Chapter 6. File Types

During a TGrid session you may need to read and write several kinds of files. You can read mesh, journal, Scheme, and domain files and write mesh, journal, transcript, and domain files. TGrid also allows you to save hardcopy files, save panel layouts, and write the current Scheme images. These files and features are described in the following sections.

6.1 Mesh Files

A mesh file includes a list of the node coordinates, boundary faces, and (sometimes) interior faces and cells. The boundary faces are grouped into zones with specified boundary condition identifiers. In addition to reading and writing TGrid mesh files, you can also import mesh files from various CAD packages and write mesh files for FLUENT. For information about the format of the CAD package files, see Appendix A: Importing Boundary and Volume Meshes, and for details on the mesh file format for TGrid and FLUENT, see Appendix B: Mesh File Format.

If the mesh information is contained in two or more separate files generated by one of these CAD packages, you can read them one by one using the Append File(s) check button in the Select File panel. You can also read them together and have TGrid assemble the complete mesh for you.

By default, TGrid saves the mesh files with the suffix `.msh`. You need not type the suffix while saving the mesh file, it will be added automatically. For example, if you enter the file name `gumby`, TGrid will write to the file `gumby.msh`.

When TGrid reads a mesh file, it first searches for a file with the exact name you typed. If a file with that name is not found, it will search for a file with `.msh` appended to the name.

6.1.1 Reading Boundary Mesh Files

To read a FLUENT boundary mesh (contained in a FLUENT mesh file created with GAMBIT or a boundary mesh contained in a FLUENT case file) into TGrid, you can do either of the following:

- Select the File/Read/Boundary Mesh... menu item to open the Select File panel and select the boundary mesh file to be read.
- Use the `file/read-boundary-mesh` text command and specify the name of the boundary mesh file to be read.

This is convenient if you want to read in a large volume mesh and recreate it by starting from the boundary mesh.

Reading Multiple Boundary Mesh Files

If the boundary mesh is contained in two or more separate files, you can read them in together and have TGrid assemble the complete boundary mesh. Alternatively, you can use the `file/read-multi-bound-mesh` text command for reading multiple boundary mesh files.

6.1.2 Reading TGrid Mesh Files

To read a TGrid mesh, you can use either of the following:

- Select the File/Read/Mesh... menu item to open the Select File panel and select the TGrid mesh file to be read.
- Use the `file/read-mesh` text command and specify the name of the mesh file to be read.

You can also use either of these commands to read a FLUENT mesh file created with GAMBIT, or to read the mesh file contained in a FLUENT case file. To do the latter, you can also use the text command `file/read-case`.

The commands discussed here are used to read a 2D mesh into the 2D version of TGrid, or a 3D mesh into the 3D version of TGrid.

i TGrid cannot read grids from solvers that have been adapted using hanging nodes. To read one of these grids into TGrid, coarsen the mesh within the solver until you have recovered the original unadapted grid.

Reading Multiple TGrid Files

If the mesh is contained in two or more separate files, you can read them together in TGrid and assemble the complete mesh. For example, if you are creating a hybrid mesh by reading in a triangular boundary mesh and a volume mesh consisting of hexahedral cells, read both files at the same time using the File/Read/Mesh... menu item or the `file/read-multiple-mesh` or `file/read-multiple-case` text commands.

You can also use the `file/read-meshes-by-tmerge` text command. This command uses the tmerge utility.

Reading 2D Mesh Files in the 3D Version of TGrid

It is also possible to read 2D meshes from FLUENT into the 3D version of TGrid. To read a 2D mesh into the 3D version of TGrid, use the File/Import/Fluent 2D Mesh... menu item or the `file/import/fluent-2d-mesh` text command.

Reading 3D Mesh Files in the 2D Version of TGrid

It is also possible to read 3D meshes from FLUENT into the 2D version of TGrid. To read a 3D mesh into the 2D version of TGrid, use the File/Import/Fluent 3D Mesh... menu item or the `file/import/fluent-3d-mesh` text command.

6.1.3 Appending Mesh Files

You can also read multiple mesh files one by one instead of reading all of them at a time. This process is called as appending the mesh files. To do so, read in the first mesh file using the Select File panel. Reopen the panel and enable Append File(s) check button and read the remaining files one by one. For details, see Section 3.2.6: Select File Dialog Box.

You can also append the files using the following TUI command:

- `file/append-meshes-by-tmerge` allows you to append the mesh files using tmerge. There is no GUI item in TGrid for this TUI command.

- `file/append-mesh` allows you to append the mesh files. This command is same as that of the Append File(s) check button in the Select File dialog box.

The Append File(s) *check button is not accessible while reading the first mesh file.*

6.1.4 Writing Mesh Files

To write a mesh file in the format that can be read by FLUENT, do either of the following:

- Select the File/Write/Mesh... menu item to open the Select File panel and specify the name of the mesh file to be written.
- Use the `file/write-mesh` text command and specify the name of the mesh file to be written.

To write a case file in the format that can be read by FLUENT, do either of the following:

- Select the File/Write/Case... menu item.
- Use the `file/write-case` text command and specify the name of the mesh file to write.

Selecting the File/Write/Mesh... and File/Write/Case... menu item will invoke the Select File panel, where you will specify the name of the mesh (or case) file to be written. By default, a binary file will be written when you write a mesh or case file. Binary files take up less memory than text files and can be read and written by TGrid more quickly.

You can disable the Write Binary Files option in the Select File dialog to write the file in text format. The text file can be edited, but it will require more storage space than the corresponding binary file. You can also use the TUI command `/file/file-format` to toggle the writing of binary files.

Since FLUENT has the same mesh file format as TGrid, you will be able to read this file back into TGrid using the File/Read/Mesh... menu item (or the `file/read-mesh` text command) if you should need to modify the mesh at a later time.

If you are writing a hexcore mesh, enable the Write As Polyhedracheck button in the Select File panel. This allows TGrid to create polyhedral cells. Polyhedral cells are created when 'hex' and 'tet' cells are merged with each other. Enabling this option allows TGrid to export these cells instead of non-conformal meshes.

⚠ Case files containing polyhydral cells cannot be read in TGrid.

6.1.5 Writing Boundary Mesh Files

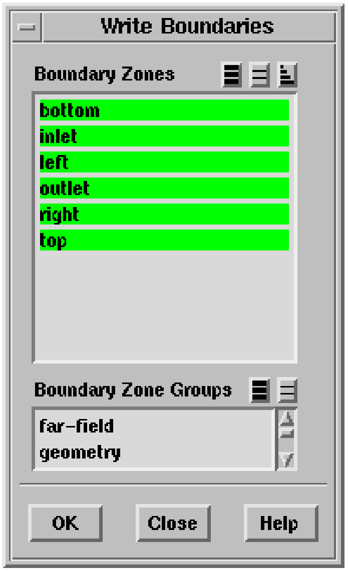
TGrid allows you to write a mesh file comprising specific boundary zones. This is useful for large cases where you may want to mesh different parts of the mesh separately and then merge them together. This allows you to avoid frequent switching between domains for such cases. You can write out selected boundaries to a mesh file and then create the volume mesh for the part in a separate TGrid session. You can then read the saved mesh into the previous TGrid session using the Append File(s) option and merge the part with the rest of the mesh.

To write a mesh file comprising selected boundaries, do either of the following:

- Use the File/Write/Boundaries... menu item to invoke the Write Boundaries panel (see Section 6.1.5: The Write Boundaries Panel) and specify the boundaries to be written.
- Use the `/file/write-boundaries` text command and specify the name of the file to be written and the boundaries to be written.

The Write Boundaries Panel

The Write Boundaries panel allows you to specify the boundaries to be written to a mesh file.



Controls

Boundary Zones contains a list of the boundary zones available.

Boundary Zone Groups contains a list of the boundary zone groups and user-defined groups available.

6.2 Compressed Files

TGrid allows you to read and write compressed files. Use the Select File dialog box to read or write the files that have been compressed using `compress` or `gzip`.

6.2.1 Reading Compressed Files

If you select a compressed file with a `.Z` extension, TGrid will automatically invoke `zcat` to import the file. If you select a compressed file with a `.gz` extension, TGrid will invoke `gunzip` to import the file. For example, if you select a file named `flow.msh.gz`, TGrid will report the following message indicating that the result of the `gunzip` is imported into TGrid via an operating system pipe.

```
Reading "| gunzip flow.msh.gz"...
```

You can also read a compressed file using the text interface by entering the file name. First, TGrid attempts to open a file with the input name. If it cannot find a file with that name, it attempts to locate files with default suffixes and extensions appended to the name. For example, if you enter the name `file-name`, TGrid traverses the following list until it finds an existing file:

- `file-name`
- `file-name.gz`
- `file-name.Z`
- `file-name.suffix`
- `file-name.suffix.gz`
- `file-name.suffix.Z`

where `suffix` is a common extension to the file, such as `.cas` or `.msh`. TGrid reports an error if it fails to find an existing file with one of these names.

Note: *For Windows systems, only files that were compressed with* `gzip` *(i.e., files with a* `.gz` *extension) can be read. Files that were compressed using* `compress` *cannot be read into* TGrid *on a Windows machine.*

6.2.2 Writing Compressed Files

You can write a compressed file using the text interface by entering the file name with the appropriate extension. For example, if you are prompted for a file name and you enter a file name with a `.gz` extension, a compressed file will be written. The following TUI command can be used:

```
/file> wm
mesh file name [""] "flow.gz"

Writing "| gzip -cfv > flow.msh.gz"...
```

The status message indicates that the mesh file information is being piped into the `gzip` command, and that the output of the compression command is being redirected to the file with the specified name. In this particular example, the `.msh` extension was added automatically.

Note: *The writing of compressed file will fail if the `compress` or GNU `gzip` compression routine is not available on your platform.*

6.3 Reading Scheme Source Files

A Scheme source file can be loaded in three ways:

- Through the menu system as a scheme file
- Through the menu system as a journal file
- Through Scheme itself

For large source files use the Select File panel invoked by selecting the File/Read/Scheme... menu item or the Scheme `load` function.

```
> (load "file.scm")
```

Shorter files can also be loaded with the File/Read/Journal... menu item or the `file/read-journal` command in the text interface (or its `.` or `source` alias).

```
> . file.scm

> source file.scm
```

In this case, each character of the file is echoed to the console as it is read in the same way as if you were typing the contents of the file.

6.4 Journal Files

TGrid creates a journal file by recording everything you type on the command line or enter through the GUI. The GUI commands are recorded as Scheme code lines in journal files. You can also create journal files manually with a text editor.

The purpose of a journal file is to automate a series of commands instead of entering them repeatedly on the command line. It can also be used to produce a record of the input to a program session for later reference, although transcript files are often more useful for this purpose (see Section 6.5: Transcript Files).

Command input is taken from the specified journal file until its end is reached, at which time control is returned to the standard input (usually the keyboard). Each line from the journal file is echoed to the standard output (usually the screen) as it is read and processed.

Note: *A journal file by design is a simple record/playback facility. Hence it knows nothing about the state in which it was recorded or the state in which it is being played back. Therefore, recreate the state in which the journal was written before you read it into the program.*

For example, if the journal file includes an instruction for TGrid to save a new file with a specified name, check that no file with that name exists in your directory before you read in your journal file. If a file with that name *exists* and you read in your journal file, it will prompt for a confirmation to overwrite the old file when the program reaches the write instruction.

The journal file contains no response to the confirmation request. Hence, TGrid will not be able to continue following the instructions of the journal file. Other conditions that may affect the ability of the program to perform the instructions contained in a journal file can be created by modifications or manipulations that you make within the program.

At a given time, only one journal file can be open for recording. But you can read a journal file at any time. You can also write a journal and a transcript file simultaneously.

6.4.1 Using the GUI

To start the journal file, select the File/Write/Start Journal... menu item. Enter a name for the file in the Select File panel. The journal recording begins and the Start Journal... menu item becomes Stop Journal menu item. You can stop journal writing by selecting Stop Journal, or by exiting the program.

You can read a journal file into the program using the Select File dialog box invoked by selecting the File/Read/Journal... menu item.

6.4.2 Using Text Commands

To start the journaling process using the `file/start-journal` command, and end it with the `file/stop-journal` command (or by exiting the program). To read a journal file into the program, use the `file/read-journal` command.

Note: *The* `read-journal` *command always loads the file in the main (i.e., top-level) menu, regardless of where you are in the menu hierarchy when you invoke it.*

The standard period (.) alias is the same as the `file/read-journal` definition and is defined by:

```
(alias '. (lambda () (ti-read-journal)))
```

6.5 Transcript Files

A transcript file contains a complete record of all standard: input to TGrid (specially keyboard and GUI input), output from TGrid (usually all screen output).

The GUI commands are recorded as Scheme code lines in transcript files. TGrid creates a transcript file by recording everything typed as input or entered through the GUI, and everything printed as output in the text window.

The purpose of a transcript file is to produce a record of the program session for later reference. The transcript file cannot be read back into the program because they contain messages and other output transcript files.

ℹ At a time, only one transcript file can be open for recording. But you can write a transcript and a journal file simultaneously. You can also read a journal file while a transcript recording is in progress.

6.5.1 Using the GUI

To start the transcripting process, select the File/Write/Start Transcript... menu item. Enter a name for the file in the Select File panel. The transcript recording begins and the Start Transcript... menu item becomes the Stop Transcript menu item. You can end transcript recording by selecting Stop Transcript, or by exiting the program.

6.5.2 Using Text Commands

In the text interface, start the transcripting process with the `file/start-transcript` command, and end it with the `file/stop- transcript` command (or by exiting the program).

6.6 Domain Files

Each mesh file written by TGrid has a domain section. A domain file is the domain section of the mesh file and is written as a separate file. It contains a list of node, face, and cell zone IDs that comprise each domain in the mesh.

By convention, domain file names are composed of a root with the suffix `.dom`. If you conform to this convention, you do not have to type the suffix when prompted for a filename; it will be added automatically. When TGrid reads a domain file, it first searches for a file with the exact name you typed. If a file with that name is not found, it will search for a file with `.dom` appended to the name. When TGrid writes a domain file, `.dom` will be added to the name you type unless the name already ends with `.dom`.

6.6.1 Reading Domain Files

To read the domain files into TGrid, do either of the following:

- Select the File/Read/Domains... menu item to invoke the Select File panel and specify the name of the domain file to be read.
- Enter the `file/read-domains` text command and specify the name of the domain file to be read.

If a domain that is being read already exists in the mesh, a warning message is displayed. TGrid *verifies if the zones defining the domains exist in the mesh. If not, it will display a warning message.*

6.6.2 Writing Domain Files

To write the domain files into TGrid, do either of the following:

- Select the File/Write/Domains... menu item to invoke the Select File panel and specify the name of the domain file to be written.
- Enter the `file/write-domains` text command and specify the name of the domain file to be read.

6.7 Importing Files

TGrid allows you to import the following file formats:

- ANSYS Prep7/cdb files
- CGNS files
- FIDAP neutral files
- GAMBIT neutral files
- HYPERMESH ASCII files
- I-deas Universal files
- NASTRAN files
- PATRAN neutral files
- STL files

Importing Multiple Files

You can also import multiple files into TGrid using the File/Import menu. Select the file format (e.g., ANSYS prep7/cdb) and the mesh type (surface or volume) to open the Select File dialog box. Select the appropriate files from the Files selection list and click OK. Alternatively, use the appropriate TUI command (e.g.,`file/import/ansys-surf-mesh`) and specify the names of the files to be imported.

Appending Multiple External Files

You can also add files of any external format to an existing mesh. This is known as appending files. To append external files, read or import the first file. Use the File/Import menu and select the appropriate file format (e.g., ANSYS prep7/cdb) and the mesh type (surface or volume). Enable Append File(s) in the Select File dialog box and import the necessary files.

6.7.1 Importing Mesh Files Generated by Third-Party Packages

TGrid also allows you to import mesh information generated by some CAD packages (ANSYS, I-deas, NASTRAN, PATRAN, and HYPERMESH) as well as mesh information in the CGNS (CFD general notation system) format. These files are imported into TGrid using the menu items in the Import submenu, or using the associated text commands.

Reading ANSYS Prep7 Files

1. Select the File/Import/ANSYS prep7/cdb menu item.
2. Select the Surface... menu item to read a surface Prep7 file, or the Volume... menu item to read a volume Prep7 file.

OR

Use one of the following TUI commands:

- `file/import/ansys-surf-mesh`
- `file/import/ansys-vol-mesh`

Reading CGNS Files

1. Select the File/Import/CGNS menu item.
2. Select the Surface... menu item to read a surface mesh or the Volume... menu item to read a volume mesh.

OR

Use one of the following TUI commands:

- `file/import/cgns-surf-mesh`
- `file/import/ cgns-vol-mesh`

Reading I-deas Universal Files

1. Select the File/Import/IDEAS universal menu item.
2. Select the Surface... menu item to read a surface Universal file or the Volume... menu item to read a volume Universal file.

OR

Use one of the following TUI commands:

- `file/import/ideas-surf-mesh`
- `file/import/ideas-vol-mesh`

Reading NASTRAN Files

1. Select the File/Import/NASTRAN menu item.
2. Select the Surface... menu item to read a surface file or the Volume... menu item to read a volume file.

OR

Use one of the following TUI commands:

- `file/import/nastran-surf-mesh`
- `file/import/nastran-vol-mesh`

Reading PATRAN Neutral Files

1. Select the File/Import/PATRAN neutral menu item.
2. Select the Surface... menu item to read a surface mesh, or the Volume... menu item to read a volume mesh.

OR

Use one of the following TUI commands:

- `file/import/patran-surf-mesh`
- `file/import/patran-vol-mesh`

Reading HYPERMESH ASCII Files

1. Select the File/Import/HYPERMESH Ascii menu item.
2. Select the Surface... menu item to read a surface mesh or the Volume... menu item to read a volume mesh.

OR

Use one of the following TUI commands:

- `file/import/hypermesh-surf-mesh`
- `file/import/hypermesh-vol-mesh`

Reading STL (Stereo Lithography Format) Files

Select the File/Import/STL... menu itemor use the `file/import/stl` TUI command.

For information about the format of these files and details about importing them (if the import commands are not available on your computer), see Appendix A: Importing Boundary and Volume Meshes. For information about changing the options related to grid import see Section 6.7.4: Grid Import Filter Options.

6.7.2 Importing FIDAP Neutral Mesh Files

To read a FIDAP neutral file, use either of the following:

- Select the File/Import/FIDAP neutral menu item, and then select the Surface... menu item to read a surface mesh, or the Volume... menu item to read a volume mesh.
- Use the `file/import/fidap-surf-mesh` or `file/import/fidap-vol-mesh` text interface commands.

6.7.3 Importing GAMBIT Neutral Mesh Files

To read a GAMBIT neutral file, use either of the following:

- Select the File/Import/GAMBIT neutral menu item, and then select the Surface... menu item to read a surface mesh, or the Volume... menu item to read a volume mesh.
- Use the `file/import/gambit-surf-mesh` or `file/import/ gambit-vol-mesh` text interface commands.

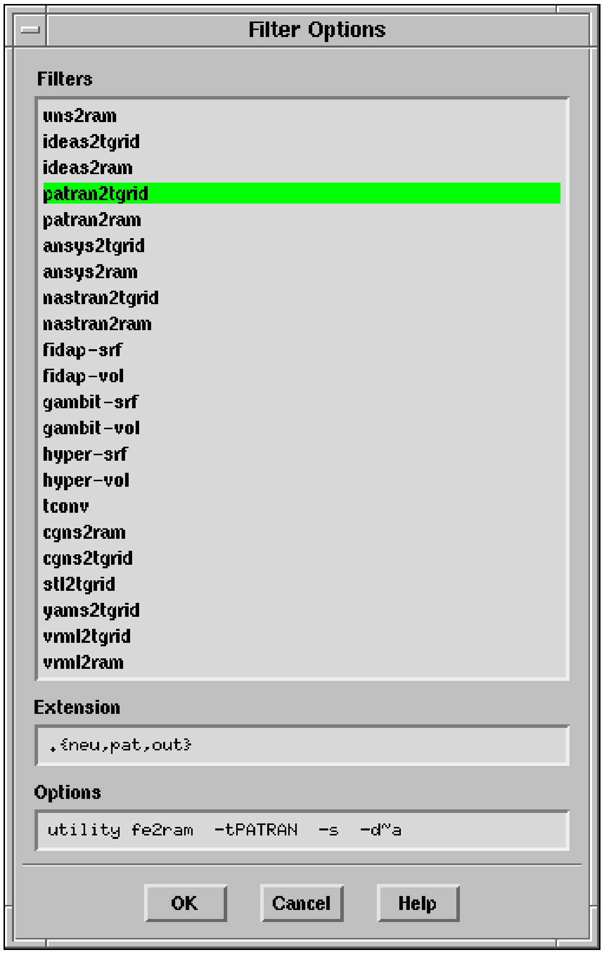
6.7.4 Grid Import Filter Options

The filter is used by TGrid to import mesh files from third-party packages can take different arguments (see Appendix A: Importing Boundary and Volume Meshes). You can control these arguments and the extensions of the files to be converted, using the Filter Options panel or the related text commands.

File ⟶ Import ⟶ Options...

The Filter Options Panel

The Filter Options panel allows you to change the extension (e.g., `.cas`, `.msh`, `.neu`) and arguments used with a specified filter.



Filters

uns2ram
ideas2tgrid
ideas2ram
patran2tgrid
patran2ram
ansys2tgrid
ansys2ram
nastran2tgrid
nastran2ram
fidap-srf
fidap-vol
gambit-srf
gambit-vol
hyper-srf
hyper-vol
tconv
cgns2ram
cgns2tgrid
stl2tgrid
yams2tgrid
vrml2tgrid
vrml2ram

Extension

```
.{neu,pat,out}
```

Options

```
utility fe2ram  -tPATRAN  -s  -d~a
```

Controls

Filters contains a list of the available file converters. When you select a name from this list, the corresponding Extension and Options will be displayed.

A single filter (fe2ram) is used for files from all CAD packages. The arguments of the fe2ram filter will indicate the CAD package used to create the file. The name selected in the Filters list (e.g., ideas2tgrid) identifies the CAD package that created the file, and whether the file is a boundary mesh or volume mesh.

For example, ideas2tgrid is used for importing a boundary mesh from I-deas, and ideas2ram is used for importing a volume mesh from I-deas. When you select a filter name in the Filters list, the appropriate file extension and arguments for the fe2ram filter will appear under Extension and Options.

Extension is the extension of the third-party mesh file to be imported.

Options are the arguments used by the filter selected in the Filters list (or by the fe2ram filter).

For some filters, one of the arguments will be the dimensionality of the grid. For such filter, TGrid will show a default dimensionality argument of `-d~a`. TGrid will automatically determine if the grid is 2D or 3D. So you need not substitute `2` or `3` for `~a`. For information about the import filters and their related arguments, see Appendix A: Importing Boundary and Volume Meshes.

Text Commands for Setting Filter Options

To import mesh files from third-party packages, use the following text commands:

`file/filter-list` lists the names of the converters that are used to change third-party mesh files to TGrid format.

`file/filter-options` allows you to change the extension (e.g., `.cas`, `.msh`, `.neu`) and arguments used with a specified filter.

For example, if you have saved the PATRAN files with a `.NEU` extension instead of `.neu`, you can either substitute or add `.NEU` to the extension list. For some filters, one of the arguments will be the dimensions of the grid. When you use the `filter-options` command.

6.8 Exporting Files

You can save the mesh to a file that can be read by HYPERMESH, NASTRAN, PATRAN, and ANSYS.

6.8.1 Exporting HYPERMESH Files

To save your mesh to a file that can be read by HYPERMESH, select the File/Export/HYPERMESH... menu item or use the `file/export/hypermesh` TUI command and specify the name for the HYPERMESH file.

6.8.2 Exporting NASTRAN Files

To save the mesh to a file that can be read by NASTRAN, select the File/Export/NASTRAN... menu item or use the `file/export/nastran` TUI command and specify the name for the NASTRAN file.

6.8.3 Exporting PATRAN Files

To save the mesh to a file that can be read by PATRAN, select the File/Export/PATRAN... menu item or use the `file/export/patran` TUI command and specify the name for the PATRAN file.

6.8.4 Exporting ANSYS and STL Files

To save the mesh to a file that can be read by ANSYS, select the File/Export/ANSYS... menu item or use the `file/export/ansys` TUI command and specify the name for the ANSYS file.

To save the mesh to a file that can be read by third-party packages, use the `file/export/stl` TUI command and specify the name for the STL file.

Note: *This option is only available in TUI.*

6.9 Saving Hardcopy Files

Graphics window displays can be saved in various formats such as TIFF, EPS, and PostScript. There may be slight differences between the hardcopies and the displayed graphics windows. This is because hardcopies are generated using the internal software renderer, while the graphics windows may utilize specialized graphics hardware for optimum performance.

Many systems provide a utility to dump the contents of a graphics window into a raster file. This is generally the fastest method of generating a hardcopy (since the scene is already rendered in the graphics window), and guarantees that the hardcopy will be identical to the window.

6.9.1 Using the Hardcopy Panel

Use the Hardcopy panel to set the hardcopy parameters and to save the hardcopy files. The controls and the equivalent text interface commands are described in Sections 6.9.2 and 6.9.3. The procedure for saving a hardcopy file is as follows:

1. Open the Hardcopy panel.
2. Select the appropriate file format.
3. Specify the file type, if applicable (optional).
4. Set the coloring.
5. Define the resolution, if applicable (optional).
6. If you are generating a window dump, specify the command to be used for the dump.
7. Preview the result (optional).
8. Click the Save... button and enter the filename in the resulting Select File dialog box.

Click Apply instead of Save... to save the current hardcopy settings, instead of saving a hardcopy. The applied settings will become the defaults for subsequent hardcopies.

Choosing the Hardcopy File Format

To choose the hardcopy file format, select one of the following options in the Format list:

EPS (Encapsulated PostScript) output is the same as PostScript output, with the addition of Adobe Document Structuring Conventions (v2) statements. Currently, no preview bitmap is included in EPS output. Often, programs which import EPS files use the preview bitmap to display on-screen, although the actual vector PostScript information is used for printing (on a PostScript device). You can save EPS files in raster or vector format.

IRIS Image is the native raster image file format on SGI computers. The IRIS Image driver may not be available on all platforms.

JPEG is a common raster file format.

PPM output is a common raster file format.

PostScript is a common vector file format. You can also choose to save a PostScript file in raster format.

TIFF is a common raster file format. The TIFF driver may not be available on all platforms.

PNG is a common raster file format.

VRML is a graphics interchange format that allows export of 3D geometrical entities that you can display in the TGrid graphics window. This format can commonly be used by VR systems and in particular the 3D geometry can be viewed and manipulated in a web-browser graphics window.

Non-geometric entities such as text, titles, color bars, and orientation axis are not exported. In addition, most display or visibility characteristics set in TGrid, such as lighting, shading method, transparency, face and edge visibility, outer face culling, and hidden line removal, are not explicitly exported but are controlled by the software used to view the VRML file.

Window Dump (UNIX systems only) selects a window dump operation for generating the hardcopy. With this format, you will need to specify the appropriate Window Dump Command.

Choosing the File Type

To save a PostScript or EPS file, you can choose either of the following file types:

- Raster: A raster file defines the color of each individual pixel in the image. Raster files have a fixed resolution. Raster supports IRIS image, JPEG, PostScript, EPS, and TIFF formats.
- Vector: A vector file defines the graphics image as a combination of geometric primitives like lines, polygons, and text. Vector files are usually scalable to any resolution. Vector supports PostScript, EPS, and VRML formats.

Note: *For the quickest print time, save vector files for simple 2D displays and raster files for complicated scenes.*

Specifying the Color Mode

For all formats except the window dump you can specify which type of Coloring you want to use for the hardcopy file.

- Select Color for a color-scale copy.
- Select Gray Scale for a gray-scale copy.
- Select Monochrome for a black-and-white copy.

Most monochrome PostScript devices will render Color images in shades of gray. Select Gray Scale to ensure that the color ramp is rendered as a linearly-increasing gray ramp.

Defining the Resolution

For raster hardcopy files, you can control the resolution of the hardcopy image by specifying the size in pixels. Set the desired Width and Height under Resolution. If the values of Width and Height are both zero, the hardcopy is generated at the same resolution as the active graphics window.

Note: *For PostScript and EPS files, specify the resolution in dots per inch (*DPI*) instead of setting the width and height.*

Options

For all hardcopy formats except the window dump, you can control two additional settings.

- Specify the orientation of the hardcopy using the Landscape Orientation option. If this option is enabled, the hardcopy is made in landscape mode (default); otherwise it is made in portrait mode.
- Control the background color using the White Background option.

  This feature allows you to make hardcopies with a white background and a black foreground, while the graphics windows are displayed with a black background and white foreground.

TGrid also provides options that allow you to save PostScript files that can be printed more quickly. These options are available in the `display/set/hardcopy/driver/post-format` text menu.

Window Dumps (UNIX Systems)

If you select the Window Dump format, the program will use the specified Window Dump Command to save the hardcopy file. For example, if you want to use `xwd` to capture a window, set the Window Dump Command to

```
xwd -id %w >
```

TGrid will automatically interpret `%w` to be the ID number of the active window when the dump occurs.

- To save the file in the Select File dialog box enter the filename for the output from the window dump (e.g., `myfile.xwd`)
- To make an animation, save the window dumps into numbered files, using the `%n` variable. To do this, enter the Window Dump Command `xwd -id %w >` and type `myfile%n.xwd` as the filename in the Select File dialog box.

Note: *Each time you create a new window dump, the value of %n will increase by one, so you do not need to track numbers to the hardcopy filenames manually.*

If you use the ImageMagick animate program, saving the files in MIFF format (the native ImageMagick format is more efficient. In such cases, use the ImageMagick tool `import`. For the Window Dump Command enter the default command:

```
import -window %w
```

Specify the output format to be MIFF by using the `.miff` suffix at the end of the filename.

The window-dump feature is both, system and graphics-driver-specific. The commands available for dumping windows depends on your system configuration.

> *i* The window dump will capture the window exactly as it is displayed, including the resolution, colors, transparency, etc. For this reason, all of the inputs that control these characteristics are disabled in the Hardcopy panel when you enable the Window Dump format.

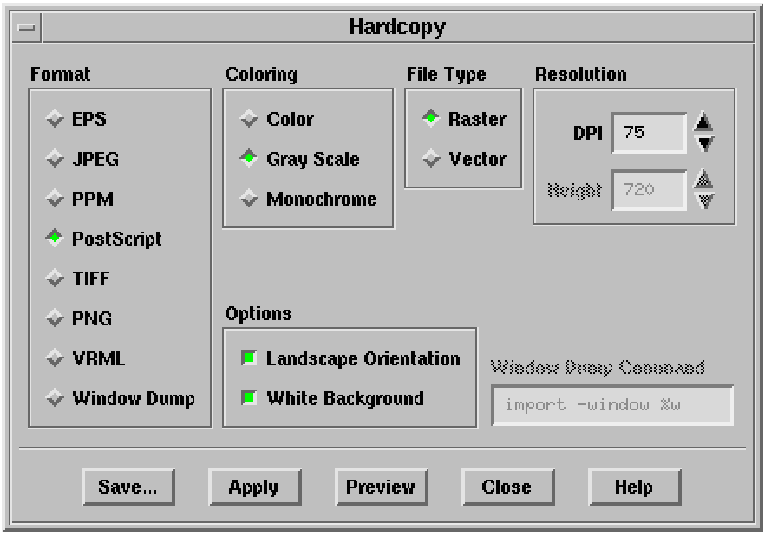
If you are using an 8-bit graphics display, you might want to use one of the built-in raster drivers (e.g., TIFF) to generate higher-quality 24-bit color output rather than dumping the 8-bit window. On the other hand, if your hardware supports transparency, a window dump is the only method to generate a hardcopy with transparent surfaces. If you save any other type of hardcopy file, the transparency effects will not be captured in the resulting image.

Previewing the Hardcopy Image

Before saving a hardcopy file, you can preview the image to be saved. Click Preview to apply the the current settings to the active graphics window so that you can see the effects of different options interactively before saving the hardcopy.

6.9.2 The Hardcopy Panel

The Hardcopy panel allows you to set graphics hardcopy parameters and save hardcopy files of graphics windows.



Format: ● PostScript

Coloring: ● Gray Scale

File Type: ● Raster

DPI 75

Height 720

☑ Landscape Orientation

☑ White Background

Window Dump Command: import -window %w

Format allows you to select the format of hardcopy files.

Coloring specifies the color mode for hardcopies. Hardcopies may be made in Color, Gray Scale, or Monochrome (black and white). Most monochrome PostScript devices will render Color images in shades of gray.

Select Gray Scale to ensure that the color ramp is rendered as a linearly-increasing gray ramp.

File Type specifies whether a Raster or Vector type hardcopy file is to be saved. You can choose either of these if you are saving a PostScript or EPS file. See Section 6.9.1: Choosing the File Type for details.

Resolution specifies the size of raster hardcopies in pixels. If the Width and Height are both zero, the hardcopy is generated at the same resolution as the active graphics window. For PostScript and EPS files, specify the resolution in dots per inch (DPI).

Options specifies whether or not to use the following options:

Landscape Orientation specifies the orientation of the hardcopy. If selected, the hardcopy is made in landscape mode; otherwise, it is made in portrait mode.

White Background specifies that the white background should be used in the hardcopy. This feature allows you to make hardcopies with a white background and a black foreground, while the graphics windows are displayed with a black background and white foreground.

Save... opens the Select File dialog box, where you can specify a name for the hardcopy file and then save it. The resulting file will contain a hardcopy of the active graphics window.

Apply saves the current settings. TGrid will use these settings when making subsequent hardcopies.

Preview applies the saved settings to the currently active graphics window so the effects of different options may be investigated interactively.

6.9.3 Text Interface for Saving Hardcopy Files

Text commands for saving hardcopy files and modifying hardcopy options are:

`display/hard-copy` saves a hardcopy file of the active graphics window.

`display/set/hardcopy/color-mode/` contains the available color modes.

`display/set/hardcopy/color-mode/color` selects full color.

`display/set/hardcopy/color-mode/gray-scale` selects gray scale (i.e., various shades of gray).

`display/set/hardcopy/color-mode/mono-chrome` selects black and white.

`display/set/hardcopy/color-mode/list` displays the current hardcopy color mode.

`display/set/hardcopy/driver/` contains the available hardcopy formats.

`display/set/hardcopy/driver/dump-window` sets the command to dump a graphics window to a file.

`display/set/hardcopy/driver/eps` sets Encapsulated PostScript format.

`display/set/hardcopy/driver/image` sets IRIS image format.

`display/set/hardcopy/driver/jpeg` sets JPEG image format.

`display/set/hardcopy/driver/post-script` sets PostScript format.

`display/set/hardcopy/driver/ppm` sets PPM format.

`display/set/hardcopy/driver/tiff` sets TIFF format.

`display/set/hardcopy/driver/list` displays the current hardcopy format.

`display/set/hardcopy/driver/options` allows you to set hardcopy options, such as landscape orientation, and physical size. The options may be entered on one line if you separate them with commas.

`display/set/hardcopy/driver/post-format/` contains commands for setting the PostScript driver format.

`display/set/hardcopy/driver/post-format/fast-raster` enables a raster file that may be larger than the standard raster file, but will print much more quickly.

`display/set/hardcopy/driver/post-format/raster` enables the standard raster file.

`display/set/hardcopy/driver/post-format/rle-raster` enables a run-length encoded raster file that will be about the same size as the standard raster file, but will print slightly more quickly. This is the default file type.

`display/set/hardcopy/driver/post-format/vector` enables the standard vector file.

`display/set/hardcopy/invert-background?` toggles the exchange of foreground and/or background colors for hardcopy files.

`display/set/hardcopy/landscape?` toggles between landscape or portrait orientation.

`display/set/hardcopy/preview` applies the settings of the color-mode, invert-background, and landscape options to the currently active graphics window to preview the appearance of printed hardcopies.

`display/set/hardcopy/x-resolution` sets the width of raster format images in pixels (0 implies that the hardcopy should use the same resolution as the active graphics window).

`display/set/hardcopy/y-resolution` sets the height of raster format images in pixels (0 implies that the hardcopy should use the same resolution as the active graphics window).

6.10 Saving the Panel Layout

The Save Layout command in the File pull-down menu allows you to save the existing panel and window layout. You can arrange panels and graphics windows on your screen in a preferred configuration and invoke the Save Layout command.

A `.cxlayout` file is written in your home directory. If you subsequently arrange different panels and save the layout again, the positions of these panels will be added to the positions of the panels that you saved earlier. If you move a panel for which a position is already saved, and then you save the layout, the new position will be written to the `.cxlayout` file.

In subsequent sessions, when you invoke a panel or create a graphics window, it will be positioned based on the saved configuration. Any panel or window not specified in the saved configuration will use the default position.

The `.cxlayout` file in your home directory applies to all Cortex applications (i.e., TGrid, FLUENT, FLUENT/UNS, RAMPANT, NEKTON, and MixSim).

6.11 The `.tgrid` File

When starting up, TGrid looks in your home directory for an optional file called `.tgrid`. If it finds the file, TGrid loads the file with the Scheme `load` function. This file may contain Scheme functions that customize the operation of the code.

6.12 Exiting TGrid

To exit TGrid do one of the following:

- Select Exit in the File pull-down menu.

  File ⟶Exit...

- Type `/exit` within any menu.

If the present TGrid state has not been written to a file, you will receive a warning message. You can either cancel the exit and write the mesh or continue to exit without saving the file.