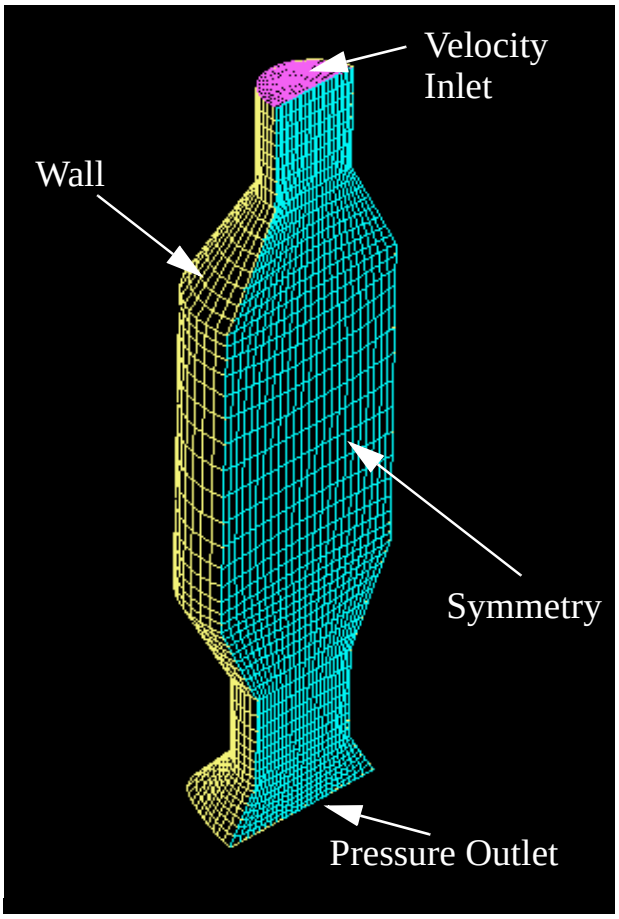
Tutorial 7: Property Setter

The **GridPro Property Setter** allows the user to set volume or boundary condition labels to be exported into a commercial CFD package or into an in-house developed code.

What You Will Create

You will learn how to set grid properties on surfaces and volumes.



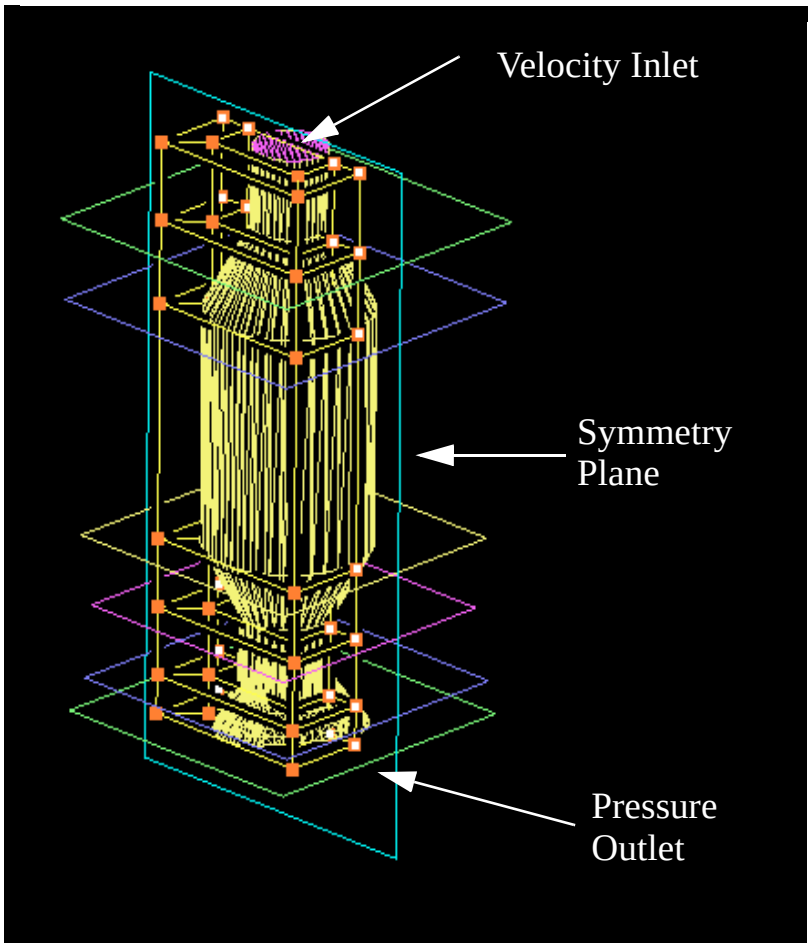
What You Will Learn

- How to use surfaces in the topology builder to set properties automatically
- How to check properties
- How to set property labels manually
- How to set property labels for a user defined code

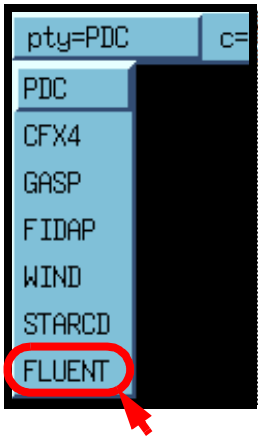
Part I: Setting Boundary and Volume Labels Automatically

Step 1 Automatic Boundary Labeling

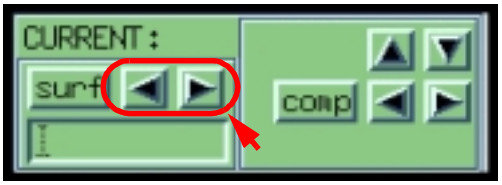
GridPro allows the user the convenience of labeling the boundaries in the Topology panel before the grid is generated. The majority of users will want to access this technique, however, it is limited to uniform properties that exist only on the boundary surfaces. Let's work on a case where we will apply a symmetric, velocity inlet and pressure outlet boundary condition on a cylindrical model. Start **GridPro** by double clicking on the **Tutorial_7.fra** file or enter **az** at the prompt and load the file from the **topo** sub-menu using **TIL: read**. We want to label boundary surfaces according to the picture below.



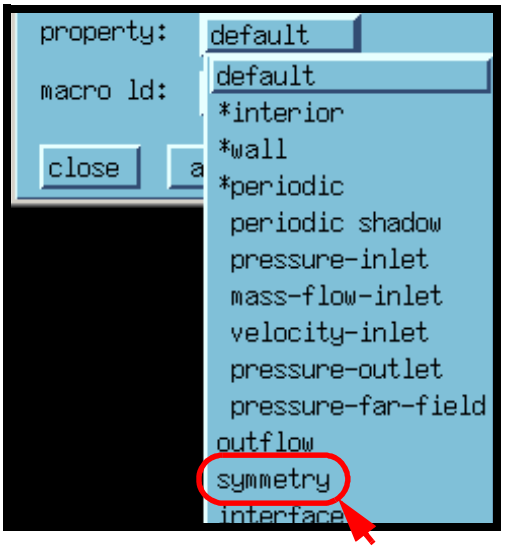
By default all of the surfaces are labeled as wall and will remain as walls unless they are changed by the user. To automatically set the property labels for this grid, choose the code you are using on the **pty=PDC** sub-menu at the top. For this case, let's use **Fluent** as our CFD code.



Now, set symmetry plane as the current surface by scrolling through the **CURRENT** sub-command panel. Make sure it is highlighted in light blue.



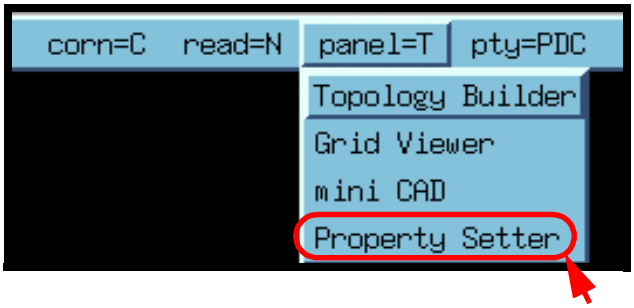
Go to the **surf** sub-menu at the top and choose **reload:current**. To set the property as symmetry, pull down the property sub-menu and choose symmetry and then click **apply**.



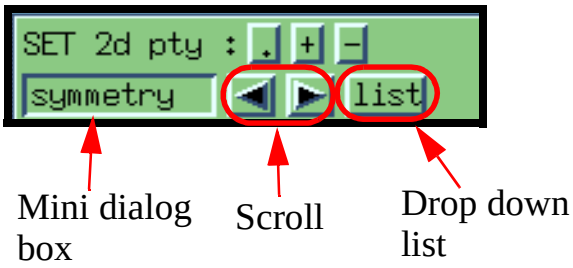
The same method can be used to apply the **velocity-inlet** and **pressure-outlet** boundary properties.

Step 2 Checking Properties

The property labels can always be checked and modified in the property setter. After running the grid load it into **GridPro** and switch the panel to the **Property Setter**.



Go to the **Set 2D Property** sub-command panel at the bottom and scroll throughout the list. Each property will appear in the mini-dialog box and each corresponding property label will be highlighted in light blue. You can also quickly choose the property in the drop down list.

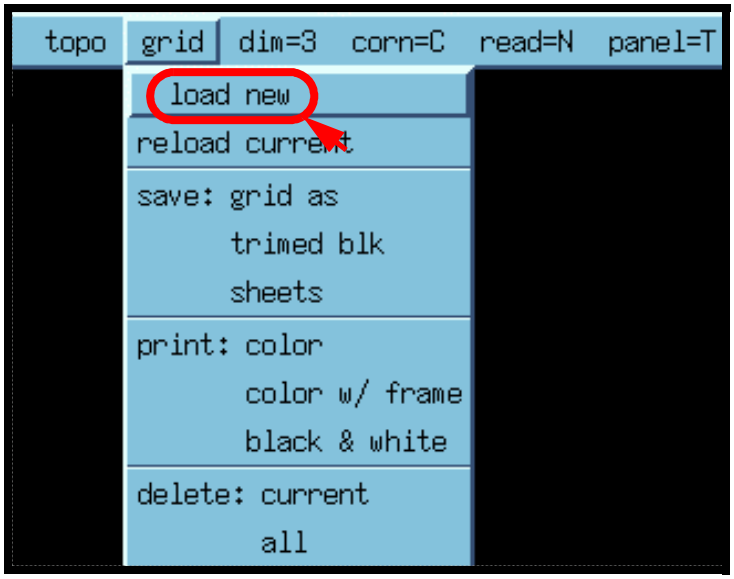


Part II: Setting Boundary and Volume Labels Manually

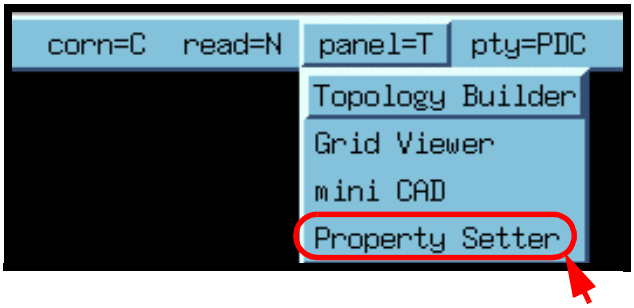
GridPro allows you to manually set the boundary and volume property labels within the **Property Setter** panel. In this case, let's set the boundary conditions that we set in Part I manually and add solid volume label in the center of the tube.

Step 1 Starting the Property Setter

Start **GridPro** by double clicking on the **Tutorial_7.fra** file or enter **az** at the prompt and load the file from the **topo** sub-menu using **TIL: read**. Load in the grid by using the grid sub-menu at the top.



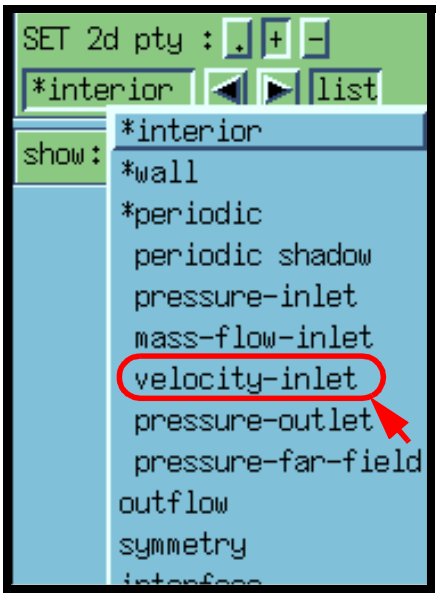
The grid is saved in the **blk.tmp** file. Go to the Property Setter by switching to **Panel=P** in the sub-menu at the top.



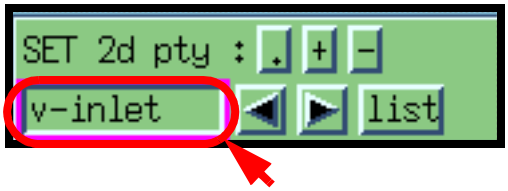
Choose a CFD code located in the **pty=PDC** sub-menu, in this case we will choose **Fluent**.

Step 2 Setting the Velocity Inlet Boundary Label

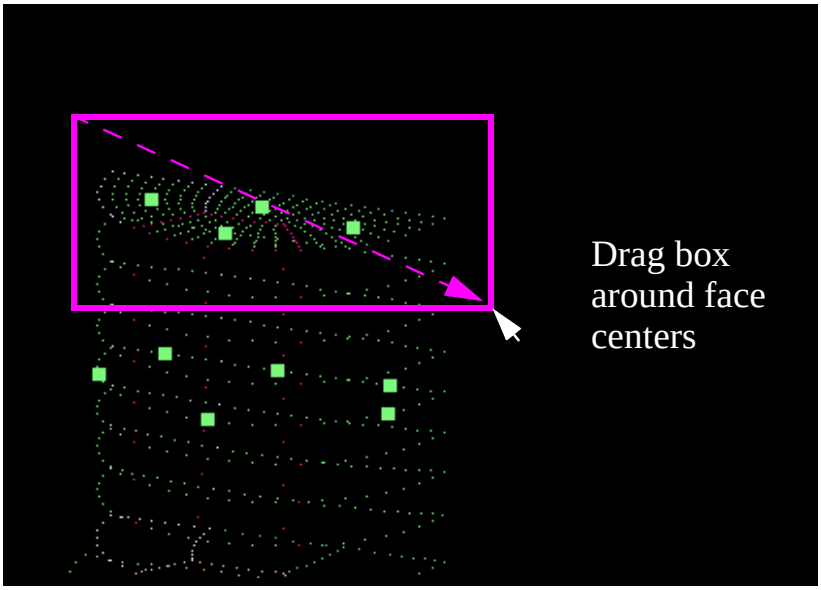
We want to set the velocity inlet boundary label at the top of the cylindrical model. First, go to the **Set 2d Property** sub-command panel and choose the **velocity-inlet** from the drop down list.

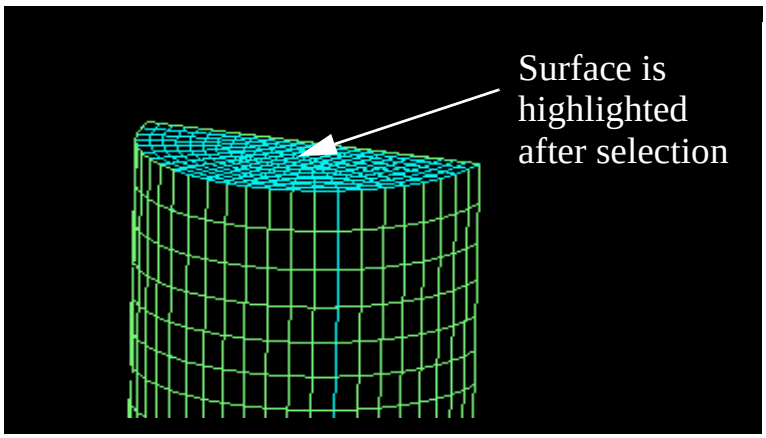


The button on the mini-dialog box can be toggled on to view the surface after it is assigned a property, or it can be turned off. Make sure that the button is pressed on so that the surfaces can be viewed.



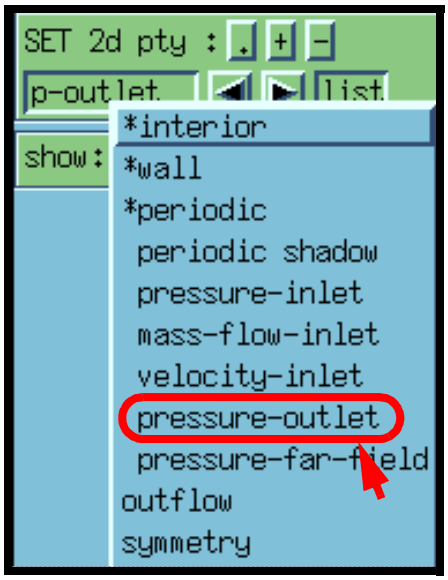
Zoom-in on the top of the model and hit the + button and drag a purple box around the highlighted face centers. The face will be automatically assigned as a velocity inlet property and will be highlighted in light blue.



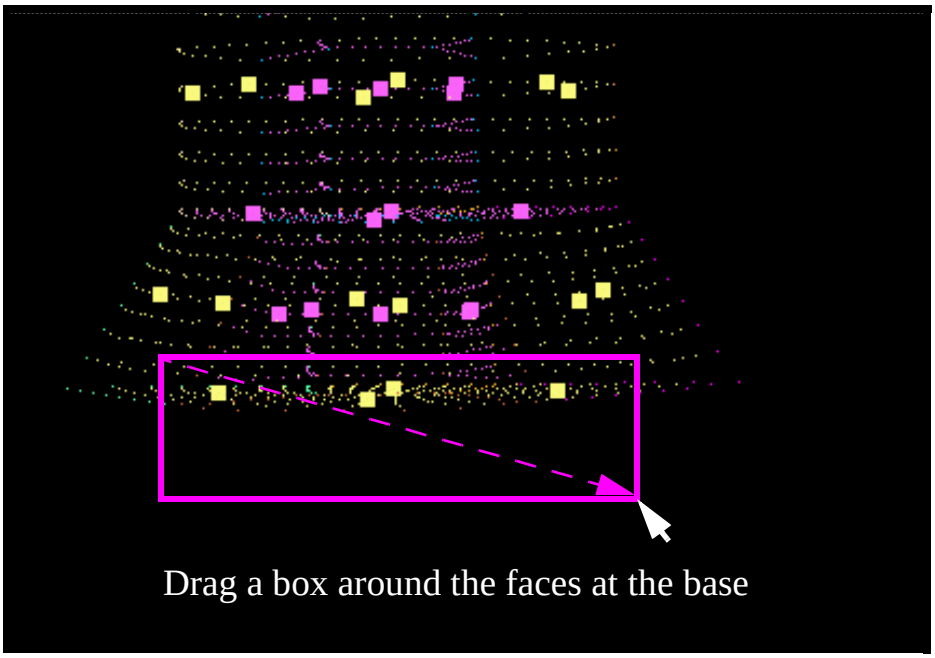


Step 3 Setting the Pressure Outlet Boundary Label

Assign the pressure outlet on the base of the model using the same method as in **Step 2**. Choose the proper label from the drop down list:

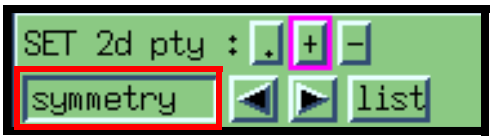


Right click the mouse button and drag a purple box around the faces at the base.

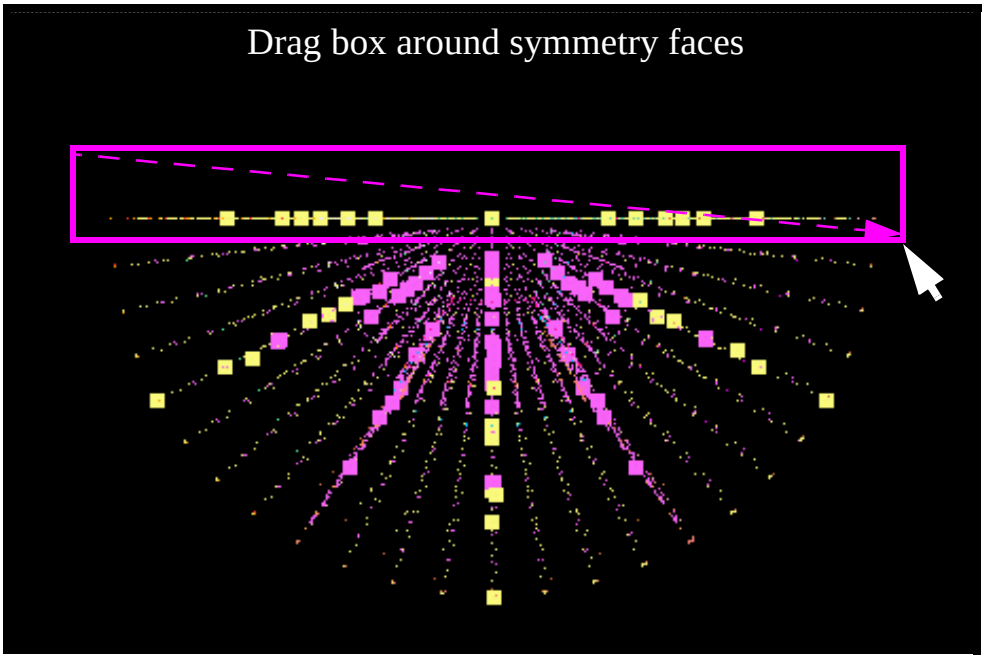


Step 4 Set the Symmetry Boundary Labels

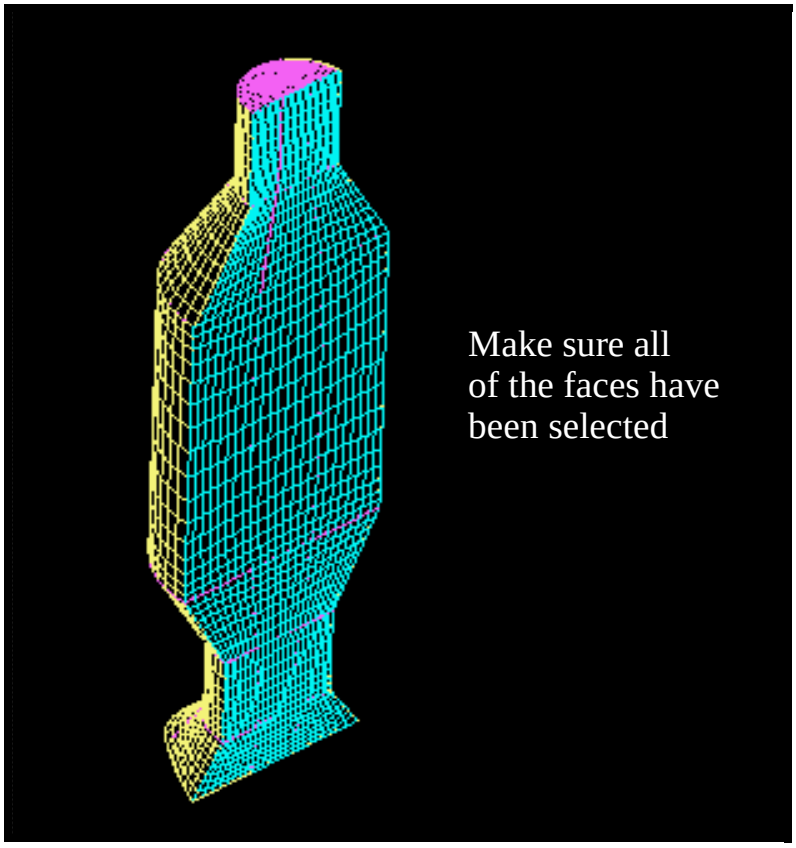
Make sure symmetry has been chosen in the drop down list and is displayed in the mini dialog box.



Snap the grid in a top view and drag a box around the faces.

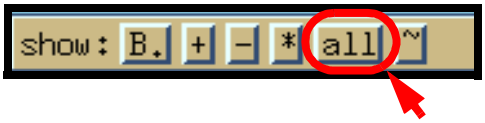


Rotate the view to make sure that all of the faces have been selected. If some faces have been selected that are not part of the symmetry plane, you can remove them by using the  and dragging purple box around the face centers you would like to remove.

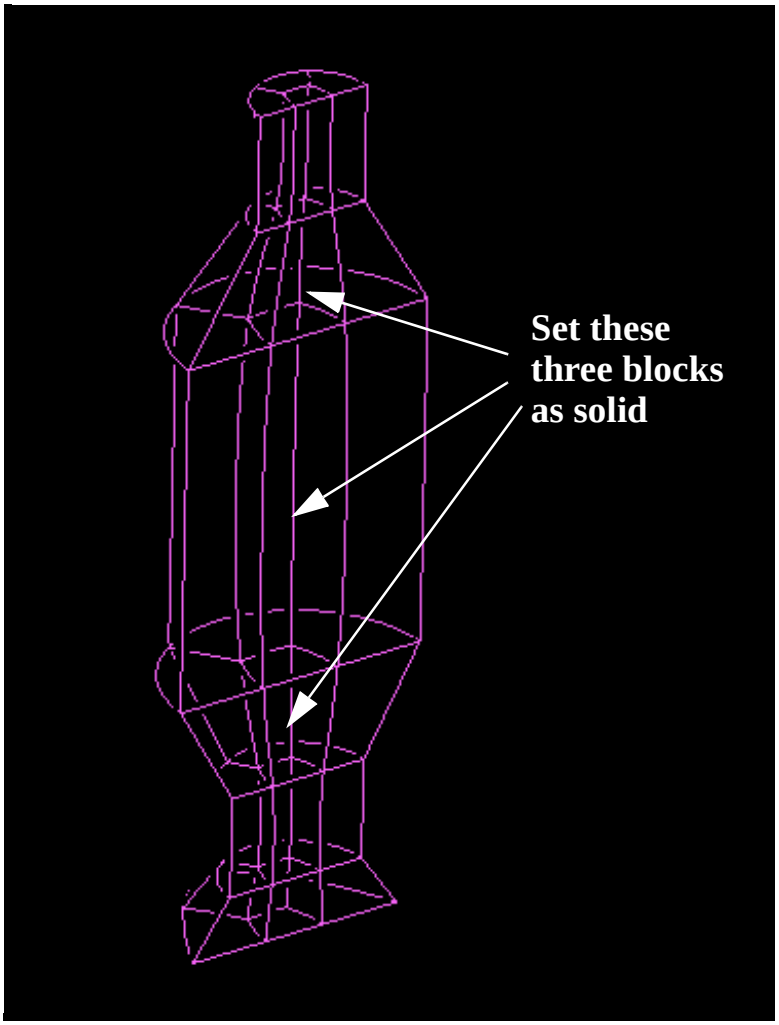


Step 5 Set Volume Boundary Properties

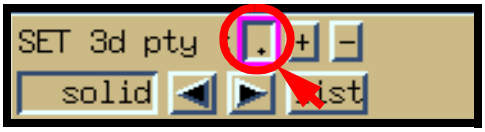
The volume boundary properties are set the same way as boundary properties. Make sure that the blocks are showing by turning off the display for all of the sheets. Go to the **show** menu in the **2D Property Setter** and click on the  button and drag a purple box around all of the faces. To turn on the blocks, go to the show menu in the **3D Property Setter** and click on the **all** button to display all of the blocks.

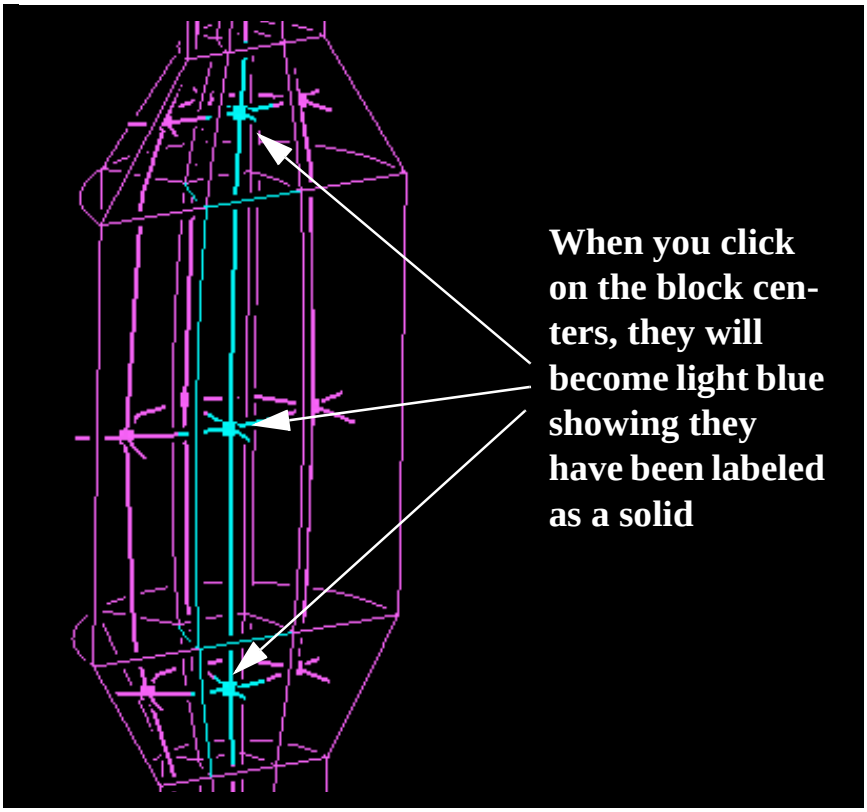


Let's set the inner core of the model as a solid, and the remaining blocks will retain their default setting as fluid.

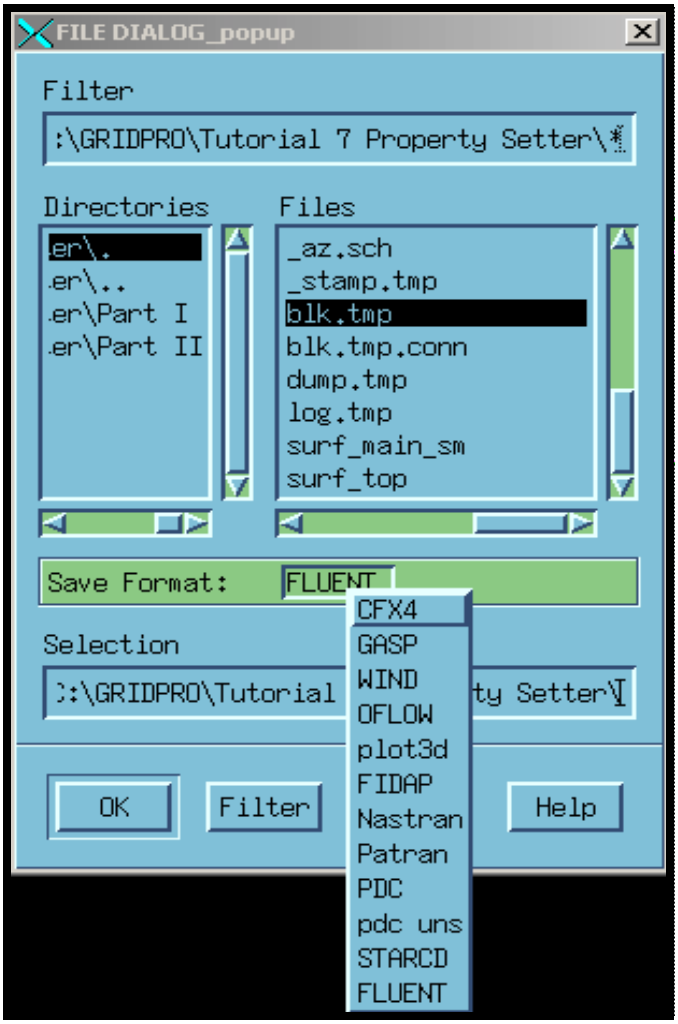


Make sure that the property setting is for **Fluent** and that the **solid** property is setting has been chosen from the drop down list. Click on the toggle button and left click on the corresponding block centers and they will be assigned as a solid. The  buttons can also be used to add and delete blocks from a chosen property.





Now that the faces and blocks have been properly labeled save the file under the chosen format by choosing **grid: save as** in the **grid** sub-menu.

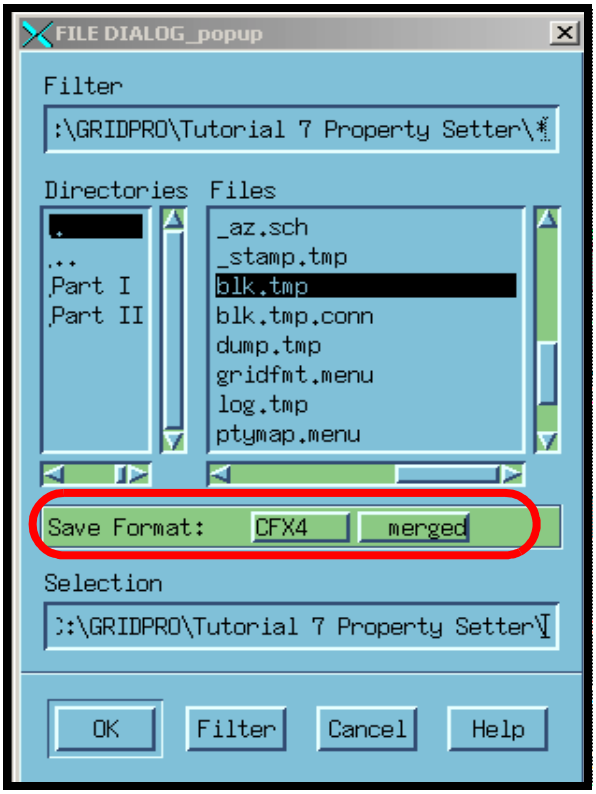

FILE DIALOG_popup
Filter
:\GRIDPRO\Tutorial 7 Property Setter*
Directories
Files
er\.
er\..
er\Part I
er\Part II
_az.sch
_stamp.tmp
blk.tmp
blk.tmp.conn
dump.tmp
log.tmp
surf_main_sm
surf_top
Save Format:
FLUENT
CFX4
GASP
WIND
OFLOW
plot3d
FIDAP
Nastran
Patran
PDC
pdc uns
STARCD
FLUENT
Selection
:\GRIDPRO\Tutorial
ty Setter\
OK
Filter
Help

Part III: Setting Properties for a User Defined Code

The property setter is most useful for independent codes that are being developed by the user. **GridPro** implements functions for outputting a grid in any given solver format as a plug-in script and executable. For example, the user can add to the **GridPro Graphics Manager** the capability of outputting the grid in a given solver format without the requirement of knowing, changing or recompiling the **GridPro** source code.

Step 1 Set the Solver Format

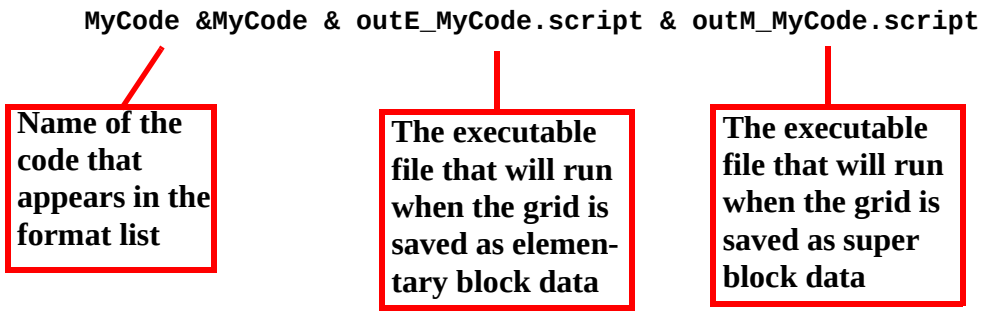
After the grid is created and you have set the properties, you must save the grid in the format of your choice. An example of the menu is shown below.



In this example, we will save the grid in CFX4 format. Modifying the **gridfmt.menu** will allow you to put the name of your solver in the **Save Format** menu list. The **gridfmt.menu** file is located in the **GridPro/ az_mngr** directory. A copy of the current file is listed below:

```
#----  FILE: gridfmt.menu ----
#menu lbl & head & elem b script to run & super b script to run
CFX4   &CFX4test & outE_cfx4.script  & outM_cfx4.script
GASP   &GASP   & outE_gasp.script  & outM_gasp.script
plot3d &plot3d & outE_p3d.script   & outM_p3d.script
FIDAP  &FIDAP  & outU_fidap.script & unused
Nastran&Nastran& outU_nast.script  & unused
Patran &Patran & outU_patran.script& unused
Fluent &Fluent & unused            & unused
pdc    &pdc    & outE_pdc.script   & outM_pdc.script
pdc uns&pdc uns& outU_pdc.script   & unused
#----  END OF FILE ----
```

Each line corresponds to a code, and let's say that your code's name is MyCode, it should be written in this format:



The `&MyCode` will not be used at this time. The name of your solver should not be longer than 10 characters. Elementary block data is saved as unmodified blocks whereas the superblock data is saved when the elemen-tary blocks have been merged into larger blocks (see next tutorial for more information about superblocks). If your code cannot handle superblock data then put `unused` in place of `outM_MyCode.script`. If you are running GridPro on a PC you must replace *.script* with *.bat.*

When a grid is saved, it will always be saved in GridPro block format in a *.pty* file then the following system call is executed:

```
system ('outE_MyCode.script grid_file_name output_file_name')
```

Where the parameters are defined as:

Table 1:

outE_MyCode.script	The name of the specific file to run
grid_file_name	The file name of the current grid (i.e. blk.tmp)
output_file_name	The output file name typed in by the user when saving the grid

Step 2 Writing the Output Script

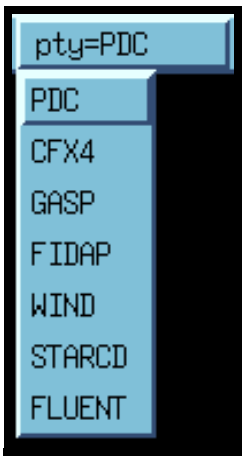
As mentioned in **Step 1**, when a grid is saved the **GridPro** graphics manager makes a system call to execute the `outE_MyCode.script` (or the `outM_MyCode.script`). It is the users responsibility to write the script, however, an existing script can be copied, edited and saved in the `GridPro/az_mngr/` directory. The script should read in the elementary block grid, the `.conn` and `.pty` file, and the property mapping `ptymap.MyCode` file (which will be discussed later) and output the grid in MyCode format. A user can also use GridPro utilities such as trf, mrgb (merge block) or chfmt (change format) in the script to accomplish some of the tasks involved. In general, the user needs to write one C or FORTRAN program to finish the final conversion. Typically, the program will read GridPro files, i.e. `blk.grd`, `blk.grd.conn`, and optionally a `blk.grid.pty` file for elementary blocks or `blk.grd.conn_n` for super blocks. An example script for **GASP** is listed below.

```
#!/bin/csh -fe
#---- FILE: outM_gasp.script ----
mrgb $1 -s 1 -gasp ──► Merge block and export in GASP format
mv $1.tmp.inp $2.inp ──► Rename the output file to a user name
chfmt $1.tmp -f p3d ──► Change the grid file to Plot 3D format
mv $1.tmp.tmp $2.p3d ──► Rename the ouput file to user name
\rm $1.tmp ──► Clean up temporary files
exit
#---- END OF FILE ----
```

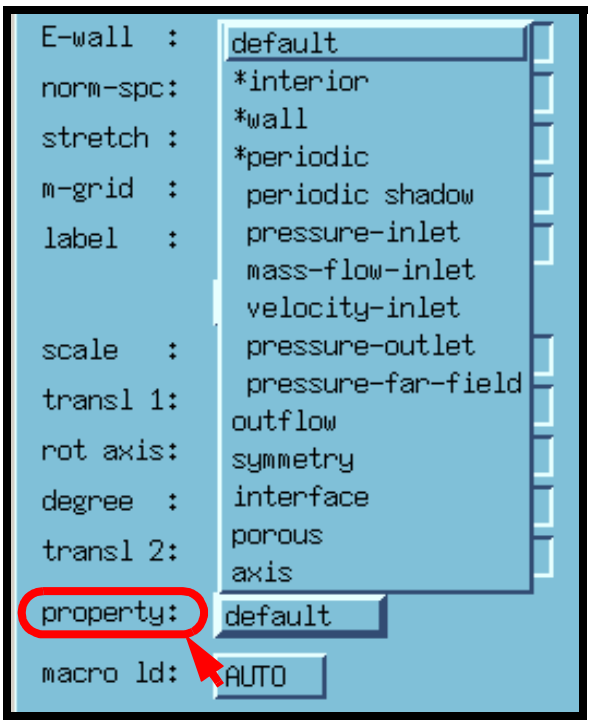
Step 3 Adding Your Solver Name into the Property Sub-Menu

The property mappings map the internally used property identifications (in number format) to solver specific internal names and appear in the **pty=SOLVER** sub-menu at the top and in the **property** settings pull down menu located in the **set surface parameters-popup** dialog box.

Property sub-menu



Set Surface Parameter Popup Menu



Each solver has its own property mapping file which is also a part of the corresponding solver entry in the `ptymap.menu` file in the `GridPro/az_mngr/` directory. Each code occupies one line in the file and each line contains three items separated by the "&" symbol. The current `ptymap.menu` file is listed below.

```
PDC    &pty=PDC    & ptymap.default
CFX4   &pty=CFX4   & ptymap.cfx4
GASP   &pty=GASP   & ptymap.gasp
FIDAP  &pty=FIDAP  & ptymap.fidap
WIND   &pty=WIND   & ../contrib/wind/ptymap.wind
STARCD &pty=STAR   & ptymap.starcd
FLUENT &pty=FLUENT & ptymap.fluent
```

The format for your solver entry is:

```
MyCode &pty=MyCode & ptymap.MyCode
```

The first item is your solver name appearing in the menu item list appears as an entry in the **pty=SOLVER** and in the **set surface parameters-popup** dialog box. The second item is the name label appearing on the two menu buttons once the property is selected from the menu item list. The first two items should not be longer than 10 characters. The third item is a file containing the property mappings for the given solver which will be discussed in the next step.

Step 4 Writing the ptymap.MyCode file

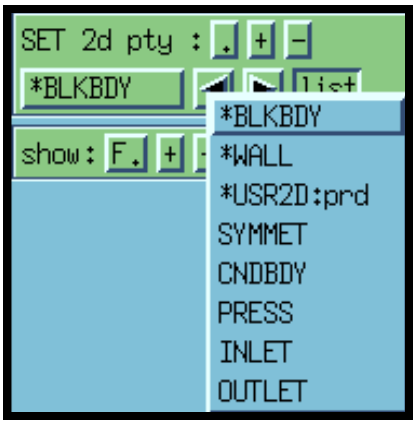
It is the user's responsibly to create the property mapping file to be saved in a file named `ptymap.MyCode`. The internal property names will be what the user sees when using the **GridPro Graphics Manager**. These names or labels will be assigned to the grids in the user defined format. Note that the GridPro grid is solver independent in that the properties are represented as numbers. It is the `ptymap.MyCode` file that must make the connection from the numbers to the internal names that are specific to the solver. To create a `ptymap.MyCode` file you can either copy and edit the `ptymap.template` or the `ptymap.code_name` of another solver.

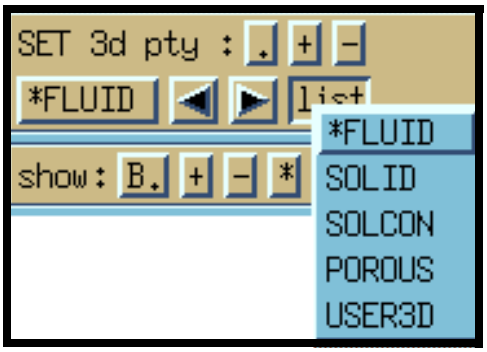
Let's look at the CFX4 file as an example. The file is listed below.

```
#---- FILE: ptymap.cfx4 ---
10     # 2d-ptys.
#pdc-id& pdc-name & mapped_name &label #comments
1      & INTERBLK & BLKBDY   &*BLKBDY    #grid generic basic
2      & PERIODIC & USER2D   &*USR2D:prd#grid generic basic
3      & BOUNDARY & WALL     &*WALL      #grid generic basic
4      & SYMMETRY & SYMMET
5      & user5    & unused
6      & user6    & CNDBDY
7      & user7    & PRESS
8      & user8    & INLET
9      & user9    & OUTLET
10     & user10
#---------------
7      # 3d-ptys.
#pdc-id& pdc-name &mapped-name  &label #comments
1      & BULK     &default &*FLUID  #grid generic
2      & user2    &SOLID   &
3      & user3    &SOLCON  &
4      & user4    &POROUS  &
5      & user5    &USER3D  &
6      & user6
7      & user7
#---- END OF FILE ----
```

Labels that appear on the buttons in the pull-down list. If you want them to be the same as the mapped name you can leave this blank

This code generates a pull-down list in the 2D and 3D property setter:





As you can see, there are two sections for the mapping - one for the surfaces (2D maps) and the other for the volumes (3D maps). Each line

defines a map for one property. Both sections have the same syntax. The first line defines the number of properties listed and is followed by the property mapping list.

Rules for Mapping

There are four items for each mapping line, they are:

1) pdc-id: Must be greater than or equal to one and must be less than or equal to the maximum number on the list. It does not have to be in any particular order but it must be unique

2) pdc-name: These are the generic property names which can be renamed. However, **2D Boundary Labels Id's** = 1 .. 3, and for the **3D Volume id** = 1 have a special role when the properties of the surfaces, block faces and block volumes are initialized. These following rules are applied during property initialization:

Rules Assigning Properties to Surfaces in the Topology Builder

If a surface is not assigned to a property it has the DEFAULT value of 0 and will automatically be reassigned according to the type of surface.

2-Sided Surface: pty-id=1, pdc-name=INTERBLK
1-Sided Surface: pty-id=2, pdc-name=BOUNDARY
Periodic Surface: pty-id=3, pdc-name=PERIODIC

Rules for Block Faces

a) If the block face is on the surface, it is reassigned to the property of the surface.
b) If the face is attached to two blocks (one on each side of the face) the face is reassigned to pty-id=1, pdc-name=INTERBLK
c) For anything else, the face is reassigned to the pty-id=2, pdc-name=BOUNDARY

Rules for Overlapping Properties

If two property idees are mapped to the same mapped-name, the faces with the larger idees are reassigned to the lower property idee.

3) mapped-name: These are solver specific names. Routines for a particular solver may use the GridPro names. A blank mapped_name means it is unused. Due to the property initialization rules, the mapped names for **2D**

Property (pty-id=1..3) should have the same corresponding meaning as the GridPro names.

4) label: Should be less than or equal to 10 characters and can have spaces (the labels are what is shown on the buttons)

5) &: A blank item should be '& &' instead of '&&'

For a predefined GridPro mapping file a user can:
1. Change the order of the lines
2. Change the labels
3. Can change the pdc-name for id >= 4 (2d cases) and id >= 2 (3d cases)
4. Needs to understand the initialization rules

For a user defined property mapping file, a user needs to map the INTER-BLK correctly since the merge block will merge all of the faces on the block throughout the domain.