SurfaceWorks 6.0 Reference: What's New

P.O. Box 684 / 54 Herrick Rd.
Southwest Harbor, Maine 04679 U.S.A.
voice   207-244-4100
fax   207-244-4171
email   support@aerohydro.com
website   www.aerohydro.com

What's New in SurfaceWorks 6.0

The "new" functionality added to the release of SurfaceWorks 6.0 is the following:

- View/Display/Set View
- View/Display/Render
- Error Details
- Insert/Copies
- New Weight Schedule Options
- New Commands
- Layer Dialog Improvements
- Dragging constraint on 3D points
- Access to all RGK entities through the Command Line interface
- Porcupine curvature profile
- Curvature profile for Snakes
- New On-line Help

View/ Modify/ Set View/ Parameters

Gives you explicit control of the camera location and viewing direction. This is to allow for visibility problems, e.g. What is the view normal to a particular curve at a specific point? or to get the camera back to a specific position for reproducing a rendered view. When you use View/ Set View, you effectively set a whole new orbit sphere for the camera.
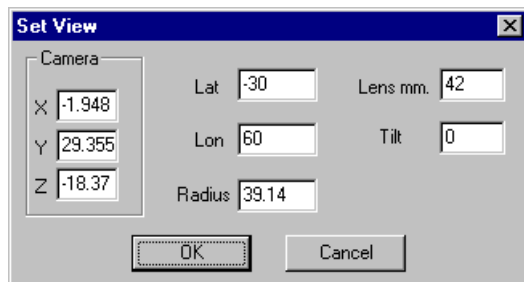
In the Set View dialog box, you specify:

Camera. Set the (X, Y, Z) position of the camera in the global coordinate system.

Lat. Set the latitude, in degrees, of the viewing direction.

Lon. Set the longitude, in degrees, of the viewing direction.

Radius. Set the distance from the camera to the center of the camera's orbit sphere.

Lens mm. Set the effective focal length of the camera lens. 42 mm is the default lens when a view is opened. A shorter focal length is like a closeup (30-35 mm) or fisheye (<30 mm) lens on a 35 mm camera, producing progressively stronger perspective. A longer lens is like a telephoto, which enlarges the image but reduces perspective. (Orthographic views correspond to the limit of infinite focal length.) <Home> restores the lens to the default 42 mm.

Tilt. Set the tilt of the camera sideways while keeping it aimed at the same look point. This action is like tilting (*not* twisting) your head from side to side — you see things from an angled perspective (relative to the vertical). Specify tilt in degrees:

- positive tilt rotates the camera counterclockwise (the image rotates clockwise)

- negative tilt rotates the camera clockwise (the image rotates counterclockwise)

The default values offered in the Set View dialog are those of the current viewing status.

If you specify a new camera location (X, Y, and/or Z), you shift both the camera *and* its orbit sphere, and you therefore also define a new point at which the camera is aimed and around which the camera will rotate. If you pick a small radius (say 0.1 ft), the action of the right and left arrow keys is like standing with your eye at the camera point and turning your head right and left.

Additional Viewing Options

While working in wireframe view there is sometimes a need to set the view parameters so the viewing direction is either normal (perpendicular) to, or parallel to, some geometric element. For example, suppose you have constructed a window on a flat cabin surface that is not parallel to any of the coordinate planes. In order to extract a CAD drawing of the window outline for manufacturing, you want to set a viewing direction normal to the plane of the cabin side surface.

View/Modify/Set View/Normal To allows setting the view "normal to" a variety of selected entities:

- A Plane

- Any three points ("normal to" means normal to the plane of the three points)

- A Frame (normal to any of the 3 coordinate planes of a Frame)

- A Magnet or Ring ("normal to" means normal to the host surface, at the location of the magnet or ring)

- A Curve (with the exception of a straight line) or Snake that is at least approximately planar ("normal to" means normal to the plane of the curve or snake)

- A Surface that is at least approximately planar ("normal to" means normal to a plane that is least-squares fit to the tabulation of the surface.)

View/Modify/Set View/Along allows setting the view "along" a variety of selected entities:

- A Line

- Any two points ("along" means along the line connecting them)
- A Plane (along some direction parallel to the plane)
- Bead or Ring ("along" means along the tangent to the supporting curve or snake, at the location of the bead or ring)
- A Frame (along any of its 3 axes)

View/ Display/ Render

Shortcut:

Presents a Rendering Options dialog, then displays a rendered, shaded, lighted view of the model in which surfaces appear as solids with hidden surfaces removed. This view always includes symmetry images. Initially, the rendered view defaults to a perspective viewpoint; but you can change back and forth between perspective and orthographic using the View menu options, toolbar buttons, or keyboard controls (<p>).

The initial rendered view takes its viewpoint from that of the active wireframe, or if invoked from a non-wireframe window, it takes the view of the most recently opened wireframe view. Specifically:

if the wireframe is a *perspective* view, the initial rendered viewpoint is exactly the same as the wireframe viewpoint

if the wireframe is an *orthographic* view, the initial rendered image uses the Lat, Lon of the wireframe, but creates a perspective image and does a Zoom All on it

Rendering options
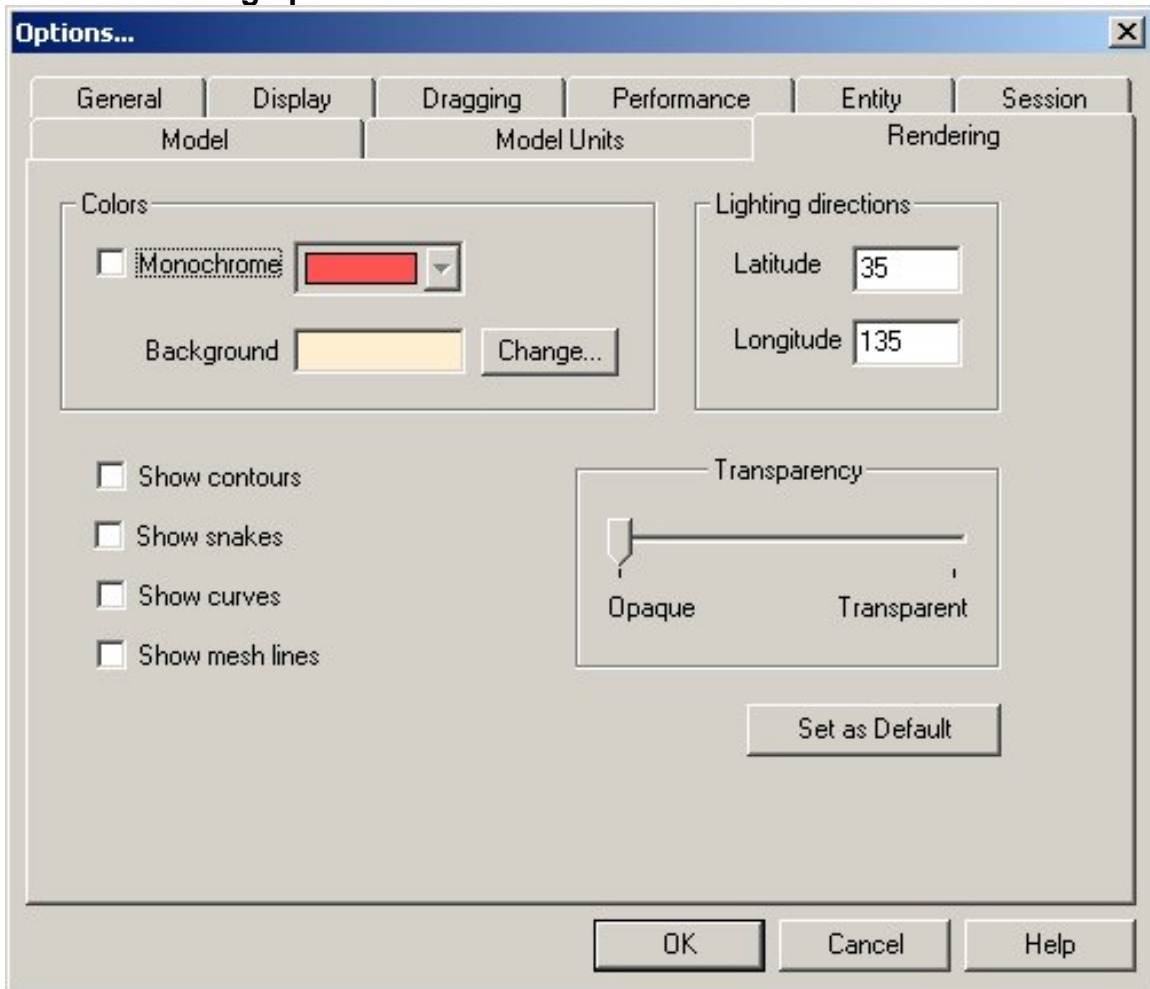
Colors. Wireframe colors or monochrome (radio buttons); if monochrome, specify the color (by number). The program default is wireframe colors.

Lighting directions. Set the latitude and longitude for the pair of rendering light sources. The program default is Latitude = 35, Longitude = 135. For details, see "

Rendered view lighting" on page 3-8.

Background color. Specify red, green, blue components with values between 0 and 155 — 0, 0, 0 is black; 255, 255, 255 is white.

Show contours, snakes, curves, and mesh lines. Check boxes toggle display of these lines on/off; the program default is off.

Tansparency. Select from fully opaque to…..invisible or anything in between.

<Make Default>. To customize the default settings for the render options, set up the dialog as you want it, then choose <Make Default>.

To abort displaying a rendered view, just cancel the window (you can do this while the program is drawing).

Rendered view lighting

In the rendered display, the brightness and hue of any region of a surface depends on a combination of:

> the surface's intrinsic color
> the color of light sources
> the surface's orientation with respect to light sources.

SurfaceWorks uses two light sources, both colored white, and of equal brightness. They are infinitely far away, so the light from each source consists of parallel rays. They are always located opposite each other in the global coordinate system, so they shine towards one another from opposite sides of the model. The light direction is fixed with respect to the model; it does not change when you move around the model with the cursor keys. You can control the light source directions by setting the latitude and longitude of one light source in the Render Options dialog. (The second light source will remain opposite.) The initial default direction is latitude 35, longitude 135.

With this lighting system, portions of a surface that are nearly perpendicular to the light direction are bright, and portions which are nearly tangent to the light direction are dark. For most models consisting of curved surfaces, this shading provides a nice level of contrast which makes the surfaces appear attractively three-dimensional. However, a result of the lighting system is that a flat surface is displayed with uniform coloration.

The lighting system also contains some "specular reflectivity", which produces white highlights in areas where the surface is suitably oriented to reflect light in a mirror-like fashion. Occasionally you can get in a position where specular reflection off a flat or nearly flat area dominates the image. This can always be corrected by changing the lighting direction. Producing the most attractive rendered views often requires two stages of setup: (1) moving the camera to the best position with respect to the model, considering framing, proportion and visibility of various model features; then (2) adjusting the light source direction to produce the nicest shading and highlights.

More realistic effects

In a rendered view of SurfaceWorks's type, no shadows are cast, and no reflections of one surface in another occur. These more realistic effects require ray tracing techniques, which are significantly more complex and laborious than shaded, lighted views. Ray tracing is not performed in SurfaceWorks, but is available in many add-on graphics packages such as 3D Studio (Autodesk, Inc.) and AccuRender (Robert McNeel Associates). See also *Reference* chapter "Making Hardcopy Pictures of SurfaceWorks Models".

ErrorView Enhancements

Error Details window

The error view has a window that reports additional useful diagnostic information about some entity errors. For example, when a ProjSnake which has failed is highlighted in the error view listbox, the Error Details window will display "Projection failed at t = XXX", where XXX is the t parameter location where the problem occurred. With this clue, you might be able to go to that particular place (for example, put a bead at that t position on the supporting curve) and understand why the projection failed at that location.

As another example, if your TrimSurf gets error 317, "Snakes don't connect up end-to-end to make closed loops", the Error Details window might display "Snakes 4 and 5 failed to join up; u,v-distance .00241 vs. tolerance .00010." This reveals the location of the problem, so you can investigate that particular junction and discover why it isn't sufficiently accurate.

Insert / Copies

Insert/ Copies is a SurfaceWorks tool for "mass production" of entities that have similar constructions. You basically say, "I want to create a bunch of entities that are like *this* one (the same entity type, in particular), but constructed using *this* and *this* and *this*, etc., as supports".

(See previously documented funtionality in on-line help in "Commands" section. A tutorial is provided as well.)

You select the entity to be copied (the "pattern" entity) first, then the set of new supports, and pick Insert/ Copies from the menu. (or the Keyboard Shortcut <C>) This opens the Insert Copies dialog, as follows:
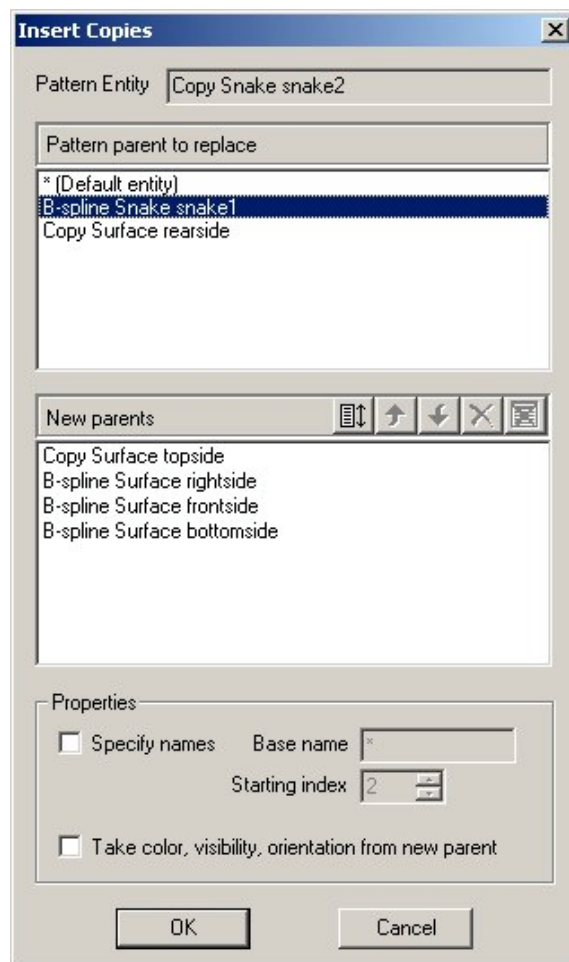
The "Pattern entity" window shows the name of the pattern entity -- the first entity selected, in this case a CopySnake. Each of the copies will be a CopySnake.

The upper list box, "Pattern parent to replace", shows all the parents of the pattern entitity, in order. The one that is highlighted is the one that will be replaced in making the copies. You can click on a different parent to change the selection. (In this case, since the "New parents" are snakes, BSnake snake1 is the only parent that makes sense for replacement; the new parents cannot serve as either relabels or surfaces.)

The lower list box, "New parents", shows the remainder of the selection set. If there are more entities selected than can be displayed, the window will have a vertical scroll bar; you can scroll up and down to review the list. In this case there are 3 new parents, so we are making 3 copies.

In case, on reviewing the "new parents" list, you find you have included one or more entities that shouldn't be there, you can highlight the incorrect entity and click the "Remove" button, to remove it from the set.

The "Specify names" checkbox allows control over the naming of the copies. Names are constructed from a base name by adding indices; you can choose the base name and the starting index. In this case the copies will be named 'D2', 'D3' and 'D4'. (If any of these names is already in use, the dialog will inform you and allow other choices.)

If you leave "Specify names" unchecked, the program will choose an available base name for you, and use 2 as the starting index.

The final checkbox "Take color, visibility, orientation from new parent" transfers these attributes from the new parent. (The alternative is to take them from the pattern entity.)

Weight Schedule Options

Tools/ Weight Schedule now has several options that allow weight, center of gravity and moment of inertia calculations to be made with subsets of entities.

When no weight schedule is open and Tools/ Weight Schedule is selected from the menu, the result is a dialog like this one:
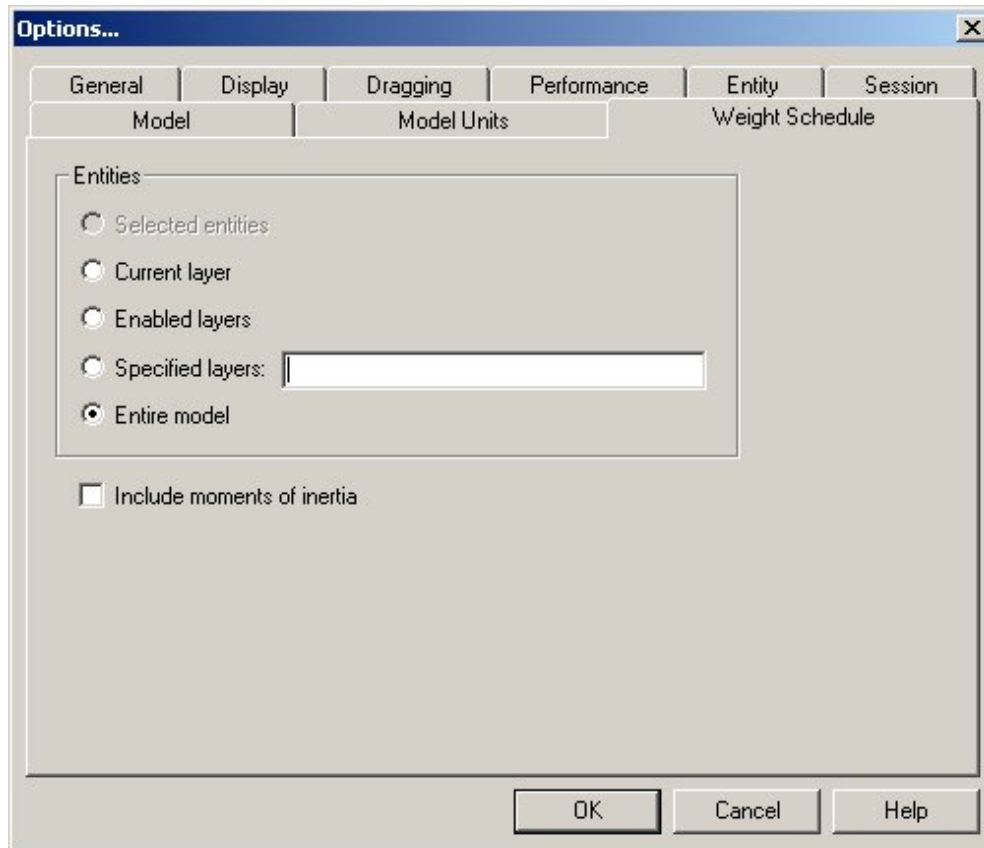
This dialog contains a number of possibilities, so let's see how they work.

The radio buttons in the "Entities" group are the principal controls. Each puts the Weight Schedule view into a different mode, which will be maintained until you come back to this dialog and change it. There are five modes, some of which utilize or interact with other controls in the dialog.

"Selected entities" mode

Sometimes it's useful to monitor the weight and c.g. of a single entity, a small group of entities, or a large set of entities that is still a subset of the entire model. An Entity List can be used to store a single list of *weight schedule entities* that you have selected for this purpose. The Entity List would need to be the only entity in the selection set. The weight schedule report will reflect the contents of the list even though it is the only selected item, i.e. the Entity List is "expanded" (all generations) before use for the weight schedule.

"Current layer" mode

In this mode, the weight schedule is calculated only from entities that are on the current layer (set in the Settings/ Layers dialog). If you change the current layer setting while in this mode (for example, from layer 9 to layer 12), the weight schedule will stop showing the entities on the old current layer and start showing the entities on the new current layer.

"Enabled layers" mode

In this mode, the weight schedule is calculated using only entities that are on currently enabled layers (layers that are checked "On" in the Settings/ Layers dialog). If you change the choice of which layers are on, the Weight Schedule will generally display a different set of entities.

"Specified layers" mode

In this mode, you can choose any combination of layers, regardless of whether they are on or off. The selection of layers is specified in the edit window labeled "Layers", described below. This selection of layers stays constant when the Weight Schedule is switched to other modes, but has no effect on operation in the other mode.

"Layers" window
The "Layers" control is the key to specifying the set of layers to be used for weight schedule calculation in the "Specified layers" mode. Use single layer numbers or ranges, separated by commas or spaces. Ranges are indicated by the first and last indices, separated by a single hyphen (and no spaces). Duplicates, negative indices, and indices outside the range 0-255 will be ignored.

"Entire model" mode

In this mode, weights are calculated and summed for all entities in the model that have nonzero unit weight, regardless of what layer they are on, or whether their layers are enabled, or are listed in the "Layers" window. ("Entire model" mode is like the way Weight Schedule has always worked in versions of SurfaceWorks before 6.0.)

"Include Moments of Inertia" checkbox

This checkbox makes moments of inertia calculations optional. (While moments of inertia are critical for some forms of analysis -- for example seakeeping, motions in waves -- they are irrelevant for most weight schedule applications.) The default setting for this checkbox is "Off" (unchecked).

OK and Cancel Buttons

"OK" accepts the settings of the dialog, and launches the Weight Schedule view, with those settings. "OK" also causes the program to store the current settings and choices made in the dialog, as the default settings for the next time Weight Schedule is invoked.

"Cancel" bypasses launch of the Weight Schedule view. The dialog simply closes, and any changes in settings that were made in the dialog are discarded.

Automatic updating of the Weight Schedule View

The Weight Schedule is organized as a view, so it updates itself whenever anything in the model changes, even if the Weight Schedule is behind other views. When it updates, it uses the current mode and other specifications that were set in the Weight Schedule Options dialog when the view was last opened.

Empty Weight Schedule

It's entirely possible for the Weight Schedule to open with no entities at all displayed, and zeroes for the weight and c.g. This will happen if no entities are currently selected for the weight schedule to use, that have non-zero Weight attributes. For example:

1. If you are in "Selected entities" mode, and have selected no entities.

2. If you are in "Selected entities" mode, and none of the selected "weight schedule entities" has a Weight attribute.

3. If you are in "Current layer" mode, and there are no entities on the current layer that have Weight attributes assigned.

4. If you are in "Entire model" mode, and there are no entities in the entire model that have Weight attributes assigned.

Weight Distribution File

Whenever a Weight Schedule is generated, a text file named %wt.dist.txt is written in the same folder as the model file. This is the weight distribution file required for longitudinal strength calculations in GHS.

New Commands

Introduction

SurfaceWorks 6.0 is organized to process a limited set of commands, which can be entered through the dialog box under Tools/ Command Window.

In the Command Window, you can type in a command; for example: file.open transl.ms2. Issue the command by pressing <Enter> or clicking the "Ok" button. Commands generally execute "silently", i.e., SurfaceWorks does not put up the usual dialog boxes or message boxes which occur when the same commands are issued from the menus.

One principal purpose of the Command Window is to provide access to new and experimental program features that have not yet been provided with a graphical user interface. There are also some undocumented commands, which we use during development to test various aspects of program operation.

The command window has a dropdown that allows ready access to the 32 most recently used commands. Selecting a command from the MRU list puts it in the command window, where it can be edited, or just executed. You still have to click "Ok" or press <Enter> to execute the command.

Filenames and pathnames

Many commands have filenames for arguments. You can use either forward slashes '/' or backslashes '\' to separate folders and filenames. If a path has any spaces in it, you must use double quotes around the entire path; for example:

```
file.open "2002 projects\65 cat\65cat.ms2"
```

The SetPath command is a convenient way to direct commands to particular folders. For example, following the command

```
setpath "2002 projects\65 cat"
```

you can open the above specified model file with

```
File.open 65cat
```

Command Message File (%cmdmsg.txt)

If a valid path has been registered through a SetPath command, the results of any command -- that is, the text contents of any message box displayed on completion of the command -- are written out to a file named "%cmdmsg.txt", in the folder specified in SetPath. This is useful if you want to transfer the results to a document, or print a copy.

Documented Commands

Note: There may be more commands than are shown in the list below. Use the Help command to see the full list, and to learn the syntax for the command you are interested in.
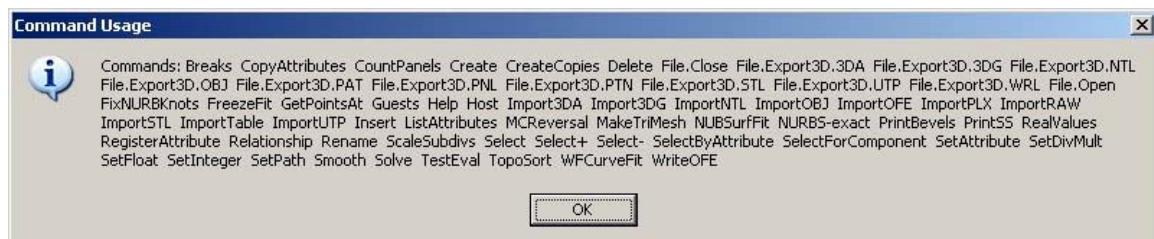
The periods and spaces indicated are required. The command is broken into "tokens" by spaces. If the spaces are missing, the command cannot be read correctly. Tabs can't be substituted for spaces. The command keywords are *not* case sensitive; some tokens that follow the keywords are case sensitive, such as entity names.

Breaks

with one curve, snake or surface selected.

Display an output dialog and prints a file showing breakpoint data (degrees and locations) for a curve or snake, or breakline data for a surface.

Countpanels

CountPanels
Counts the panels on visible surfaces.

Create

Given a complete syntax string, e.g.

```
Create BCurve MC2  11  1 10x1 / * 2 { P21 P22 P23 P24 } ;
```

this command creates a new entity from the given string, just as if the *string* had been read from a model file. The syntax has to be correct, and any supporting entities must already be present, otherwise the create operation is aborted.

CreateCopies

See On-line Help for full documentation plus a tutorial. Insert/Copies was added to the menu this release rendering this command obsolete for normal modeling.

Delete

Deletes all entities in the selection set, if such deletion is legal; i.e., if no entities in the selection set have dependents outside the selection set. If it can't delete the entire selection set, it does nothing. If "LastEdited" entity is in the selection set, this pointer is set to NULL. "LastSelected" pointer is set to NULL.

File.Close

Closes the current model, just like File/Close on the menu.

File.Export3D.3DA

File.Export3D.3DA [filename [.ext]]

Exports a 3DA wireframe file, just like File/Export3D/3DA, but without dialogs. If the selection set is empty, the file will have include all visible entities; otherwise, only the selected entities are output. Symmetry images can be turned on or off with the SetSymmetry command. If filename.ext is absent, SurfaceWorks will use the .MS2 filename, with .3DA extension. If no extension is given, SurfaceWorks will use .3DA.

File.Export3D.3DG

File.Export3D.3DG [filename [.ext]]

Exports a 3DG FormZ triangle mesh file. If the selection set is empty, the file will include all visible surfaces and triangle meshes; otherwise, only the selected entities are output. Symmetry images can be turned on or off with the SetSymmetry command. If filename.ext is absent, SurfaceWorks will use the .MS2 filename, with .3DG extension. If no extension is given, SurfaceWorks will use .3DG.
\

File.Export3D.NTL

File.Export3D.NTL [filename[.ext] [tol]]

Exports NTL (Patran Neutral File) file of the selection set (or all visible entities, if nothing is selected). The default extension is .NTL. The default tolerance is .0005 times the model size

File.Export3D.OBJ

File.Export3D.OBJ [filename [.ext] [tol]]

Exports an Alias Wavefront OBJ file. If the selection set is empty, the file will include all visible entities; otherwise, only the selected entities are output. Symmetry images can be turned on or off with the SetSymmetry command. If filename.ext is absent, SurfaceWorks will use the .MS2 filename, with .OBJ extension. If no extension is given, SurfaceWorks will use .OBJ. *tol* is the tolerance; default *tol* is .0005 * size.

File.Export3D.PAT

File.Export3D.PAT [filename [.ext]]

Exports a PAT patch file, just like File/Export3D/PAT, but without dialogs. If the selection set is empty, the file will include all visible surfaces; otherwise, only the selected entities are output. Symmetry images can be turned on or off with the SetSymmetry command. If filename.ext is absent, SurfaceWorks will use the .MS2 filename, with .PAT extension. If no extension is given, SurfaceWorks will use .PAT.

File.Export3D.PNL

File.Export3D.PNL [filename [.ext]]

Exports a .PNL file. If the selection set is empty, the file will include all visible surfaces; otherwise, only the selected entities are output. Symmetry images can be turned on or off with the SetSymmetry command. If filename.ext is absent, SurfaceWorks will use the .MS2 filename, with .PNL extension. If no extension is given, SurfaceWorks will use .PNL.

File.Export3D.PTN

File.Export3D.PTN [filename [.ext]]

Exports a PTN patch file. If the selection set is empty, the file will include all visible surfaces; otherwise, only the selected entities are output. Symmetry images can be turned on or off with the SetSymmetry command. If filename.ext is absent, SurfaceWorks will use the .MS2 filename, with .PTN extension. If no extension is given, SurfaceWorks will use .PTN.

File.Export3D.STL

File.Export3D.STL [filename[.ext] [ascii]]

Saves an STL (stereolithography) file.

If anything is selected, uses the Selection Set; else all visible surfaces and triangle meshes.

Use ascii = 1 to save in ASCII format; default is binary.

File.Export3D.UTP

File.Export3D.UTP [filename[.ext]]

Saves a .UTP (triangular panel) file.

If anything is selected, uses the Selection Set; else all visible surfaces and triangle meshes.

File.Export3D.WRL

File.Export3D.WRL [filename[.ext] [kind]]

Saves a VRML (Virtual Reality Modeling Language) file

If anything is selected, uses the Selection Set; else all visible surfaces and triangle meshes.

 kind = 1 or 2 for VRML 1.0 or 2.0 format. (Default = 2.)
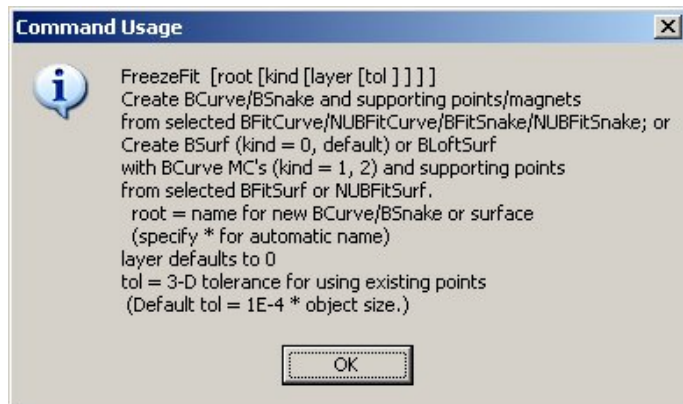
File.Open

File.Open filename [.ext]

Opens a model file. The model must be in the directory specified by a previous SetPath command, or a fully qualified path must be supplied in the command. If no extension is specified, .MS2 is assumed.

FixNurbsKnots

FixNURBKnots [places]
Makes minimal adjustment to knot spacing of NURBS surface,
Light NURBSsurface, NURBS Curve or NURBS Snake
to allow evaluation without error.

FreezeFit


```
Command Usage
FreezeFit  [root [kind [layer [tol ] ] ] ]
Create BCurve/BSnake and supporting points/magnets
from selected BFitCurve/NUBFitCurve/BFitSnake/NUBFitSnake; or
Create BSurf (kind = 0, default) or BLoftSurf
with BCurve MC's (kind = 1, 2) and supporting points
from selected BFitSurf or NUBFitSurf.
  root = name for new BCurve/BSnake or surface
   (specify * for automatic name)
layer defaults to 0
tol = 3-D tolerance for using existing points
   (Default tol = 1E-4 * object size.)

                    OK
```

[Available when the selection set contains exactly one object from the following list of entities:

BFitCurve	NUBFitCurve
BFitSnake	NUBFitSnake
BFitSurf	NUBFitSurf

The common feature of these entities is that they all embody curve or surface fitting operations, using either uniform or non-uniform B-spline math.]

This command accesses the curve-fit or surface-fit results from the fitted entity, creating free-standing geometry (consisting of NURBS curves, snakes and/or surfaces, and the point entities needed to support them) that duplicates the fitted curve or surface. The term "freeze" implies that a support relationship is broken: the "frozen" curve or surface and its generated parents do not depend on the fitted curve or surface or their parents.

Applications of Freeze Fit

Simplifying a "heavy" NURBS surface

Frequently surfaces that arrive in IGES files have a totally obscene number of control points. This makes them slow to load and edit, and tedious or impossible to modify. Frequently you can obtain a very close BFit or NUBFit approximation with a modest number of control points, and then Freeze Fit to replace the heavy surface with a much lighter and more easily editable one.

Preparing geometry for IGES export

BFit and NUBFit approximations prior to IGES export allow management of the approximations on a surface-by surface basis. Freeze Fit can be part of this process. However, note it is usually just as effective to export the BFit or NUBFit surface directly – you get exactly the same results in the IGES file.

Access to NUBFit knots

The knot lists used in a NUBFit entity are stored internally and not directly accessible. Freeze Fit will turn them into Knot List entities, which can be used in NUBFitting adjacent surfaces to achieve accurate joins in the exported IGES. If all you want from a Freeze Fit operation is the Knot Lists, you can easily select and delete the other generated entities.

Replacing an expensive intersection or procedural object

IntSnakes generated by surface-surface intersections, and Procedural objects involving intersections and projections, are computationally intensive and may make your model slow and unresponsive. They are also vulnerable to radical failures caused by possibly small geometry changes. When the supporting geometry is in a final configuration, consider replacing that expensive snake or surface with a light, robust BSnake or NURBSurf, generated by fitting followed by Freeze Fit.

Breaking a connection between parts of a model

Suppose you have a complex hull and want to build a deck and superstructure on it. The hull is already under construction and isn't going to change. A natural first step is to make a snake on the hull for the deck edge, then everything beyond depends on that snake. But that means the whole complex hull model has to be part of your superstructure model. If you fit and freeze that snake as a curve, then build onward from the curve, you can isolated the superstructure in a separate model.

Breaking dependence on external files

Tabulated Curves and Tabulated Surfaces supported by 3DA files are a good way to import curve and surface data, for example from a DXF file (via DXF2MSF utility). But if you build directly on the TabCurves and TabSurfs, your model will always depend on those files. If you fit and freeze the TabCurves and TabSurfs, you can delete them from your model and build on the standalone curves and surfaces that result. Moreover, you will be able to edit and modify the "frozen" entities, whereas the Tabulated entities can only be modified by changing the data in their files.

GetPointsAt

GetPointsAt [mask [layer layer …]] (with one point selected)

                         List points at the location of the selected point.
                         Mask is a string of subclass chars, e.g. "pm"
                         * admits all points.

Guests

Displays a message box which shows all the guests of the selected entity (if any). ("Guests" is the inverse relationship of 'host'.)

Help, or Help *command*

With no argument, displays a dialog box listing all the documented commands, in alphabetical order.

Help with a *command* argument (e.g. "Help File.Open") displays a dialog box that explains the syntax for the specified *command*.

Host

Displays a message box which identifies the host(s) of the selected entitity.

Import3DA

Import 3DA filename[.ext] [kind [layers]]

      Imports 3DA file into points and/or curves.
      Kind = 0, points only
      Kind = 1, Type-1 BCurves
      Kind = 2, Type-3 CCurves

Import3DG

Import3DG [filename[.ext]]

Imports a FormZ 3DG triangle file, creating one or more Triangle Mesh entities from the data.

ImportNTL

ImportNTL filename[.ext] [kind [layer]]

Imports geometric data from a Patran Neutral file into the current model. kind is inactive.

ImportOBJ

ImportOBJ [filename[.ext]]
Imports triangular facets from an Alias WaveFront OBJ file,
creating one or more Triangle Mesh entities from the data.

ImportOFE

ImportOFE filename[.ext] [kind [layer]]

Imports .OFE offsets file as type-1 BCurves.
kind = 0, all on one layer (default);
kind = 1, each station on a separate layer.
default layer = 0.

With this command, in conjuntion with WriteOFE, SurfaceWorks can be used as an OFE Editor.

ImportPLX

ImportPLX [filename[.ext]]

Imports a PLX (AeroHydro plate expansion) file, creating one or more Triangle Mesh entities from the data.

ImportRAW

ImportRAW [filename[.ext]]

Imports a POV-ray raw triangle file, creating one or more Triangle Mesh entities from the data.

ImportSTL

ImportSTL [filename[.ext]]

Imports an STL (stereolithography) file, creating one or more Triangle Mesh entities from the data

ImportTable

ImportTable filename[.ext] [kind [layer]]
Imports table file into points and/or curves. If no extension is given, .TXT is used.
Kind = 0, points only
Kind = 1, Type-1 BCurves
Kind = 2, Type-3 CCurves

ImportUTP

ImportUTP [filename[.ext]]

Imports a UTP (unordered triangular panels) file, creating one or more Triangle Mesh entities from the data.

Insert

Insert type_name

Creates a new entity of the type specified by type_name. (type-name is NOT case sensitive.)

Selection set can contain preselected parents.

MCReversal

MCReversal [on|off]
Changes the way surfaces are created in order to correct the twist caused by conflicting orientation of supporting curves. It does not actually change the orientation of supporting entities. If no parameter is supplied, "on" is assumed; the system default is also "on". Turning the command off allows the creation of a twisted surface in the event that this is what the user had in mind.

NOTE: Be aware that if you use this command to repair a twisted surface created with MCReversal turned off, the surface will not be corrected when the command is entered and MCReversal turned on. In order to correct the surface you must edit the attributes for the curve.

MakeTriMesh

MakeTriMesh [light]

With 1 surface OR with at least 3 lines selected.

Makes a Triangle Mesh

(light = 0, default) or a Light Triangle Mesh (when light = 1).

NUBSurfFit

Fully documented in On-line Help. See "Commands"

PrintBevels

PrintBevels [filename[.ext] [ds]]

With 1 surface plus 1 or more X=constant contour entities selected.

Writes a file of bevel angles with respect to X-direction.
default extension = .TXT
ds = arc length interval between points (default 1.)

PrintSS

PrintSS [filename[.ext]]

Prints the names of the selected entities to a file.

RealValues

RealValues command reveals the current values and unit dimensions of real-valued entities (Variables and Formulas) in your model.

Relationship

The Relationship command reveals the dependency relationship (if any) between two selected entities. It is useful for tracking dependencies in a complex model, especially one that you're unfamiliar with, or have forgotten the details of.

Example: Open Demo.ms2; select XContours 'stations' and AbsPoint 'P22'. Window/ Command and issue the command:

Relationship

(This command has no parameters or modifiers.) You get a message box with this information:

> XContours stations is a 3rd generation descendant of AbsPoint P22
>
> AbsPoint P22
>
> BCurve MC2
>
> CLoftSurf hull
>
> XContours stations
>
> File saved: c:\%Relation.TXT

If you look up the referenced file, you'll find it contains other useful information to help you find and identify the intermediate entities:

```
          DEMO   23-Jun-2004  17:08:01
          color vis layer  name
            14    1    0   AbsPoint P22
            11    1    0   BCurve MC2
            10    1    0   CLoftSurf hull
            12    1    0   XContours stations
```
Note that the path between two entities is not necessarily unique (as it is in this example). Relationship displays the first path it finds connecting the two entities.

Rename

Rename [*root* [*base*]]

Renames the selection set. *root* = base name; use * for automatic root. *base* = starting number.

For example, with four points selected, the command

RenameEntities P 5

will attempt to rename the points as 'P5', 'P6', 'P7', 'P8'. If any of these names are unavailable (i.e, already in use by an entity that is not currently selected), no names are changed.

ScaleSubDivs

ScaleSubDivs [scale factor]

Multiplies the sub-divisions of curves and surfaces, in the model, by the scale factor. This command is commonly used to lessen the sub-divisions in a model to improve performance.

Select

Select [name name ....]

This clears the working selection set and selects the named entity(s).

Select+

Select+ [name name ....]

This adds the named entity(s) to the working selection set, without clearing the set.

Select-

Select- [name name ....]

This removes the named entity(s) from the selection set.

SelectForComponent

SelectForComponent is a convenience for collecting together the set of entities to be included in a component (MC2) file. This is especially convenient when the component is a complex one with many entities, or when you are refining the component and need to repeatedly save revised and improved versions.

To use the command, first make two EntitityLists:

(1) a list of the supports needed for the component (we'll call this EntitityList 'supports') and

(2) a list of "products" -- the essential entities that the component needs to produce (we'll call this EntitityList 'products').

Then select 'supports' and 'products', in this order (the order is important), Window/Command and give the command:

SelectForComponents

(This command has no parameters or modifiers.) Following the command, the selection set should contain exactly the set of entities that are either products or ancestors of products *and* are descendants of the "supports". Then you're all set for File/Component/Save.

SetDivMult

SetDivMult [*value*]

Sets the divisions multiplier for the current model. *value* is screened to be > 1 and < MAX_DIVMULT; no change is made if *value* is out of bounds. If *value* is omitted, divmult is set to 1.

SetFloat

SetFloat entity index value [0]

'entity' is an entity name. This command sets a new value for one floating point number in the entity data, the one denoted by index. If the optional 0 is appended to the command, no update is performed following the data change. Otherwise, the entity and all its descendants (to any generation) are updated, and the change is registered with the Undo system.

You can figure out how many floating point numbers an entity has in its data, and the order in which they occur, by looking up its Entity Specification. For example, a Frame Point has three floats: its X, Y and Z coordinates, in that order. To set Z of P11 to 3.85, you would give the command SetFloat P11 3 3.85. A RevSurf has two floats: angle1 and angle2, in that order. To set angle1 of RevSurf body to -90, you would give the command SetFloat body 1 -90.

SetInteger

SetInteger entity index value [0]

'entitity' is an entity name. This command sets a new value for one integer in the entity data, the one denoted by index. If the optional 0 is appended to the command, no update is performed following the data change. Otherwise, the entity and all its dependents (to any generation), the model is updated, and the change is registered with the Undo system.

You can figure out how many integers an entity has in its data, and the order in which they occur, by looking up its Entity Specification. Note that:

the integers in all curves and snakes start with 2 integers for t-divs and t-subd;

the integers in all surfaces start with 5 integers for u-divs, u-subd, v-divs, v-subd and orientation.

the integers in all solids start with 7 integers for u-divs, u-subd, v-divs, v-subd, w-divs, w-subd, and orientation.

For example, a BCurve has three integers: t-divs, t-subd, and type, in that order. To set t-subd of MC1 to 4, you would give the command:

SetInteger MC1 2 4

A CLoftSurf has six integers: u-divs, u-subd, v-divs, v-subd, orientation and type. To set type of CLoftSurf hull to 3, you would give the command:

SetInteger hull 6 3.

SetPath

SetPath *path*

This establishes a directory for subsequent File commands. If you don't set a path, SurfaceWorks will use the root directory.

SetSymmetry

SetSymmetry [on | off]

Sets symmetry images on/off, for export commands.

If no argument, sets symmetry ON.

SmoothWireframe

SmoothWireframe **[1|0]**

Toggles Smooth Wireframe on and off.

Solve

Solves an N-Degrees Of Freedom geometric problem stated by the Selection Set.

SelSet: free points with N Degrees of Freedom, plus N [point, plane/surface] pairs.

TestEval

TestEval [N]

Re-evaluates the selected entity N times and reports the time taken.

Toposort

Sorts the model into a top-down or "topological" order, in which any parent is listed before any of its childen. This makes a model file much easier to read. (Theoretically, model evaluation is independent of the order the entities are listed in, but there have been occasional bugs that have broken this rule.)

WFCurveFit

WFCurveFit [nv [type]]

Fits Bcurves to polylines of selected WireFrame entitity.

Defaults: nv=6; type = 3

WriteOFE

WriteOFE filename[.ext]

With a set of transverse type-1 B-Spline Curves selected.

Writes an offset file; default extension is .OFE.

With this command, in conjuntion with ImportOFE, SurfaceWorks can be used as an OFE Editor.

Layer Dialog Improvements

The following features have been added to the SurfaceWorks layer dialog:

1   The Current Layer's name is displayed alongside its number, at the top of the dialog.

2   "Current" button. You can choose a new layer for the current layer by highlighting its number/ name in the list box and clicking the "Current" button.

3   To insert one or more layers, use the "Move Down" button would be used. You can open space for a new layer by highlighting a layer (say, layer 5) and clicking the "Move Down" button. The contents and name of layer 5 are moved down one layer; if layer 6 is already occupied, its contents and name are moved to layer 7 etc, until we reach an empty layer, where the chain reaction stops. Thus, an empty layer is created at layer 5.   If no insert can be done, the Insert Layer button is deactivated.

With a layer selected, and the cursor held over the "Move Down" button, arrows appear in the layer list indicating how many layers will be affected by the move. Multiple layers can be selected to be moved. In this way a block which may contain blank layer separators, can be moved as a group.

The "Move Up" button operates in the same manner as "Move Down", but in the opposite direction.

4   "Move Layer…" button. You can move the contents and/or name of any selected, non-empty, layer to another selected layer. This button opens a dialog where you set check boxes that determine whether you move the layer name and the layer on/off state with the layer contents. When a layer is empty and cannot, therefore, be moved the Move Layer button is deactivated.

5   "Swap" The contents and names of two selected layers can be swapped with each other.

Layer Settings dialog and Undo

As long as you stay in the Layer Settings dialog, individual "moves" and "inserts" are not recognized as undoable events. If you leave the dialog by the "Cancel" button, all changes made while in the dialog are undone, and the model is exactly the same as when you entered the dialog. Leaving the dialog by accepting (clicking OK) registers as one undoable event, covering all the changes made.

Point Dragging and Nudging

The Point entity has always had a *dragging constraint or type*, 0 to 7, but it has not previously had any significance. In SurfaceWorks 6.0, the type now has an important role: it controls *constraints* applied to the point during dragging and nudging. Only when a point uses Cartesian coordinates is the dragging constraint available. If a point employs Polar coordinates, the dragging constraint is not available. The 8 FramePoint types are as follows:

Type	Dragging/nudging behavior
0	Free to drag in any direction
1	x is pinned; the point is draggable in y and z
2	y is pinned; the point is draggable in x and z
3	x and y are pinned; the point is draggable only in z
4	z is pinned; the point is draggable in x and y
5	x and z are pinned; the point is draggable only in y
6	y and z are pinned; the point is draggable only in x
7	x, y and z are pinned; the point is not draggable

The order of the types in the above table may look a bit odd, but the logic is simple, and easy to remember with the following recipe:

Start with *type* = 0; if x is pinned, add 1; if y is pinned, add 2; if z is pinned, add 4.

"Pinned" means that the corresponding frame coordinate x, y, or z will not change during dragging and nudging. FramePoints of types 1, 2, and 4 are constrained to planes normal to the x, y and z frame axes respectively. FramePoints of types 6, 5, and 3 are constrained to lines parallel to the x, y and z frame axes respectively.

The "pinned" coordinates can still be edited through the Property Manager; this editing mode is not restricted by type.

As an example application of this feature, consider a boat hull design consisting of a lofted surface supported by a number of master curves in transverse planes. If you use FramePoints of type-1 (x pinned) to support these transverse master curves (in a frame parallel to the global coordinates, e.g.'*'), these control points can be dragged or nudged in any oblique view, and will remain in their respective transverse planes. If the forward master curve lies in the centerplane, you could use type-2 FramePoints (y pinned) to allow dragging in any view without moving it out of the centerplane. The last (centerplane) control point on each transverse master curve could be type-3 (x and y pinned), to keep it both in the centerplane, and in its correct transverse plane.

More Entities

There are many more RGKernel Entities than are on the menu and fully documented. The Insert Command, documented above, gives you access to quite a few more tools. For a number of reasons thay have been left off the menu, ranging from obsure usage to lack of robustness. Many entities are familiar to long time SurfaceWorks users and we want to make sure they are accessible to them and any other advanced users. A list and short description follows. The name in **bold** is the RGKernel name which should be used in the command syntax. Example: Insert rbfillet

Breakbead   A bead located at a breakpoint of a curve.

Breakring   A ring located at a breakpoint of a snake.

Centerpt   Center of curvature of any curve at the location of the parent bead.

ContourCurv   A curve fitted to a given index of contour.

IntSnake2   Alternate intersection method. Should be used when menu driven Intersection Snake give poor results.

PolyCurve   The original version of PolyCurve and required by some users for certain applications. The biggest difference is the ability to set the location of the ending t-values of the various segments of the PolyCurve.

PolySnake   Snake version of the PolyCurve explained above.

PolySurf   Surface version of the PolyCurve explained above. This is an entity which can cause more problems than it solves. The biggest thing to look for is the introduction of breaklines into the model which could have detrimental effects further down the dependancy tree.

RBFillet   A rolling ball fillet between 2 surfaces. 2 Magnets or Surfaces, a curve and a radius value are needed for parents.

TMTrim1   Trimmed TriMesh

Porcupine Curvature Display
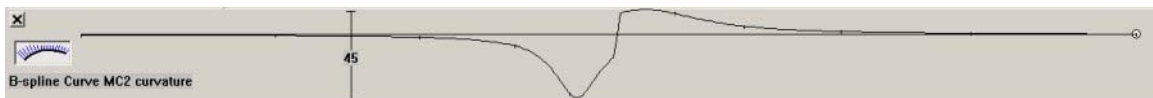
 Clicking on the button to the left (seen above in the Curvature Profile view) will enable an alternate method of displaying a curves curvature profile. This is commonly called Porcupine Curvature Display. The spines of the curve represent the amount and direction of curvature. See below for an example from the model view.

Another new feature in this view is the ability to display the curvature profile of Snakes. Previously only curves were eligible.