouse button on the first item on the list and drag (while pressing down the left mouse button) across the subsequent items that you want to select.

Abort Selection

If you don't want to execute a command but have already gotten into the selection mode, click in any empty space in the 3D Camera to abort the command.

A.2.3 Object Movement in the 3D Camera

Remember that everything that we describe here is about "looking at Objects from different angles".

Nothing we do here actually change the physical location of the Objects in space.

To use the Camera Movement Buttons in the Camera:

1.   *Select a Movement button.*

2.   *Set the Increment.*

3.   *Select the direction. This is when the movement is carried out.*
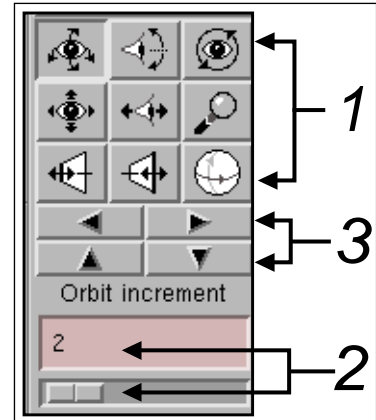
Or you can use the mouse buttons:

1.   *Turn*
     Move the mouse in the desired direction in the Camera while pressing down the Shift-Key and the *Left* mouse button.

2.   *Shift*
     Move the mouse in the desired direction in the Camera while pressing down the Shift-Key and the *Middle* mouse button.
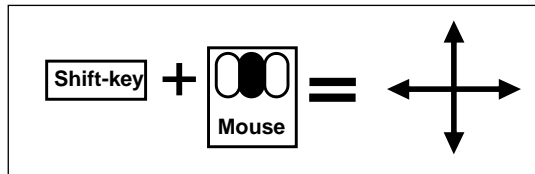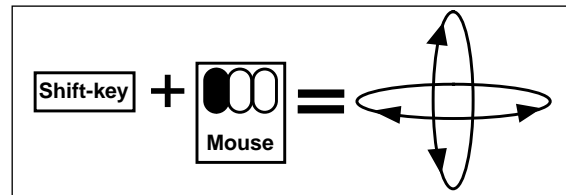
3.   *Zoom*
     Move the mouse up (zoom out) or down (zoom in) the Camera while pressing down the Shift-Key and the *Right* mouse button.
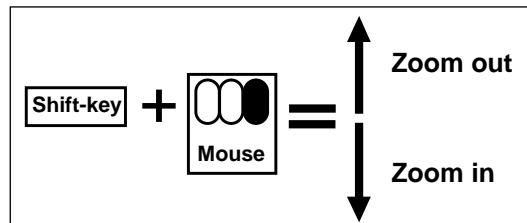
A.3 GOCAD OBJECTS

Currently there are 7 fully-implemented GOCAD Object Classes (Types):

The five Subtypes of the Atomic Type are also loosely referred to as Types instead of Subtypes:

The are also some less common Object Types. See the `GOCAD quick Reference`.

A.3.1 Atomic

Atomic Objects are made of Atoms that are connected in a specific manner. There are 5 Types of Atomic Objects:

But first, one should understand the basic terminology of an Atomic Object:

Atoms (Vertices)
Atoms are the most elementary components of any Atomic Object. Atoms are points in space defined by their (X, Y, Z) coordinates.

Properties
Atoms can carry Properties. All Atomic Objects carry the Properties X, Y and Z by default; which contain the coordinate values at each Atom. Properties can be scalar or multi-components, such as a vector Property (3 components).

Connectivity
The manner in which the Atoms in an Object are connected to one another is called the Connectivity.

Control Nodes
Any Atom in an Object can be specified to be a Control Node. Once declared a Control Node, the Atom is not allowed to move (change its spatial location) during an Object Interpolation Operation such as Fit Geometry.
There are 7 types of Control Nodes that can be declared in the data file of an object. The user can also set atoms to be Control Nodes using GOCAD Menu commands, but the Control Nodes that are set this way can only be ***All-direction Control Nodes (CN)*** *(page A-11)*.

1. **All-direction Control Nodes (CN)**
   These atoms are anchored in every direction. Their spatial locations cannot be changed during an interpolation operation.
   This is the only type of Control Node that the user can set from the GOCAD command Menu.

2. **Semi-restricted Control Nodes**
   These Control Nodes are fixed in certain directions and allowed to move freely in other directions during an interpolation operation.
   For instance, an Atom marked as **CNX** is fixed in the X direction, but is allowed to move in the YZ plane, while a **CNYZ** Atom is fixed in the YZ plane but allowed to move in the X direction.

Property Control Nodes
Similar to Control Nodes, these are Atoms whose Property Values are not allowed to change during a Property Interpolation or initialization operation. They have not been fully implemented yet.

If you need more information on this subject, see the GOCAD Quick Reference.

A.3.2 PointsSet (VSet)

The recommended suffix for a PointsSet file is "**.vs**". A PointsSet is a set of atoms that are not connected in any way.

A PointsSet in GOCAD is usually brought in as the output from other computer applications.

Datapack (VSet)
If you have data files from C-Gocad, you may get a "Datapack" from time to time.
A Datapack in C-Gocad is a Points (PointsSet) Object that carries Property information; a regular Points (PointsSet) Object was not allowed to carry Property information in C-Gocad.
In GOCAD++, a PointsSet can either carry or not carry Property information, therefore the concept of datapack is obsolete.
If you have Datapack files from C-Gocad, just make sure you select the old2new Convertor when you load the file into GOCAD++. GOCAD++ will treat this file like a PointsSet with Property information.

If you need more information on this subject, see the GOCAD Quick Reference.

A.3.3 Curve (PLine)

The recommended suffix for a Curve file is "**.pl**". A Curve is made of connected (and/or disconnected) Segments; each Segment connects two Atoms.

Segment
Two Atoms that are connected form a Segment. An Atom cannot belong to more than two Segments (there cannot be branching in a Curve).

ILine
A Curve may consist of multiple disconnected pieces. Each piece is called an ILine.

Extremity
The two end Atoms of an ILine are called the Extremities of the ILine.

Closed Curve
An ILine can be a closed loop. There are no Extremities in a closed ILine.

If you need more information on this subject, see the *GOCAD Quick Reference*.
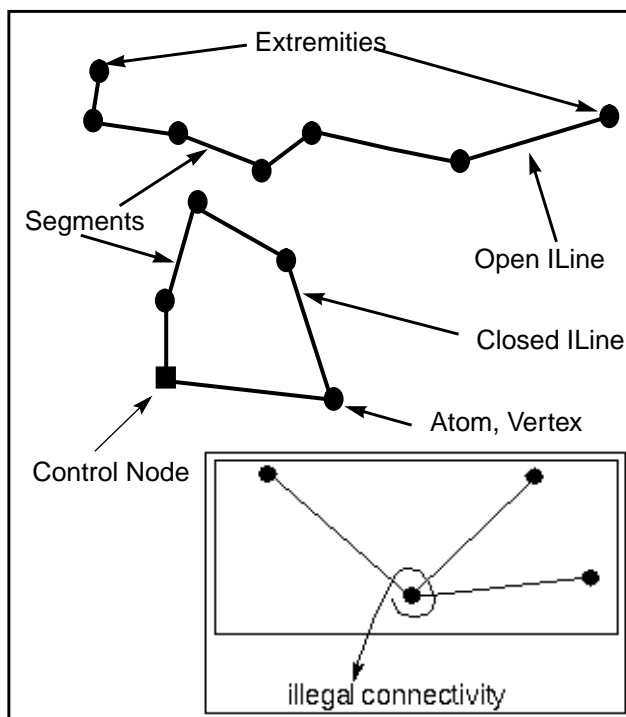
A.3.4 Surface (TSurf)

A Surface is made of connected (and/or disconnected) Triangles; each Triangle is made up of three Atoms, one on each corner. The recommended suffix for a Surface file is "**.ts**".

Triangles
Triangles are composed of three connected Atoms; each Atom forms one corner (Vertex) of the Triangle. A Surface cannot have wings, which means no more than two Triangles can share an edge.

TFace
Every patch of connected Triangles is called a TFace. A Surface can have one or more TFaces.

Border
The boundary of a TFace is called a Border. A single boundary can be made up of multiple Borders; each Border defines a piece of the boundary.

Border Stone
Borders that are on the same boundary are separated from each other by Border Stones.

Mesh
The triangulation network of a Surface.

If you need more information on this subject, see the `GOCAD Quick Reference`.
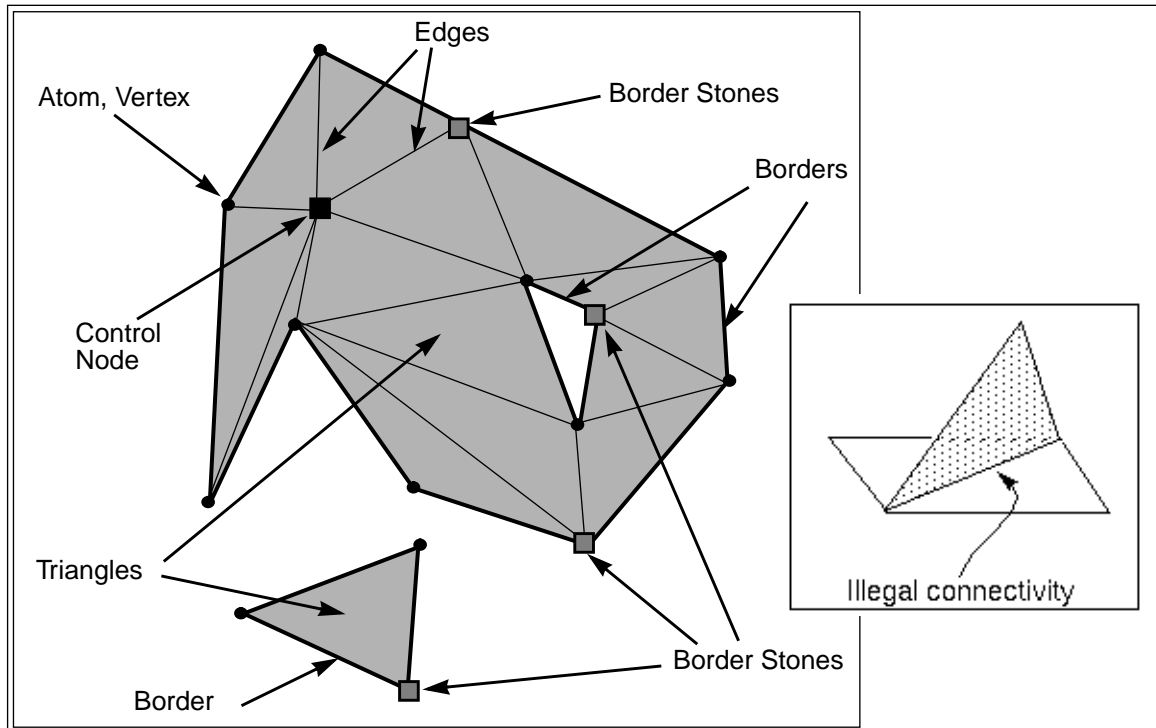
A.3.5 Solid (TSolid)

A Solid is made of connected (and/or disconnected) Tetrahedra; each Tetrahedron is made up of four Atoms, one at each corner. The recommended suffix for a Solid file is "**.so**".

Tetrahedron
Four connected Atoms form the basic element of a Solid, called a Tetrahedron. A Tetrahedron is a three-sided pyramid. A Solid cannot have branching in the 4th dimension; no more than two Tetrahedra can share a Tetrahedron face (we will not try to draw a 4-D picture here).

A Tetrahedron.
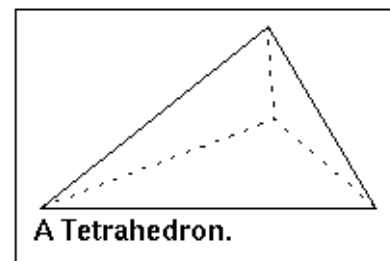
TVolume
A Solid can be made up of multiple disconnected chunks. Each chunk is called a TVolume.

Skin
The bounding surface of a Solid is called the Skin. Sometimes the Skin of a Solid is also referred to as the Border of the Solid.

If you need more information on this subject, see the *GOCAD Quick Reference*.

A.3.6 GShape

One of the Tube lines   The Backbone

Sections

A Rib Point   A Rib Vector
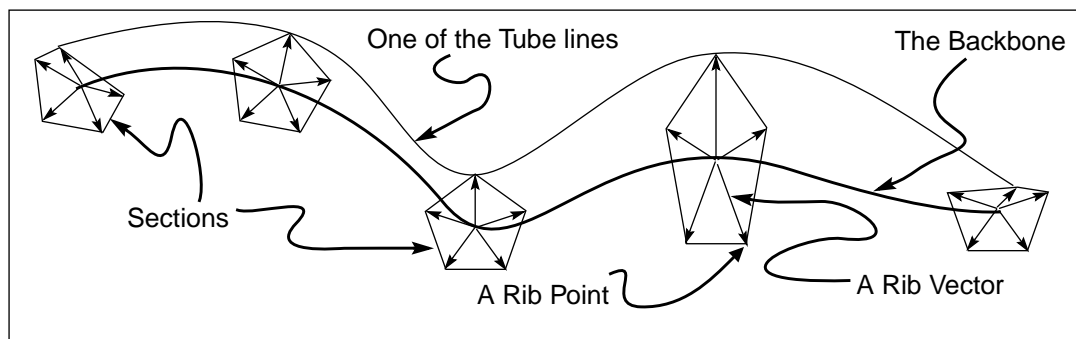
A GShape Object is defined by a Backbone, and by Sections placed along the Backbone. This Object is created specifically to describe long and meandering geological features, such as channels. The recommended file suffix for GShape is "**.gs**". It is an Atomic Object Type.

The shape of each Section is defined by a set of Ribs that radiate from the Backbone. The fourth component of a GShape is the Tube, made of lines that connect the correlated corners of each Section. The fifth component is the skin that wraps around the fibrous structure; this is not shown here.

Like other Atomic Objects, a GShape Object can consist of multiple disconnected pieces, where each piece is as described above. Each disconnected piece in a GShape Object is sometimes referred to as an IShape (since IGShape is hard to pronounce).

Backbone
A line that describes the main orientation of a GShape Object. You can edit this element using commands you find in the ***Curve*** Mode.

Section
A Cross-Section along the Backbone that defines the shape of the Object at a particular point.

Ribs
The points that define the outline of a Section are referred to as the **Rib points**, or simply the Ribs. The Ribs at a Section are indexed from zero to the total number of Ribs at the Section (0, 1, ..., n-1). The number of Ribs per Section is a constant for the entire GShape Object, because Ribs of the same

index number are correlated from one Section to another to define the shape of the Object. The lines of correlation of the Ribs form the Tube. The line that connects the Rib points in sequence at a particular Backbone Atom forms the outline of the Section.

Tube
A set of longitudinal lines that connect the correlated Ribs (Ribs with the same Index number) of the Sections along the ***Backbone*** *(page A-14)*.

Skin
The outer boundary of a GShape Object. Think of it as the plastic wrap that wraps around the ***Tube*** *(page A-15)*.

Rib Vectors
A set of vectors that radiate from the Backbone to the Ribs *(page A-14)*.

GShapeBundle
Like other Atomic Object Types, a GShape Object can consist of multiple disconnected pieces. However, the real name for a GShape Object is actually GShapeBundle, while individual pieces in the Object is called a GShape (Surface-TFace, Curve-ILine, Solid-TVolume, GShapeBundle-GShape). Sorry about the confusion, but you will learn to live with it like the rest of us :-).

If you need more information on this subject, see the `GOCAD Quick Reference`.

A.3.7 Well

A Well Object in GOCAD includes

1. ***the physical location of the path of the Well, and***

2. ***Well logs (called Well Curves) in that Well.***

The recommended suffix of a Well file is "**.wl**"

The Well path data file (which also includes Marker and Zone information) is in ASCII. The Well Curves data file can be in either ASCII or binary, since Log-data files can be huge.

If you need more information on this subject, see the `GOCAD Quick Reference`.

A.3.8 Voxet

A Voxet Object is a 3D regular grid made of volume elements called Voxels, or simply Cells. Each grid point (Node) is the center of a Cell. The recommended suffix for a Voxet file is "**.vo**".

The Cells in a Voxet are arranged along grid lines parallel to three axes (directions); u, v, and w. The three axes do not have to be orthogonal. You can name the three axes something other than u, v, and w; this is done in the Voxet data file (please see the *GOCAD++ Developer's Guide*).

a.) a Voxet and its step-vectors

b.) a Voxet and its end-points
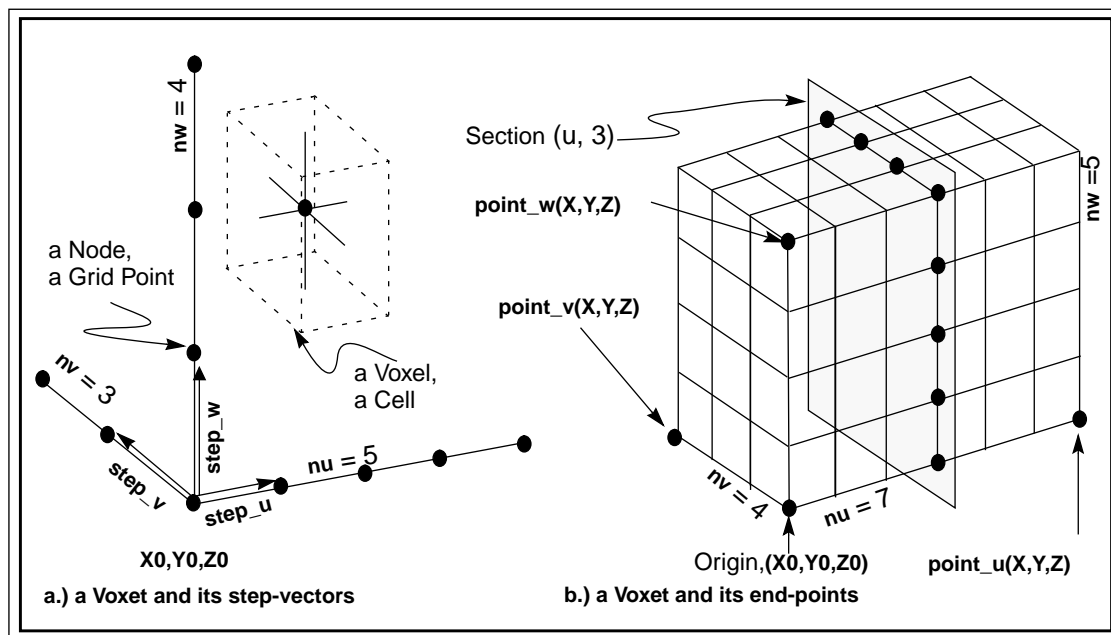
The distance between Grid Points along the u, v, and w axes are step_u, step_v, and step_w respectively.

The numbers of Grid Points in the directions u, v, and w, are nu, nv, and nw respectively.

Dimensions

The size and position of a Voxet are defined by the following:

the Voxet's Origin: (X_0, Y_0, Z_0);

the directions of the three axes (two possibilities are given because the GOCAD Menu provides two different ways of defining a Voxet.):

either the end points of the axes: point_u(X,Y,Z), point_v(X,Y,Z), point_w(X,Y,Z); or

the step vectors of each Cell: step_u, step_v, step_w; and

the number of Grid Points along each axis: nu, nv, nw.

1. **Origin**

The Origin of a Voxet is similar to the origin of a Coordinate system. The Origin of a Voxet anchors the Voxet in space. It is defined by its coordinates O (X_0, Y_0, Z_0).

2. **u, v, w**

The three axes of a Voxet are named u, v, and w. There is no specific naming convention as to which one should be u, v, or w; they do not have to be orthogonal. If the Voxet is associated with a data set (e.g. seismic data), u is the fast axis and w is the slow axis.

3. **nu, nv, and nw**

The number of Grid Points along the three axes.

4. **step_u, step_v and step_w**

The three vectors that define the sides of a single Cell. They are not coordinates of three points, nor are they simple lengths; they are vectors.

5. **point_u (x, y, z), point_v (x, y, z), and Z (of point_w)**

The coordinates of the end points of the three axes of a Voxet. They are not vectors; they are simply coordinates of points.

Cage

The outer edges of a Voxet. This is not necessarily a cube, since a Voxet is not necessarily orthogonal, nor are the sides necessarily all of the same length.

Region

You can divide a Voxet volume into multiple overlapping sub-volumes. Each sub-volume is called a Region.

Grid Connectivity

During a global Property Interpolation (or Simulation) process, the Property value at any Grid Point is affected by the Values at Grid Points that it is directly or indirectly connected to. Such connectivity is symbolized by the grid structure.

Connectivity discontinuities (boundaries) can be created by building a VoxetModel or by executing the command **Cut With Surfaces**.

Property values in different connectivity-wise-isolated volumes will not affect one another in Interpolation or simulation processes.

In a Region-wise operation, the operation is always confined within the selected Region whether the Region boundaries are connectivity boundaries or not. Furthermore, if there are connectivity discontinuity within the Region, they will always be honored.

Section

A Section on a particular Voxet axis is a plane that is parallel to the other two axes of the Voxet.

A Section is defined by an axis and by an index number, which starts at zero in a Voxet.

For example, the Section denoted by (u, 3) is the plane parallel to v and w, whose distance from the Origin is 3 times the distance of step_u. The Section consists of all the grid points whose coordinates are (3, v, w).

Voxet outline

Section outline

a Voxel,
a Cell in
a Voxet

a 2d Voxel

A Voxet Node sits at the center of a Voxel, not at its corner.
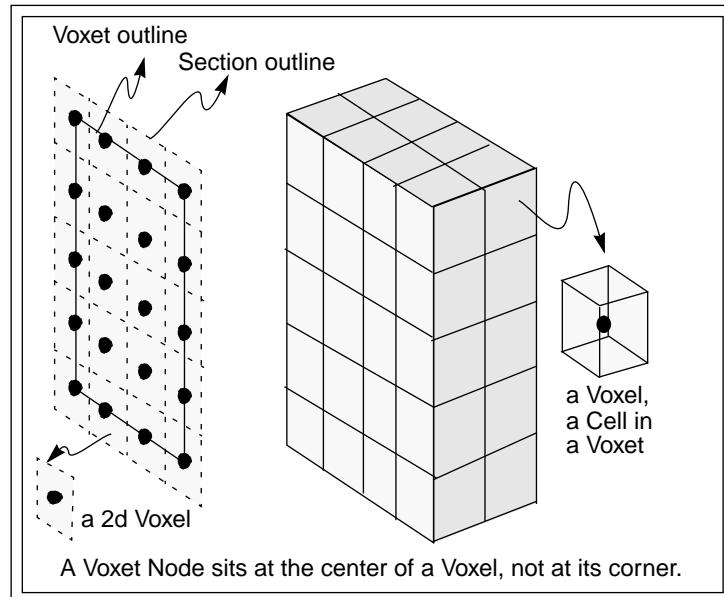
Property

A Voxet may contain multiple sets of information. Each set is called a Property (or Grid) of the Voxet and contains a value at each Grid Point of the Voxet. For this reason, sometimes the Property in a Voxet is referred to as a Grid. When you display a Voxet, you

Voxet Grid Point Structure

Voxet Cell Structure

Voxet Property Structure

need to select a Property to display, otherwise you see only an empty **Cage** *(page A-17)*.
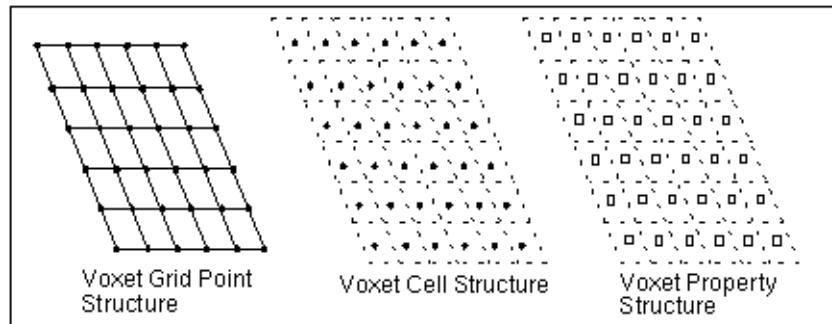
In a Voxet the Cell boundaries do not coincide with the lines connecting the Grid points; the Grid points are the centers of the Cells. Therefore the Grid Structure is smaller than the Cell Structure, and a Section is slightly (half of a Cell on each side) wider than the cage boundary of the Voxet.

The Property values are carried at the grid points, i.e. the center of the Cells. The entire Cell has the same Property value. That is why when you display a Property on a Voxet Section (if the antialiasing toggle is off), it consists of tiny little blocks. A Property data file of a Voxet can be a separate binary or ASCII file.

If you need more information on this subject, see the GOCAD Quick Reference.

A.3.9 SGrid

An SGrid is a Stratigraphic Grid. It is more sophisticated than a ***Voxet*** *(page A-16)* in that the user can conform the SGrid to bounding Surfaces and split the SGrid along fault Surfaces. The recommended suffix for an SGrid file is "**.sg**".

This Object is important in Reservoir Geology and Engineering where anisotropy plays a significant role in hydrocarbon migration and accumulation.

a) an SGrid and its step-vectors

□ Property point
● Grid point (Node)

b) an SGrid and its end-points
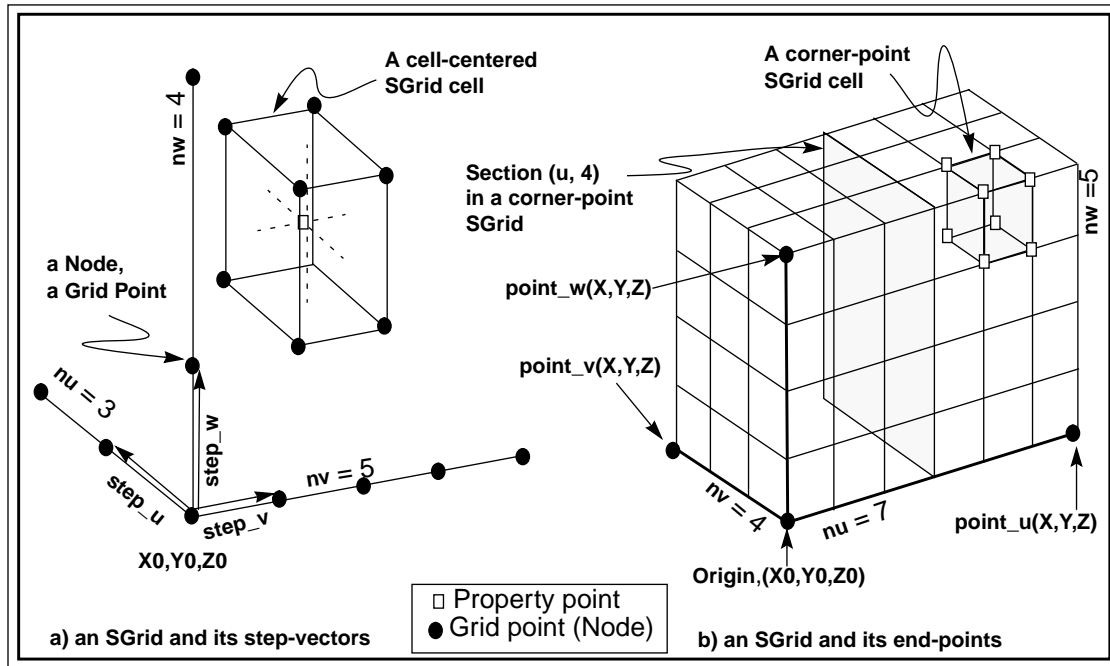
Dimensions
When you first create an SGrid in GOCAD, it has the same components as a Voxet. However, <u>***UNLIKE***</u> the Voxet, the SGrid's grid structure is coincident with its cell structure; i.e., the grid points are the corner points of the cells, not the centers of the cells.
The dimensions of the SGrid are defined by its Origin, its three axes, the number of Nodes along each axis, and the three step vectors of each Cell, which are explained as follows:

1. ***Origin***
   The Origin of the SGrid is similar to the origin of a Coordinate system. The Origin of an SGrid anchors the SGrid in space. It is defined by its coordinates (X_0, Y_0, Z_0).

2. ***u, v, w***
   The three axes of an SGrid are named u, v, and w. There is no specific naming convention as to which one should be u, v, or w.

3. ***nu, nv, and nw***
    The number of Nodes along the three axes.

4. ***step_u, step_v and step_w***
    The three vectors of an SGrid Cell. These are vectors, not the coordinates of three points.

5. ***point_u (x, y, z), point_v (x, y, z), and w (x, y, z)***
    The coordinates of the end points of the three axes of an SGrid. They are not vectors; they are simply coordinates of points.

Cell
Unlike a Voxet, an SGrid has its grid points at its Cell corners instead of at its Cell centers. Therefore the Cell structure and the grid structure of an SGrid are identical.

Cage
The edges of an SGrid.

Region

You can divide a SGrid volume into multiple overlapping sub-volumes. Each sub-volume is called a Region.

Section
A Section on a particular axis in an undeformed SGrid is similar to a Section in a ***Voxet*** *(page A-16)*.
A Section in a conformed SGrid has a similar definition. For example, a Section on axis u is a plane that contains all the grid points (constant, v, w). However, in a conformed SGrid, u, w, w are no longer vector directions, and a Section is not necessarily a flat surface.

In an SGrid, you have the option to specify the Section-numbering convention (whether you want to start counting from 0, or 1, or whatever number), which is done in the Attribute Manager, Misc Attribute Category.

Property
An SGrid Object can carry multiple Properties. The user can specify whether the Property information in an SGrid is to be carried at the centers of SGrid Cells (cell-cen-

A cell-centered Section | *A corner-point Section*
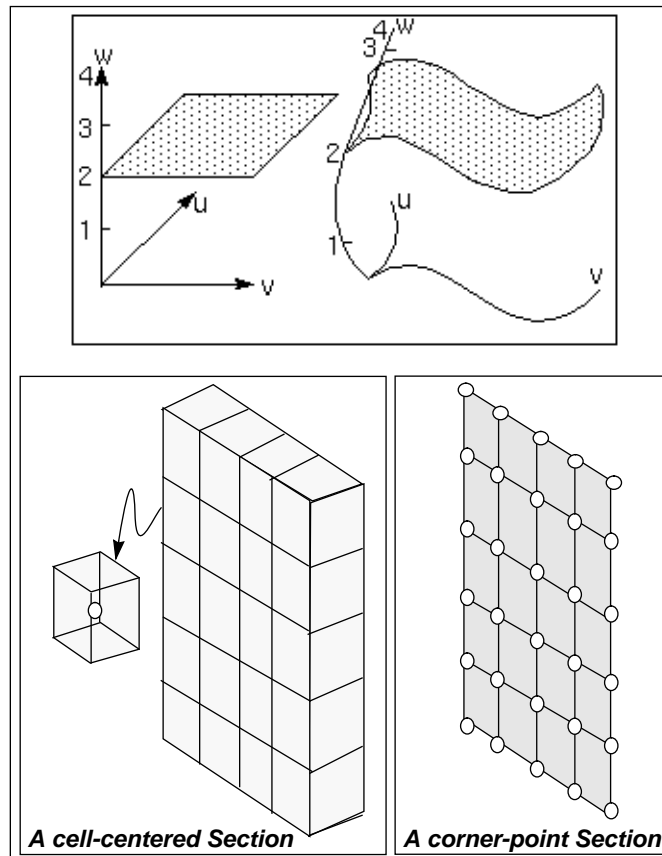
tered SGrid) or at the corners of SGrid Cells (corner point SGrid). This is **_DIFFERENT_** from a Voxet, where the Property information is always carried at the Grid points, which are always the centers of Voxet Cells.
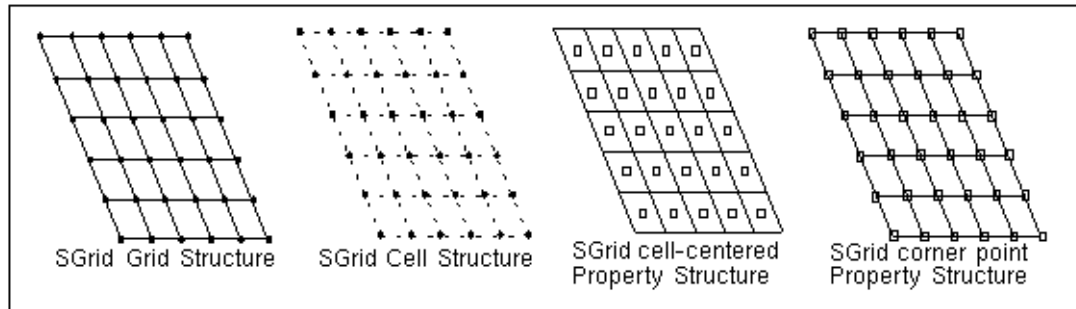
A Property data file of an SGrid is a separate binary or ASCII file.

    1. **_cell-centered_**
        Property Values are carried at the cell centers. The Property value in a Cell has a single value. When you display a Section, it consists of tiny little blocks.

    2. **_corner point_**
        Property Values are carried at the cell corners, the grid points. The Property value in a Cell is linearly interpolated from the eight corner points of the Cell. When you display a Section, it is always appears blurred because the Property values are always interpolated.

If you need more information on this subject, see the `GOCAD Quick Reference`.

A.3.10 Model

A Model Object in GOCAD consists three major parts:

    1. **_Members_**
        Objects that the Model is built of;

    2. **_Topology_**
        relationships between the Model members; and

    3. **_Property_**
        a set of mathematical equations that define Properties in various portions of the Model.

If you need more information on this subject, see the `GOCAD Quick Reference`.

A.3.11 Group

A Group Object in GOCAD is basically a group of Objects that are declared to have a single identity. In other words, it is an Object made up of other Objects. There are two types of Group Objects; the type is determined when the Group is first created:

1. **HomogeneousGroup**
   All members in the Group are of the same Object Type (TSurf, VSet, etc.).

2. **Heterogeneous Group**
   Members in the Group are of different Objects Types.

If you need more information on this subject, see the GOCAD Quick Reference.

Appendix B: DATA FORMATS

This chapter covers the GOCAD data file format. It is based on information provided by Jean-Claude Dulac. Like all GOCAD documentation and GOCAD itself, descriptions in this appendix do **CHANGE QUICKLY**. If you have any questions or problems, or need more information, please contact Jean-Claude Dulac directly at ***jcdulac@t-surf.com.***

B.1 BASICS

A GOCAD Object file consists of three parts: the Header, the Body and the End marker. A GOCAD Object file may also contain Comments that start with a # sign on each comment line..

B.1.1 Header

The header gives the Object Type name (GOCAD TSurf) and the version number (0.01).

The case of the Object Type name matters. The version number is not necessary. The Object Type name and version number are used to call up the correct GOCAD Object Reader when the file is loaded.

Also stored in the Header section are Attribute values, which include the Object's name (HEADER{...}).

```
GOCAD TSurf 0.01

HEADER {
*name: H1
*color: red
..

}
..

..

..

#comments

..

END..
```

Header
Body
Comment
End marker

Example 1: **Basic GOCAD Object file.**
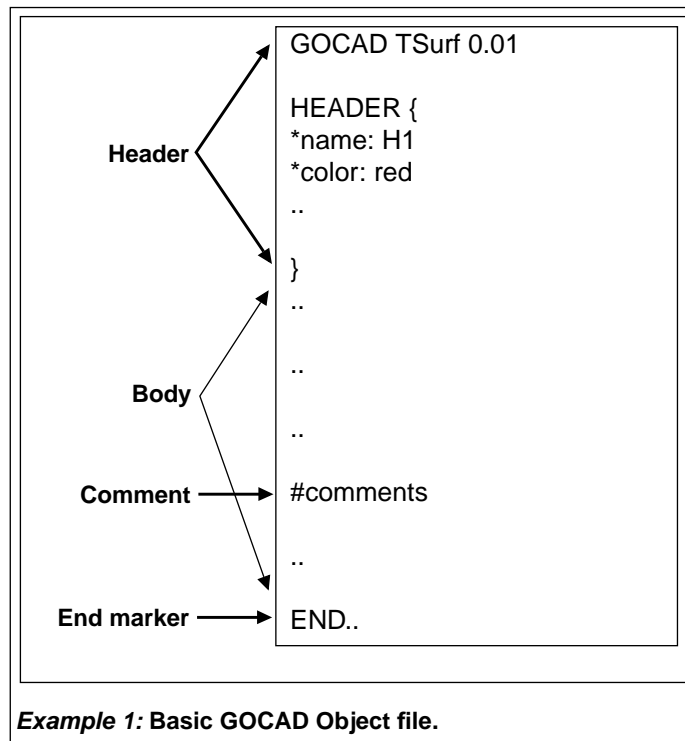
B.1.2 Body

The Body contains the geometrical and Property information of the Object.

Since GOCAD Objects are inheriting in nature, so are the Object files.

For example, TSurf (Surface) is derived from the Atomic Object Class (Type), so a TSurf file contains all the file elements of an Atomic Object, plus some extras.

All GOCAD Objects are derived from the GObj Object Class, therefore all GOCAD Objects contain the file elements described in B.2 GObj *(page B-4)*.

When describing the ASCII format of the different Classes (Types) of GOCAD objects, only the extra elements are described to avoid redundancy.

B.1.3 END

The trailer, END, provides an end-of-object marker. Because of the existence of the **End** marker, multiple Objects can be stored in a single file. See Example 2 *(page B-3)* .

B.1.4 Comments

GOCAD allows comments in any part of an Object file, as long as the first column of the line is a # sign.

B.1.5 Units
Units are not fully implemented in the current version of GOCAD, but theoretically, Units may be specified as shown below:

> seconds
>
> meters
>
> meters/seconds^2
>
> meters/seconds
>
> seconds*meters
>
> seconds
>
> 3600*seconds
>
> 0.44704*meters/seconds
>
> 0.3048*meters/seconds
>
> 0.3048*seconds*meters

Restrictions: at most one scalar in front, at most one division.

```
                    GOCAD TSurf 0.01

                    HEADER{
Object 1            name:FirstOne
                    }
                    ....
                    END

                    GOCAD PLine 0.01

                    HEADER{
                    name: SecondOne
Object 2            }
                    ..
                    END
```
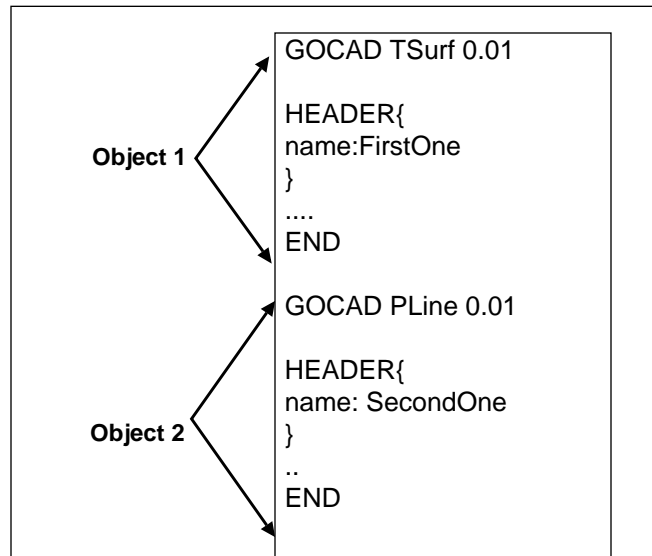
***Example 2:* Multiple GOCAD Objects file.**

B.2 GOBJ

The GObj file elements define the GObj style and the GObj coordinate space and Units. Style Attributes

The HEADER block, delimited by { and }, contains the Style definition. Each line defines an Attribute of an Object. Attributes are strictly optional, except for the name of the Object. An Object must have a name.

If preferred, an Attribute can be given on a separate line outside of the HEADER block; in that case the line should begin with the **HDR** keyword as shown in Example 3 *(page B-4)*.

An example of GObj Style element is:

B.2.1 Coordinate System

```
                    HEADER {
                    *name : H1
                    *color: red
Header              ..
Block
                    ..}
                    HDR *solid : on
Header              HDR *solid*color : green
Lines
                    HDR ..
```
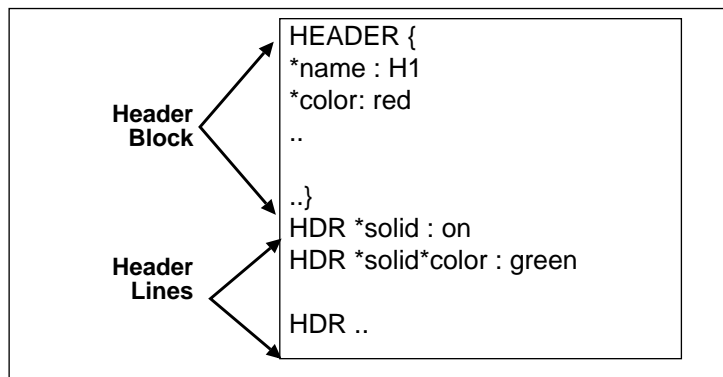
Example 3: **GOBj Style file element.**

The GObj Coordinate System (basis name) is optional for all Objects. It is used to translate an Object from its file coordinate system to the "real world" coordinate system.

For example, a surface maybe described in terms of CDP-Line-Time in the data file. With the proper GObj Coordinate System description, the Surface can be read in the real-world XYZ to be compatible with other Objects.

The GObj Coordinate System is given after the HEADER block and before the first VRTX data line either as:

       BASIS_NAME *Basename*

or as:

       COORD_SYS_NAME *CSname*

There is no known reason why there are these two options, since there is only one way to define the Coordinate System itself.

The Coordinate System must have been defined previously, in the same file or in its own separate file but loaded into GOCAD before being referenced by an Object, as follows:

    GOCAD_COORD_SYS
    AXIS_O *X Y Z*
    AXIS_U *x y z*
    AXIS_V *x y z*
    AXIS_W *x y z*
    AXIS UNIT *unit_u unit_v unit_w*
    AXIS_NAME *name_u name_v name_w*
    NAME *CSname*
    END

The coordinates are read in double precision converted to simple precision after coordinate system conversion.

B.3 ATOMIC

The Atomic data format defines the points, locations and Property values, of an Atomic (currently, this includes PointsSet, Curve, Surface, Solid and GShape). The Atomic inherits all of the **GObj** *(page B-4)* file format elements, plus Atomic data.

B.3.1 Atomic

This section describes the recommended Atomic file format.

Since Property values are optional, a basic Atomic data line consists of three parts: Type, ID number, and data:

> VRTX *ID X Y Z [CN]*

where the XYZ describes the location of that point. Additional information maybe attached at the end of each VRTX line to specify the Interpolation Restriction on that Atom (Control Node information).For example:

```
                        GOCAD VSet 0.01
                        HEADER{
      HEADER            name:WithoutProperty
                        }


                        VRTX 1 1000 20 35 CN
                        VRTX 2 120 18 90 CNXY PCN
      Atomic            VRTX 3 310 5 21 CNX
      Data              ...


                        ...
                        END
```

Example 4: Atomic data file, with no Properties.
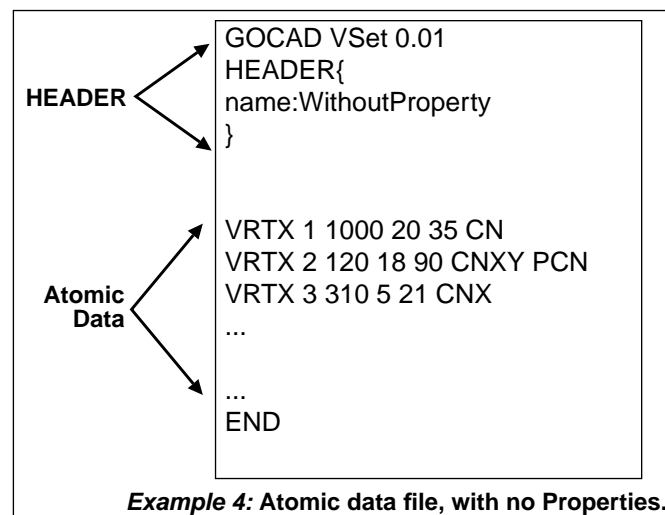
If there are Properties, they must be declared after the HEADER block and before the first point data line as:

> PROPERTIES Pname1 Pname2 ...

PROPERTY_CLASSES class_name1 class_name2 (optional, but if you declare one, you must declare all)

UNITS unit1 unit2 ...(optional, but if you declare one, you must declare all)

NO_DATA_VALUES v1 v2 ...(optional, but if you declare one, you must declare all)

The data lines then look like:

PVRTX ID X Y Z PV1 PV2 ...

where PV1, PV2... are Property values. For example.

```
GOCAD VSet 0.01
HEADER{
name:WithProperty
}

PROPERTIES Poro Perm Ampl
PROPERTY_CLASSES poro perm seis.....
NO_DATA_VALUE -1 -1 -999...

PVRTX 1 100 20 35 0.2 4000 ..
PVRTX 3 120 18 20 0.1 3500 ...
...
#SUBVSET
PVRTX 17 200 13 21 0.35 4300...
ATOM 18 3 0.22 3900...
...
.....
END
```

Header → GOCAD VSet 0.01 ... }

Property Declarations → PROPERTIES Poro Perm Ampl ...

Atomic Data →

Part Declaratoin → #SUBVSET

Atom statement → ATOM 18 3 0.22 3900...

Example 5: **Atomic data file, with Properties.**

B.3.2 VRTX vs. ATOM

See B.4.2 <u>VRTX vs. ATOM</u> *(page B-11)*.

B.3.3 Parts (Subset)

An Atomic Object can have its VRTX grouped into subsets. In a VSet, the keyword is **SUBVSET**; in

PLine, it is **ILINE**; in TSurf, it is **TFACE**; and in TSOLID, it is **TVOLUME**.

B.3.4 Atomic - Old from C-GOCAD

This section describes an older Atomic format, mainly made to be compatible with C-GOCAD file format.

The format consists of two sections, the location section and the Property section. The Property section is optional.

The location section is identical to the new format:

> VRTX ID X Y Z

The Property value section contains a header and the list of Property values associated to a point defined in the previous section. The header defines the names of the DataPackFields (Properties) associated with the Atomic. The following example declares two DataPackFields/Properties.

> FIELDS Pname1 Pname2
>
> (or PROPERTIES Pname1 Pname2 ...)
>
> PROPERTY_CLASSES class_name1 class_name2 (optional, but if you declare one, you must declare all)
>
> UNITS unit1 unit2 ...(optional, but if you declare one, you must declare all)
>
> NO_DATA_VALUES v1 v2 ...(optional, but if you declare one, you must declare all)

The Property section define, for each point defined in the location section, the Property values associated with the Atomic's DataPackFields.

> REC *ID1 ID2 PV1 PV2 ...*

Associated with a REC definition are two identifiers needed to be specified for compatibility reasons with earlier version. The two identifiers must be identical.

An example of a GObj file identical to the one given in <u>Example 5</u> *(page B-7)*, but in this old data format

is given below:.

HEADER	GOCAD VSet 0.01
	HEADER{
	name:WithProperty
	}

Location Data	VRTX 1 100 20 35 VRTX 3 120 18 90 ...
Property Declarations	PROPERTIES Poro Perm Ampl PROPERTY_CLASSES poro perm seis..... NO_DATA_VALUE -1 -1 -999...
Property Data	REC 1 1 0.2 4000 ... REC 3 3 0.1 3500 ... ... END

Example 6: **Old Atomic data file, with Properties.**
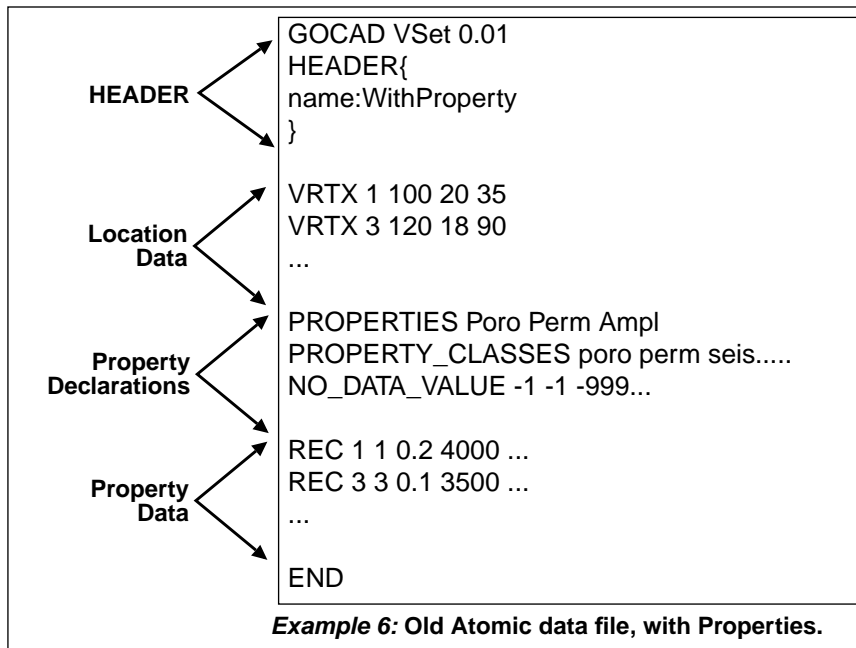
B.4 TSURF (SURFACE)

The data file of a TSurf includes the inherited file elements of a **GObj** *(page B-4)* and of an **Atomic** *(page B-6)*. The Atoms (Vertices) of the TSurf must be defined before the triangles.

B.4.1 Triangles

Each Triangle is then defined by its three Vertices (Atoms) in the following format:

        TRGL *id1 id2 id3*

where *ids* are the IDs of the already-defined Atoms (Vertices). In addition, the geological type of a TSurf can be specified using the following format:

        GEOLOGICAL_TYPE *geological_type_name*
        STRATIGRAPHIC_POSITION *age time*

where *geological_type_name* can be one the following: **top, intraformational, fault, unconformity, intrusive, topography, boundary, and ghost**. **age** is the name of a stratigraphic unit. **time** is the numerical value of age (35000, etc.).

For example, the ascii file corresponding to a TSurf named SQUARE, containing four vertices, and two triangles is shown in Figure 7.
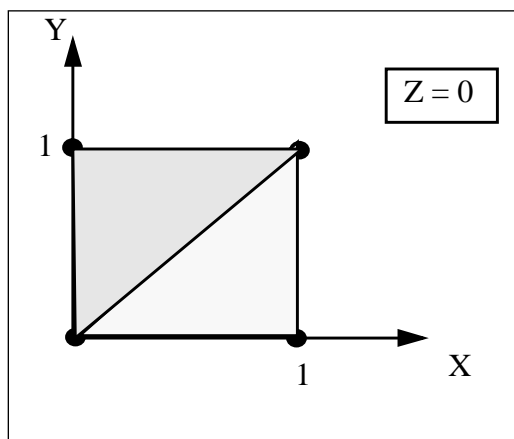
Example 7: **An example of a TSurf and its data file.**

```
GOCAD TSurf 0.01
HEADER{
name:SQUARE
}
GEOLOGICAL_TYPE top

VRTX 1  0.  0.  0.
VRTX 2  1.  0.  0.
VRTX 3  1.  1.  0.
VRTX 4  0.  1.  0.
TRGL 1  2  3
TRGL 1  3  4

END
```

B.4.2 VRTX vs. ATOM

In addition to the VRTX or PVRTX point/vertex descriptor find in the Atomic format definition, on can find the additional line inside a TSurf ascii file:

> ATOM *id1 id2* (where id1 > id2)

where **id1** is the index number of this new node and **id2** is the index of an existing VRTX node whose XYZ this Atom shares.

The goal of the ATOM keyword is to create a new Node that shares the XYZ of an already existing vertex. An Atom node has its own Property values but is spatially linked to the existing VRTX node. In other words, an ATOM and its referenced VRTX are collocated, but not connected.

An example of the use of such Atom record is given below:


```
GOCAD TSurf 0.01
HDR name:s1
VRTX 1 0 0 0
VRTX 2 0 2 0
VRTX 3 1 1 0
VRTX 4 2 2 0
VRTX 5 2 0 0
TRGL 1 2 3
TRGL 3 4 5
END
```

$Z = 0$

```
GOCAD TSurf 0.01
HDR name:s1
#TFACE
VRTX 1 0 0 0
VRTX 2 0 2 0
VRTX 3 1 1 0
TRGL 1 2 3
#TFACE
VRTX 4 2 2 0
VRTX 5 2 0 0
ATOM 6 3
TRGL 4 5 6
END
```

Example 8: **An example of the use of ATOM in TSURF.**
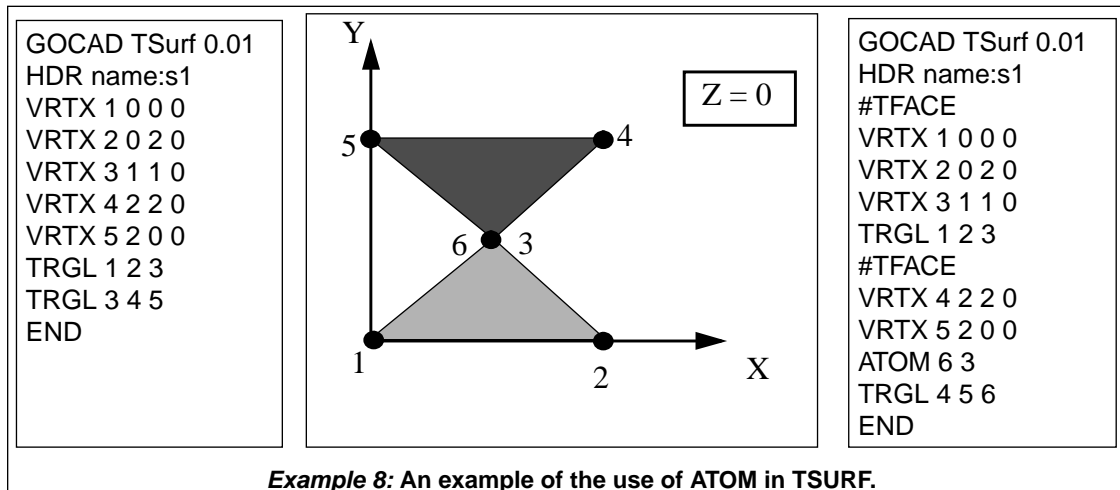
The input Tsurf file on the left will create a Surface shown at the center. The Surface has two TFaces, because two Triangles need to share an edge to be considered connected. When the Surface is saved, GOCAD will output the file on the right. It recognizes that the two Triangles both have one vertex at an identical location, but topologically, they can not be the same point. So, GOCAD creates a new point, ATOM 6, that shares the identical location of VRTX 3, but is an independent point in the sense that it has its own Property values (not in this case). However, had you have two more triangles in your input file, TRGL 2 4 3 and TRGL 6 5 1, the ATOM line would not have been created because the four triangles are then connected through direct or indirect edge-sharing.

This applies to all Atomic Objects.

B.4.3 Parts/TFace

The **_TFACE_** keyword is proposed to ensure identical indexing of the VRTX every time you save an Surface whose mesh you have not modified.

In <u>Example 8</u> *(page B-11)*, the file on the right (the one output by GOCAD) not only has the extra ATOM line, it also two TFACE lines to separate VRTX (and ATOM) and TRGL from different TFaces. They are currently commented out for their full implementation is still to be determined by the GOCAD Committee (the statement is not utilized currently).

B.4.4 Borders

An edge of a TSurf can be a single piece or can be divided into multiple pieces. Each piece is called a Border, and it is separated from its neighboring Border by a Border Stone. A Border Stone is a designated node (VRTX, ATOM) on an edge of a TSurf that separates two Borders.

Such information is recorded at the end of the Surface file, in the format shown below:

        BSTONE vrtx_id
        BORDER border_id vrtx_id1 vrtx_id2

The **BSTONE** line decalres an VRTX as a border stone.

The **BORDER** statement defines the border, **border_id**, as starting at **vrtx_id1**, which must have been declared as a Border Stone in preceeding **BSTONE** lines, and continuing in the direction of **vrtx_id2** , which must be adjacent to **vrtx_id1** (this border ends when it runs into another Border Stone).

The beginning border stone VRTX is always included in the list of points along that border.

B.5 PLINE (CURVE)

The Ascii PLine format inherits from the **GObj** *(page B-4)* and from the **Atomic** *(page B-6)* formats. The Atoms of the Atomic must be defined first. Each Segment in the PLine is then defined by its 2 apices in the following format:

        SEG *id1 id2*

where *ids* are the IDs of the already-defined Atoms.

For example, the ascii file corresponding to he PLine represented in Figure 9 may look like:


```
GOCAD PLine 0.01
HEADER{
name:TwoSeg
}

VRTX 1  0.  0.  0.
VRTX 2  1.  1.  0.
VRTX 3  0.5   0.5 0.
SEG 1  3
SEG  3 2

END
```
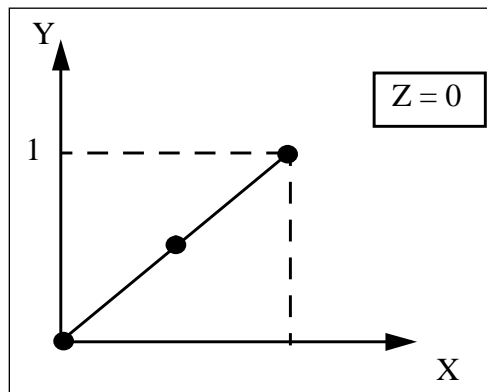
***Example 9:* An example of a PLine and its data file.**

B.6 TSOLID (SOLID)

The Ascii TSolid format inherits from the **GObj** *(page B-4)* and from the **Atomic** *(page B-6)* formats. The Atoms of the Atomic must be defined first. Each Tetrahedron in the Solid is then defined by its 4 apices in the following format.:

        TETRA *id1 id2 id3 id4*

where *ids* are the IDs of the already-defined Atoms.

For example, the ascii file corresponding to he Solid represented in Figure 10 may look like:
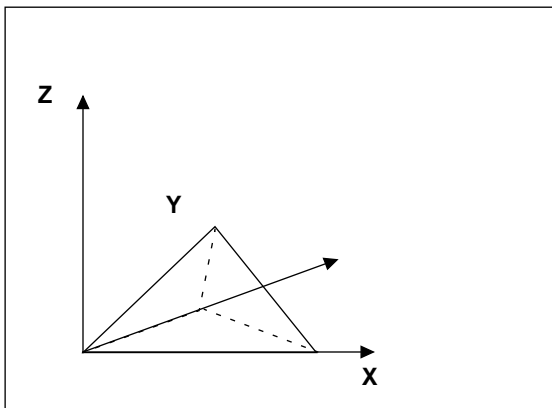

```
GOCAD TSolid 0.01
HEADER{
name: Pyramid
}

VRTX 1  0.  0.  0.
VRTX 2  1.  0.  0.
VRTX 3  0.  1.  0.
VRTX 3  0.5   0.5 0.8
TETRA 1 2 3 4
END
```

***Example 10:* An example of a TSolid and its data file.**

B.7 WELL

The Well ASCII format contains (in addition to the sections relative to a GObj definition) 4 sections: a header section, a wellpath section, a well curves (log) section and a zone/marker section. Well files are complex; contact **_jcdulac@t-surf.com_** immediately when you encounter problems.

B.7.1 WellPath

The WellPath is composed of a header and a coordinate section: The header defines the reference point and the datum plane. The coordinate section defines the well path by giving either a series of PATH definition or VRTX definition.

> WREF *X Y Z*
>
> DPLN *datum_plane*
>
> PATH *Zm Z dX dY*
>
> (VRTX *X Y Z*)

WREF gives the X and Y coordinates of the reference point for the well path; usually the surface location of the well. The Z value of the WREF is used as the depth for drawing the derrick and the 0 position of the Well Name. If the Well Path is given in the **VRTX** format, the **WREF** point is used as the Zm**=0** point to construct measured depths for displaying Well Curves and other information that is given in measured depth format. The **WREF_z** is not necessary the KB or RT of the well; and it is never subtracted from the **PATH_z** value.

PATH describes a point of the WellPath, giving a measured depth, Zm, a real depth (with KB or RT elevation **_ALREADY TAKEN OUT_**; the "real world" Z coordinate), Z and x, y deviations, dX and dY relative to the X and Y of the WREF point. The Z and dX, dY of the PATH is used to draw the well path, while the Zm is needed to interpolate a Curve point (REC, which is given in measured depth; see <u>WellCurve Internal Data</u> *(page B-19)*) position. The PATH format is the perferred format (GOCAD's output format).

VRTX describes a point of the WellPath given in absolute (real world) coordinates. In other words, **VRTX_X=PATH_dX+WREF_X**; **VRTX_Y=PATH_dY+WREF_Y**; **VRTX_Z=PATH_Z**. If the Well Path is given in the VRTX format, the **WREF** point is used as the Zm**=0** point to construct measured depths Well Curves

and other information that is given in measured depth format.

New path descriptions

Two new keywords will be introduced to the WellPath specification in the near future:

1. **TVSS_PATH**

   TVSS_PATH Zm Zss dX dY

   TVSS_PATH is almost as the same as the original PATH statement. GOCAD expects 4 values following this keyword, Zm Zss dX dY.

   However, the sign of Zss is still to be determined by the GOCAD Committee.

2. **TVD_PATH**

   TVD_PATH Zm Ztvd dX dY

   This format is designed to accommodate well path information that does not have KB or other elevation values subtracted from its directional survey. When GOCAD encounters the keyword **TVD_PATH**, it expects 4 values following this keyword, Zm Ztvd dX dY, and will subtract (or add) **WREF-Z** from Ztvd to be the Z value for that point.

   Again, the sign of Ztvd is still to be determined by the GOCAD Committee.

3. **PATH_CURVE_X**

   The WellPath information can also be specified as three special WellCurves. Each set of coordinates is stored in a WellCurve. Values in these files must be in real world coordinates. The WellPath is specified as follow:

   WREF X Y Z

   PATH_CURVE_X   name_of_well_curve_containing_X_information.

   PATH_CURVE_Y   name_of_well_curve_containing_Y_information.

   PATH_CURVE_Z   name_of_well_curve_containing_Z_information.

B.7.2 WellZone and WellMarkers

WellZone and WellMarkers data are both optional, and are given in the following format:

    ZONE *name Zm1 Zm2 index*
    MRKR *Marker_name measured_flag Zm*

[DIP *azimuth dip*]

[NORM *X Y Z]*

[MREF *horizon_name*]

[UNIT *rock_layer_name*]

Additional information (between [and]) can be attached to each Marker. This extra information includes the orientation (given as DIP or NORM) of the Marker, the Marker Reference Horizon (currently not utilized by GOCAD) and the Rock Unit (currently not utilized by GOCAD).

The **DIP** information *MUST* be given in *Grads*. What are Grads? It is an ancient French measurement of angles; there are 100 grads in a right angle (as opposed to 90 degrees in a right angle).

The **NORM** information is always given in XYZ, representing the normal vector of the Marker. This is the preferred representation for Marker orientation (this is the way GOCAD will output Marker orientation information, instead of DIP).

The optional information can be on the same line as the marker itself, or on lines that immediately follow the marker ascii code (MRKR.....).

New WellMarker dip specification

To accommodate the fact that no one uses Grads any more, GOCAD introduced a new DIP specification keyword, **DIPDEG**, where the two angles, azimuth and dip, are given in *degrees*.

DIPDEG azimuth dip

B.7.3 WellCurves

The WellCurves section consists of two parts, the **WellCurve Header** *(page B-19)* section and the **WellCurve Internal Data** *(page B-19)* section. The header section defines the format of the Well Curves. The WellCurve data can be given in the same file, or in an external binary or ascii file.

External Binary Data Declaration

If the WellCurve data are stored in an external binary file, you must provide the name of the file, *BEFORE* any WellCurve Header information, using the following statement:

BINARY_DATA_FILE *filename*

The above statement gives the name of the binary file in which Well Curves are stored as a series of Z (measured depth) values and data values:

>   *P1Z1, P1Z2, ...,P1Z20, P1V1, P1V2, ...P1V20, P2Z1, P2Z2, ...,P2Z12,*
>   *P2V1, P2V2, ..., P2V12, P3Z1, ...etc.*

External ASCII Data Declaration

If the data are stored in an external ASCII file, you must provide the following information before any WellCurve Header information:

>   ASCII_DATA_FILE *filename*
>   DEPTH_COLUMN *index*
>   NCOLUMNS *ncol*
>   NROWS *nrow*

The data are read in as a matrix of floating-point numbers and records are separated by blanks. The dimension of the matrix is defined by NCOLUMN and NROWs.Each column is a Property and NROWS specifies the number of data points per Property. The DEPTH_COLUMN specifies which column contains the measured depth data, which also means that in the external ASCII format, different Properties must all have data values at the same measured depth point.

For example, the following file, named *AsciiExample.wl.dat*,

P1V1	P2V1	P3V1	Z1	P4V1
P1V2	P2V2	P3V2	Z2	P4V2
P1V3	P2V3	P3V3	Z3	P4V3
.....				
P1200	P2V200	P3V200	Z200	P4V200

should be specified as

>   ASCII_DATA_FILE *AsciiExample*
>   DEPTH_COLUMN *4*
>   NCOLUMNS *5*
>   NROWS *200*

An optional no_data_value can be given to describe holes or invalid values. The format is given below:

>   NO_DATA_VALUE *whatever_u_like*

WellCurve Header

Each WellCurve is defined inside a block beginning with WELL_CURVE and ending with END_CURVE. The format of the header is:

    WELL_CURVE
    PROPERTY *type_name*
    UNITS *z_unit f_unit*
    .....
    END_CURVE

The PROPERTY defines the name of the well log. The UNIT statement defines the unit of the measured depth and the unit of the Property.

You can also define the Property Class of the Property using the following statement:

    PROPERTY_CLASS *property_class_name*

Defining the Property Class places the Property in its proper statistical database for future interactive manipulations inside GOCAD. Furthermore, the data will be transformed into the unit of the Property Class. If the Property Class is not given, the data will keep in the unit defined in the UNITS specification.

WellCurve Internal Data

If the WellCurve data are given in the same file as the WellCurve Header, the data values are given in the following format:

    REC *Zm V*

where Zm is the measured depth and V is the Property (log) value at that point.
If V is equal to the no_data_value (which must be declared before any Well Curve Header), the data is treated as missing. Alternatively, you can specify a HOLE, i.e. a discontinuity in the well curve. For example:

    REC 3. 1
    HOLE
    REC 10. 1

indicates that there is a hole (no data) in the curve between depth 3. and 10 and GOCAD will not inter-polate this particular Property between those tow point. See ***Single Well File Example*** *(page B-21)*.

WellCurve External ASCII Data

If a WellCurve data is defined in an external ASCII File, the WellCurve Data portion comprises only one line:

> COLUMN *ncol*

which tells GOCAD which column in the ASCII file contains the data points of this Property. This infor-mation along with the DEPTH_COLUMN information,

> DEPTH_COLUMN *index*

will be used to retrieve data from the ASCII file to create the WellCurve. See ***Well with an External ASCII Data File Example*** *(page B-23)*.

WellCurve External Binary Data

If a WellCurve is defined in an external binary file, the user must specify the starting point (in the binary file) of this Property and the number of data points in this Property by including the following statement in the WellCurve Header:

> SEEK *seekpos*
>
> NPTS *npts*

This information will be used to read *npts* z values and *npts* data values from the binary file starting at Byte position *seekpos*. See ***Well with an External Binary Data File Example*** *(page B-22)*.

B.7.4 Single Well File Example

The first example is a well file that includes the WellCurve data:

Well GObj Header	GOCAD Well 0.01
	HEADER {
	name:WellDemo1
	}
	WREF 0 0 0
	DPLN 0
	VRTX 1 0 1
Well Path	VRTX 1 .1 3
given in absolute XYZ	VRTX 1.2 .5 4
	VRTX 1.7 1.5 7
	VRTX 3.2 1.5 8
	VRTX 5 2.0 20
	VRTX 5.6 2.3 34.
	MRKR marker 1 2 0.3048*meters
Marker and Zone Info	NORM 0 0 1
	ZONE zone 4 7 1
First Well Curve Begins	WELL_CURVE
WellCurve Header	PROPERTY Pro1
	UNITS meters meters/seconds
WellCurve Data	REC 3 2
	REC 5 2.2
	REC 6 2.3
First Well Curve Ends	END_CURVE
	WELL_CURVE
	PROPERTY Pro2
	UNITS meters meters/seconds^2
	REC 3 60
	REC 6.5 67
Second Well Curve	REC 7 17
	REC 20 58
	END_CURVE
	END

***Example 11:* Single Well File format..**

B.7.5 Well with an External Binary Data File Example

The second example is a well with the WellCurve data stored in the external binary file described in
External Binary Data Declaration *(page B-17)*:

Well GObj Header	GOCAD Well 0.01
	HEADER {
	name:WellDemoBinary
	}

Binary File Specification	BINARY_DATA_FILE WDB.wl.dat
Well Path given in absolute XYZ	WREF 0 0 0 DPLN 0 VRTX 1 0 1 VRTX 1 0 20
Marker and Zone Info	MRKR marker 1 2 NORM 0 0 1 ZONE zone 4 7 1
First Well Curve Begins	WELL_CURVE
WellCurve Header	PROPERTY type1 UNITS meters meters/seconds
Binary File References	NPTS 10 SEEK 0
First Well Curve Ends	END_CURVE
Second Well Curve	WELL_CURVE PROPERTY type2 UNITS meters meters/seconds^2 NPTS 12 SEEK 80 END_CURVE .. END

Example 12: **Well with an External Binary Data File.**

B.7.6 Well with an External ASCII Data File Example

The third example is a well with the WellCurve data stored in the external ASCII file described in **External Binary Data Declaration** *(page B-17)*:

Well GObj Header

    GOCAD Well 0.01
    HEADER {
    name:WellDemoAscii
    }

ASCII File Specification

    ASCII_DATA_FILE AsciiExample.wl.dat
    DEPTH_COLUMN 4
    NCOLUMNS 5
    NROWS 200

**Well Path
given in absolute XYZ**

    WREF 0 0 0
    DPLN 0
    VRTX 1 0 1
    VRTX 1 0 20

Marker and Zone Info

    MRKR marker 1 2
    NORM 0 0 1
    ZONE zone 4 7 1

First Well Curve Begins   →   WELL_CURVE
WellCurve Header   →   PROPERTY Property1
                                  UNITS meters meters/seconds
ASCII File Reference   →   COLUMN 1
First Well Curve Ends   →   END_CURVE

    ..

Last Well Curve

    WELL_CURVE
    PROPERTY Property4
    UNITS meters meters/seconds^2
    COLUMN 5
    END_CURVE

    END

Example 13: **Well with an External ASCII Data File.**

B.8 GRID3DVOXET

A Voxet is a rigid 3D Grid Object; it can carry multiple Properties. A Property in a Voxet is often referred to as a grid3d. A Voxet is derived from the **GObj** *(page B-4)* Class.

A common confusion comes from not realizing that UVW can mean the (i,j,k) indexing of the Voxet Nodes, but it can also mean the coordinate system in which the Voxet resides.

In addition to its inherited Gobj file elements, a Voxet is further defined in three parts: the header section (which defines the geometry), the grid3d sections (which defines the Properties, grid3ds) and the Region section (which gives the Region storage and Region information). Each Property in the Voxet has its own grid3d section that defines the Property and Property values.

B.8.1 Geometry Section

```
                                          AXIS_O OX OY OZ
Voxet Coordinate System                   AXIS_U UX UY UZ
UVW                                       AXIS_V VX VY VZ
given in XYZ space                        AXIS_W WX WY WZ
                                          AXIS_MIN UMin VMin WMin
        Voxet Geometry                    AXIS_MAX UMax VMax WMax
        given in UVW Space                AXIS_N NU NV NW
                                          AXIS_D DU DV DW
                                          AXIS_UNIT "Unit_U" "Unit_V" "Unit_W"
Voxet Axes Optional Info                  AXIS_NAME "Name_U" "Name_V" "Name_W"
                                          AXIS_TYPE U_type V_type W_type (either
                                          even or uneven)
Axis-U Sampling Points                    AXIS_COORD_U ...
Axis-V Sampling Points                    AXIS_COORD_V ...
Axis-W Sampling Points                    AXIS_COORD_W...
```

Example 14: **Voxet Geometry format.**
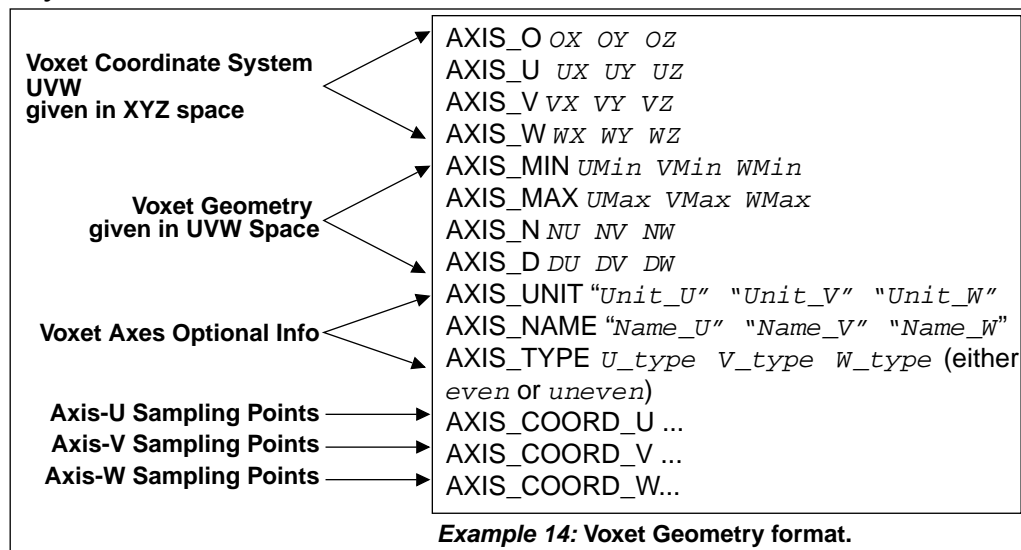
UVW Coordinate System and Voxet Dimension

A Voxet is never directly defined in the XYZ space. The user must first define the UVW coordinate sys-

tem (which may be identical to the XYZ space) in terms of the XYZ coordinate system; then define the Voxet geometry in the UVW space. See Example 14 *(page B-24)* and Example 15 *(page B-25)*.
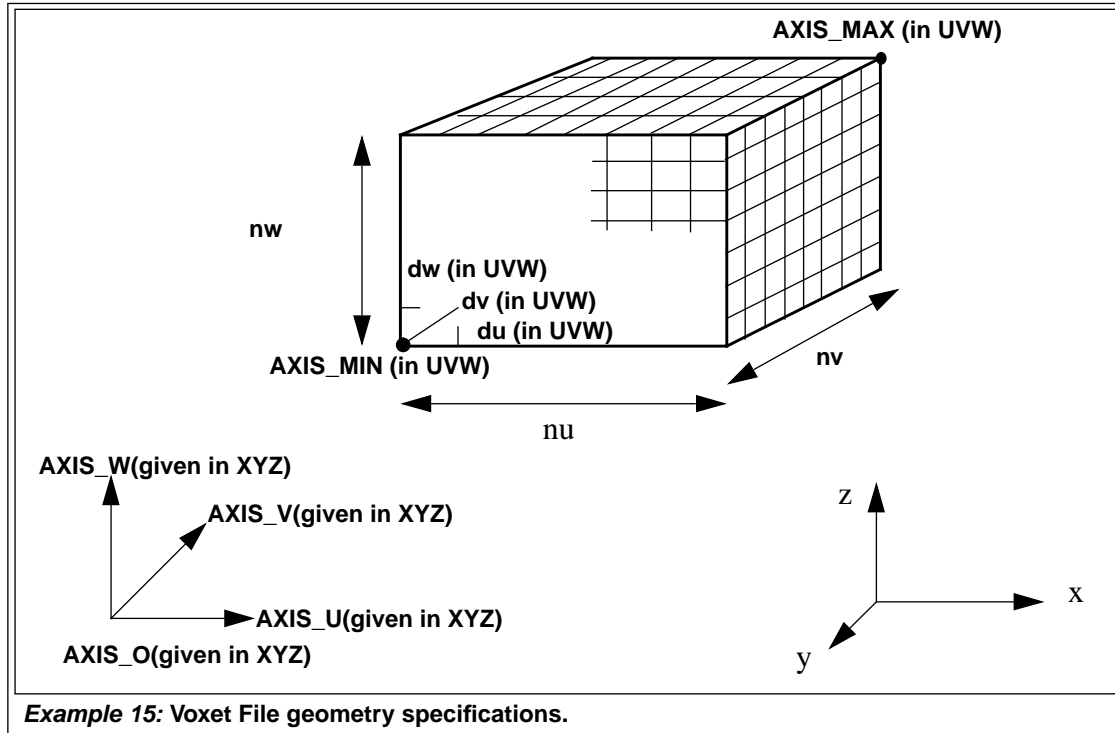
***Example 15:* Voxet File geometry specifications.**

AXIS_O, **AXIS_U, AXIS_V, AXIS_W** are required information; they define the UVW coordinate system in which the Voxet resides.

AXIS_MIN and **AXIS_MAX** are also required information; they define the dimension of the Voxet in the UVW space.

Either **AXIS_D** or **AXIS_N** must be present; it (they) defines the grid structure (node-spacing or number of nodes).

AXIS_O specifies the origin of the Voxet coordinate system. **AXIS_U, AXIS_V,** and **AXIS_W** represent the direction and the length of the vectors, U, V and W respectively in the (X,Y,Z) coordinate system (See Example 15 *(page B-25)*).

A common confusion comes from not realizing that this set of vectors, **AXES_OUVW**, only defines the UVW coordinate system in which the Voxet resides; they do *__NOT__* define the dimension of the Voxet. The dimension of a Voxet is defined by another set of parameters, **AXES MIN-MAX**.

AXIS_MIN defines (in terms of UVW) the origin of the Voxet. **AXIS_MAX** defines (in terms of UVW) the maximum point of the Voxet. In other words, **AXIS_MIN** and **AXIS_MAX** define the bounding box of the Voxet in the (U,V,W) space. **AXIS_MAX** does *__NOT__* have to be literally greater than **AXIS_MIN**.

Example 16: **Multiple Voxets in the same UVW coordinate system.**
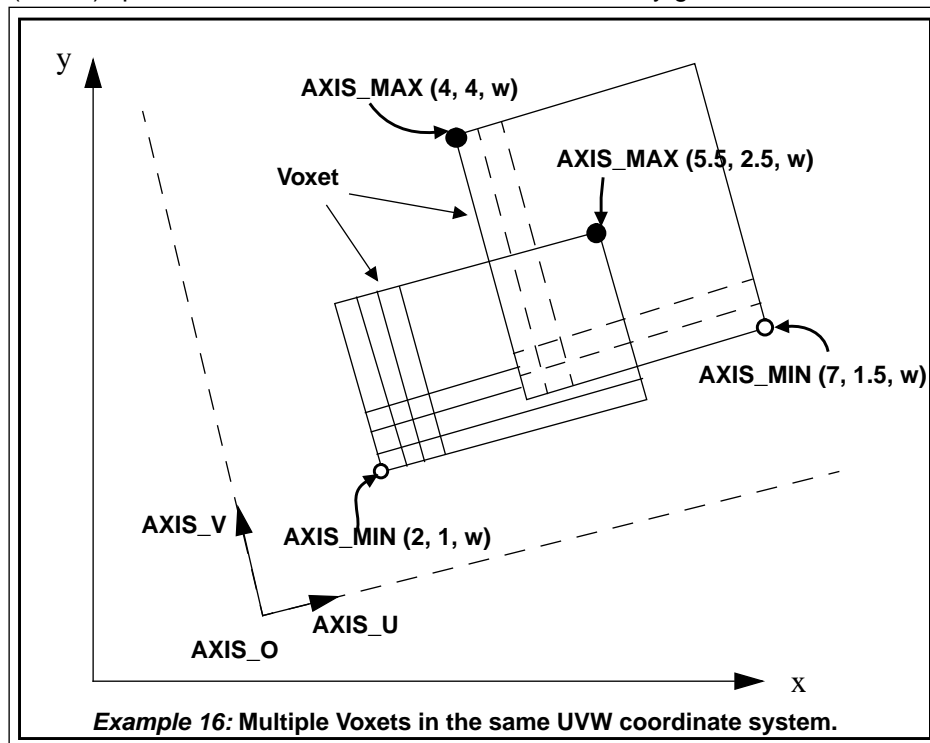
These two sets of specifications, **AXES UVW** and **AXES MIN-MAX**, are not unique. For example, you can specify **AXIS_UVW** to be (1, 0, 0) (0,1, 0)(0, 0,1) and **AXIS_MIN** and **MAX** to be (0,0,0) (10, 8, 6);

Or you can specify AXIS_UVW to be (10, 0, 0) (0, 8, 0)(0, 0, 6) and AXIS_MIN and AXIS_MAX to be (0,0,0) (1, 1, 1);

Or you can specify AXIS_UVW to be (5, 0, 0) (0, 2, 0)(0, 0, 2) and AXIS_MIN and AXIS_MAX to be (0,0,0) (2, 4, 3).

These three specifications are identical (in terms of the Voxet dimension). See <u>Example 17</u> *(page B-27)*.

For practical reasons (less confusing), most users prefer to set the **AXES UVW** to be the dimension of a Cell (data spacing), or to be the dimension of the entire Voxet volume; as shown in the two upper examples in Example 17 *(page B-27)*. The third example in Example 17 *(page B-27)* is rarely utilized.

AXIS_V (0,8,0)
Y 9
V
AXIS_MAX (1,1,w)
AXIS_O 0 1 2
AXIS_U 10 0 0
AXIS_V 0 8 0
AXIS_U (10,0,0)
AXIS_W 0 0 6
AXIS_MIN (0,0,0)
AXIS_MIN 0 0 0
AXIS_MAX 1 1 1
AXIS_N 11 9 6
1
U
AXIS_O (2,1,2)
X
2   12

Y 9
AXIS_MAX (10,8,w)
AXIS_O 0 1 2
AXIS_U 1 0 0
AXIS_V 0 1 0
AXIS_V (0,1,0)
AXIS_W 0 0 1
AXIS_MIN 0 0 0
V   AXIS_MIN (0,0,0)
AXIS_MAX 10 8 6
AXIS_U (1,0,0)
1
AXIS_N 11 9 6
U
AXIS_O (2,1,2)
X
2   12

Y 9
AXIS_MAX (2,4,w)
AXIS_O 0 1 2
AXIS_U 5 0 0
AXIS_V 0 2 0
AXIS_W 0 0 2
AXIS_V (0,2,0)
V   AXIS_MIN (0,0,0)
AXIS_MIN 0 0 0
AXIS_U (5,0,0)
AXIS_MAX 2 4 3
AXIS_N 11 9 6
1
U
AXIS_O (2,1,2)
X
2   12

Example 17: **An example of three specifications that give the identical Voxet geometry.** The upper two are the most common cases; i.e. the AXES UVW either define the entire volume dimension or they define a cell dimension. When GOCAD outputs a Voxet file, it uses the upper most format; i.e. the UVW axes define the entire volume. Do not confuse Voxet grid lines with Voxet cells. Grid lines connect Voxet nodes while Voxet cells are centered at each Node. Shown above are grid lines, not cells.

Voxet Grid Density

AXIS_D and **AXIS_N** are redundant. Only one needs to be provided.

AXIS_N (NU, NV, NW) defines the number of points in the Voxet along each of the three axes.

AXIS_D (DU, DV, DW) defines the node-spacing (cell-dimension, sampling rate) along the three axes of the Voxet in the UVW space.

This following equation describes the relationship between these two parameters:

(NU-1)*DU = AXIS_MAX (UMax) - AXIS_MIN (UMax)

One thing to remember is that Voxet cells are centered at each data point, so the number of cells along each axis is the number of data points along each axis, _**NOT**_ one less.

Each axis can be unevenly sampled and the sampled point coordinates are defined in the **AXIS_COORD_{U,V,W}** section. However, uneven sampling is not implemented yet.

B.8.2 Voxet Regions

Currently, Region information _**MUST**_ be given before Property information.

Region and connectivity information in a Voxet is saved on a per data point basis (values in the file increase first in U, then V, then W) in an external binary file with the default name "__**flags@@**". For each Region, a point is either in it or not in it.

In the main file itself, there are two sections describing the Region-related information.

The first section defines how the Region information for each Voxet point is stored in the external file.

FLAGS_ARRAY_LENGTH length_of_the _array

FLAGS_BIT_LENGTH number_of_bits (never less than 7, the first 6 bits for connectivity info)

FLAGS_ESIZE number_of_bytes (should equal (number_of_bits+1)/8 rounded up)

FLAGS_OFFSET offset_inside_the_binary_file_for_the_first_data_point

FLAGS_FILE binary_file_name (values in the file increase first in U, then V, then W)

The second section gives the name of each Region and its flag bit position.

REGION region_name region_bit_marker_position_number (must be less than number_of_bits)

....

An example of an Voxet ASCII file with Region information is given in <u>Example 18</u> *(page B-29)*.:

Voxet GObj Header

GOCAD Voxet 0.01
HEADER {
name:MyVoxet
}

Voxet UVW Specification

AXIS_O 0 0 0
AXIS_U 1 0 0
AXIS_V 0 1 0
AXIS_W 0 0 1

Voxet Dimension and Density (Grid Structure)

AXIS_MIN 9 9 9
AXIS_MAX 0 0 0
AXIS_N 10 10 10

Optional Axis Info

AXIS_NAME "X" "Y" "time"
AXIS_UNIT none none none
AXIS_TYPE even even even

....

Region External File Specification

FLAGS_ARRAY_LENGTH 1000
FLAGS_BIT_LENGTH 11
FLAGS_ESIZE 2
FLAGS_OFFSET 0
FLAGS_FILE MyVoxet__flags@

Region Name and flag position

REGION RegionExample 7
REGION HighPorosity 8
...
REGION LowPerm 9
REGION BadBuy 10

***Example 18:* Voxet Region specificaitons.**

B.8.3 Grid3d/Property Section

The grid3d section defines the properties in the Voxet. Currently, Region information **_MUST_** be given before Property information.

For each property there is a unique identifier, *id*, which is used to relate different Property statements to the same Property. The data can be in an external file in binary format, or inside the file in ascii format.

For each Property, the declaration must be the first line (PROPERTY *id "property name"*) and the Property file name (PROP_FILE *id filename*) or data (DATA) must be the last line.

> PROPERTY *id "property name"*
>
> PROPERTY_CLASS *id "property-class_name"*
>
> PROP_UNIT *id "unit"*
>
> PROP_LEGAL_RANGE *id min max* (where *min/max* is a float or **none**)
>
> PROP_SAMPLE_STATS *id n x x2 min max* (# of samples, mean, sigma, min, max of current values)
>
> PROP_NO_DATA_VALUE id value
>
> PROP_ETYPE *id data_type* (either IEEE or IBM)
>
> PROP_FORMAT *id file_format* (either RAW or SEGY)
>
> PROP_ESIZE *id element_size* (either 1 or 4)
>
> PROP_OFFSET *id offset*
>
> PROP_FILE *id filename*
>
> DATA

PROP_OFFSET indicates that the array of float value begins at the given offset in the **PROP_FILE**.

PROP_ETYPE specifies the type of floating point value that is in the file. It can be IBM floating point value or IEEE floating point value.

PROP_FORMAT specifies the format of the file. SEGY format implies that the data resides in a standard SEGY file. RAW specifies that the data is formatted as a C array where the fast axis is the Axis_U.

The keyword **DATA**, if present, indicates that the Property data are to be read following that token. If this is the case, the floating point array of data must be in ascii RAW format, and no Property File specifications are necessary (**PROP_OFFSET**, etc.).

See Example 19 *(page B-31)*.

Voxet GObj Header	GOCAD Voxet 0.01
	HEADER {
	name:VoxetInProperty
	}
Voxet UVW Specification	AXIS_O 0 0 0
	AXIS_U 10 0 0
	AXIS_V 0 10 0
	AXIS_W 0 0 10
Voxet Dimension and Density (Grid Structure) Specifications	AXIS_MIN 0 0 0
	AXIS_MAX 1 1 1
	AXIS_N 2 2 2
First Property Declaration	PROPERTY 1 OnlyEight
First Property Data values	DATA
	1 2 3 4 5 6 7 8
	END

Example 19: **Voxet with an internal Data File.**

An example of a Voxet with an external Property file.

Voxet GObj Header

GOCAD Voxet 0.01
HEADER {
name:VoxetExProperty
}

Voxet UVW Specification

AXIS_O 2128403. -79200. 0.
AXIS_U 0. 0. 55.
AXIS_V 0. 220. 0.
AXIS_W 220. 0. 0.

Voxet Dimension and Density (Grid Structure) Specifications

AXIS_MIN 0.0 0.0 0.0
AXIS_MAX 549. 227. 149.
AXIS_N 550 228 150
AXIS_D 1. 1. 1.

Optional Axes Info

AXIS_NAME "Z" "Y" "X"
AXIS_UNIT "m" "m" "m"
AXIS_TYPE even even even

First Property Declaration

PROPERTY 1 "Seismic"

Optional First Property Info

PROPERTY_CLASS 1 "amp"
PROP_UNIT 1 none
PROP_NO_DATA_VALUE 1 -999
PROP_LEGAL_RANGE 1 -128 127
PROP_SAMPLE_STATS 1 1000 0 2 -112 109
PROPERTY_CLASS_HEADER 1 "amp" {
*low_clip:-128
*high_clip:128
*pclip:100
}

First Property External File Specificaion

PROP_ESIZE 1 4
PROP_ETYPE 1 IEEE
PROP_FORMAT RAW
PROP_OFFSET 1 0
PROP_FILE 1 VoxetExProperty_Seismic__@@

Example 20: **Voxet with an external Data File.**

B.9 STRATIGRAPHIC GRID

The geometry of an SGrid is defined by a cube of Point3d (a set of points in space) and its Properties by a Cube of floating point values. In addition to the inherited *GObj* *(page B-4)* file elements, an SGrid file consists of three sections: the header section (which gives the geometry storage information), the region section (which defines Region storage information and Region information), and the property section (which defines Property and Property storage information).

Be aware of the deadly differences between SGrid terminology and Voxet terminology!

B.9.1 Header Section

SGrid GObj Header	GOCAD SGrid 0.01
	HEADER {
	name:MySGrid
	}

SGrid Dimension	AXIS_N 5 5 5
SGrid Property Point type	PROP_ALIGNMENT CELLS
SGrid Poind3d **External File**	POINTS_OFFSET 0 POINTS_FILE MySGrid__points@@
SGrid Split Node **External File**	FLAGS_OFFSET 0 FLAGS_FILE MySGrid__flags@@ END

Example 21: **An SGrid Geometry format example.**
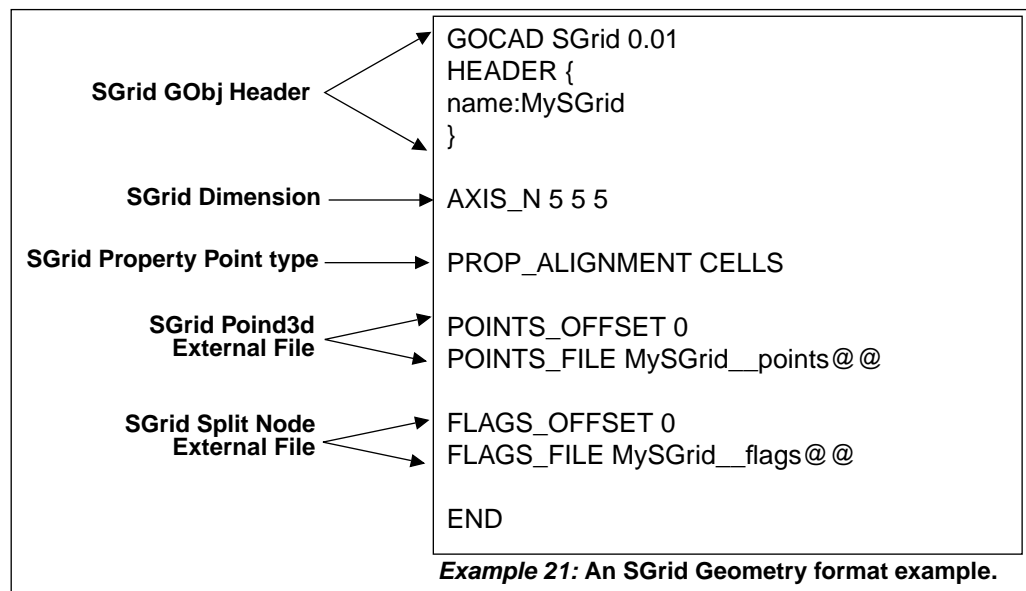
The statement

    AXIS_N *NU NV NW*

specifies the dimension of the stratigraphic grid. This dimension will be the dimension of the Point3d array (which specifies the location of each node of the SGrid) and more or less the dimension of the Property arrays (which specify properties).

The Point3d array is read in from a file specified by the POINTS_FILE, and the starting position of the Point3d array is given by POINTS_OFFSET. The size of the array is NU*NV*NW.

> POINTS_OFFSET *offset*

> POINTS_FILE *filename*

The statements

> FLAGS_OFFSET offset

> FLAGS_FILE filename

specify the filename and offset in the file where the flags array lies. The flags array contains the **split-node** information. This is _**DIFFERENT**_ from a Voxet where the **FLAGS** information is related to the **Regions**.

The statement

> PROP_ALIGNMENT   (CELLS or POINTS)

specifies whether the Stratigraphic Grid is cell -centered or corner-point.

If the SGrid is cell-centered, the size of the point3d array will be NU*NV*NW but the size of each of the Property arrays will be (NU-1)*(NV-1)*(NW-1).

If the SGrid is a corner-point SGrid, the size of the property array will be NU*NV*NW, the same as the Point3d array.

B.9.2 SGrid Region Format

Currently, Region information _**MUST**_ be given before Property information.

Region information in an SGrid is saved on a per data point basis (values in the file increase first in U, then V, then W) in an external binary file with the default name "__**region_flags@@**". For each Region, a point is either in it or not in it.

In the main file itself, there are two sections describing the Region-related information.

The first section gives the name of each Region and its flag bit position in the external file.

> REGION region_name region_bit_marker_position_number (must be less than number_of_bits)

The second section defines how the Region information for each Voxet point is stored in the external

file.

REGION_FLAGS_ARRAY_LENGTH length_of_the _array

REGION_FLAGS_BIT_LENGTH number_of_bits (at least 1 greater than the number of Regions)

REGION_FLAGS_ESIZE number_of_bytes (should equal (number_of_bits+1)/8 rounded up)

REGION_FLAGS_OFFSET offset_inside_the_binary_file_for_the_first_data_point

REGION_FLAGS_FILE binary_file_name (values in the file increase first in U, then V, then W)

An example of an SGrid file format with Property and Region information is given below:

SGrid GObj Header	GOCAD Voxet 0.01
	HEADER {
	name:MySGrid
	}
SGrid Geometry Specification	AXIS_N 10 10 10
	PROP_ALIGNMENT CELLS
	POINTS_OFFSET 0
	POINTS_FILE MySGrid__points@@
	FLAGS_OFFSET 0
	FLAGS_FILE MySGrid__flags@@
Region Name and flag position	REGION Reg_top_1 1
	REGION Reg_1_2 2
	REGION Reg_2_bot 2 3
	REGION Facies_Region_1 4
Region External File Specification	REGION_FLAGS_ARRAY_LENGTH 26691
	REGION_FLAGS_BIT_LENGTH 5
	REGION_FLAGS_ESIZE 1
	REGION_FLAGS_OFFSET 0
	REGION_FLAGS_FILE MySGrid__region_flags@@
Property Information	PROPERTY 1 "Geol_Facies_1"
	PROPERTY_CLASS 1 "facgm"
	....

***Example 22:* SGrid Region specifications.**

B.9.3 Property Section

The Property section defines the property information. Currently, Region information **_MUST_** be given before Property information.

Each Property has its unique identifier, id, which is used to relate different Property statements to that Property. The data can be in an external file in a binary format, or inside the header file in ascii format.

The definitions of the Property format is identical to Voxet. Please see B.8.3 <u>Grid3d/Property Section</u> *(page B-30)*.

        PROPERTY id "$property\ name$"

        PROPERTY_CLASS id "$property\_class\_name$"

        PROP_UNIT id "$unit$"

        PROP_LEGAL_RANGE id min max (where min/max is a float or none)

        PROP_NO_DATA_VALUE id $value$

        PROP_SAMPLE_STATS id n x $x2$ min max

        PROP_ETYPE id $data\_type$ (must be IEEE or IBM)

        PROP_FORMAT id $file\_format$ (either RAW or SEGY)

        PROP_ESIZE id $element\_size$ (either 1 or 4)

        PROP_FILE id $filename$

        PROP_NO_DATA_VALUE id no_data_value

        PROP_OFFSET id $offset$

        DATA

The fast axis is Axis_U. The data are specified as a C array.

If the SGrid is cell-centered, the size of the point3d array will be NU*NV*NW but the size of each of the Property arrays will be (NU-1)*(NV-1)*(NW-1).

If the SGrid is a corner-point SGrid, the size of the property array will be NU*NV*NW, the same as the Point3d array.

B.9.4 Example

An example of an SGrid with one Region, one Property and external binary files (lines that start with

an * are optional Attribute lines):

SGrid GObj Header	GOCAD SGrid 0.01
	HEADER {
	name:s3
	*skin:off
	}
SGrid Geometry	AXIS_N 32 32 32
	PROP_ALIGNMENT CELLS
Specification	POINTS_OFFSET 0
	POINTS_FILE s3__points@@
	FLAGS_OFFSET 0
	FLAGS_FILE s3__flags@@
Region Name and flag position	REGION BM_Inactive_Region 1
	REGION_FLAGS_ARRAY_LENGTH 26691
	REGION_FLAGS_BIT_LENGTH 2
Region External File Specification	REGION_FLAGS_ESIZE 1
	REGION_FLAGS_OFFSET 0
	REGION_FLAGS_FILE s3__region_flags@@
Property Declaration	PROPERTY 1 "SGS_simulation_1"
	PROPERTY_CLASS 1 "porndx"
Optional Property Info	PROP_UNIT 1 none
	PROP_SAMPLE_STATS 1 62883 0.24624 0.000661522 0 0.415
	PROP_ESIZE 1 4
	PROP_ETYPE 1 IEEE
Property External File Specifications	PROP_ALIGNMENT 1 CELLS
	PROP_OFFSET 1 0
	PROP_FILE 1 s3_SGS_simulation_1@@
	END

Example 23: **SGrid Region specifications.**

B.9.5 Ascii External File

The geometry and the properties can both be read from an Ascii data file. This file must be specified in the header as follows:

> ASCII_DATA_FILE *filename*

The format of that file is:

> x y z p1 p2 p3 ... flag i j k

where x, y, z specify the location of that node; p1, p2, p3, ... are the Property values at that node; flag specifies the connectivity flag at that node; and i, j, k specifies the uvw index of the node.

If the SGrid is cell-centered, Property values are then attached to the node in that cell that has the lowest ijk indices. Those nodes that do not have Property values attached to them will get No_data_value in those Property fields. When the file is read in, GOCAD knows how to sort things up.

B.10 MODEL3D

This file specification has not been fully verified. Contact *dulac@ensg.u-nancy.fr* if you encounter difficulties.

A Model file contains the description of its Model Members and its Property (Property Model).

B.10.1 Model Members

A Model3D consists of Surfaces, TFaces, Regions, Layers, Faultblocks and Property-Model database. The ASCII description of the Model therefore consists of 5 parts:

1. **TSURF description**

    TSURF tsurf_name

2. **TFACE description**

    TFACE identifier X1 Y1 Z1 X2 Y2 Z2 X3 Y3 Z3

    X1,Y1,Z1,... describes a key triangle of the TFace. The TFace will be then referenced inside the Regions description by the unique identifier.

3. **REGION description**

    REGION identifier {[+-]TFACE_identifier] } 0

    Each Region is given a unique identifer, and the list of the oriented TFaces composing this Region. The list of tfaces ends with a 0. The orientation is specified by either a + or a - sign.

4. **LAYER description**

    LAYER layer_name {region_identifier} 0

    Each Layer is given a name and the list of Regions which compose the layer. The list ends with a 0.

5. **FAULTBLOCK description**

    FAULTBLOCK faultblock_name {region_identifier} 0

    A faultblock is given a name and the list of regions which compose the faultblock. The list ends with a 0.

B.10.2 PROPERTY_MODEL_DB description

Definitions of a ModelProperty and its Variables are given in a block that begins with BEGIN_PROPERTY_MODEL_DB_{ , ends with }_END_PROPERTY_MODEL_DB.

Since a Property in a Model3d Object is defined on a per Property per Layer basis, the DataBase is described using a Style syntax. The Style syntax allows us to easily describe the scope of the definition. The rules are as follows:

> *DB* : the following definitions are global to the DataBase.
>
> *LDB*layer_name* : the following definitions are given for al(never less than 7, the first 6 bits for connectivity info)l Properties in the specified Layer.
>
> *PDB*property_name* : the following definitions are valid for the particular property in all layers.
>
> *PLDB*propery_name*layer_name*: the following definitions are defined for a particualr Property in a particular Layer.

For example:

> *DB*V0*type: constant
> *DB*V0*value: 1800.0

defines the variable $V0$ inside the DataBase to be a constant equal to 1800. Similarly,

> *LDB*Layer1*Vs*type:constant
> *LDB*Layer1*Vs*value:1800.0

defines the variable Vs inside the Layer $Layer1$ to be a constant equal to 1800.

B.10.3 Example

> GOCAD Model3d 0.01
> HEADER {
> name:saltmodel
> tsurf_file_is_separate:true
> }
> TSURF Box
> TSURF F1
> TSURF H1

```
TSURF H2
TSURF H3
TSURF H4
TSURF H5
TSURF H6
TFACE 1 none Box
 17014.9824 17460.1855 -2548.08936
 16134.4531 17460.1855 -1782.04761
 17014.9824 17460.1855 -1016.00592
TFACE 2 none Box
 18776.043 15775.0977 5112.32764
 18776.043 17460.1855 5112.32764
 17895.5117 16617.6406 5112.32764
......
REGION 21
 +1  +10 -10 +4 +13
 -13 +3 +18 -18 +5
 -9 +9 +2 0
REGION 22
 -1 +11 +15 -8 -19  -20 0
REGION 23
 -2 +8 0
SURFACE Box
 1 3 4 5 2 0
SURFACE F1
 7 6 0
........
LAYER UNIVERS
 21 0
LAYER Layer1
 22 0
LAYER TOP
 23 0
```

LAYER Salt
 24 27 28 0
......
FAULT_BLOCK no_fault_block_name
 21 22 23 24 25
 26 27 28 29 30 0
....
PROPERTY Vp PROPERTY_CLASS vp
PROPERTY Vs PROPERTY_CLASS vs
BEGIN_PROPERTY_DB_{
*DB*variables:Vs Vp K V0 Z0
*DB*Vs*type:c_script
*DB*Vs*arguments: Vp
*DB*Vs*value:{Vs=1.5*Vp;}
*DB*K*type: constant
*DB*K*value: 0.4
*DB*V0*type: constant
*DB*V0*value: 1800.0
*DB*Z0*type: constant
*DB*Z0*value: 500.0
*DB*Vp*type: linear
*DB*Vp*arguments: V0 K Z0
*LDB*Layer1*variables: V0 K Vs
*LDB*Layer1*V0*type: constant
*LDB*Layer1*V0*value: 2500.0
*LDB*Layer1*K*type: constant
*LDB*Layer1*K*value: -0.400000
*LDB*Layer1*Vs*type: constant
*LDB*Layer1*Vs*value: 1800.0
*LDB*Layer2*variables: Vp Z0
*LDB*Layer2*Z0*type: top_shoot
*LDB*Layer2*Z0*target: Layer1
*LDB*Layer2*Z0*property:Z

```
*LDB*Layer2*Vp*type: c_script
*LDB*Layer2*Vp*arguments:  V0 K Z0 Z
*LDB*Layer2*Vp*value: {Vp=V0+K*(Z-Z0);}
*LDB*Salt*variables: Vp
*LDB*Salt*Vp*type: constant
*LDB*Salt*Vp*value: 4800.0
*LDB*Shell2*variables: Vp
*LDB*Shell2*Vp*type:    constant
*LDB*Shell2*Vp*value:   4800.0
*PDB*Vs*variables:     Vi
*PDB*Vs*Vi*type:       interpolation
*PDB*Vs*Vi*top:        Layer1
*PDB*Vs*Vi*bottom:     Salt
*PDB*Vs*Vi*property:   Vp
*PLDB*Vs*Layer2*variables: Vs
*PLDB*Vs*Layer2*Vs*type:   c_script
*PLDB*Vs*Layer2*Vs*arguments: Vi
*PLDB*Vs*Layer2*Vs*value:  {Vs=Vi+2000.0;}
}_PROPERTY_MODEL_DB
```

END
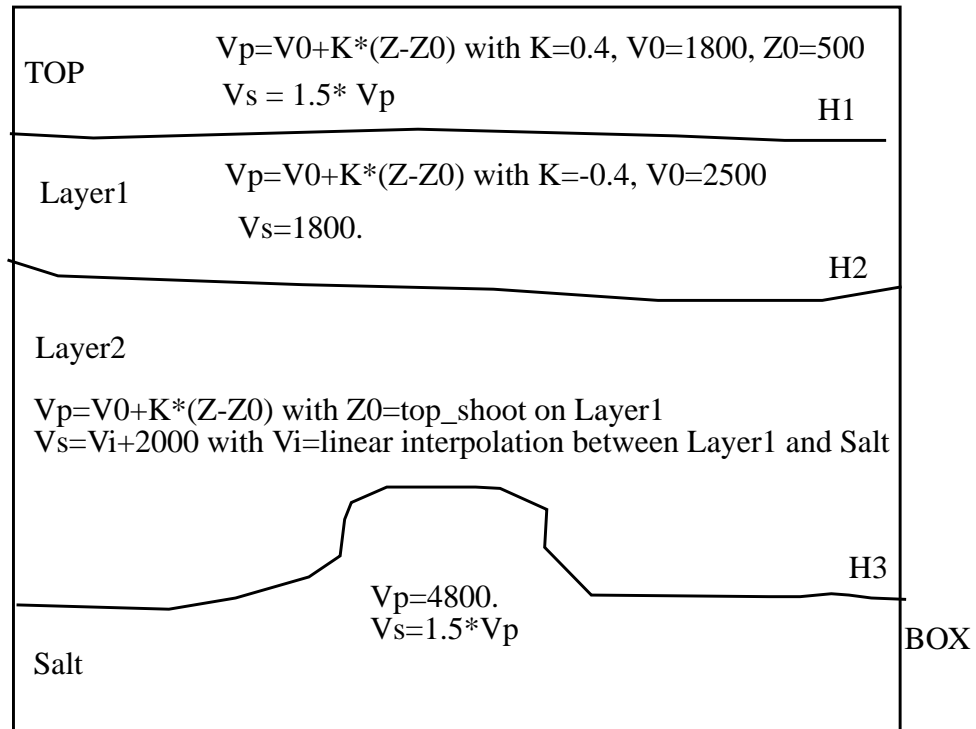
Example 24: **Illustration of the example given in B.10.3** Example
(page B-40) .

B.11 GSHAPE

A GShape is a Curve with Sections defined by Ribs along the Atoms of the Curve. Its file format there-fore consists of a ***PLine (Curve)*** *(page B-13)* and

> NBRIB *nbribs*
>
> TOP *rib_begin rib_end*
>
> SEC *atom_id x0 y0 z0 x2 y2 z2 .... x_nbribs-1 y_nbribs-1 z_nbribs-1*

NBRIBS defines the number of ribs at each Atom of the BackBone (the Curve).

TOP defines the top portion of the GShape and is optional information.

SEC gives the XYZ coordinates of the vectors that define the profile of the GShape at the specified Atom of the BackBone.

CSEC specifies that the Section is a Control Section.

```
GOCAD GShapeBundle 0.01
HEADER {
name:ThreeOnly
}
GEOLOGICAL_TYPE top
#ILINE
VRTX 1 0 9 0
VRTX 2 0 3 0
VRTX 3 0 0 0
SEG 1 2
SEG 2 3
NBRIB 4
TOP 0 1
SEC 1
      1 0 1
      -1 0 1
      -1 0 -1
      1 0 -1
SEC 2
      1 0 1
      -1 0 1
      -1 0 -1
      1 0 -1
SEC 3
      1 0 1
      -1 0 1
      -1 0 -1
      1 0 -1
CSEC 2
END
```
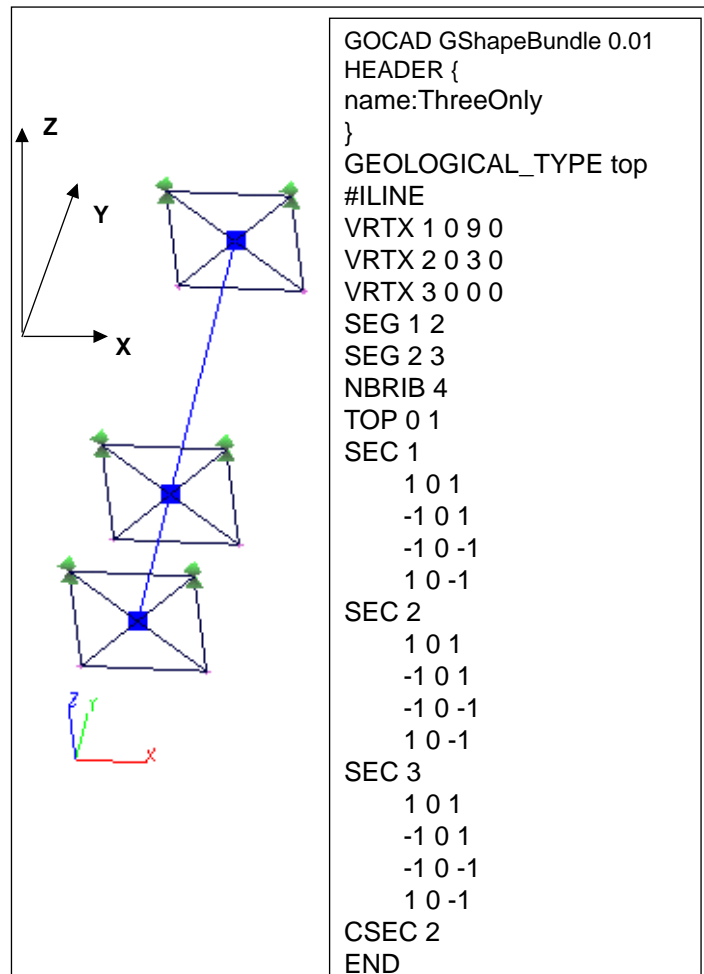
Example 25: **An example of a GShape and its data file.**

B.12 GEOSTATISTICAL FILES

GOCAD's goal is to provide geostatistical applications for users who are already familiar with the terminology and concepts of geostatistics. Therefore we assume that the user does not need detailed definitions of the variables used in this section, such as CUT_OFF, COVARIANCE_MODEL, etc.

Almost all geostatistical applications in GOCAD requires one or more ASCII files that contain variogram and/or other geostatistical information.

This section is based on information provided by Tom Tran. If you have any questions about this or other GOCAD geostatistical applications, please contact him directly at jtttr@chevron.com.

These file formats change rapidly as the geostatistical applications are being modified and developed rapidly. If your files are not accepted by GOCAD during an application, please contact Tom Tran at jtttr@chevron.com or Arben Shtuka at shtuka@ensg.u-nancy.fr.

B.12.1 An Example of a Variogram File

```
===========================================================================
# coordinate system can either be XYZ, XYW, or UVW

# in this example, XYW is chosen; therefore, the areal correlation

# ranges are in real-world coordinates and the vertical correlation

# range is in normalized (0 to 1) coordinate

COORDINATE_SYSTEM XYW

# not used

KRIGING_TYPE 0

# maximum number of nearby data for kriging

MAX_CLOSE 16

# 0 = simple kriging;  1 = ordinary kriging

KRIGING_OPTION 0

# search ellipsoid
```

generally, one should make the ranges of the search ellipsoid
larger than the correlation ranges
format: angle1 angle2 angle3 range1 range2 range3
SEARCH_ELLIPSOID 0. 0. 0. 4000. 4000. 0.4
covariance model
format COVARIANCE_MODEL sill number_of_nested_structures
model_type angle1 angle2 angle3 range1 range2 range3 contribution
model type can either be SPHERICAL EXPONENTIAL GAUSSIAN or POWER
in this example, there is only 1 nested structure, so only one model line
is needed
COVARIANCE_MODEL 1. 1
SPHERICAL 0. 0. 0. 2000 2000 0.2 1.
END
==

B.12.2 An Example of an Indicator_Variogram File

==
comments line start with this symbol
coordinate system can either be XYZ, XYW, or UVW
in this example, XYW is chosen; therefore, the areal correlation
ranges are in real-world coordinates and the vertical correlation
range is in normalized (0 to 1) coordinate
COORDINATE_SYSTEM XYW
not used
KRIGING_TYPE 0
maximum number of nearby data for kriging
MAX_CLOSE 16
0 = simple kriging; 1 = ordinary kriging
KRIGING_OPTION 0
3 cutoffs
NB_CUT_OFFS 3
first cutoff is 0.268 with cumulative probability of 25%
second cutoff is 0.300 with cumulative probability of 50%

third cutoff is 0.332 with cumulative probability of 75%

CUT_OFFS 0.268 0.25 0.300 0.50 0.332 0.75

search ellipsoids corresponding to each cutoff

generally, one should make the ranges of the search ellipsoids

larger than the correlation ranges

format: angle1 angle2 angle3 range1 range2 range3

SEARCH_ELLIPSOID 45. 0. 0. 5000. 2500. 0.2

SEARCH_ELLIPSOID   0. 0. 0. 5000. 5000. 0.2

SEARCH_ELLIPSOID 135. 0. 0. 5000. 2500. 0.2

covariance model for each cutoff

format COVARIANCE_MODEL sill number_of_nested_structures

model_type angle1 angle2 angle3 range1 range2 range3 contribution

model type can either be SPHERICAL EXPONENTIAL GAUSSIAN or POWER

in this example, there is only 1 nested structure, so only one model line

is needed

covariance for first cutoff

COVARIANCE_MODEL 1. 1

SPHERICAL 45. 0. 0. 3000. 500. 0.1 1.

covariance for second cutoff

COVARIANCE_MODEL 1. 1

SPHERICAL   0. 0. 0. 2500. 2500. 0.1 1.

covariance for third cutoff

COVARIANCE_MODEL 1. 1

SPHERICAL 135. 0. 0. 3000. 250. 0.1 1.

END

==

B.12.3 An Example of a Column_Average_Map File

This file is needed in Block-Kriging and Simulated Annealing. In this example, **an sgrid with dimension 138x33x100**. The first two lines must start with NX and NY which specify the number of cells in u and v direction, followed by NX*NY lines of data (one data per line) representing the column averages.

==

```
NX      138
NY      33
0.23
...
0.30
```
==

B.12.4 An Example of a Scattergram File
This file is needed in Cloud Transform (w/ P-Field). This file has no header lines, just a series of data lines in the format of
Independent_Variable Dependent_Variable
==

```
0.1     1.5
0.1     1.3
0.1     1.6
...
1.2     3.9
```
==

B.12.5 An Example of an External_Histogram File
This file is needed in Simulated Annealing and Continuous Histogram Correction. This file has no header lines, just a serious of data lines in the format of
Data_Value
GOCAD then constructs the Histogram using the data read in from the file.

==
```
1.0
0.9
...
1.2
```
==

B.12.6 An Example of a Facies_Map File
This file is needed in Fill From Facies Map. This file has a header line for the user's own identification

(not used in algorithm), followed by a serious of data lines in the format of

```
X_coord   Y_coord      Facies_value
```

```
=====================================================================
    1000000.0    23455066.0    2
    1230303.0    39393939.0    1

    ...

    3030303.0    10393030.0    3
=====================================================================
```

B.12.7 An Example of a Annealing_Schedule File

This file is needed in Simulated Annealing.

```
=====================================================================
# comments line start with this symbol
#but there is no real comments because if the variables
#are not obvious to the users, it would take too long to define them
# comments
INITIAL_TEMPERATURE        0.001
REDUCTION_FACTOR           0.1
# comments
MAX_PERTURB            1000.
MAX_PERTURB_PER_TEMP       10.
# comments
MAX_SUCC_PERTURB_PER_TEMP  3.
MIN_OBJECTIVE          0.001
# comments
REPORT_INTERVAL            0.2
STOPPING_NUMBER            5
=====================================================================
```

B.13 OTHER TEXT FILES

B.13.1 Color File

A color file contains color assignments for Well Marker and Well Zone.

Not all Well Markers or Zones in the GOCAD session that calls this file need to be listed in the file and not all listed Markers or Zones need to exist in the GOCAD session that calls this file. The color can be in RGB code, or an ASCII name that GOCAD recognizes. If a Marker is assigned a color that GOCAD does not recognize, such as *pacific sunset*, it will be displayed in black.

```
name_1 color_1

name_2 color_2

Top_of_Mesozoic green

Jurrasic_Interval 0.3 0.42 0.7

...
```

Where *name_n* is the name of a Marker or a Zone, and *color_n* is its assigned color either as RGB code, e.g. *0.3 0.42 0.7*, or an English name, such as *green*.

B.13.2 ASCII Curve File

GOCAD allows Well Curves to be added to an existing Well (during a GOCAD session) from an ASCII file. Such a file has 2 header lines followed by data lines:

```
Number_of_Fields

field_1_name field_2_name I_am_Z field_4_name...

data_1_1 data_2_1 data_3_1 data_4_1...

data_1_2 data_2_2 data_3_2 data_4_2...

...
```

Where *Number_of_Fields* specifies how many fields (columns) in each data line, and *field_name_n* is the Proeprty name of each field (column), which will be used to name the Well Curve created from that field (column) of data.

One of these fields (columns) must contain the Z value (depth) for that data line. In the GOCAD command **Add Curve From Ascii** dialog, the user will be asked to enter the Property (field) name that contains the Z values., e.g. *I_am_Z*. Z values can be in either measured depth or true depth, which is also specified by the user in the **Add Curve From Ascii** command dialog.

Appendix C: GLOSSARY

3D Camera
The main viewing window in GOCAD. Both parallel view (no distortion) and perspective view are available.

3D Grid
A generic term for GOCAD Grid Objects, Voxet and SGrid, a three-dimensional Object that is defined by an origin point and three axes, with regularly or irregularly spaced nodes.

3D Property
The same as a Vector Property. A Property that has three components (in the X, Y and Z directions) such as Velocity; as opposed to a Scalar Property, which has only one component, such as Amplitude. See also **Property** *(page C-12)*.

A?
A quick button that brings up (in the Attribute Manager) the Attributes of the Object you click on next.

Add...
A common command that adds a specific component to an exist Object.

Antialiasing
A process applied to a Camera or Object display to make things look smooth.

Apply
A execution button in many command Dialog window, which executes the command but keeps the Dialog window open.

Atom
The fundamental component in an Atomic Object (Surface, Curves, etc.). They are discrete points in space, defined by XYZs. Besides their own coordinate information, they can also carry extra information, such as porosity, temperature, etc., called **Property** *(page C-12)*.

Atomic...
An adjective, indicating that the subject is related to Objects made of Atoms.

Attribute
An item that describes a particular behavior of an Object, usually related to the physical appearance of the Object, e.g. color, size, etc.

Attribute Manager
The lower left portion of the GOCAD window, where you can edit the Attributes of the selected Object.

Autosetup
> A command that brings the Camera to the viewing angle that provides the best view (GOCAD-calculated) of the displayed Objects.

Axes
> Either refer to the axes of the coordinate system in the Camera, or the coordinated system of an SGrid or Voxet Object.

Azimuth
> The azimuth of a Surface or the orientation of the display of a well log.

Back Plane
> The rear view limit in a 3D Camera.

Backbone
> The Curve component of a ***GShape*** *(page C-7)* Object.

Beautify...
> A command that automatically modifies the specified component of an Object without the user's input.

Beauty Index
> A number that indicates the symmetry of the specified Component, i.e. Triangle.

Border
> A boundary or a portion or a boundary of a Surface or a Solid Object.

Border Stone
> An Atom that is at the end point of a Border.

Bounding Box
> A rectangular box that depicts the maximum and minimum XYZ extents of an Object. A term used in Model-building, meaning a closed box-shaped Surface. Also means the bounding walls of a Voxet or an SGrid. See also ***Cage*** *(page C-2)*.

Build
> A command that uses other Objects to create a new Object or to add components to an existing Object.

Built-in
> Something that came with GOCAD, usually means you can not delete it.

Cage
> The six edges of a Voxet or an SGrid; or the spatial extent of a group of Objects. See also ***Bounding Box*** *(page C-2)*.

Camera
> A viewing window.

Cancel
> An execution button in many Dialog window, which means "Close the Dialog window and do not execute the command".

CDF
> Cumulative Distribution Function, used a lot in GOCAD Geostatistical Applications.

Cell
> In an SGrid, the semi-cubical volume defined by 8 adjacent grid points. In a Voxet, the volume defined by expanding a Node until it touches its expanding neighboring Nodes.

Cell-centered
> An SGrid that carries Property values at cell centers. See **Corner-point** *(page C-4)*.

Clip
> Used as a noun to define the display range of a Property, as in High-clip, Low-clip; or as a command to delete all specified components except for the one you click on next, as in Clip Around Connected Triangle.

Closed
> Used as an adjective to describe that the following Object does not have an opening or free end, i.e. closed Curve.

Collapse...
> A command that delete the selected component and merge its remaining neighbors.

Color
> A generic Attribute item, describing the display color of a component of the selected Object.

Color Bar
> Exists in any Property Attribute Category; the color column that describes how a color relates to a Property value. It is detachable from the Attribute Manager.

...color file
> An ASCII file that contains color assignments for the specified component of the selected Object.

Color Scale
> Same as **Color Bar** *(page C-3)*.

Colormap Attributes
> A set of Attributes that control the color display of a Property; in Property Attribute Category.

Compute...
> A command, calculating the values of the specified Property.

Connectivity
> How Atoms in a specific Atomic Object or Nodes in a specific Grid Object are connected. It defines (describes) how the Atoms (Nodes) are related to each other in an Object and therefore how modifications (**DSI** *(page C-5)*) will propagate in that Object.

Constrain...
> A command that sets a certain type of Constraint.

Constraint
>   Specifications on the geometry (e.g. Control Points) or movement directions (e.g. On Straight Lines) of an Object. A Constraint has no effect until you actually modify the Object using **DSI** *(page C-5)* (Interpolation).

Control Points
>   Used in **Constraint** *(page C-4)*. Atoms that symbolize the desired geometry (location) of another Object.

Converter
>   A feature in the Object/Load command that converts data from other computer applications into GOCAD format. Currently, there are three most commonly used converters, old2new (C-Gocad to GOCAD++), uncompress (uncompress compressed files), and zmap2surf (ZMap ASCII file to GOCAD Surface).

Convex Hull
>   A termed used in a command to indicates that the created Object, viewing from the specified direction, will have no apparent concave portion.

Copy...
>   Duplicate the specified Object or Object components.

Corner-point
>   An SGrid that carries Property values at grid points. See **Cell-centered** *(page C-3)*.

Corner point
>   The 8 edge points of a Voxet or an SGrid; the 8 grid points of an SGrid cell.

Create...
>   Generating a new Object, or a new component for an existing Object.

CrossPlot (cross plot)
>   A Well Curve display that shows the main curve as displacement and the secondary curve in color. A GOCAD window that plots one Property against another.

CrossSection
>   A generic term, meaning a cross-section. A type of GOCAD Object, consists of sets (sections) of digitized Curves.

Cube
>   The entire volume of a Voxet Object. A display symbol for Atoms or Grid Nodes.

Curve
>   An Atomic Object, consists of Atoms connected in a one-dimensional way, a line. Its internal name is PLine; file suffix is ".pl". Often used to represent Fault cuts, Well data, and other linear geological features.

Curve resolution
>   The resolution of a Well Curve (log). 0 means no sampling, the highest resolution possible.

Cut...

A command that cuts the specified Object with other Objects, e.g. Cut By Surfaces.

Datapack

An obsolete Object Type from C-Gocad; a PointsSet that has one or more Properties.

Default values

Parameter or Attribute values that are provided by GOCAD. Usually overwrittenable by you.

Delayed Mode

Not a real toggle, it is the Mode when the Immediate Mode is toggled off. In this Mode, you need to press the Apply button to implement any changes that you make.

Delete...

Remove the specified Object or Object component. Gone forever.

Densify...

To increase the density of the specified component in an Object.

Detach

A toggle that allows you to create a duplicate of the specified Attribute item and display it any where you want on the screen.

Dialog

A pop-up window that allows you to specify parameter values needed to execute the selected command.

dir_shoot

The direction along which GOCAD searches for a specified Object. This is a very important component in many Constraint and Property commands. See the GOCAD User's Manual.

dir_Z

The selected command will be carried out along the specified Z-direction.

Discrete Property

A Property whose values are only defined at discrete points in space. Most GOCAD Objects carry discrete Properties at Atoms and Grid Nodes. **Model** *(page C-9)* Objects carry continuous Properties called **Model Property** *(page C-9)*.

Dolly

Moving the Camera closer to or away from the displayed Objects.

Draft Mode

How an Object (usually a Well or an SGrid) should be displayed while you are changing the view in the Camera using the mouse buttons.

DSI

The GOCAD interpolation engine, Discrete Smooth Interpolator. Invented and patterned by the founder of GOCAD, Professor Jean-Laurent Mallet.

Edit

Modifying a component of an Object.

Enable...
> Usually a toggle, meaning to apply the specified function.

Expand Macros
> A command in the History Menu, it records an executed command using the real name of an Object instead of the generic name "d". Relevant only if you selected the target Object was by clicking on it in the Camera.

Extremity
> The end Atom of an open Curve.

Fiber
> In a Voxet or an SGrid, the intersection between two Sections.

Fill...
> Usually related to Properties, meaning to copy Property values from another Object using the indicated method.

Filter...
> Eliminates some of the specified component from the selected Object. There is usually a pop-up Dialog for you to specify the criteria of eliminations.

Fit....
> To modify the specified component of an Object, using ***DSI*** *(page C-5)*.

Flat Shading
> GOCAD will not attempt to make Object look smooth, i.e. the Triangular facets on a Surface will be very visible.

From...
> A command that creates a new Object from the specified existing Object.

from_inside
> A toggle in some commands that involve ***dir_shoot*** *(page C-5)*. It specifies that the starting point of search is at the point of interest.

from_outside
> A toggle in some commands that involve ***dir_shoot*** *(page C-5)*. It specifies that the starting point of search is from outside of the layer towards the point of interest, which is inside of the layer.

Front Plane
> The front view-limit in a 3D Camera.

Fuzzy Control Points
> Same as ***Control Points*** *(page C-4)*.

Gaussian...
> The specified function or command uses the Gaussian (Normal) distribution function.

Get...
> A command that calculates the size of the specified component of the selected Object.

Glyph
> The display size of a component.

GObj
> Short for GOCAD Object, which includes all the Objects you can visualize in a Camera, plus some more.

GObj Style Manger
> The Manager window that allows you to manage all existing GOCAD Object Styles in your current GOCAd session.

Grid
> A Voxet or an SGrid; a Property in a Voxet or an SGrid; lines that connect Nodes in a Voxet or an SGrid; coordinate grid lines in a 2D MapView and other windows.

Gridded Surface
> A generic gridded surface is defined by an origin, delta-X, delta-Y and a series of Z values. A GOCAD gridded Surface is a triangulated Surface, but its triangulation and the distribution of its Atoms shows a regular gridded pattern.

Group
> A Group Object is a specific GOCAD Object Type, consists of multiple Objects that can be added to or removed from the Group. The term is also casually used to mean a set of Objects.

GShape
> A GOCAD Object Type, short for GeologicalShape. A GShape Object is an Atomic Object. It has a backbone, which is basically a Curve, and a series of Section along the Backbone to define the shape of the Object at those locations. It often used to model a meandering channel. Its internal name is GShapeBundle and its file suffix is ".gs".

GShapeBundle
> Internal name for GShape.

HeterogerneousGroup
> A Group Object that consists of Objects of different Object Types.

Hide
> To remove a currently displayed Object from the Camera.

History
> An ASCII file automatically generated by GOCAD when you start your GOCAD session, in which GOCAD records all successfully-executed commands; the Menu that contains commands allowing you to manipulate the history-recording.

Home
> A saved viewing angle; a command that brings the Camera to a saved viewing angle.

HomogeneousGroup
> A Group Object that consists of Objects of a single Object Type.

ILine
An individual piece of Curve in a Curve Object.

Immediate Mode
In the Attribute Manager and in the Object Display List, a toggle that specifies that any changes must be applied immediately.

Impact Point
The point on an Object where a search line (shoot line) intersects the Object.

Increment
A slider bar in the 3D Camera that allows you to specify the amount of change each time you press the Movement Direction button.

Index Number
Of a Voxet or SGrid Section is the number of Nodes between the Section and the Origin of the Grid Object; of a Rib in a GShape Section is internal ID of that Rib.

Info
The area at the bottom of the GOCAD window.

Init...
A preliminary step in modifying the geometry or a Property of an Object.

Interpolate...
An interpolation process utilizing ***DSI*** *(page C-5)*; or a simple linear interpolation.

Isovalue Shells
Usually means a Surface created from points (Atoms or Nodes) in an Object that has the same Property value.

Kill...
To delete the specified portion or component from the selected Object.

Ladder
A Well Curve display style, where the space between a Curve point and the Well Path is filled with a solid color.

Lathe
A Well Curve display style, where each Curve point is revolved around the Well Path to form a 3D display of the Curve.

Layer
A term used in a Model Object to describe a closed space bounded by members of the Model (Surfaces, usually).

Load
A command that allows you to bring information into GOCAD.

Locate
See XYZ?

Main Control Area
> The upper left portion of the GOCAD window, above the Attribute Manager and left of the Camera.

maximum
> The maximum value for the specified operation or display.

Menu
> A collection of commands that allows you to perform operations related to a specific Subject as indicated by the name of the Menu.

Menu Bar
> The area in a window where Menus are located. It is always at the top portion of a window.

Merge...
> To create a Object from other Objects of the same Type (e.g. Merge SGrids), or to "connect" isolated components in the same Object (e.g. Merge Parts).

Mesh
> The triangulation of a Surface.

Messages
> An area in the Main Control Area where GOCAd prints out error and warning messages.

minimum
> The minimum value for the specified operation or display.

Model
> A GOCAD Object. There are three types of Model Objects, 3dModel (Weiler Model), Voxet-Model and SGridModel. They share a common trade that they contain ***Layer*** *(page C-8)* information.
>
> Also a generic term meaning, model, any kind of model; Property Model, Structural Model, etc.

Model Property
> Not to be confused with Property Model, which is a Geostatistical model of a Property; A Model Property is continuous Property defined by mathematical equations, in a Model Object.

Move...
> A command that moves the specified component of an Object or even the entire Object. There are three types of Move: by specifying the direction and the amount of move (e.g. Move Node With Vector), by specifying the new location (e.g. Move Node To Point), or by clicking at the new location using the left mouse button (e.g. Move Node by Mouse).

Movie
> A feature in the Location Attribute Category of Voxet or SGrid that automatically loop through the Sections in that Grid Object.

Multi-Z-Value Surface
> A Surface, for a given X,Y location, it may have two different Z values, e.g. a closed Surface. Such a surface cannot be presented in a grid-form, but it can be represented by triangulation.

nbiter
> A common parameter in a interpolation Dialog window, short for "number of iterations".

nblevel
> A common parameter in a New Surface Dialog window, short for "number of levels of Triangles".

nbrings
> A common parameter in a triangle-related operation, short for "number of rings" (how many layers of triangles, starting from the one you click on, should be included in the operation).

New...
> A command that creates a new Object, a new Window...

No-data Value
> A value specific to a Property and an Object that signals to GOCAD that the specified Property does not have a known value at this Atom (Node).

Node Operation
> A sub-menu that contains a set of commands that allow you to modify the Atoms in an Atomic Object.

Normal
> The normal vector (normal to the Object) at the specified point.

North View (looking To The North VIew)
> A specific Camera viewing angle, in which you are looking into the Objects along the Y-axis of the Camera world, X-positive to the right and Z-positive to the top (unless you change the scaling of the Camera with a negative number).

nu, nv, nw
> The number of Nodes or cells along the U, the V, or the W axis of a Voxet or an SGrid.

Object
> An entity. There are many classes of Objects in GOCAD, the one you are likely to encounter is the GObj, GOCAD Object, which include such common Types as PointsSet, Curve, Surface, Solid, GShape, Well, SGrid, Voxet, Model, Cross-Section, RaySet and other less common ones.

Object Type
> See ***Object*** *(page C-10)*.

Object-Display List
> A list on the left side of a Camera displaying existing Objects, from which you can select Objects to be displayed in the Camera.

Operation Menu Bar

The wide and thin area on top of the 3D Camera. The first Menu is always the Operation Mode and the last Menu is always the Help Menu. In between, the Menus are updated depending which Operation Mode you are in. Commands here allow you to modify or create an Object, and perform geophysical, geological and geostatistical operation.

Optimize...

An automated modification process, applicable to the specified component of an Object. Some have a dialog window for you to specify optimization criteria, some use distance as the criterion (e.g. Optimize Shoot Directions).

origin

In a Voxet or an SGrid, the origin is the point whose internal (u,v,w) coordinates are (0,0,0).

Output Mode

Most GOCAD data files are in the ASCII format, but for Well and Grid Objects, you are allowed to select binary as the output file format because of the potentially huge file size.

Paint...

A command that copy a Property (or a Property-like component) from one Object onto another Object.

Parallel View

In this mode, parallel lines remain parallel. There is no distortion of lengths or areas. There is no sense of depth.

Panel

In a Camera, all the buttons and stuff to the left of the viewing area.

Ancestor Style

In the GOCAD Style Instance Tree (see the `GOCAD User's Manual`), every higher Style along the linear heritage of the current Style is an ancestor Style of the current Style.

Perspective View

In this mode, parallel lines are drawn to intersect at a distant point. There are distortions of lengths and areas. There is an illusion of depth. This is the default view when a 3D Camera is first created.

PLine

GOCAD internal name for Curve.

PLineFrame

A GOCAD Object, consists of a set of PLines (Curves) that intersect each other, usually the frame work of a Surface.

PointsSet

An Atomic Object, consists of discrete points that are connected in any way. Its internal name is VSet and it file suffix is ".vs". Often used to represent scattered Well data. Mostly imported as data points to create Surfaces.

Project
> The record of an entire GOCAD session.

Property
> Information that an Object carries, such as porosity, temperature, etc. There are two type of Properties, ***Discrete Property*** *(page C-5)* and ***Model Property*** *(page C-9)*.

Quick Buttons
> The buttons in a Camera panel that allows you quick access to commands or creating Objects.

Quit
> Exiting GOCAD.

Region
> Same as a Subset: a portion of an Object. You can create a Region using the subset editor, or the New Region command. A subset can scatter across different Parts of an Object. A closed space in a Model Object, bounded on all sides by Surfaces.

Remove...
> Usually it means permanently delete the specified components from the Object.

Rename....
> Give the specified component a new name.

Resolution
> Usually in the Attribute Manager, meaning how detailed you want the specified component to be drawn.

Rib
> The points in a GShape Section that defines the outline of the Section.

Save...
> Keep a record of the specified component; could be an external ASCII file (e.g. Save Object) or an internal file (e.g. Save View).

Script
> There are two types of Scripts in GOCAD, Command Script and Property Script. A command Script consists of a series of GOCAD commands that you can execute by selecting the Run Script command in the History Menu. A Property Script is a series of mathematical expressions that you can apply to a Discrete Property.

Search
> A generic term in GOCAD meaning finding a point on another Object. The direction of search is often called ***dir_shoot*** *(page C-5)*. The point found is often called ***Impact Point*** *(page C-8)*.

Section
> In a Voxet or an SGrid, it is layer of Nodes or cells. In a GShape, it is the shape of the Object at each Atom of its Backbone.

Segment
> Two connected Nodes in a Curve forms a Segment.

Set...

Assign a value or a role to the specified component or Object.

SGrid

A GOCAD 3D Grid Object, short for StratigraphicGrid. When you first create an SGrid in GOCAD it is a regular Grid Object, meaning the axis directions and the Node intervals are fixed. But sequential modification of the SGrid allow it to become completely limber: variable axis directions and variable Node intervals. Its internal name is also SGrid and its file suffix is ".sg". Mostly used to perform **Simulations** *(page C-13)*.

SGS

Sequential Gaussian Simulation.

shoot

To search for, see **Search** *(page C-12)*.

Show...

To display the specified component, in a Camera or in the Attribute Manager.

Simulations

Geostatistical realization methods available in GOCAD, found in the SGrid/Simulation and Voxet/Simulation Menus.

SIS

Sequential Indicator Simulation

Skin

The bounding surface (not a GOCAD Surface) of a Solid or a GShape, or even a Grid Object. It is also sometimes used to describe the creation of a Surface from two Curves.

Solid

An Atomic Object, consists of discreet points connected by tetrahedra. Its internal name is TSolid and its file suffix is ".so".

Split..

Usually used in **Mesh** *(page C-9)* modifications, meaning to divide a Triangle into 2 or four new ones.

Split Node

An SGrid Node that has been displaced (faulted) and are now concurrently existing at two different locations.

Stratigraphic Grid

See **SGrid** *(page C-13)*.

Style

The way an Object behaves. A Style is defined by numerous individual items called **Attribute** *(page C-1)*. Usually the user refer to the Style as the physical appearances of an Object, but internally it also includes all default values for any applicable operations that may be applied to the Object, and more.

Style Manager
> The Manager window where you can manage all existing Styles in your current GOCAd session.

Subset
> Same as a Region: a portion of an Object. You can create a subset using the subset editor, or the New Region command. A subset can scatter across different Parts of an Object.

Surface
> The most common Object in GOCAD. It is an Atomic Object, consisting of Atoms that are connected through triangulation. Its internal name is TSurf and its file suffix is ".ts".

Switch...
> A command that switches two adjacent components or directions.

Tetrahedron
> A three-sided pyramid. The basic unit of the mesh work in a solid, four connected Atoms.

Texture mapping
> Display a Property from a Voxet onto a Surface. It is a temporary display and the property dos not become part of the Surface.

Texture Extraction
> A type of **Texture mapping** *(page C-14)*. Property values along the location of a Surface is extracted from a Voxet and displayed on the Surface.

Texture Draping
> A type of **Texture mapping** *(page C-14)*. Property values from a Voxet Section is projected and displayed on a Surface.

Texture Projection
> A type of **Texture mapping** *(page C-14)*. Property values from a Voxet Section is projected and displayed on the Surface.

TFace
> An individual piece of surface in a Surface Object.

Transparency
> In the Attribute Manager, if you apply Transparency to an Object, it becomes transparent so you can see the Objects "behind" it.

Triangle
> The basic unit in the Mesh work of a Surface, three connected Atoms.

TSolid
> GOCAD's internal name for **Solid** *(page C-13)*.

TSurf
> GOCAD's internal name for **Surface** *(page C-14)*.

Tube

A component of a **GShape** *(page C-7)* Object; it is a set of lines that connects the correlated Rib along the Sections of the Object. It is more like the wire structure of a tube.

TVolume

An individual, disconnected "chunk" in a Solid Object.

Unset...

Usually, to remove an assigned role from an Atom.

Voxel

A term rarely used now a day, a cell in a Voxet Object.

Voxet

A GOCAD 3D regular Grid Object. It is defined by an origin, three axis vectors, and the number of Nodes along the axes. Often used to store seismic information.

VoxetModel

A Model Object created and related to a Voxet. It is automatically created when a Voxet is created but it is empty, until you define Regions (**Layer** *(page C-8)*) in it by cutting the Voxet with Surfaces (there are other ways to create Regions).

Well

A GOCAD Object. It is models what we call Wells in daily life. IA Well has a Well Path, showing the location of the Well; it can also have logs (called Well Curve in GOCAD).

Well Curve

Well log information. Part of a Well Object.

XYZ?

The query button, it prints out the XYZ and Property information of the point you click on.

Zoom

Changing the Camera Aperture to get an enlarged (not closer) image of the displayed Objects.

Appendix D: FROM C TO ++

This Appendix offers C-Gocad user a quick transition from GOCAD9.3 to GOCAD++. The layout of this Appendix follows the window layout of C-Gocad.

D.1TERMINAL WINDOW

D.1.1 File

New Project	*Close Project*
Open Project	*Main Control Area/File/Load Project*
Close Project and Save	*Main Control Area/File/Close Project*
Close Project w/o Saving	*Main Control Area/File/Close Project*
Save Project	*Main Control Area/File/Close Project*
Save Project As...	*Main Control Area/File/Save Project As...*
Run Script	*Main Control Area/History/Run Script*
Quit	*Main Control Area/File/Quit*

D.1.2 Object

Load	*Main Control Area/File/Load Objects...*
Save (ASCII)	*Main Control Area/File/Save Object*
Save As (ASCII)	*Main Control Area/File/SaveObject As...*
Attribute	*Attribute Manager*
	Select the Object, select the Attribute Category (usually quite obvious) to access the Attribute item then modify it.
Show/Hide	*3D Camera Object Display List*
	Lower left corner of the 3D Camera. To display a single Object, click the Object's name; to display multiple Object, press down the Ctrl-Key then click (toggle).
	Also 3D Camera Menu Area/Display Menu
Hide	*3D Camera Menu Area/Display Menu/Hide Object*
Move to	*NA*
Move with vector	*Operation Menu Bar/Edit/.../Move with Vector*
Rotate with mouse	*NA*
Rotate with vector	*Operation Menu Bar/Property/Apply Script on Property*
Scaling	*Operation Menu Bar/General Mode/Property/Apply Script on Property*
Dissociate	*N/A*
Info	*3D Camera/Quick Button Area/XYZ? button*
Measure	*Operation Menu Bar/Surface Mode/Property/Compute.... print out in the Info Area*
	Operation Menu Bar/Curve Mode/Length/
	3D Camera/Quick Button Area/<-? -> button
Name	*3D Camera/Quick Button Area/XYZ? button*
Copy	*Main Control Area/Edit/copy*

New Elementary	*?????*
New Group from Camera	*Main Control Area/Edit/New Group then Main Control Area/Edit/Add To Group*
New Group from mouse	*Main Control Area/Edit/New Group then Main Control Area/Edit/Add To Group*
Edit Group	*Main Control Area/Edit/UnGroup and Main Control Area/Edit/Add To Group*
Destroy	*Main Control Area/Edit/Delete*

D.1.3 Camera

New	*Camera #0 part of window.*
	Main Control Area/Windows Menu
Show/Hide Object	*3D Camera Object Display List*
	Lower left corner of the 3D Camera. To display a single Object, click the Object's name; to display multiple Object, press down the Ctrl-Key then click (toggle). *Also 3D Camera Menu Area/Display Menu, Main Control Area/Windows/ Show-Hide Objects...*
Hide Object	*3D Camera Menu Area/Display Menu/Hide Object*
Pilot On	*3D Camera/ Camera Movement buttons*
	Upper left portion of the 3D Camera.
Antialiasing	*Main Control Area/Windows/Camera Style Manager...*
	Select the Camera then press the View Attribute button; in the Attribute Manager, toggle on the *Antialiasing* toggle.
Autosetup	*3D Camera/Quick Button Area/Globe button*
	or 3D Camera Menu Area/Settings Menu/Autosetup
Vision Parallel	*3D Camera Menu Area/View Menu/Parallel*

Vision Perspective	*3D Camera Menu Area/View Menu/Perspective*
Scaling	*3D Camera Menu Area/Setting Menu/Scaling...*
Axis	*3D Camera Menu Area/Setting Menu/Show Axes (toggle)*
Select	*????????????*
Clear One	*In the particular Camera/Camera Menu Area/display/Hide All*
Clear All	*NA*
Set Background Color	*3D Camera Menu Area/Setting Menu/Background color...*
Switch On	*?????????????????????*
Switch Off	*?????????????????????*
Destroy	*Click on upper left-hand corner of the Camera frame and select Exit.*

D.1.4 Options

Modeler Menu On	*Automatically part of GOCAd window*
Customize Modeler Menu	*Operation Menu Bar/Mode Menu*
Application Menu On	*Automatically part of GOCAd window*
Customize Application Menu	*NA*

D.1.5 Help

About g0cad	*GOCAD logo in the GOCAD window*

Prompter On	*Info Area, automatically part of GOCAD window*
	Also in Operation Menu Bar/any Mode/Help/Toggle Context Usage
Help Path	*Look into the Operation Menu Bar/any Mode/Help Menu*
Atom	*??????????????????????*
Simplex	*??????????????????*

D.2MODELER MENU

Unless otherwise specified, every GOCAD++ command listed in this section is in the Operation Menu Bar area.

D.2.1 Surface

D.2.1.1 Surface New

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Surface Mode, New Menu.

From Points   *From Set of Points*

                                      ,From Points and Curve

From Curves	*From Closed Curve*
	From Points and Curve
	From Two Curves
	From Several Curves
	From Two Parallel Sets of Curves
From Skeleton	*From PLineFrame*
From Surfaces	*From Surfaces*
From Solid Skin	*From Volume (Solid) Border*
	From Volume (Solid) Borders
From model face	*?From Model Region Borders?*
From model region	*From Model Region Borders*
From GShape skin open	*From GShape*
From GShape skin closed	*From GShape*
From cage plane	*?From PointsSet Medium Plane?*
From isosurfaces	*From Isovalues*
From slope disk	*NA*
From Dynamic Surface one state	*???????????????????*
From Dynamic Surface current state	*???????????????????*
From Dynamic Surface all states	*???????????????????*
From Voxet	*Cube From Voxet Cage (in Built in Form sub-Menu)*

D.2.1.2 Surface Cut

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Surface Mode, Edit or Mesh Menu.

Surface by Surface	**Cut By Surfaces**
Surface by Surface with border on surface	**Cut By Surfaces,** with **build_constraint** on.
Surface and Surface	**Cut And Surfaces**
Surfaces	**Cut And Surfaces**

D.2.1.3 Surface Cut By Scissors

Unless otherwise specified, all GOCAD++ commands are in the Camera, Create Buttons Area.

open line	**Open-Line Create Button,** select **Scissors.**
closed line	**Closed-Line Create Button,** select **Scissors.**

D.2.1.4 Surface Border

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Surface Mode, Border Menu.

Beautify Border, all border	**Beautify Border**
Beautify Border, part of a border	**NA**
Smooth Border	**Smooth Border**
Move Border	**Move Border**
Add BorderStone	**Add Border Stone**
Remove BorderStone	**Remove Border Stone**

D.2.1.5 Surface Curvature

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Surface Mode, Property Menu.

compute	*Compute Curvature*
Remove	*Remove Property*

D.2.1.6 Surface Compute Thickness
No equivalent in GOCAD++.

D.2.1.7 Surface Texture With Voxet
In Attribute Manager, select the Surface, then change the Attribute Category to Texture.

D.2.1.8 Surface Normals
No equivalent in GOCAD++.

D.2.2 Curve

D.2.2.1 New
Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Curve Mode, New Menu.

From surface, all border	*From Surface Borders*
From surface, part of a border	*From Surface Border*
From Two Surfaces	*From Surface Intersections*
From Surface and Plane	*NA*
From Surfaces	Run *"From Surface Borders"* multiple times with the same Curve name*.*
From isocurves	*From Surface Contours*

From Points Convex Hull, given normal	*From Convex Hull and Normal*
From Points Convex Hull, computed plane	*From Convex Hull*
From curves	*From Curves*
From Points	Use any of the **Create Button**, in a Camera, select **Curve**

D.2.2.2 Segments

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Curve Mode, Edit Menu.

Filter Segments, by angle	*Filter Small Angle*
Filter Segments, by minimum size	*Filter Small Segments*
Densify Segments	*Densify*
Densify Segment	*Break Segment*
Extend Curve	*Extend Node*

D.2.3 Points

D.2.3.1 New

From Object	*New From Points*
From Objects	*New From Points*
From Points	*No one-step equivalent.*

D.2.3.2 Move, Add and Remove
GOCAD++ commands are scattered in different Operation Mode.

Move Points	**Surface Mode/Mesh Menu/Node Operation/Move Node...;**
	Curve Mode/Edit Menu/Node Operation/Move Node...,
	PointsSet Mode/Edit Menu/Move points...
Add Points	**Surface Mode/Mesh Menu/Triangle Operation/Break Triangle;**
	Curve Mode/Edit Menu/Break Segment.
Remove Points	**Surface Mode/Mesh Menu/Node Operation/Kill Node;**
	Curve Mode/Edit Menu/Node Operation/Kill Node,
	PointsSet Mode/Edit Menu/Delete points.
Get Location	**3D Camera/Quick Button Area/XYZ? button**
Get Distance	**3D Camera/Quick Button Area/<-?-> button**

D.2.4 DynamicSurface

New From Surface	**??????????????????**
Manager	**???????????????????????**

D.2.5 GShape

D.2.5.1 New
Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, GShape Mode, New Menu.

From Curve	**From Curve**

From Closed Curves (Sections)	**From Sections**

D.2.5.2 Edit

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, GShape Mode, Edit Menu.

Resize Section	**Section/Resize**
Rotate Section	**Section/Rotate**
Set Section, at node	**Section/Set...**
Set Section, at	**Sections/Set....**
Copy Section, at node	**Section/Copy...**
Copy Section every-where	**Sections/Copy Section**
Set Control Section Orthogonal	**Section/Set Orthogonal**
Set Channel Section	**Section/Set Channel**
Set Bounds	**?????????????????**
Inverse	**??????????????????**
Correl Rib	**Section/Correl Ribs....; Sections/Correl Ribs....**
Move Rib	**Rib/Move With Vector**
Add Rib	**Change Number of Ribs**

D.2.6 Solid

D.2.6.1 New

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Solid Mode, New Menu.

From Closed Surface	*From Closed Surface*
From Surfaces	*NA*
From Points	*From Points*
From Cage	*NA*

D.2.6.2 Edit

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Solid Mode, Edit Menu.

Densify	This is now done when the Solid is created.
Fit To	*NA*
Break By Surfaces	*NA*
Cut By Surfaces	*NA*

D.2.7 MESH

Unless otherwise specified, all GOCAD++ commands are in the Operation Menu Bar, Surface Mode, Mesh Menu.

Subset Editor	*General Mode/Region/New region; Other Global Operation sub-Menu/Subset Editor*
New Subset From Property Range	*General Mode/Region/New Region; then General Mode/Region/Property Range Region*

Beautify triangles	*Beautify sub-Menu/Beautify Triangles*
Switch Triangles, for beauty	*Beautify sub-Menu/Switch Triangles*
Switch Triangles, for roughness	*?????????????????*
Switch Triangles, for Gauss curvature	*NA*
Switch Triangles, for Mean curvature	*NA*
Remove Flat Triangles	*Beautify sub-Menu/Remove Flat Triangles*
Optimize, for measure	*Other Global Operations/Optimize mesh/Criteria: Area*
Optimize, for ControlPoints	*Other Global Operations/Optimize mesh/Criteria: Control points*
Optimize, for roughness	*Other Global Operations/Optimize mesh/Criteria: Roughness*
Split	*Split All*
Unsplit	*Unsplit*
Move Node To	*Node Operations/Move Node d To d*
Move Node to Point	*Node Operations/Move Node By Vector?????????*
Move Node to Impact	*Node Operations/Move Node To Point* (without specifying destination XYZ)*.*
Move Node to Location	*Node Operations/Move Node To Point*
Collapse Node	*Node Operations/Collapse Node*
Kill Node	*Node Operations/Kill Node*
Extend Node	*Node Operations/Extend Node; Curve Mode/Edit/Extend Node*
Bridge Nodes	*Node Operations/Bridge Node; Curve Mode/Edit/Bridge Nodes*
Switch Two Triangles	*Triangle Operations/Switch Two Triangles*

Kill connected simplexes	*Kill Connected Triangles; Curve Mode/Edit/Kill Connected Segments*
Clip connected simplexes	*Clip around Connected Triangles; Curve Mode/Edit/Clip Connected Segments*
Move connected simplexes	*NA*
Dissociate in connected simplexes	*Dissociate Vertices*
Collapse Simplex	*Triangle Operations sub-Menu/Collapse Triangle; Curve Mode/Edit/Collapse Segment; Curve Mode/Edit/Collapse Segment*
Kill Simplex	*Triangle Operations sub-Menu/Kill Triangle; Curve Mode/Edit/Kill Segment*
Extend Simplex	*Node Operations sub-Menu/Extend Node; Curve Mode/Edit/Extend Node*
Break Simplex	*Triangle Operations sub-Menu/Break Triangle; Curve Mode/Edit/Break Segment*

D.2.8 Constraints

D.2.8.1 Control Nodes

In GOCAD++, each Object Type (Operation Mode) has its own *Constraints Menu*. The GOCAD++ equivalent commands given here are scattered in each Operation Mode. The user should change the Operation Mode to the appropriate Object Type to locate the command. If it is not there in the *Constraints Menu*, then the specific command is not available for that Object Type. *For Surface-only commands, the entire path will be given.*

Set One	*Control Nodes/Set Control Node*
Set on part of surface border	*Control Nodes/Set Control Nodes on Border*
Set on all border	*Control Nodes/Set Control Nodes on All Borders*
Set on current subset	*NA*

Set on connected sim-plexes	***NA***
Set directional	***NA***
Unset One	***Control Nodes/Unset Control Node***
Unset on part of surface border	***Control Nodes/Unset Control Nodes on Border***
Unset on all border	***Control Nodes/Unset Control Nodes on All Borders***
Unset on current subset	***NA***
Unset on connected sim-plexes	***NA***
Unset directional	***NA***
optimize Control Nodes	***NA***

D.2.8.2 Control Points

Set Control Datapack	***Set Control Points***
Set Control Points	***Set Control Points***

D.2.8.3 Other Constraints

Set Border On Surface	***Surface Mode/Constraints Menu/Set Border on Surface***
Set Border On Line	***Surface Mode/Constraints Menu/Set Border on Straight Line***
Set Control Thickness	***Surface Mode/Constraints Menu/Other Constraints/Set Thickness Con-straint***
Set BorderStone on Bor-der	***Surface Mode/Constraints Menu/Other Constraints/Set Border Stone on Border***
Set Control Slope	***Surface Mode/Constraints Menu/Set Control Slope and Points???***

Set Parallel	*Surface Mode/Constraints Menu/Other Constraints/Set Thickness Constraint???*
Set Veclink	*Set Veclink*
Unset Constraint from Surface	*Delete Constraint*
Unset Constraint from Solid	*NA*
Activate Constraint, one	*Activate Constraint*
Activate Constraint, all	*NA*
Inhibit Constraint, one	*Unactivate (Deactivate) Constraint*
Inhibit Constraint, all	*NA*
Compute Error on Constraint	*Compute Error*

D.2.8.4 Shoot Direction

Set Control	*Local Editing Functions sub-Menu/Set Control Dir*
Unset Control	*Local Editing Functions sub-Menu/Unset Control Dir*
Set Impact	*Local Editing Functions sub-Menu/Set Shoot Dir With Impact Point*
Move Direction	*??????????*
Switch Direction	*???????????*
Reset Direction	*Initialize Shooting Direction; Local Editing Functions sub-Menu/Set Shoot Dir With Vector*
Smooth	*Smooth Shooting Direction*
Optimize	*Optimize Shooting Direction*

D.2.8.5 Misc.

Control Section, set	***GShape Mode/ Set Control Section***
Control Section, unset	***GShape Mode/ Unset Control Section***

D.2.9 Interpolation

In GOCAD++, each Object Type (Operation Mode) has its own Interpolation Menu. The GOCAD++ equivalent commands given here are scattered in each Operation Mode. The user should change the Operation Mode to the appropriate Object Type to locate the command. If it is not there, then the specific command is not available for that Object Type.

Fit Geometry	***Fit Geometry***
Smooth Geometry	***Fit Geometry*** with the ***smooth*** toggle on***.***
Fit Property	***Interpolate Property***
Fit Voxet	***Voxet Mode/Interpolation/Voxet Interpolate***
Smooth Voxet	***Voxet Mode/Interpolation/Voxet Interpolate*** with the ***smooth*** toggle on
	Voxet Mode/Interpolation/Voxet LS-Smooth
	Voxet Mode/Interpolation/Grid LS-Smooth
Init Voxet, classic	***Voxet Mode/Interpolation/Voxet Init***
Init Voxet, multigrid	***Voxet Mode/Interpolation/Grid Init***
Fit GShape Sections	***GShape Mode/Interpolation/Interpolate All Sections***
Set Number of Iterations	In GOCA++, this is done in each ***Interpolation*** command.
Set Geometry Moveability	***NA,*** except for ***Surface/Constraints/Other Constraints/Set On Straight Lines***

D.2.10 Properties

In GOCAD++, there is a Property Menu in every Operation Mode (Object Type). The GOCAD++ equiv-

alent commands given here are scattered in each Operation Mode. The user should change the Operation Mode to the appropriate Object Type to locate the command. If it is not there, then the specific command is not available for that Object Type.

In GOCAD++, Property Display is handled in the **Attribute Manager**. Select the Object, then select the **Property** Attribute Category, then select the Property you want to work on.

D.2.10.1 Managers...

In GOCAD++, there is a Property Menu in every Operation Mode (Object Type). The GOCAD++ equivalent commands given here are scattered in each Operation Mode. The user should change the Operation Mode to the appropriate Object Type to locate the command. If it is not there, then the specific command is not available for that Object Type.

Selection	This is done in each **Property** command dialog window.
Variable Add	**Property Menu/New Property**
Variable Select	In GOCAD++, Property Display is handled in the **Attribute Manager**. Select the Object, then select the **Property** Attribute Category, then select the Property you want to work on.
Variable Show Colormap	In GOCAD++, Property Display is handled in the **Attribute Manager**. Select the Object, then select the **Property** Attribute Category, then select the Property you want to work on.
Variable Show Histogram	**Property Menu/Statistics-Univariate**
Variable Remove	**Property Menu/Remove Property**
Variable Rename	**Property Menu/Rename Property**

D.2.10.2 Paint From Voxet
Surface Mode/Property Menu/Init from Volume Property Server

D.2.11 Isovalues

In GOCAD++, Surface contours are handled in the **Attribute Manager**. Select the Object, then select the **Contours** Attribute Category. To create a new set of contours:

1. ***Add a new Contour-Set name***
2. ***Select the Property you want to contour***
3. ***Specify the Contour Interval.....***

D.2.12 ColorMap

In GOCAD++, color bar is handled in the **Attribute Manager** and it is always associated with a particular Property. Select the Object, then select the **Property** Attribute Category.To edit the color bar of a Property:

1. ***Select the Property***
2. ***Select a built-in colormap***
3. ***Specify the Low Clip and High Clip Values***
4. ***Specify Transparency applications if you have any***
5. ***Specify No-Data display if you so desire***
6. ***Click the Detach button if you want to move a colorbar into the Camera***

D.2.13 Cage

NA

D.2.14 CrossSection

CrossSection Applications in GOCAD++ are not finalized yet. Currently, there are no C-Gocad equivalent commands. However, there are many GOCAD++ CrossSection applications that you can find in C-Gocad. Please see the `GOCAD User's Manual`. Currently, a Cross Section in GOCAD++ requires a Voxet to define the display volume.

D.2.15 Partitions

There is no exact equivalent of this Menu or its commands in GOCAD++. In GOCAD++, however, you can create Regions in Voxet and SGrid, and thus divide a bounded space into different portions. If you are interested in these options, see the *GOCAD User's Manual* or the *GOCAD C to ++*.

D.2.16 Model

There are three types of Models in GOCA++, Model3d (Wilier Model), VoxetModel and SGridModel, and PLineFrame. You may want to explore these options, see the *GOCAD User's Manual* or the *GOCAD C to ++*.

New	**Model Mode/New Menu/From Surfaces**
Locate Point	**NA**

D.2.17 Well

Well Objects in GOCAD++ are very different from wells in C-Gocad, because you can have Well Log information in GOCAD++ Wells. To learn about GOCAD++ Well Objects, see the *GOCAD User's Manual* or the *GOCAD C to ++*.

New Well From Curve	**NA**
Load Color Table	In GOCAD++, Well Display is handled in the **Attribute Manager**. Select the Well; select the **Graphic** Attribute Category, then modify individual display Attributes there.
Compute Marker	**Well Mode/Edit Menu/Compute Well Markers**
Get Info On Well	**3D Camera/Quick Button Area/XYZ? button**
Get Info on Marker	**NA**
Fit Surface To Well	**Surface Mode/Mesh Menu/Fit To Well Marker sub-Menu**

D.2.18 Skeleton

The **Skeleton Objects** become an internal Object Type in GOCAD++. The user no longer has access to Skeleton Objects.

D.2.19 Voxet

Unless otherwise specified, all GOCD++ commands listed in this section are found in the **Operation Menu Bar** in **Voxet Mode**.

D.2.19.1 New

Unless otherwise specified, all GOCD++ commands listed in this section are found in the **Operation Menu Bar**, **Voxet Mode**, **New Menu**.

Classic	**Voxet**
Diet	**Voxet From Points;** use the **Rectangle-Create button** in a Camera and select **Voxet.**
From Cage	**Voxet From Objects Cage**

D.2.19.2 Cut

Unless otherwise specified, all GOCD++ commands listed in this section are found in the **Operation Menu Bar**, **Voxet Mode**, **Edit Menu**. There are also many options in the **Region Menu** that allows you to create and define Regions in a Voxet. If you are interested in these options, see the `GOCAD User's Manual` or the `GOCAD C to ++`.

Cut By Surface	**Cut With Surfaces**
Cut By Cage Plane	**NA**
Cut By Cage	**NA**
Cut By Closed Scissors	use the **Open-Line Create Button** in a Camera and select **Scissors.**
Cut By Open Scissors	use the **Closed-Line Create Button** in a Camera and select **Scissors.**
Reset Connectivity	**NA,** but if you use **Regions** options, you will not need this command.

D.2.19.3 Paint

With Solid	**NA**
With Surface	**Edit Menu/Paint With Surface Property**
With Datapack	**Edit Menu/Paint With pointsSet Property**
With GShape	**Edit Menu/Paint With GShape Property**
From Point	**??????????????????????**

D.2.19.4 Region

The **Region Menu** in GOCAD++ is very different from the C-Gocad Region Menu. You may want to check out the GOCAD++ **Region Menu** directly.

Region Editor	**NA**
From Point	**NA**
Reset Full	No one-step equivalent. Use **General Mode/Region Menu/Empty Region** then **General Mode/Region Menu/Complement Region**
Reset Empty	**General Mode/Region Menu/Empty Region**
Get Volume	**Voxet Mode/Region Menu/Get Volume**
Get Volume From Point	**????????????????**
Get Volume From Curve	**????????????????**

D.2.19.5 Display Manager

In GOCAD++, Voxet Display is handled in the **Attribute Manager**. Select the Voxet, then select the desired Attribute Category, then modify the display Attributes as needed.

D.2.19.6 Region Editor
NA

D.2.19.7 Region Movie

NA

D.2.20 VVoxet

NA

D.3APPLICATION MENU

D.3.1 Simulation

In both *Voxet Mode* and *SGrid Mode* in GOCAD++, there is a *Simulation Menu* with very sophisticate Simulation options. You should investigate those options directly instead of trying to translate C-Gocad *Simulation* concept into GOCAD++, because they are so different.

D.3.2 Rays

In GOCAD++, there is a *Geophysic Mode* with very sophisticate Velocity and Ray Tracing options. You should investigate those options directly instead of trying to translate C-Gocad *Rays* concept into GOCAD++, because they are so different.

GOCAD C to ++