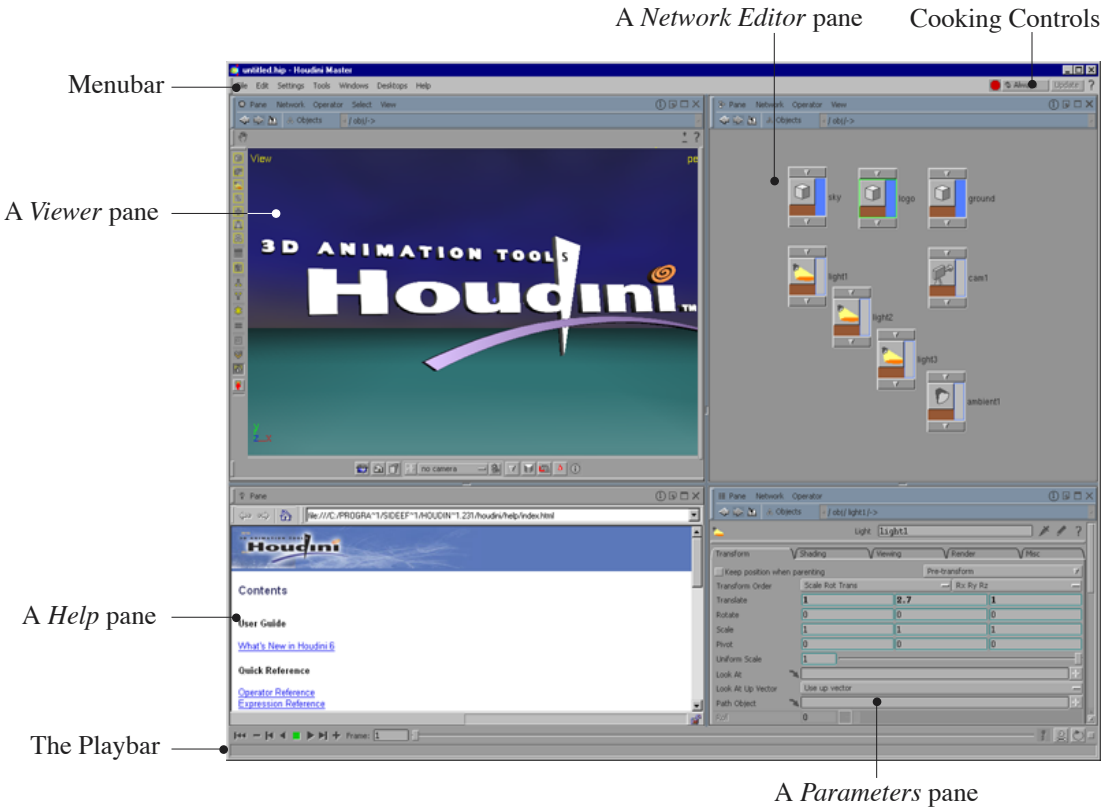
I Interface

I INTRODUCTION

PANES AND DESKS

Houdini's interface is comprised of a number of configurable regions, or *Panes*. Each pane provides access to editors which allow you to view and change different parts of Houdini's database. Panes can be reconfigured and resized, providing you with a completely customisable interface. Below is a sample of one way in which the panes in Houdini's interface can be configured:

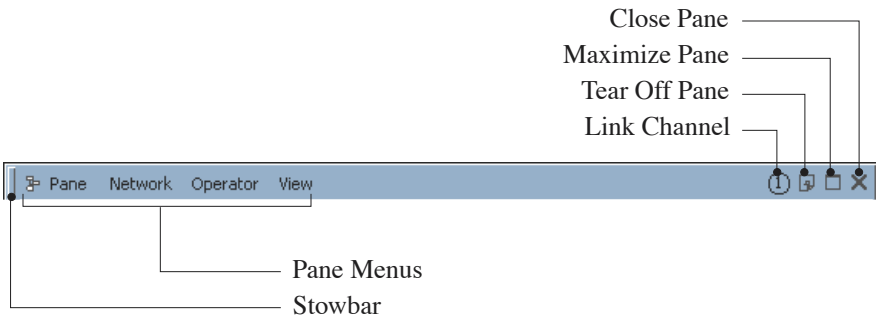


In this example there are four panes; they are setup to be:

- A *Viewer* – for displaying a 3D view of your animation
- A *Network* Editor – for displaying the logical organisation of your project
- A *Channel* Editor – for tweaking animation channels
- A *Parameters* display – for changing parameters of things in your animation

The way all the panes are setup is called a *Desk*. You can have multiple desks, each with a different setup of panes. You switch between different desks by selecting from the *Desks* menu.

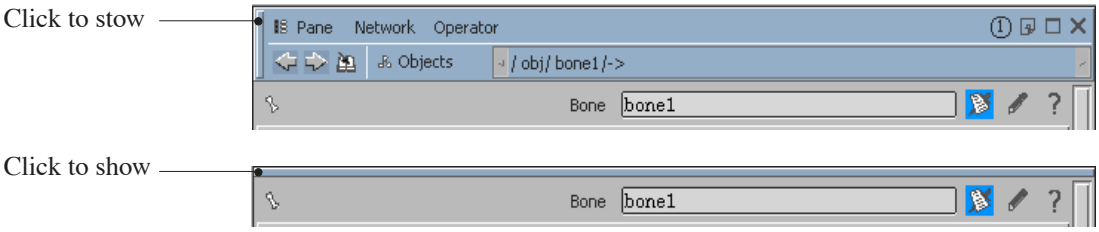
2 PANE CONTROLS



Along the top of each pane are a set of *Pane Controls*. They allow you to configure the content of a pane as suits your needs.

2.1 PANE CONTROLS

STOWBAR

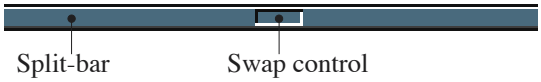


Stowbars allow you to collapse a toolbar down to a single line. This conserves screen real-estate, and gets rid of the clutter of seldom used controls. These are used throughout the Houdini interface.

If you don't see a Toolbar as mentioned somewhere, it is probably just stowed. Look for a Stowbar where you think the toolbar should be, and unstow it. If you have Help tags enabled (*Help* menu > *Help Tags*), pausing the cursor over a stowed item will reveal its name.

ADJUSTING PANE SIZE

The amount of space each Pane occupies relative to others is adjustable. To alter their proportions, position your cursor over the Split-bar (located between any two adjacent Panes) and drag your cursor to adjust the size of the pane.



swap control

Using the Split-bar you can also swap any two adjacent panes by clicking on the Swap control located at the centre of any Split-bar. After clicking, the two Panes will exchange locations.

LINK CHANNEL

It is often desirable to have the contents of one pane follow another. For example, if one pane is set to be a *Network Editor*, and you select a different OP, then you want another pane which is set to display *Parameters* to change also – following the change in the *Network Editor* pane. This is possible by linking the panes to a common channel. If you set one pane to Channel *2*, and the other also to Channel *2*, then they will synchronise with each other as you work within one or the other pane. Very useful. As long as any pane is set to the same channel, it will follow changes in any other pane set to the same channel.

You have up to three channels to which you can set your panes to. If you don't want your panes to be changing in sync with activities going on in another pane, simply set the channel for that pane to: *No Link*.

COPY / TEAR OFF PANE

Creates a copy of the current pane in a new separate window that floats overtop of the Houdini window. This floating pane contains the full functionality of the pane it was copied from, and the original pane is left in the Houdini window as it was.

Clicking with the Alt key on this button removes the pane from the Houdini window, and moves it entirely into the floating window. The space which the pane occupied in the Houdini window is freed up as if you had selected to close that pane.

MAXIMIZE PANE

Zooms the current pane to take over the entire Houdini window (eliminating the display of all other panes). This gives you a quick full-screen mode for a pane. Clicking again toggles the pane back down to its former size, and restores the other panes as they were.

CLOSE PANE

Removes the pane from the desk, freeing-up the space it occupied for other panes.

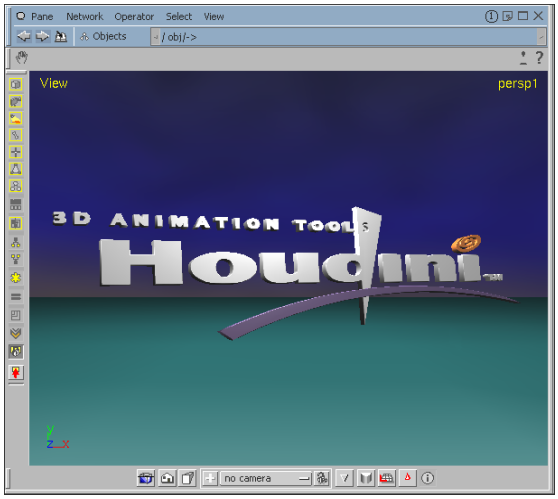
3 PANE MENUS

3.1 PANE MENU

The *Pane* and *Network* menus allow you to specify the behaviour of a pane. They allow you to set the function a pane performs by switching between the different types of Editors that can exist within a pane. They set the function of a pane and the context in which they will operate.

Depending on which sort of *Network*, a *Pane* may provide a different display. For example, the Viewer for 3D geometry Objects will be different than a Viewer for 2D COP images.

VIEWER

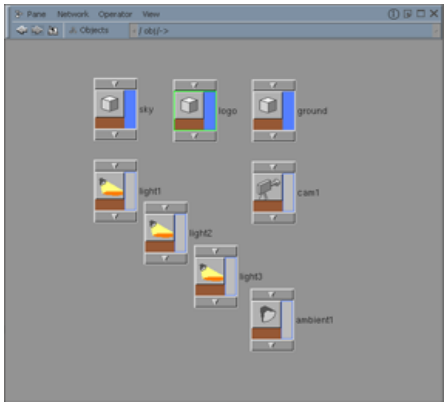


Viewers allow you to view and sometimes manipulate the data which you have created. For example, in a typical 3D Viewport, you can view the geometry in your scene through a camera, and even directly manipulate that geometry within the Viewport. The kind of Viewer you see depends on the context. For example, if you're viewing 2D images, a 2D Viewport is presented instead. The kinds of Viewers available are:

- *3D-Viewers* p. 110
- *2D-Viewers* p. 156
- *Shader Viewer* p. 158
- *CHOP Viewer* p. 160

Setting a pane's *Editor* menu to *Viewer* displays the resulting view of a given object or OP as it appears within a scene. The type of viewer is context sensitive to the type of data being displayed. If the data is 3D, then a 3D Viewport is displayed; if the data is a 2D image, then a 2D Viewer is displayed; if the data is a channel or graph, then the waveform is displayed.

NETWORK EDITOR

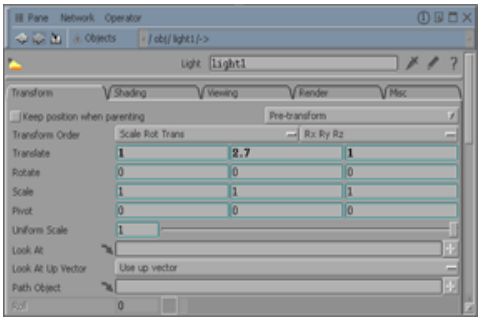


Setting a pane to *Network Editor* displays a view of the objects at the current level of the scene hierarchy as tiles or as a list. If you happen to be in objects, then the Objects in your scene are displayed; if you happen to be in CHOPs, then the CHOP Operators are displayed; if you happen to be in COPs, then the COP tiles that make up the procedural flow of 2D graphics is displayed.

This view allows you to access and change the parametric flow of data and relationships between individual components that make up your scene.

See *Network Editor* p. 167 for more information.

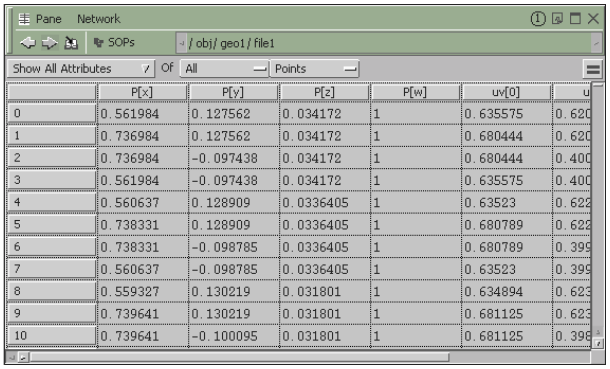
PARAMETERS



Closely related to the Network view, this Editor allows you to change individual parametric values of OPs displayed in a Network View.

See *Parameters* p. 208 for more information.

GEOMETRY SPREADSHEET

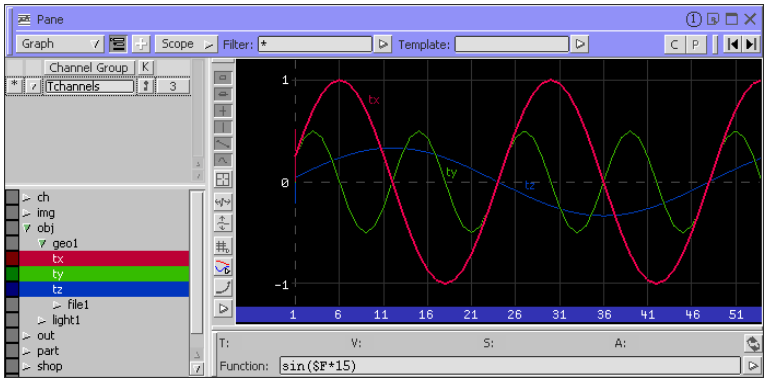


	P[x]	P[y]	P[z]	P[w]	uv[0]	u
0	0.561984	0.127562	0.034172	1	0.635575	0.620
1	0.736984	0.127562	0.034172	1	0.680444	0.620
2	0.736984	-0.097438	0.034172	1	0.680444	0.400
3	0.561984	-0.097438	0.034172	1	0.635575	0.400
4	0.560637	0.128909	0.0336405	1	0.63523	0.622
5	0.738331	0.128909	0.0336405	1	0.680789	0.622
6	0.738331	-0.098785	0.0336405	1	0.680789	0.399
7	0.560637	-0.098785	0.0336405	1	0.63523	0.399
8	0.559327	0.130219	0.031801	1	0.634894	0.623
9	0.739641	0.130219	0.031801	1	0.681125	0.623
10	0.739641	-0.100095	0.031801	1	0.681125	0.396

This editor is specific to geometry types. It allows you to access and change the individual values and attributes of points and vertices within the geometry as absolute rather than parametric values generated by the OPs procedurally.

See *Geometry Spreadsheet* p. 221 for more information.

CHANNEL EDITOR

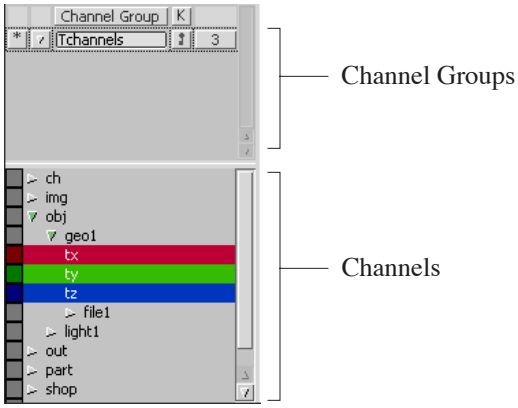


The Channel editor displays channel data for parameters over time graphically. It also shows segment functions which mathematically interpolate between keyframe values, and lets you adjust these values.

See *Channel Editor* p. 223 for more information.

Note: This should not be confused with the CHOP Channel Editor which is displayed when you set the *Editor* menu to: *Viewer > CHOP!* The main difference being that the CHOP editor edits raw waveform sample data (bits), whereas the Channel Editor works with keys and segment interpolation functions between them (vectors).

CHANNEL LIST



The Channel List is integrally tied to the Channel Editor. By selecting it as a pane type, you can just the channels without the graph display. It displays a listing of all currently active channels for parameters in the Houdini OP hierarchy. By clicking on any of the channels in the list, you set the *Scope*. Houdini will display the channels specified in the scope in a linked Channel Editor pane.

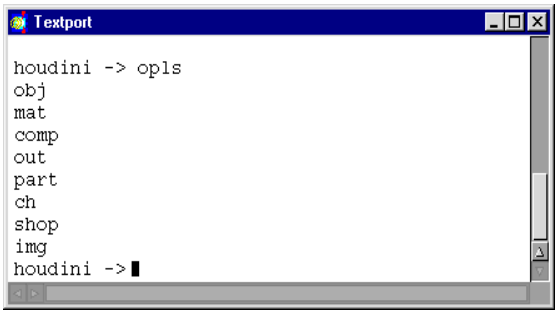
See *Channel List* p. 244 for more information.

CUSTOM PANEL

The Custom Panel allows you to setup a pane with custom wired controls linked any parameter within Houdini. You can layout your own buttons, edit fields, popup menus, and completely create your own parameters dialogue. Once this is done, you can wire your controls to any specific Houdini parameter in an OP. For example, above, a custom panel has been created where a slider (“slider 2”) is wired to control the Translate-X parameter of a *logo* object.

See *Custom Panel* p. 248 for more information.

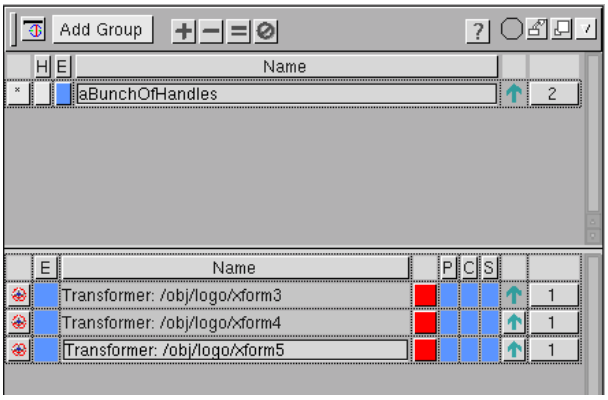
TEXTPORT



Displays a Textport for direct entry of scripting commands. Similar to a UNIX shell, but for entering *hscript* commands to Houdini.

See *Textport* p. 250 and the *12-Scripting* section for more information.

HANDLE LIST



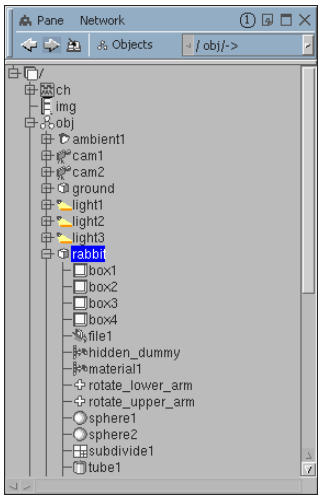
Displays a list of Handles which have been made persistent in the Viewport. A Handle is a device which allows you to tweak and edit geometry during a Modelling Operation in the Viewport.

making handles persistent

While you are modelling (say, with an Edit or Transform OP), you can use the pop-up menu on a Handle to select an option called *Persistent*. This makes the Handle stay there, even while you go on to do other things. The *Handle List* allows you to view and manage a list of these Persistent handles.

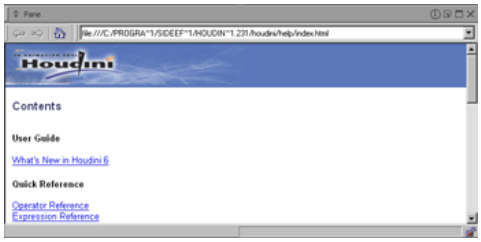
See *Geometry > Handles* p. 408 for more info on Handles.

TREE VIEW



Provides a traditional tree view of the object hierarchy.

HELP BROWSER



For online HTML help, select a *Help Browser*. It allows you to explore Houdini's wide range of online built-in help.

LINKING

Link Channel
: You have up to three channels to which you can set your panes to. Panes set to the same channel will change in sync with activities going on in another pane

Tie to This Network's Current Node
: If selected, changes (such as scolling or zooming) in any pane will be reflected automatically in this pane as well.

Tie to Parent Network's Current Node
: Enable this menu item to have this pane follow changes in the parent network.

TOOLBARS AND CONTROLS

Toggles for the display of the *Network Controls* and the *Operator Toolbar*.

EDIT TOOLBAR

Brings up the Toolbar Manager dialog, in which you can add custom groups of Operators to a Toolbar for each sort of viewed OP type.

MOVE PLAYBAR HERE / RESET PLAYBAR Alt P

Moves the Playbar to be located within the bottom area of this pane.

If the Playbar is already within the pane, an additional option appears – reset the Playbar back to the bottom of the Houdini window by selecting *Reset Playbar*.

COPY PANE

Creates a copy of the current pane in a new separate window that floats overtop of the Houdini window. This floating pane contains the full functionality of the pane it was copied from, and the original pane is left in the Houdini window as it was.

TEAR OFF PANE

Same as *Copy Pane*, except it removes the pane from the Houdini window, and moves it entirely into the floating window. The space which the pane occupied in the Houdini window is freed up as if you had selected to close that pane.

MAXIMIZE

Zooms the current pane to take over the entire Houdini window (eliminating the display of all other panes). This gives you a quick full-screen mode for a pane. Clicking again toggles the pane back down to its former size, and restores the other panes as they were.

SPLIT LEFT/RIGHT - TOP/BOTTOM Alt [/ Alt]

Splits the pane into two new separate panes - vertically or horizontally.

CLOSE PANE Alt /

Removes the pane from the desk, freeing-up the space it occupied for other panes.

3.2 NETWORK MENU

OBJECTS (OBJ)

Objects allow you to set up the objects in your scene in relation to each other – geometry, lights, cameras, microphones, fog, and the parenting relationships between them. For information, see the *03-Objects* section.

GEOMETRY (SOP)

When you create geometry in the Viewport using *OPerations*, Houdini will build and maintain an underlying set of Surface-OPerators (i.e. SOPs). This allows you to manually access and define these networks of SOPs and their interconnections which procedurally define geometry. For information, see the *04-Geometry* section.

PARTICLES (POP)

This icon allows you to edit POPs. These allow you to create procedural networks to control Particle behaviour which is birthed directly from existing geometry. For information, see the *05-Particles* section.

SHADERS (SHOP)

Allows you to work with SHader OPerators. These provide you with a versatile RenderMan™ like language for shading geometry. For information, see the *07-SHOPs* section.

VEX BUILDER (VOP)

Selecting this icon allows you to create and modify networks of VOPs.
For information, see the *06-VOPs* section.

MOTION AND AUDIO (CHOP)

This icon displays CHOPs, which allow editing of Motion and Audio using Procedural Networks. For information, see the *08-Channels* section.

COMPOSITING (COP)

This icon displays the current COPs. These allow you to perform complex compositing operations on individual or a sequences of images. For information, see the *09-Compositing* section.

OUTPUTS (ROP)

Clicking here accesses your Render Outputs. These control both rendered and composite outputs. For example, you can specify output operators which control *mantra* and Renderman™ output, the frame range to output, and the name(s) of the rendered image files. For information, see the *10-Outputs* section.

JUMP UP / DOWN U / Enter

Allows you to navigate up or down in the Houdini object Hierarchy.

JUMP BACK / FORWARD Alt → / Alt H

Moves back and forth in the history of places you've viewed in the object hierarchy.

MOVE TO PREVIOUS / NEXT OPERATION K / J

Moves to the next or previous OP in an OP chain.

MOVE TO LEFT / RIGHT SIBLING Shift PgUp / Shift PgDn

Moves the selection to the OP left or right of the current OP having to click the OP explicitly.

ADD BOOKMARK Alt Shift K

Setting a bookmark allows you to easily come back to the place in the object hierarchy where you were previously. This is accessible from the *Bookmarks* icon in the toolbar.

EDIT BOOKMARKS Alt B

Brings up a list of bookmarks where you can delete items in the list.

3.3 OPERATOR MENU

ADD OPERATOR Tab / Bksp

Adds an OPerator to the Layout Area or List. A menu pops-up allowing you to choose an operator to place.

ADD GROUP Shift G

Adds the curently selected OPs to the group in the Group List. See *Open Group List* X p. 178. Groups allow you to associate a number of OPs together, so you can enable, disable, select, deselect, and otherwise operate on them together.

ADD NEW NETWORK Shift N

Allows you to add a new Network within a branch of a particular type of OPs. This is one level higher in the Houdini directory structure than the individual OPs. For example, you might add a new network (e.g. /ch2) which contains a number of CHOPs. Individual OPs are added within the network.

CUT / COPY Alt C

Cuts or Copies all selected objects to the Editor's clipboard.

PASTE Alt V

Makes a copy of the clipboard's contents and adds it to your scene. If the names of the copied objects are already taken, the new objects are named with a different sequence number appended.

Tip: If you are having problems Copy/Pasting between two .hip files, you might need to use the Scripting commands *opscript* or *opwrite* to save out the OPs, and the *source* command to get your OPs loaded back into another .hip file. See *OPscript* p. 189, *opwrite* p. 191, and *source* p. 119 in the *Scripting* section for details.

DELETE Del

Deletes all the currently selected operators.

SELECT ALL... Alt A

Selects all operators that exist in the Editor.

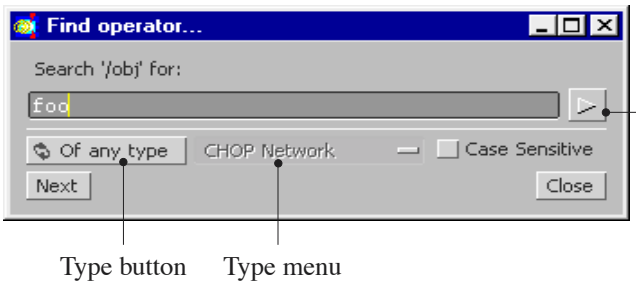
SHOW OPERATOR TREE W

Displays a hierarchical list of Operators (similar to the: *Pane > Tree View* menu).

SHOW PARAMETERS P

Displays the parameters within the Network Editor's pane as an integral part of it rather than as a seperate pane.

FIND OPERATOR... Alt F



Displays a dialog that lets you select an operator by name and scrolls the Layout Area or List view to the given operator making it current and selected in the Layout Area, the Object List (in the Object Editor), and in the Parameter Editing area.

If an operator with the exact name specified isn't found, then operators merely containing the specified text are found.

Tip: Typing / in a network editor acts like Alt F. However, it is a fast OP find which allows you to type in your OP name and then Enter without using the mouse or explicitly closing any dialogs.

wildcards in search

For further search flexibility, the edit field accepts standard wildcard characters as part of the search pattern. For example, searching for “left*bone” will find operators whose name starts with “left” and ends with “bone”.

type button / menu

By clicking on the *Type* button and specifying a type from the Type pop-up menu, searching takes place by operator type. i.e. if you have selected an Add SOP and the name field has: *t** in it, the search will find all Add SOPs that start with *t* .

When you switch to *Of any Type* the edit field automatically changes to contain * (match all names) to avoid the confusion that can be caused when it fails to find anything when no name is typed there.

pop-up menu

Clicking on the pop-up menu button displays a list of operators available in the current editor. This makes it easy to pick an operator, and have *Find* locate and centre the object in the Layout Area.

next button

The *Next* button continues the search using the search string specified in the edit field. This is useful for finding objects that have similar names.

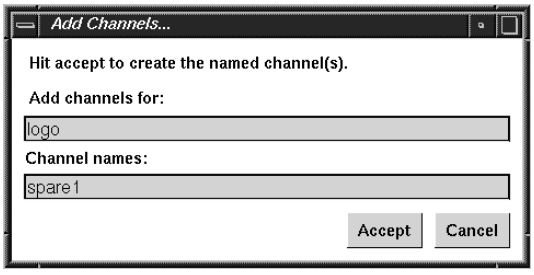
close button

Closes the Find dialog.

RENAME SELECTED NODES

Allows you to rename the nodes, including intelligent Prefix and Suffix handling.

ADD SPARE CHANNELS...



Displays a dialog which enables you to create additional channels for an operator or group of operators. You can view and edit your channels from in a Channel Editor pane.

example of a use for spare channels

Spare channels are useful within expressions. For example, you can multiply a channel with a spare channel to scale the values within it. Say you have a *sin()* expression controlling the */ty* parameter of an object, but you wanted to scale that *sin()* expression over time. You could change the expression for */ty* to something like:

*ch("spare1") * abs(sin($F*12))*

and then graphically modify the envelope of the *sin()* expression in a Channel Editor pane by tweaking the spare channel. Bracketing the *sin()* function with the *abs()* function makes the whole thing bounce. The spare channel can then control the height of the bouncing. Nifty, eh?

renaming channels

In order to rename channels that you've already added, you have to edit the .hip file that contains them. For information on how to do this, please see, *Editing hip Scripts* p. 195 of the *Scripting* section.

deleting a spare channel

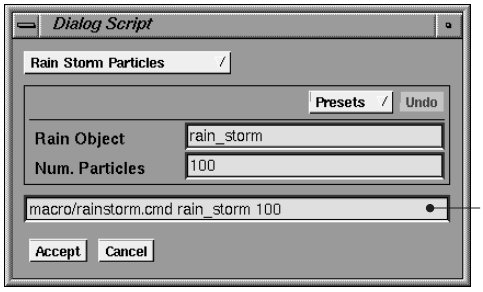
Delete any channel using the *chrm* command in a Textport. For example:

```
chrm /obj/geo1/spare1
```

SCOPE CHANNELS

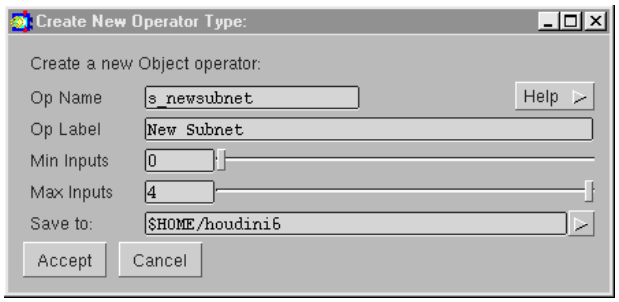
Takes the channels of the selected OP, and scopes them in a Channel Editor, allowing you to see them graphically.

EXECUTE OPERATOR SCRIPTS...



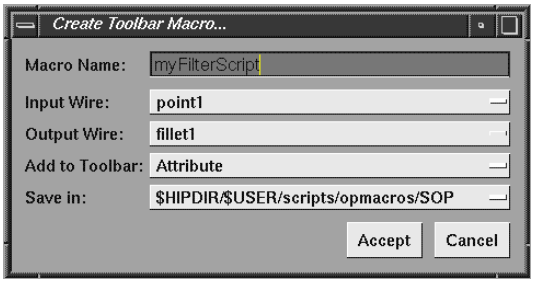
Operator Scripts are macros which provide a graphical interface to command scripts. The macros allow you to change parameters for the script. These scripts can do a variety of different tasks to simplify common procedures or processes. For more information on Dialog Scripts, which include Macros, see *Dialog Scripts* p. 219.

CREATE SUBNET OPERATOR TYPE...



Scripted subnets are defined in a text file located in HOUDINI_PATH. This dialog will create a new table entry in location of your choosing. Following this, you can add parameters to the scripted sub-network. To complete the process, choose Make Permanent Defaults from the Preset menu of the OP. Then when a scripted sub-network is created in the future, it will have the proper setup.

CREATE TOOLBAR MACRO...



Allows you to to create a macro out of the selected OPs. Your macro will show up as if it were just another OP. To do this, simply select the OPs you want in your macro, then select this command. In the dialog that appears, supply a name (how you want it to show up in the toolbar), specify the input and output to your “blackbox”, and click Accept. You will also need to specify a Toolbar into which your macro should be placed. Toolbar macros are saved as *opscripts* which are saved in your Houdini directory.

EXPOSE ALL Shift E

Exposes all hidden operators previously hidden with the *Hide Selected* command.

HIDE SELECTED Shift D

Hides the selected operators while in network or list modes, reducing screen clutter.

COLLAPSE SELECTED INTO SUBNET Shift C

Collapses the selected OPs down into a Sub-network as described in *Sub-Networks* p. 190.

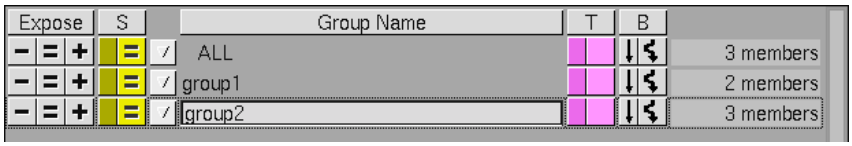
3.4 VIEW MENU

SWITCH TO WORKSHEET / LIST T

Changes the display from a Layout Area full of OP tiles to a List.
See *Tables and Lists* p. 187.

OPEN GROUP LIST X

Displays OP Groups within the Layout Area as a list. Groups allow you to associate a number of OPs together, so you can enable, disable, select, deselect, and otherwise operate on them together.



expose controls

Click on the Expose flags to set the groups of OPs you wish to display:

- **=** Set the display to equal that which is in the group.
- **+** Add what is in the group to what is displayed.
- **–** Subtract what is in the group to what is displayed.

select “s” controls

Click on the "=" to set the items within the group to be selected within the Layout Area or in the List. Click the slightly darker yellow button beside it to deselect.

pop-up menu

Use this menu to add, remove, replace, or delete items from the group according to the current selection.

group name

Displays the name of the group, and allows you to rename the group.

template “t” controls (sops, pops only)

Click on the Pink flag to template the objects in the Viewport. Click on the darker pink flag to disable templating for objects in that group.

display “d” controls (objects, chops only)

Use these two buttons to enable/disable the display flag for objects in the group. Click the blue flag to enable display; and click on the darker blue flag to disable the display of objects in that group.

bypass “b” controls

Use these two buttons to Bypass or Not Bypass the objects for cooking.

number of items

Tells you how many items (“members”) are contained within a group.

HOME ALL / SELECTED H / Shift H

Homes the display of OP tiles within the Layout Area. *Home Selected* homes only the selected OPs. Homing changes the amount of Zoom such that it will fill the Layout Area with the OPs.

FRAME ALL / SELECTED F / Shift F

Frames the display of OP tiles within the Layout Area. *Frame Selected* frames only the selected OPs. As opposed to Homing, Framing leaves the Zoom amount alone.

LIST ORDER

Determines the ordering of the List View: *User Defined*, *Alphabetical*, [by] *Operator Type*, and *Hierarchical*. When set to *User Defined*, use the button and drag to reorder the list.

LAYOUT ALL / SELECTED L / Shift L

Takes the existing tiles in the Layout Area and rearranges them in rows and columns starting in the bottom left corner. This operation can be undone by selecting *Undo* from the *Edit* menu.

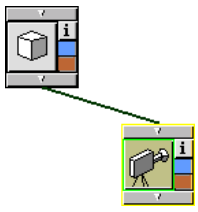
Layout Selected rearranges only the selected Operators.

ALIGN ALL / SELECTED TO GRID

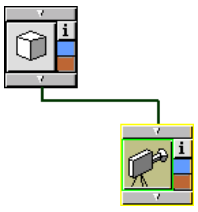
Snaps the OP tiles in the Layout Area to the grid specified in *Display Options*.

LINK STYLE S

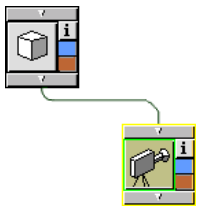
Three styles available are: *Right Angles*, *Straight*, and *Rounded*. Selecting one changes the display of connections between Object Tiles:



Link Straight



Link Right Angles



Link Rounded

SNAP TO GRID

Causes OPs to snap to a grid when you move them – this makes it easier to have a tidy desktop. You can adjust the grid in the *Grid* page of the *Display Options*.

DISPLAY GRID

Displays the grid that OPs snap to when you have *Snap to Grid* enabled.

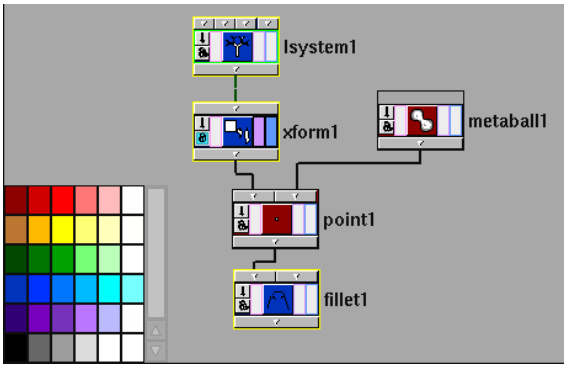
PREVIOUS ZOOM LEVEL B

Gets you back to the previous level you were Zoomed (when changed with).

OPERATOR NAMES N

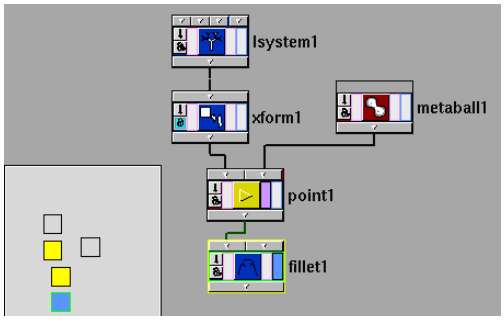
Enables the display of names for OPs beside the tiles.

COLOR PALETTE C



Allows you to colourise individual OP tiles in the Layout Area. Simply select (or Shift select multiple OPs if desired), and then click on the colour you want them to be. The tiles then take on that colour. In the above example, the Lsystem, Xform, and Fillet SOPs are coloured blue, and the Metaball and Point SOPs are coloured red. Colour-coding OPs in this manner can be very useful for managing your project.

NETWORK OVERVIEW O

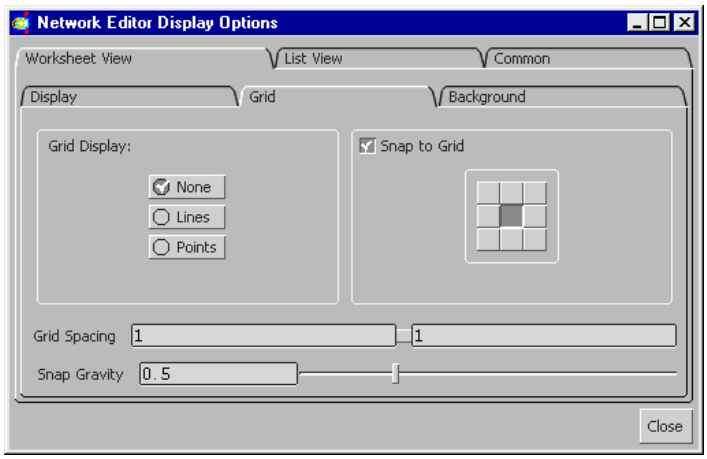


The Layout Area may show only a portion of all the operators contained within it. Enabling the Net Overview, displays a small 'radar' in a corner of the Layout Area, providing a smaller, outline representation of the operators. This is useful for locating specific groups of operators within networks with many operators.

The small white rectangle represents that portion of the Layout Area that you can currently see. Click and drag within the navigator to move the viewing area. The

Operators are coloured as assigned with the Color Palette. This changes if any flags are present – which are shown as quarter-size colour-patches within the tile.

NETWORK EDITOR DISPLAY OPTIONS



worksheet view

Display tab

Allows you to set what is displayed in the Layout View: *Operator Names*, *Network Overview*, and *Colour Palette*. You can also set the *Link Style*, and the behaviour of the *Name Highlights* (i.e. the name of an OP can be highlighted based on the specified condition).

Grid tab

You can specify a Grid for OPs to snap to by enabling the Snap to Grid option (click which anchor point you'd like the use - default is Centre). You can set the *Grid Spacing* and *Gravity* (how close the cursor must be to the grid in order to snap to it), and to display the grid as *Lines* or *Points*.

Background tab

Allows you to specify a background image for the Layout Area. If you want this image to *Pan and Zoom* along with the network, enable the option. You can also specify a *Quality* level for the image (0=Poor, 1=Full).

list view

Edit Expose Flags

Expose flags are used to set which group of OPs you wish to display in the list:

Show Implicit Operators

The default list view for SOPs is such that it only shows what you have directly done, and the order in which you did it.

If you performed an operation that involved the creation of other SOPs (e.g. Lattice, or a Merge + opera-

tion) then those SOPs will be hidden in the list, because they were 'implicit' (i.e. not directly associated with what the user did), and therefore don't have a particular place in the workflow. By hiding the Implicit operators, you get to see only those OPs which you specifically invoked.

You may want to Show the Implicit Operators to better understand what is going on beneath the hood, as an aid to debugging.

List Order — The List view can be ordered: Alphabetically, by Operator Type, Hierarchically, or by a User-defined order (use the button and drag to reorder the list).

common

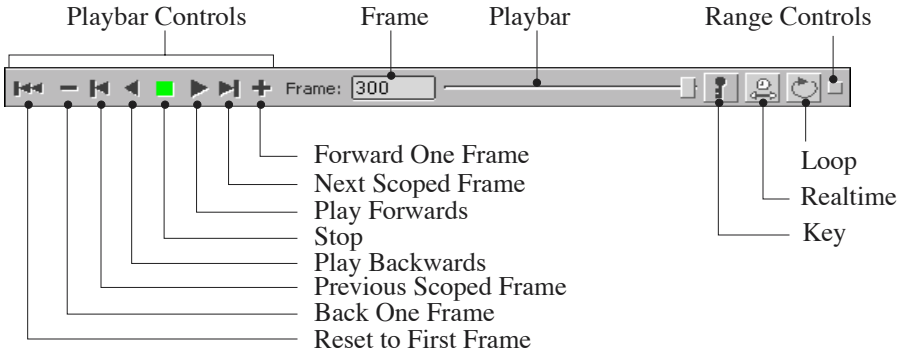
Show Operator Parameters — Displays the Parameters within the Layout Area instead of in a separate pane.

Show Group List — Displays the list of OP groups in addition to the specific OPs.

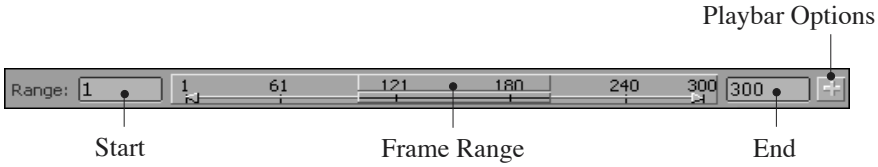
Show Operator Tree — Displays the Tree View.

4 PLAYBAR & PLAYBAR CONTROLS

The Playbar, located across the bottom of your screen, allows you to move back and forth in time – allowing you to animate parameters in the Parameter Area.



Range Controls



The Playbar provides VCR-style controls to *Play*, *Stop*, *Forward*, and *Rewind* animation playback. You can call up the Range Controls by clicking on the *Range Controls* button. These allow you to set the current playback range, looping behaviour, and *Real Time* options. Access the Playbar Options dialog by clicking the ✣ button in the Range Controls, where you can set various attributes, such as the animation length.

4.1 PLAYBAR

Dragging the thumb in the Playbar allows you to move to a different frame by moving the thumb to the desired location. The value displayed in the *Frame* field changes accordingly. The Thumb on the Frame Slider becomes yellow when the frame you are currently at is not in the frame range you specified with the Frame Range Slider.

PLAYBAR CONTROLS

keyboard shortcuts

↑	Play/Pause Forwards.
↓	Play/Pause Backwards.
← and →	Step one frame Forwards or Backwards.

play forward / backward & stop

Plays the animation forward or backward in time, displaying the results of all channels that have been animated. If no channels have been created, the displayed geometry will not change. Click *Stop* to stop playing the animation.

next / previous scoped frame

Steps to the next or previous Keyframe which has been scoped.

step forward / backward

Move the frame number forward or backward by one step, while changing the animated channel values as appropriate.

reset

Use the *Reset* button to quickly jump to the first frame of the range (as opposed to dragging the Thumb of the Playbar there manually). This is necessary when working with POPs, when the network topology or parameters of the POP network have changed. When this happens, the Viewport becomes 'stale', and only a reset of the simulation will provide accurate results. In this case, the *Reset* button turns yellow to warn you of the stale condition.

PLAYBAR – CHANNEL HOLD

	Click the Playbar's Thumb to set Frame.
	Drag thumb to hold scoped Channels.
	Click thumb for previous value.

Tip: This is the same as doing a: *chhold -b* command when you start dragging and then doing a: *chhold -e* command when you release the thumb (see *Scripting*).

FRAME

Displays the frame number of the animation currently being shown. Moving the thumb in the Playbar changes this value. Enter a value into this edit field directly to jump to a specific frame.

KEY BUTTON

Sets a key on currently scoped channels.

REALTIME

Enabling this option plays your animation back at a speed that appears realistic. Because this playback mode uses a considerable amount of your computer's processing resources, a machine with a slower processor may have to drop frames in order to meet the requirements of real time. Instead of appearing smooth and natural, your animation may appear erratic.

LOOP

This popup menu allows you to select from three modes:

loop

After clicking the *Play* button, Loop displays your animation in a continuous cycle from beginning to end and back to the beginning again. Click on the *Stop* button to end the cycle.

play once

Plays your animation from beginning to end only once, and then stops at the end.

zigzag

This plays your animation back and forth, from beginning to end and then reversing direction, back to the beginning; continuously bouncing back and forth.

4.2 RANGE CONTROLS

Click on the Frame Range stow/unstow button at the end of the Playbar to display.

FRAME RANGE SLIDER

The frame range slider represents the entire length of the animation.

- Grabbing the arrows, located on either side of the Thumb, and dragging them alters the playback frame range.
- Dragging in the upper part of the slider changes the play range, centered around the middle of the playrange.
- You can also drag the lower part of the slider itself to set the locus of the play range.

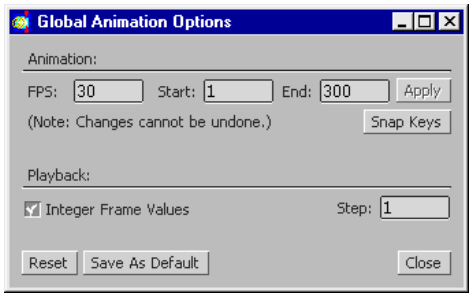
start - end

The two fields on either side of the Range Slider area display the start and end frame numbers for the playback range. You can specify your own start and end frame numbers by typing directly into these fields.

PLAYBAR OPTIONS ✣

Click to display the Global Animation Parameters dialog (below).

4.3 GLOBAL ANIMATION PARAMETERS



FPS

Frames Per Second sets the rate for animation playback when *Real Time* playback is enabled. You generally want to set the FPS for your animation only once, and do so before you begin animating anything at all. This is because when you change the FPS, the time of the last keyframe will not match the new animation length. If you want to modify your animation timing, you must do so explicitly.

The FPS rate for NTSC video is 30, for PAL video, it is 25, and for Film, it is 24.

Production Tip: If you made a mistake, you can use a Field COP with a *Pushup / Pulldown* operation to fix a sequence of rendered-out images and convert them to a different FPS after the fact.

START / END

The *Start* field displays the starting frame number of the animation sequence. By default, it contains a value of one. If you enter a value other than 1, an alert appears asking if you want to increase/decrease the animation length. Click on *Squash/Stretch*, or *Truncate/Append* as desired. After squashing/stretching, you may have to adjust some of the channels.

The *End* field displays the final frame number of the animation sequence. Entering a new value here brings up an alert asking whether you want to *Squash/Stretch* or *Truncate/Append* the data currently in the animation. Click to choose the desired function. After squashing/stretching, you may have to adjust some of the channels.

tip to squash or stretch an animation

Houdini uses floating point keyframing. Therefore there's no need to squash or stretch or use field rendering to cheat in lengthening an animation. Click on the ✣ button of the Playbar to bring up the Global Animation Parameters and turn off *Integer Frame Values*. Then you can enter decimal frames in the *Start/End/Inc* field (Output Editor > *mantra* parameters). If you specify a non-integer frame increment, it will give you a stretched or squashed animation without having to adjust the channel data. For example if you specify a frame *Inc(rement)* of 0.8, you'll get an animation which is 20% longer.

You will want to be careful with the name of the file for the image you're generating. If you use the $F variable in the filename, you will probably have one frame writing over another. In this case it is better to use the $N variable in the filename

which specifies the number of the image being rendered. For example, for the first image $N = 1, and for the second image $N = 2, and so on.

Tip: Set the environment variable HOUDINI_ENABLE_FPS_SCALE if you want your keyframes to scale as they did in Houdini 4 and prior.

SNAP KEYS

When enabled, this will snap all keys to integer frame boundaries.
Note that this does the equivalent of: *chround -a* .

INTEGER FRAME VALUES

Houdini works with floating-point frame values ($FF). When enabled, this option confines the display of frame values to integers ($F). When disabled, Houdini allows you to set the current time to any value, even a fractional time that does not lie exactly on an exact frame number.

STEP

The size of a single increment in frame values. For example, entering a value of 2 here plays every second frame.

4.4 PLAYBAR EXAMPLE

1. From a default Houdini, a Viewer and Parameters panes should be set to view *Objects*, and a geo object named '*logo*' should be placed at its default position in the centre of the Viewport, while the Playbar is set to frame 1.

2. In a Parameters pane (locate the *logo* object in a *Network Editor* pane, and type P for Parameters), click on the *Translate* parameter with with and select *Set Keyframe* – this creates a channel (green), where frame 1 contains a value of 0.

3. We also need a keyframe at the end. So move the Playbar to frame 300, and click with and Set Keyframe there also. This ensures that we end up at zero for the last frame.

4. Now move the Playbar to frame 150, and in the parameters for the *logo* object type in a new *Translate-X* value for the logo of: *3*. You should see the logo move to the right in the field of view, and the parameter turns yellow. Click with on the Translate Parameter, and *Commit Change* (If you no longer see the logo, you may have to home the View by clicking on the Home icon at the bottom of the Viewport. If homing zooms out the scene too much, use Space within the Viewport to zoom in/out as desired).

5. Click *Play* in the Playbar, and watch the logo slowly move back and forth. If you want to see a visual representation of what is happening to the channel values and the keyframe you've created, then click on the *Translate* parameter's name to display a pop-up menu, and select *Scope Channels*.

6. Since a channel now exists to accept keyframes for the *Translate* parameter, you can add more keyframes easily. Stop the animation playback (click *Stop* in the Playbar), and set the Playbar to frame 230.

7. Using the Transform Operation in the Viewport (type Tab in the Viewport for a pop-up menu of Operations), drag the logo from its present position over to the left area of the Viewport (so its Translate-X value reads something like -3).

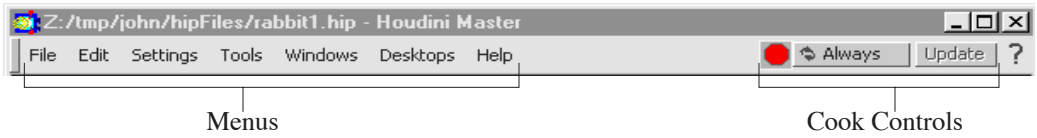
Since this translational change has not yet been committed, the Key Indicator in the Playbar once again lights up Red to warn us of the uncommited change.

8. Click the *Key* button to commit the change, and store the new translation value at frame 230. The values between the frames are automatically interpolated (you can see this if you viewed the channels by scoping the channel).

9. Once again, click *Play*, and you should see the logo move between both key-framed locations.

Tip: You can graphically edit the keyframe values (both their frame number, and parameters value) by dragging the keys within the Channel Editor's graph.

5 MENU BAR ELEMENTS

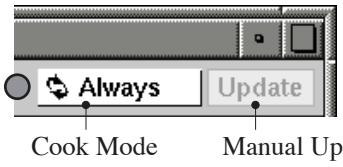


5.1 MENUS

The Menu Bar in Houdini contains several pull-down menus (*File, Edit, Options, Dialogs, Help*) that allow you quick access to global commands essential to the program. The menus display the name of the command and, if applicable, the keyboard short-cut for it.

Menus are covered in detail in the chapter: *Menus* p. 72.

5.2 COOK CONTROLS



Setting the Cook mode allows you to decide if you want to interactively see the effect of changes to parameters, or to suppress automatic cooking in order to speed up workflow.

KEY BUTTON

Depending on the current status of all channels this indicator is green or red. Normally, the indicator is green, but if you change the value of an animated parameter at a frame where the parameter's channel has no keyframe, the indicator becomes red, indicating that there are pending changes. Clicking on the button while it is red commits the pending changes by adding keys to those parameters that have been modified. See the *Playbar Example* p. 68. If you change the current frame when the indicator is red, then uncommitted changes will be erased and the indicator will become green again.

COOK MODE

always (continuous updates)

When enabled, causes OPs to update as you change their parameters, and continuously updates the Viewport display to reflect the changes. This mode allows better interactivity, but may be slower than the other modes as it continuously cooks all OPs affected by your changes.

This mode is the same as *Continuous Updates* in previous versions of Houdini.

never

Allows you to globally suppress cooking.
Viewport and OPs are only cooked upon explicitly clicking on the *Update* button.

changes

This mode forces Cooking after you have finished making parameter changes. Any changes you make appear only after you release the mouse button (this is especially noticable when using value grids).

UPDATE

Click here to manually update the cooking of all OPs and Viewport displays when the Cook Mode is set to *Never.*

When you see a yellow border around the *Never* and *Update* buttons, it indicates that something has been modified and an update will refresh the current view.

This button is disabled for the *Always* and *Changes* cook modes, as it is unnecessary for them.

5.3 MESSAGE AREA

DESCRIPTION

At the bottom of the Houdini window, just below the Playbar, is the Houdini *Message Area*. It displays brief messages from Houdini. Typical messages include: the version number of Houdini, the number of operators that have been deleted, the status of the deletion, the addition of operators, and the successful loading of a network into memory.

In the Model Editor, it provides you with context-sensitive assistance. For example, “To build a sphere, click and drag on floor, or click, or just hit Enter.”; “Left mouse to tumble, middle mouse to dolly/zoom, right mouse to scroll.

2 Menus

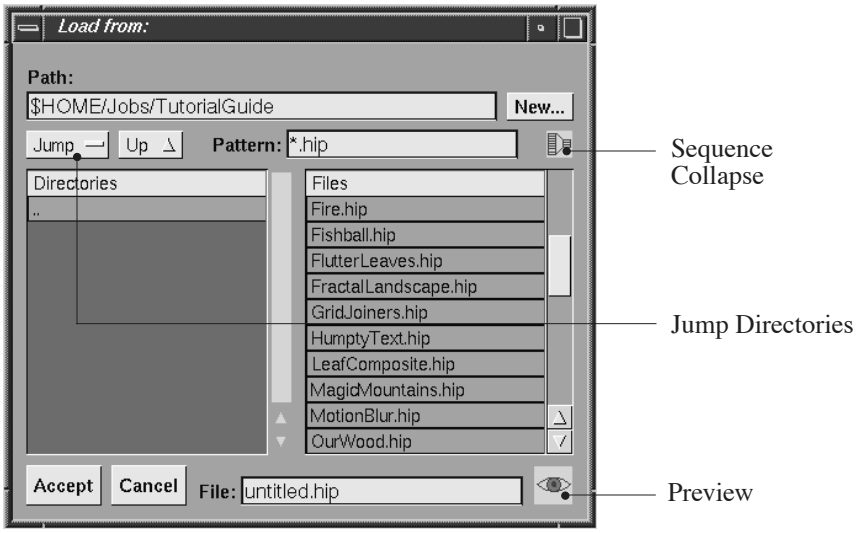
1 FILE MENU

NEW Alt N

Deletes all existing objects and channels within your project, giving you a new, untitled blank slate on which to work.

OPEN... Alt O

Presents an Open dialog asking you to select a *.hip* file to open:



If your current project in memory is not saved, an alert appears giving you the option to discard your work, save it first, or cancel the open operation altogether.

To go to a different directory, type the path in the *Path* field. Files matching the current pattern (see *Pattern* p. 73) are listed in the *Files* scrolling field. Click on a file, or type its exact full name in the *File* field, and click *Accept* to open the file.

A *.hip* file contains everything necessary to recreate the contents of your project as it existed at the time that you saved the file. This includes all OPs, OP layouts and connections, model geometry, materials, channel, and key information, and Viewport settings.

jump / up / drive

The *Jump* menu allow you to jump to a number of predefined directories. Most of these are preset, but you can set $HIP in a Textport:

```
hip /usr/staff/rick/myJobs
varchange
```

Note: The *varchange* command must be run after the *hip* command because the *hip* command is changing a the value of a variable.

The *Up* and *Drive* buttons allow you to move up a directory level, or change the drive. ***Note:*** Solaris, Linux and Irix don't need the *Drive* button, since they contain a network-centric file-system as opposed to Window's drive-centric file-system.

pattern

Allows you to specify a pattern to filter the display of files in the window. For example, a pattern of **.hip* will only list files ending in *.hip* .

sequence collapse

The *Sequence Collapse* button allows you to list multiple files with similar names as a single entry in the list. For example the files:

```
Foo1.pic, Foo2.pic, Foo3.pic
```

would be listed as: *Foo[1..3].pic* if the *Sequence Collapse* button was enabled. This button is disabled when opening *.hip* files, because you can't load a sequence of .hip files. It is usually available when opening image sequences (such as in *iplay*).

new

Creates a new subdirectory. You are prompted for a name.

preview

Clicking here presents you with a preview of the currently selected file in a separate window.

MERGE... Alt M

Opens a dialog box asking you to specify a file to open. This file's information will be combined with that of the currently open file.

SAVE Alt S

Saves your project as a *.hip* file. The data is saved to the file whose name was previously specified using the *Save As...* command. If the file has not already been given a name, it is saved as *untitled.hip*.

If the *Increment on Save* option (in the *Options* menu) is enabled, and the filename ends in a number (e.g. *untitled1.hip*) then each successive save will save as a new file, where the number in the filename is incremented by 1.

SAVE AS... Alt Shift S

Displays a dialog box which asks you to specify a filename for your work before saving it to a *.hip* file. After you have specified the filename, the *Save* command will save the current or updated state of your work to the file with this filename.

If the file already has a name, entering a new name in the dialog box saves your work to the newly specified file, leaving the original file unaltered. Using the *Save* command after specifying the new filename saves your work to the new file.

NEW OPERATOR TYPE...

Allows you to create your own OPs. See *Scripted Subnets* p. 192 for an example.

INSTALL OP TYPE...

This option allows you to read .optypes saved from an OP (by selecting *Create Type From...* from the OP's popup menu) and install them.

REFRESH OPERATOR TYPE LIBRARIES

Will update OP type libraries, based on settings in *Tools > Operator Type Manager.*

NEW DESKTOP Alt D

Creates a new blank Desktop for you to arrange and configure as desired. By default, new Desktops are named for you: *Desk1*, *Desk2*, *Desk3*, etc. You can rename them by selecting: *Dialogs > Desktops* Alt D p. 79.

You should Save your Desk once you've arranged it to your satisfaction. Do this with the *Save Desktop* command, or with the *Settings > Desktops...*

SAVE CURRENT DESKTOP

If you have made changes in your arrangement of Panes, this command saves them into the current desktop. You can rename and edit Desktops by selecting: *Settings > Desktops* Alt D p. 79.

SAVE ALL DESKTOPS

Same as *Save Current Desktop*, except changes in all Desktops are saved.

LISTING OF RECENTLY OPENED FILES...

Select one of these to quickly open a recent file.

QUIT APPLICATION Alt Q

Quits Houdini. If you choose to Quit Houdini, and your work is not yet saved, an alert appears asking if you want to save your changes.

2 EDIT MENU

UNDO / REDO... Alt U / Alt R

The *Undo* command allows you to take back the changes made in the last operation. For example, deleting a SOP, changing the layout of tiles, changing a parameter's value. The *Redo* command reapplies the changes that were undone in the last *undo* command. Not all operations can be undone.

The number of operations that you can be *undo* or *redo* is determined by the amount of memory that is consumed in storing the data necessary for the operation.

To undo changes within a parameter edit field, you can also click with the right mouse ( ) button within the edit field.

UNDO HISTORY...

Displays a dialogue listing recently completed actions, and allows you to Undo or Redo them by clicking to a given level in the list.

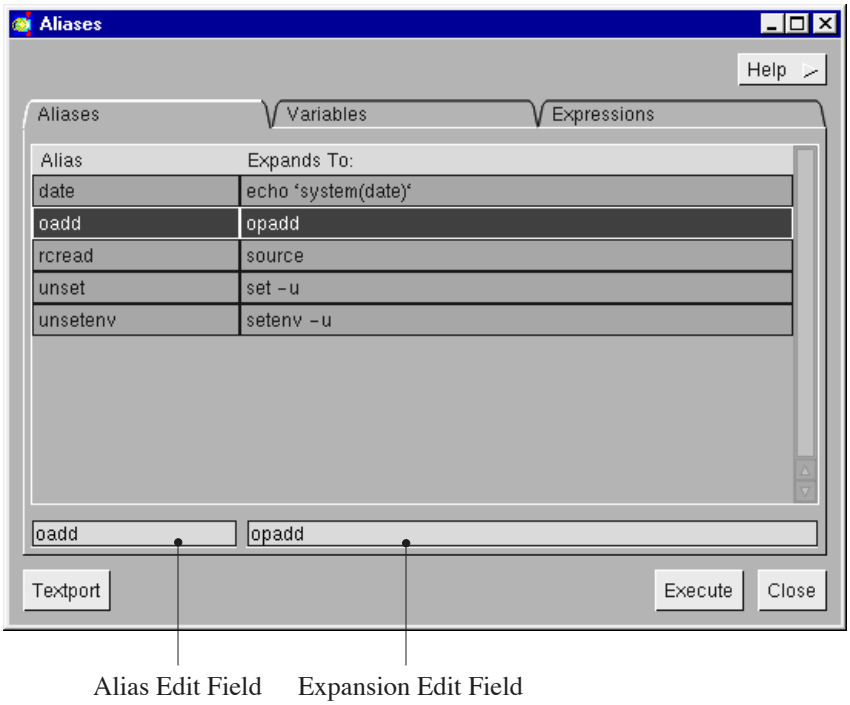
3 SETTINGS MENU

3.1 MAIN PREFERENCES

These are lengthy, and can be found here: *Preferences* p. 97.

3.2 ALIASES AND VARIABLES... Alt Shift A

Selecting this item displays the Aliases/Variables dialog, which allows you create, edit, and delete command aliases (i.e. abbreviated forms of commonly used commands), and define Varibales, and custom Expression Functions.



ALIASES PAGE

The dialog comes with aliases for some common commands (set within the *123.cmd* file). To edit an alias, click on its text box in the list. The alias and its expanded form are highlighted to indicate they've been selected. Click on an edit field to ready it for editing and type your new text in the field(s).

To add an alias, click on the *Alias* edit field and enter the text you want to serve as an alias. In the *Expansion* edit field, type in the command you want to Houdini to execute when you use the alias. Type Enter to add the new alias to the list above.

To delete an alias, select it in the list and then type the Del key.

aliases – assigning an alias to a scripting command

For scripting commands, enter the command in the second column, and the alias you want it assigned to in the first column. For example, the *opcf* command has the alias *opcd* – whenever you type *opcd* in the Textport, it is the same as if you typed *opcf*.

VARIABLES PAGE

This consolidates the definition of variables used throughout Houdini. Enter values for common values such as PI, HIP, and JOB here. You can set them here instead of explicitly setting them with the *set* scripting command in the Textport.

EXPRESSIONS PAGE

This page allows you to enter Expression Functions which can be used in any of your expressions within Houdini. For example, you can create a custom expression function that will find a minimum value between two given values:

```
# Function to find the minimum value of two
# floating point numbers
min(v1, v2)
{
  if (v1 < v2)          return v1;
  else                  return v2;
}
```

You can find the syntax for this language in: *Ref > Scripting > Custom Expression Functions* p. 36.

TEXTPORT

Clicking this button invokes Houdini's Textport, where you can enter commands, or see the output generated by executing an alias. This is equivalent to selecting the *Textport...* command from the *Dialogs* menu.

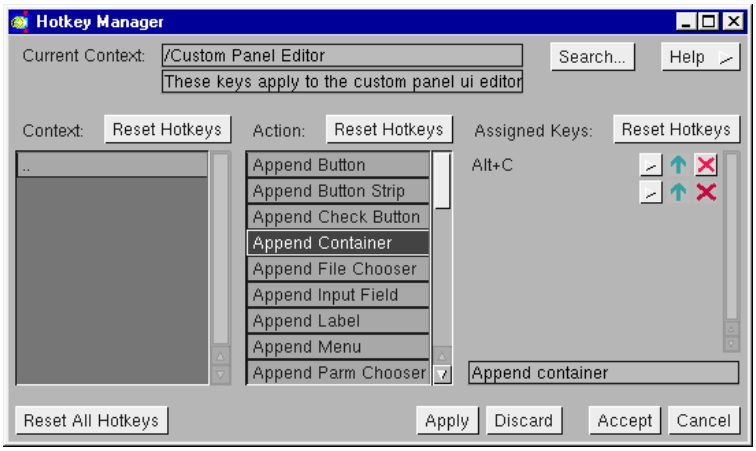
EXECUTE

This button executes the command highlighted in the Aliases list.

CLOSE

Dismisses the dialog.

3.3 HOTKEYS



This dialog allows you to assign, change or remove hotkey assignments for many of Houdini's actions. Use the Search button within the dialog to locate the action or hotkey you wish to edit,

THE SPOCK GRIP (Ctrl Alt Shift)

You can assign or reassign a hotkey to virtually any Houdini function, button or menu by typing: Ctrl + Alt + Shift and clicking on the UI widget you want to assign the hotkey to. Doing so brings up this dialog so you can edit its hotkey.

Note: Not all UI widgets support hotkeys. Hotkeys displayed in menus and popup help are not updated until Houdini is restarted.

EDITING A HOKEY

To edit the hotkey for an action, use the *Cryptic Combination*, the *Search* dialog or navigate to the proper context using the *Context* List. Then choose the action you want to edit from the *Action* list. A list of currently assigned hotkeys will appear.

REMOVING / ORDERING / CHANGING A HOKEY

To remove a hotkey, click the button with the red X next to the hotkey. You can change the order in which the hotkeys appear in the UI using the blue up-arrow buttons. You can change a hotkey by selecting the hotkey to edit, then clicking in the edit field. The next key that you type will be translated into its hotkey representation. To create a new hotkey for an action, simply edit the last entry in the hotkey list, which always appears as a blank entry.

ASSIGNING VIA THE POPUP MENU

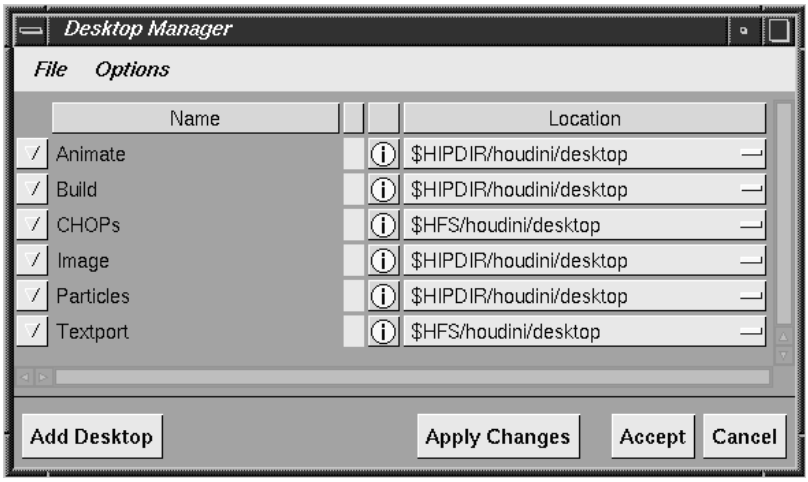
You can also edit an existing hotkey or assign a new hotkey using the ▷ popup menu, which presents a list of all possible hotkeys. Those keys which cause conflicts appear in red, but can still be selected.To assign a hotkey with some combination of the Ctrl Alt or Shift modifiers, simply hold down the appropriate keys when clicking on the ▷ menu button.

3.4 DESKTOPS Alt D

Desktops allow you to save a particular arrangement of panes as a set.

An added desktop can be saved with its particular arrangement of panes for rapid recall by selecting it from the *Desks* menu. You create a new Desk with the *File > New Desktop* command. After arranging and configuring your panes as desired, you can save a Desk by using the *File > Save- Desktop*.

For greater control over your Desktops, the *Desktop Manager* displays a dialog in which you can rename, load, and save Desktop settings:



FILE MENU

add new desktop Alt A

Adds a new desktop, creating a new file for the entry.

save modified desktops Alt S

If desktops have been modified (by resizing, changing panes, etc.) this will save those changes into each of the corresponding desktops files.

apply changes

Changes made in the Desktop Manager will not take effect until this explicitly instructed to do so by clicking this button.

discard changes

Discards any changes you have made to the setup of your desktops within the Desktop Manager.

OPTIONS MENU

show hidden files

Shows hidden files (marked with Red flag).

show all files

If Houdini's default desktop files are overridden by files of the same name in the users *$HOME/houdini* directory, then only the files in use are displayed, and the default ones are not displayed. When this option is enabled, all the files are shown, even if they are not actually in use, because they are overridden. This can be confirmed by clicking on the Info popup.

auto save changes

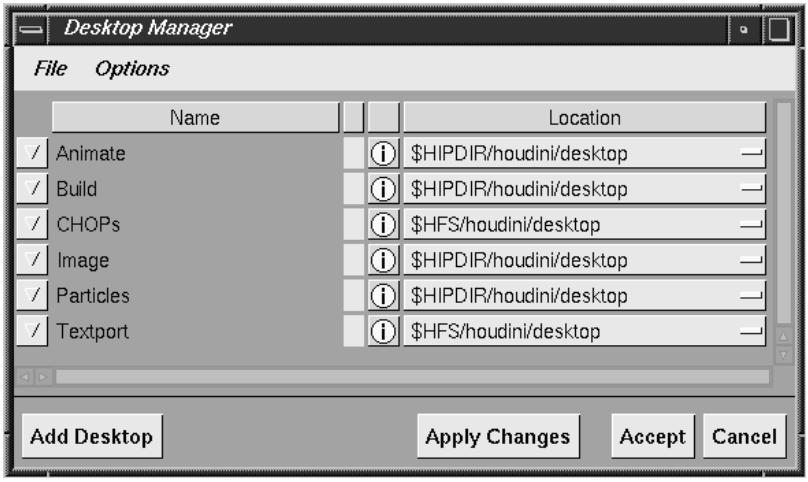
When enabled, changes to desktops are automatically saved to the relevant desktop files without having to explicititly save them.

LIST OF DESKS – ENTRIES

Each Desk contains: *Name*, *Location*, *Info*, *NoSave* flag, and a *Menu*.

The *Name* of the desk is thefilename to which the Desk info is saved. The location this file resides is displayed in the *Location* column. If the *NoSave* (green) flag is on, then the file is considered "locked". No changes will be saved to the file.

3.5 DESKTOPS



Desktops allow you to save a particular arrangement of panes as a set.

An added desktop can be saved with its particular arrangement of panes for rapid recall by selecting it from the *Desks* menu. You create a new Desk with the *File > New Desktop* command. After arranging and configuring your panes as desired, you can save a Desk by using the *File > Save- Desktop*.

For greater control over your Desktops, the *Desktop Manager* displays a dialog in which you can rename, load, and save Desktop settings:

See *Desktops* Alt D p. 79 for a description of this dialogs controls.

3.6 OBJECTS

ALWAYS HIGHLIGHT OBJECT SELECTION

When an object is selected in the Viewport, it will be lit up in yellow. You may want to turn it off when Painting your geometry, or if you want to see your applied textures (while editing the geometry) without the visual interference of the highlight.

KINEMATIC OVERRIDE

When viewing bones, setting this to Rest will show you the original positions of the bones as they were posed. If set to *Capture*, it will show you the bone's position at the capture frame. The rest position is the position of the bones when they were last edited in the bone state.

KEEP POSITION WHEN PARENTING

This option, when enabled, allows you to parent objects while maintaining the child object's world space position.

Never — When set to *Never*, objects will not maintain their world position when you parent objects together, rather, they will assume a space relative to the parent object.

Always — When set to *Always*, objects will maintain their world position and orientation even if you rewire an object node to set its parent.

Use Object Flag — This setting will maintain an object's world position according to the *Keep Position When Parenting* parameter of the individual object.

Generally, you want to make sure *Keep Position when Parenting* is set to *Always* before parenting limbs to each other so that the parenting can be changed without altering the position of the objects involved.

Note: When set to *Always* – to compensate for the new parenting relationship, Houdini may need to change one or more of the child objects *Translation Order*, *Translate*, *Rotate* or *Scale* values when the object is parented. This is undesirable if you have an animation channel in one of these with a value that may change. Thus, the *Keep Position When Parenting* does nothing if the child object has animated channels in any of the necessary *Translate*, *Rotate* or *Scale* parameters.

3.7 SKIP GADGET UPDATE

Disables refreshing of the Playbar's current status and of other animating interface controls during animation playback. This increases playback performance.

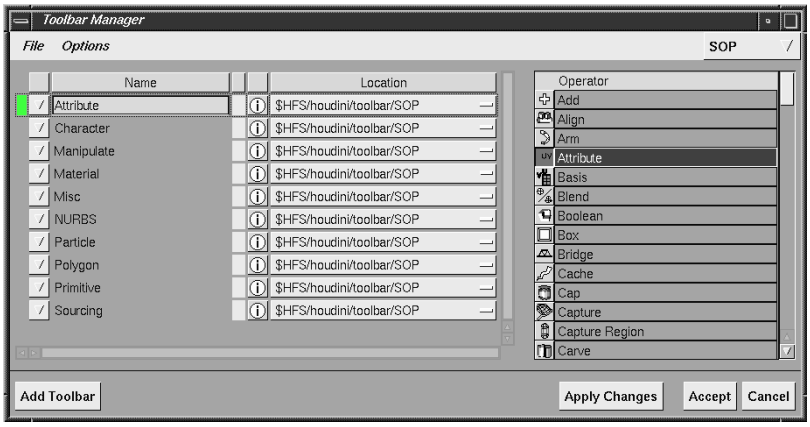
3.8 BALLOON HELP TAGS

When checked, leaving the mouse stationary over a button or control for a moment displays a small help tag with a terse description of its function.

Note: Only controls that perform direct actions will have help tags. Some others, like menu buttons, will not.

4 TOOLS MENU

4.1 TOOLBAR MANAGER...



Allows you to customise the Toolbars found in the Network Editor for each OP type. This also defines the sub-menus you see when you type Tab to place a new OP in the Layout Area.

With this, you can create new toolbars containing sub-sets of the OPs. For example, you can create a toolbar of Generator SOPs, and another one for Modifier SOPs.

creating a custom toolbar

To create a custom toolbar, select the type of OP from the OP-Type menu on the top-right. Then click on the *Add Toolbar* button. The list on the right will display all available OPs that you can add into the toolbar. Toggle the items you want included in that toolbar by clicking on items in the list. When you are done, click on the pop-up menu beside the name of the Toolbar you've added, and select *Save*. If you don't want a toolbar any longer, select *Delete*.

4.2 OPERATOR TYPE MANAGER...

Note: See also: *OTLs – Operator Type Libraries* p. 197.

OPERATORS PAGE

This dialog allows you to manage operator types and operator type libraries. For any operator definition you can view information about it by clicking on it with the middle mouse ( ). Use the right mouse ( ) on an operator definition to see a list of actions that can be performed on the definition.

If an operator definition name is displayed in black, it means it is the current definition being used for that operator type. Names in yellow are also being used as the current definiton for that operator type, but there is a more recent definition availa-

ble in another operator type library. Names in red are not the current definition for the specified operator.

Suppose an operator definition appears in red, but you want to make it the current definition for that operator type. The first step is to find the current definition, which you can do by choosing *Jump to Current Definition* from the right mouse ( ) menu on the unused operator definition. If the current definition is part of the operator type library Embedded in the Hip file, the easiest approach is to delete the operator definition from the Embedded library. This is also the way to update to the latest definition for an operator when the *Give Preference to Definitions Saved in Hip File* option is enabled. You can also delete the entrie Embedded library using the right mouse ( ) menu on the library itself. If the current definition is not part of the Embedded library, you can Update the Time Stamp of the definition that you want to use, or you can delete the definition that is being used.

Use the *Find Definition* buttons to find a particular operator type. Only the Next Definition button works if there are wildcards in the field. For example, if you enter *Shop*/* in the field, the *Next Definition* button will cycle through all SHOP operator types.

CONFIGURATION PAGE

The Configuration page allows you to set the following options:

Give Preference to Definitions from Index Files
: Tells the operator type manager that operator definitions listed in index files take precedence over definitions for the same operators found in operator type libraries.

Give Preference to Definitions Saved in Hip File
: Tells the operator type manager that operator definitions embedded in a Hip file take precedence over definitions for the same operators found in other operator type libraries.

Give Preference to Definitions with Latest Date
: Tells the operator type manager that for a given operator definition, the version with the latest time stamp should be used. The alternative is to use the definition which was loaded last.

Save Operator Definitions to Hip File
: Tells the operator type manager to embed an operator type library when saving your Hip file. The embedded library contains definitions for every operator used in the hip file.

Display Warning for Out of Date Operators
: Tells the operator type manager to display a dialogue when loading a Hip file that contains embedded operator definitions if there are more recent definitions for those operators.

4.3 PERSISTENT HANDLE EDITOR

This dialogue allows you to create and modify persistent handles. Changes to handle bindings will take effect when you click Apply or Accept. Any external changes to the handles will cause uncommitted changes to be lost.

On the left hand side is the list of all the current persistent handles, and below it a channel tree. To bind an operator's parameter to a handle, first select the handle from the list, and then drag the desired channel onto the desired binding. Note that only one channel can be bound to each handle binding.

The Type field is not editable and reflects the basic type of the handle.

The Name field is the english name which should uniquely define this handle with respect to the operator type. One may have multiple handles of type 'angle', but only one handle with the name 'angle'.

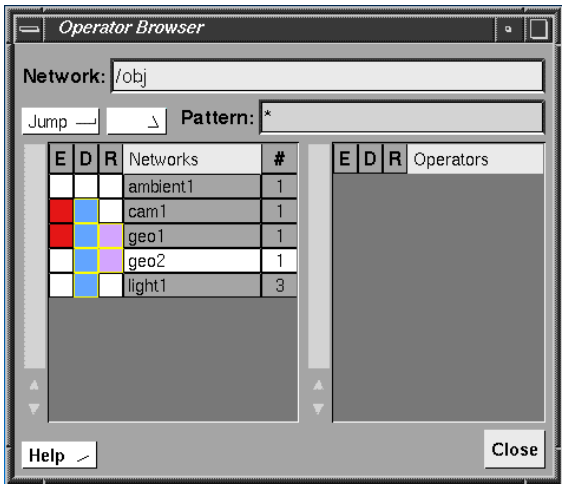
The Default field is not used for persistent handles.

The Bindings listing shows a list of the handle parameters that can be bound to operator channels. The name beside each entry is the handle parameter name. The entry field can be filled with a channel name present in the operator and the relevant handle parameter will be bound to it. To bind a handle parameter to a constant value, use: *!constantexpression* . To specify a conversion function between the handle parameter and the operator channel, use: *{ a, b, c, d, chname }* – which defines the function: $f(x) = (ax + b) / (cx + d)$.

For example, to have the handle be the negative value of the channel, use:
{ -1, 0, 00, 1, chname }

To have the handle be the reciprocal value of the channel, use:
{ 0, 1, 1, 00, chname }

4.4 OPERATOR BROWSER... Alt Shift B



This displays a dialog in which you can view and edit flags for each node in the object hierarchy – turning objects on/off, and setting display and render flags.

The flags can be modified by clicking once on each, or by dragging up or down the column to set or unset many flags together. The state of the first flag set determines the state of all the others while dragging up or down.

If you Alt click an OPerator, it will make it the Current operator within its network. If you are within that editor, this will cause its parameters to be displayed.

The *Network* field at the top shows the current path and can be edited to change locations. The *Jump* button can also be used to change networks while the *Up* button climbs up to the parent of the current network.

The *Pattern* field allows the names in the list to be filtered by a string pattern.

FLAGS AVAILABLE

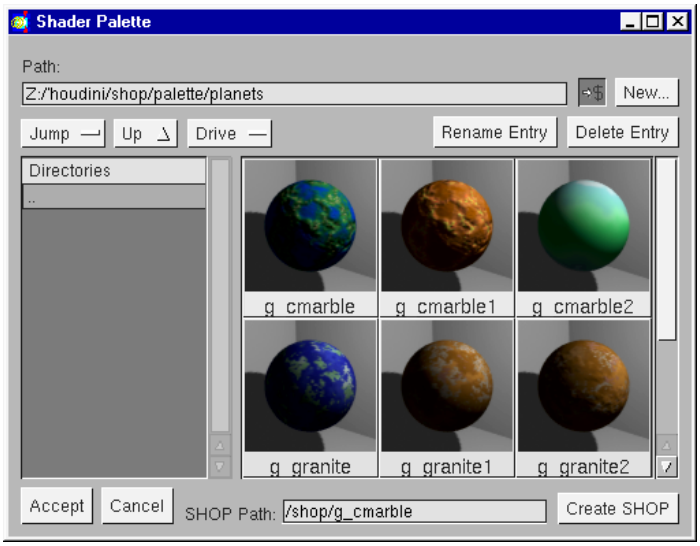
E - Expose (Red)	Displays/hides the specified OP in the Layout and List views. Note that hidden operators still cook and display in Viewports and renderings.
D - Display (Blue)	Displays/hides the specified OP in the Viewport.
R - Render (Violet)	Specifies if the OP should be the one to be rendered.
#	Number of OPs in Network.

Note: The display and render flags are not meaningful in all editors, and in these cases, changing the flag will have no effect.

EXPOSE/HIDE VS SUB-NETS

These are not the same, and should not be confused. Exposing and hiding OPs simply changes the fact that they are displayed or not. This is in contrast to Sub-nets which embed operators within a single operator in the Layout Area.

4.5 SHADER PALETTE



The Shader palette allows you to rename, delete, and create shops in a specified directory, providing you with a way to manage a library of Shaders.

PATH / JUMP / UP / DRIVE

Specify a path to the desired shops here, or by using the Jump, Up, and Drive buttons to navigate to the desired directory path.

NEW...

Specify the name for a new directory in the dialog that appears.

PALETTE

This area provides previews lists all the available palettes in the path specified above. You can choose from a large selection of pre-defined shaders.

RENAME ENTRY

Allows you to rename a shader.

DELETE

To remove a material from a Palette, select it and then type the Del key or click the *Delete* button.

Note: You need to have write permissions to delete items in a given directory.

4.6 COMPOSITE PROJECT

This dialogue allows you to specify the settings for Compositing. You can specify basic parameters such as: *Resolution*, *Aspect Ratio*, *Pixel Format* (i.e. Bit Depth), the *Frame Range* and *Rate*. The *Global Cook Resolution* allows you to work on scaled versions of the processed images to save cook time.

4.7 RESET PLAYBAR TO BOTTOM

If the Playbar is not at the bottom of the Houdini window, but has been moved to reside within a pane (using the *Pane* menu > *Move Playbar Here*), you can reset the Playbar back to the bottom of the Houdini window by selecting *Reset Playbar.*

4.8 CLEAR COMPOSITING CACHE

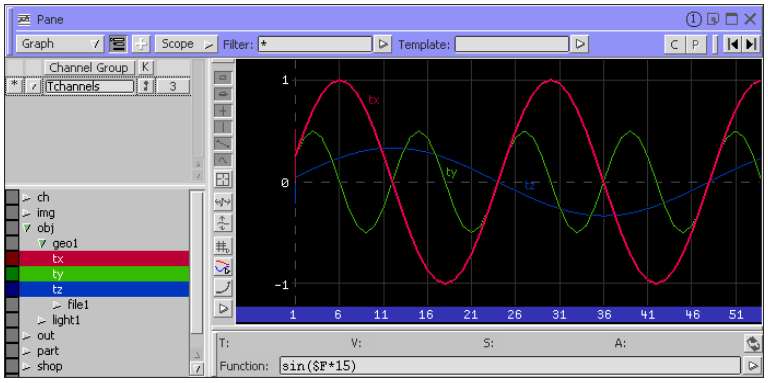
This command empties the memory cache associated with all active COP networks. Caching COP images in memory speeds up processing of COP images significantly (because they don't have to be manually recomputed each time). However this comes at the expense of memory – lots of it. Because caching large images over multiple frames can take up so much memory, this provides a way to manually clear out the memory used by the COP caches.

5 WINDOWS MENU

5.1 FLOATING PANE... Alt Shift W

Simply creates a new floating window which contains a standard Houdini pane (see *Pane Controls* p. 44). This allows you to have a separate window with a Viewport, Layout area, or parameters as you wish.

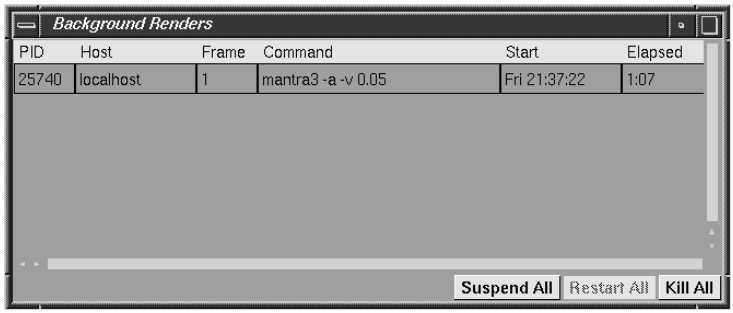
5.2 CHANNEL EDITOR



The Channel editor displays channel data for parameters over time graphically. It also shows segment functions which mathematically interpolate between keyframe values, and lets you adjust these values.

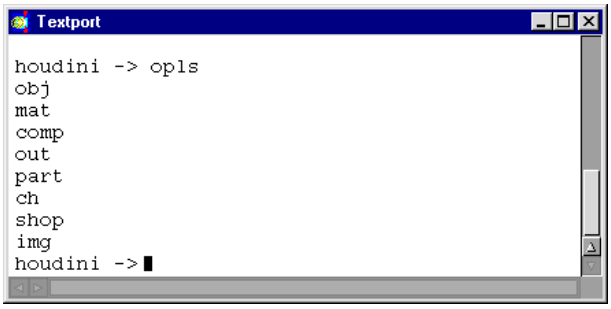
See *Channel Editor* p. 223 for more information.

5.3 RENDER SCHEDULER...



Displays a dialog in which all render processes and information about te processes are listed. By selecting a render process from the list, you can suspend, restart, or kill any rendering process.

5.4 TEXTPORT Alt Shift T



Opens a window which allows you to enter text-based Scripting commands. You can resize the Textport by dragging the edges of the window.

For more information on the Textport and the commands you can issue using it, see *Introduction* p. 79 of the *Scripting* section of the manual for more information.

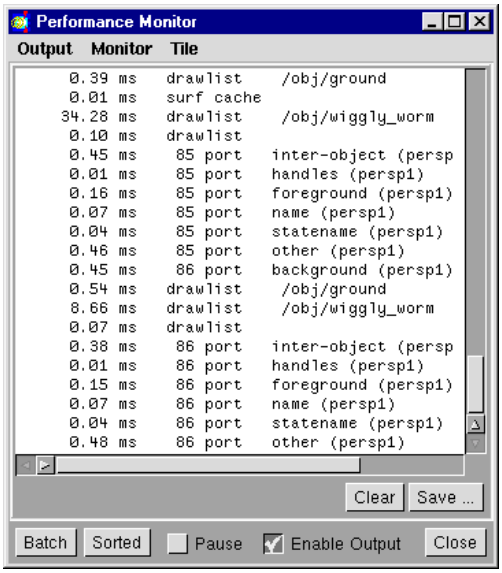
Note on Key Repeat: Due to event handling problems in the implementation of X Windows, Houdini turns the keyboard key repeat off while the cursor is inside one of the 3D Viewports and on again when the cursor leaves a Viewport. If you should find that the key repeat is off on your system, simply type the UNIX command: *xset r on* to turn the key repeat feature on again. This is unnecessary in Windows NT.

5.5 CODE VIEWER...

Drag and drop a VOP node onto the View Code window to compile and view the generated code. Click the *Refresh* button if you've altered anything.

This is the same as doing a *View Code* from a VOP's right-mouse ( ) menu.

5.6 PERFORMANCE MONITOR... Alt Y



The Performance Monitor dialog allows you to monitor and diagnose which OPs require the most processing, in order to streamline your networks. It does this in a number of ways:

- Each OP's Info pop-up – The Cook time and Cook counter of each OP can be displayed in the Info pop-up of each CHOP.
- Each OP's Tile in the Layout Area – The OPs can be made to flash red while they cook. This sometimes occurs so quickly, that the flash is hardly visible.
- Performance Dialog – The Cook time, Cook counter, Viewport display times and User interface compute times can be logged in the Performance Monitor dialog.

INFORMATION THAT IS LOGGED

The data that can be logged in the Performance dialog is:

- Order in which OPs are cooked
- OP Cook Time – Per-OP cook time, and the cook time of all the OPs it calls.
- OP Cook Count – A separate counter for each OP
  (this is indented for each level of recursion).
- The number of times the OP has been cooked
- Viewport Display Time / Count
  (Individual display time of each Object/SOP in the Viewport)
- Frame Length / Frequency

You can use this information to track which OPs use the most processing time, and to streamline networks to avoid costly operations in favour of OPs which are less taxing to the system in a given situation. This is also useful for tracking where slowdowns in processing occur in a Real-time environment.

OUTPUT MENU

Off	Disables text output of performance information.
In Window	Displays results of performance monitoring within the Performance dialog. Note: Sending log messages to the window is about four times slower than sending it to Standard Output.
Standard Output	Routes the text normally displayed in the Text field to *stdout* (in the shell from which Houdini was started).

MONITOR MENU

Determines what type of performance events will be monitored:

OP Cook	Generates a message whenever an OP is cooked and outputs the time and/or cook count.
Object Display	The length of time to display objects is logged.
Viewport Display	Generates a message whenever a Viewport is redrawn and outputs the time and/or redraw count.
Frame Length	Generates a message each time the frame bar is incremented and reports time between frames and/or playback rate. This is most apparent during simulations.
Errors	Error messages from OPs are logged.

TILE MENU

Statistics in OP Info	Causes timing to be calculated and stored for each OP. To view it, click the Info (i) pop-up on each OP tile – the timing will then be displayed there.
Highlight When Cooking	When enabled, causes each OP tile to turn orange as it cooks. useful for designing your network as to minimize cooks by placing animating geometry near the leaves of the network.

SINGLE FRAME CAPTURE

Allows messages between two consecutive frames to be caught, then turns off the enable *Enable* button, allowing you to examine all the events pertaining to a full frame of animation.

When enabled, measurements are only made between complete frame advances, so typically you'd turn on your objects of interest, turn on *Single Frame Capture* and *Enable* and then click *Play.* Some measurements will be reported, and then *Enable* will turn off by itself. Then you can optionally turn on *Enable* again. After the passage of one complete frame the *Enable* button will turn itself off again. This allows you to grab a snapshot of all the activity during that single frame.

PAUSE

Since the display of the log messages themselves take time, this option will pause the output, capture the events in memory, then unpause. The logged messages will quickly be cached into memory giving a more accurate timing description.

ENABLE OUTPUT

Enables / disables all performance monitoring.
If this is not checked, performance monitoring will not occur.

Note: Performance monitoring itself slightly decreases performance.

CLOSE

Closes the Performance dialog. Performance monitoring will still continue as long as the Enable button is still checked. You may want to route the Output to Standard Output – if you keep the Performance Tools enabled – before closing the dialog.

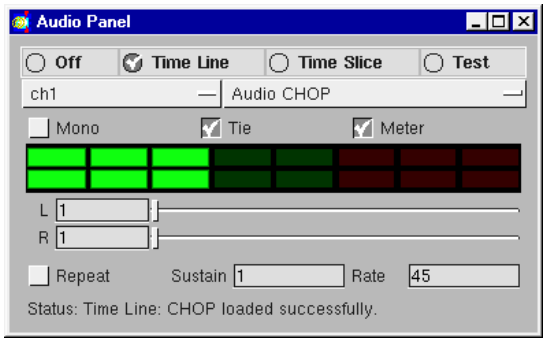
SCROLLING TEXT FIELD

Scrollable area where log messages are sent *(Output* menu must be set to *Window)*.

CLEAR / SAVE

Clears the display window. If you don't want to lose the info, save it first. If you click *Save*, it displays a File Save dialog to save contents of the Scrolling Text Field.

5.7 AUDIO PANEL... Alt I



The Audio Panel set the Volume for audio, and allows you choose which CHOP gets attached to the animation timeline.

HOUDINI AUDIO PANEL VS OPERATING SYSTEM AUDIO PANEL

The Audio Control Panel also sets the overall output level to the speakers. The Audio Panel in Houdini passes its audio through the Audio Panel in IRIX on SGI. The SGI Audio Panel is displayed through the *Toolchest > Desktop > Audio Control*

menu. The Houdini audio levels are reduced (multiplied) by the SGI audio panel levels.

THE AUDIO VU LEVEL METER

The Audio Panel VU meter, when CHOPs are in the range -1 to +1, is at 100% of the acceptable level before the sound amplitude is clipped, and displays as full green. When the meter is red, audio is still audible but the sounds are clipped.

AUDIO OFF / TIMELINE / TIMESLICE / TEST

Audio can be attached to the Timeline. In this mode, when the animation plays, the audio plays.

Off	Don't play audio.
Timeline	In this mode, you can scrub and play CHOP audio while the animation plays.
Time Slice	Plays a CHOP whose Time Slice option is ON (forwards only).
Test	If the Audio is in Test mode, it will play independently of the audio timeline.

SELECTING THE CHOPNET AND CHOP TO PLAY

The two menus select which CHOP you want to play. Changing these menus set the CHOP's Audio flag. Similarly, setting the Audio flag on any CHOP sets these two menus.

SCRUB CONTROLS

Houdini has sound scrubbing controls, which plays the audio slowly around the current frame. Its scrub decay parameter controls the decay of the sound when scrubbing stops. You can scrub backwards and forwards.

When scrubbing the audio, the "Rate" is the audio scrub rate, and defines the amount of time that will be looped when scrubbing. 45 means 1/45 second.

When stepping or scrubbing the play bar, the length of the interval is the Decay parameter in the Audio dialog. 30 means decay in 1/30 of a second.

5.8 SHELL...

Much the same as Textport, but opens a OS-level shell from which you can enter system-level commands. The difference between this and a regular shell window, is that the process is owned and spawned by Houdini instead of the system.

6 DESKTOPS MENU

6.1 LIST OF CURRENT DESKTOPS

The Desktops menu allows you to quickly customise the Houdini UI by allowing quick retrieval of preset arrangements of panes. Desk managament is covered in detail in: Desktops Alt D p. 79.

Desktops allow you to save a particular arrangement of panes as a set.

Select from this menu to choose a Desktop.

You create a new Desk with the *File > New Desktop* command.

An added desktop can be saved with its particular arrangement of panes for rapid recall by saving it with: *File > Save Desktop*. It can be recalled by selecting it from the *Desks* menu.

PANES AND DESKTOPS

See the beginning of this chapter (*Interface* p. 43) for information on configuring Panes and Desks.

7 HELP MENU

VERSION Alt Shift V

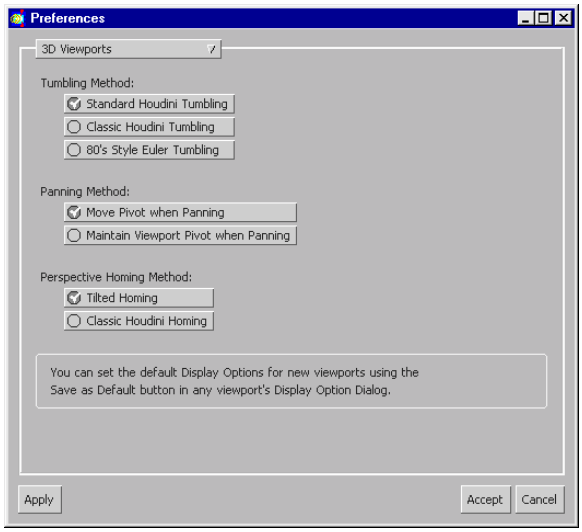
Displays the current version of the Editor in the Message Bar.

MANUAL ENTRIES

All the sections of this manual, and the Tutorial Guide are listed here. Select one to peruse its contents with the Acrobat reader.

Note: If you do not have the Acrobat Reader installed, refer to the *Getting Started > Online Help* section for info.

8 PREFERENCES



The Preferences dialog allows you to set various options for Houdini. These are split into several pages accessed from the menu at the top-right.

8.1 GENERAL USER INTERFACE PREFERENCES

ENABLE DOUBLE BUFFERING

Double buffering improves the quality of moving images on a 24 bit monitor. If Double Buffering is not being used, then your work is drawn directly to the screen without any synchronization with the monitor's refresh rate. This causes areas of the screen which are being frequently redrawn to be displayed while being only partially redrawn. This produces a "flickering" effect. Double-buffering eliminates this flicker.

When *Double Buffering* is on, the new image is drawn in an offscreen buffer in video ram and then swapped into the display almost instantaneously, thereby eliminating the display of half-drawn images, and the flicker.

Why not have it on all the time? Because some machines do not have sufficient video ram to perform double buffering in full colour. The colour resolution must be halved and then still-frame image quality is reduced due to dithering.

DRAW LINES SMOOTHLY

Determines whether lines drawn in the Viewport are anti-aliased.

COLOR SETTINGS

Allows you to specify the Houdini interface: *Gamma*, *Brightness*, *Contrast*, and *Saturation*. These settings will apply to the whole Houdini UI as a whole.

8.2 WORKSHEET TILES & TREES PREFERENCES

CENTER ZOOM ON MOUSE

Scrolls the Layout Area in the Network Editor so that the tile is right beneath the mouse cursor after placement.

USE TOOLBARS IN INSERT/APPEND MENUS

When enabled, the categories set in the Toolbars are used in the pop-up menus when adding a new OP from the input/output connector of an OP. If not enabled, then all the available OPs show up together in one huge listing.

CONNECT TILES WITH THICK LINES

Draws the lines connecting OP tiles with thicker lines.

PLACE NEW TILES IMMEDIATELY

Places tiles immediately upon selecting an OP, intstead of waiting for a second click to give the placment location of that new OP.

If the *Scroll to the New Tile* option is enabled, Houdini will adjust the display of the Network View so the newly placed tile is within view.

scroll to new tile

Adjusts the Layout Area to display newly placed tiles.

HIGHLIGHT THE CURRENTLY COOKING NODE

If enabled, then the tile that is being processed gets lit-up during when its is cooked.

INCLUDE PERFORMANCE STATISTICS IN INFO

Includes performance info in the pop-up info for the tile.

CHANGE SOP RENDER AND DISPLAY FLAGS SEPARATELY

When enabled, when you click on a SOP's render or display flags, both of them will be set at the same time. When disabled, they will change together.

DISPLAY ICONS IN OPERATOR TREE VIEWS BY DEFAULT

Will display OP tiles in a *Tree View* p. 51.

8.3 3D VIEWPORTS PREFERENCES

TUMBLING METHOD

Standard — The normal Houdini method of tumbling the display (using). It is such that the tumbled is attached to the mouse, and moving the mouse back to the same location on the screen results in the tumble returning the same result for that location. This is intuitive for most users.

Classic Houdini — With the classic method of tumbling, you move the mouse in small circular motions counter to the direction you want to turn. You will get a tumble-orientation that moves continuously counter to the direction in which you are moving the mouse. This actually provides more control over the tumble once you have got the knack of it.

80's Style Euler — Provides the option for 80's Style Euler tumbling – for those who prefer it.

PANNING METHOD

Move Pivot when Panning — The pivot about which the Viewport swivels will always be the centre of the Viewport – even if you've panned the geometry off to the side.

Maintain Viewport Pivot when Panning — The pivot about which the view swivels will be maintained around the geometry in question - even when it has been panned away from the centre of the Viewport.

PERSPECTIVE HOMING METHOD

Tilted Homing — Slightly tilts the view when homing (the Construction Plane must be on).

Classic Houdini Homing — Old Houdini 4 style homing (down the specified axis).

8.4 TOOL MENU PREFERENCES

SHOW DEFAULT TOOL CATEGORIES

Displays the default tools supplied with Houdini. Usually, this shoud be enabled – which causes Houdini to display all available toolbars in the Tab popup menu in the Viewport.

SHOW CUSTOM TOOL CATEGORIES

Whether to show user-created Custom toolbars – created with *Tools > Toolbar Manager.*

SHOW VIEWPORT STATE HISTORY

When you type Tab in the Viewport, and this is enabled, the menu will have a list of the last several used items appended to it.

TOOL HISTORY LENGTH

Allows you to specify how many tools to remember in the history.

8.5 ANIMATION PREFERENCES

KEYING

add keyframes on parameter change

When enabled, parameters changes are immediately processed (i.e. “cooked”), and the changes reflected in the geometry displayed in the Viewport.

set additional first key at frame

Will set an extra keyframe at the specified frame (typically frame 1) when a keyframe is added. If left blank, no extra keyframes will be added.

If set to ’1’ – it will approximate the behaviour of older (pre H6) versions of Houdini, which would automatically add a key at the first frame.

global set key

Set Keys on Pending Channels
: Set keys on channels where changes have been made.

Set Keys on All Scoped Channels
: Sets keys on all channels which are scoped.

INITIAL SEGMENT

default / rotation function

You can set the default segment interpolation function to be of the type you specify here. e.g. *cubic()* . You can also set the default function used for rotations.

ADDED SEGMENTS

auto function

Attempt to automatically determine the best function to interpolate between added keyframes. When disabled, adding keyframes inserts segments of the type specified here.

auto slope

Attempt to automatically match incoming / outgoing slopes on channels. When disabled, adding keyframes will set the new segment slopes to the specified value.

hold last key

When adding keyframes in the last segment of a channel, the final segment is modified to hold it at the same value between the last keyframe and the end of the animation. Enabling this parameter allows you to insert keyframes repeatedly without changing the interpolation of the last segment.

8.6 HANDLES PREFERENCES

HIDE LOCKED HANDLE PARTS

Keeps locked handle parts from being displayed.
You will need to reload the file to take effect for existing handles)

OPEN CHANNEL EDITOR WHEN SCOPING

When scoping from handles, it will open up a new Channel Editor (if one isn't already open).

SCOPE CHANNELS WHEN SETTING KEYFRAMES

Depending on the setting, it will not alter what is displayed *(Do Not Scope)*, *Add to the Scope* (in addition to whatever channels are already being displayed), or *Replace* whatever channels are scoped when adding keyframes to handles.

8.7 OBJECTS AND GEOMETRY PREFERENCES

HIGHLIGHT OBJECT SELECTIONS IN VIEWPORT

Provides visual feedback when objects are selected in the Viewport.

SCOPE CHANNELS

When objects are picked in the Viewport, its channels will be scoped.

open channel editor if closed

Opens a Channel Editor if necessary.

VIEW OVERRIDDEN PARAMETERS IN CHOP VIEWER

The channels which are overriden by CHOP control are displayed instead.

OBJECT-LEVEL SECURE SELECTION ON BY DEFAULT

Enables Object-level secure selection by default - when enabled, it prevents you from losing your selection with a stray click.

SOP-LEVEL SECURE SELECTION ON BY DEFAULT

Enables SOP-level secure selection by default - when enabled, it prevents you from losing your selection with a stray click.

SELECT ENTIRE SUBNETS ON BY DEFAULT

When enabled, the default behaviour will be to select the subnets when you select the containing node.

HIGHLIGHT SELECTED GEOMETRY IN VIEWPORT

When an object is selected, Houdini will highlight the corresponding object in the Viewport.

KEEP POSITION WHEN PARENTING

Never	When set to *Never*, objects will not maintain their world position when you parent objects together, rather, they will assume a space relative to the parent object.
Always	When set to *Always*, objects will maintain their world position and orientation even if you rewire an object node to set its parent (this is done by changing the object's Transform/Rotate/Scale parameters).
Use Object Flag	This setting will maintain an object's world position

according to the *Keep Position When Parenting* parameter of the individual object.

8.8 PARTICLES PREFERENCES

SKIP PARTICLE INFO GENERATION (FASTER)

Bypasses the generation of info in the POP tiles' info pop-up (click tile with). This speeds up the particle simulation, at the expense of generating more informative feedback.

8.9 MOTION AND AUDIO PREFERENCES

For information on Time Slicing, see: *CHOP*s > *Time Slicing* p. 298.

MAXIMUM TIME SLICE SIZE (FRAMES)

This option allows you to limit the size of a Time Slice, expressed in frames. The default maximum is 60 frames. This is useful if you only need a few frames of history, and the Playbar is jumping ahead by large intervals. If a Time Slice is larger than this maximum size, it will be clipped from the current frame backwards (causing a gap between this slice and the previous one). The slice will always end on the current frame.

ENABLE TIME SLICE FLAG WHEN CREATING CHOP

Sets the default behaviour for when you place a CHOP in the Layout Area.

All CHOPs with Time Slice capability created while this option is on will have their Time Slice flags enabled by default. Otherwise, all CHOPs are created with the Time Slice flag turned off. This is a convenient option to leave on when creating and working with large Time Sliced networks. This option is off by default.

MINIMAL TIME SLICE COOKS

Some Time Sliced CHOPs can minimize their cooking, rather than cooking every frame. When cooking is unnecessary, or it produces the same output, these CHOPs will stop cooking until an input changes.

CHOPs that have this feature are:

- Copy CHOP
- Count CHOP
- Envelope CHOP
- Lag CHOP
- Spring CHOP
- Trigger CHOP

Minimal Cooking will not apply to these Time Sliced CHOPs if they have non-constant CHOP inputs or time dependent CHOP inputs. This option is off by default. It

can speed up puppeteering and CHOP networks that respond to isolated inputs or events.

8.10 COMPOSITING PREFERENCES

CACHE PAGE

This page contains all the preferences associated with the global compositing cook cache. All COPs in all COPNets share the same global cache.

cache size

The size of the tile cache, in Mb. If you use the compositor constantly, you should set this to a fairly large value (half the available memory for systems with more than 256Mb).

cache compression

If memory is short, you may want to apply lossless compression to the tiles before caching them. This produces a slight to moderate performance hit (fast to best, respectively). Faster processors and multi-processer machines may benefit from compression if large images are being processed.

tile size

The size of the tiles. The image is divided into these tiles and processed in chunks. If you are always working at a specific resolution, you may want to specify a tile size that exactly divides the horizontal and vertical resolution. Changing this value will flush all the caches. A tile must have at least 1024 pixels in it.

resolution limit

Allows you to specify the max. size an image in the compositor is allowed to be.

use performance statistics to influence caching

If enabled, the cook times of tiles are recorded. Tiles that took longer to cook are held in the cache with a higher priority than quickly cooked tiles.

clear cache

Clears all the compositing caches.

COOKING PAGE

This page contains all the preferences which relate to COP cooking, and any updating that happens during normal or interactive cooking.

use collapsable pixel operators

If enabled, collapsable pixel operators in series (blue COPs) will be combined into one operation. If off, they are all processed individually.

multi-threaded cooking

If enabled, multi-threaded cooking is enabled. Otherwise, only 1 thread is used to cook images (the main Houdini process).

number of threads

Specifies the number of threads to use when cooking. If set to 1, only the main Houdini thread cooks images (as if Multi-threaded cooking was off). To saturate a multiprocessing computer, use twice as many cooking threads as processors.

use fast interactive cooking

If enabled, the resolution of images is automatically down-scaled to a smaller resolution while an interactive edit is in progress (dragging a handle, sliding a slider or XCF slider). This gives much quicker feedback at the expense of quality. If off, the image will be cooked at normal resolution.

only use fast cooking on slow operations

If enabled, fast operations (like Bright, Gamma and Timing COPs) do not cause the interactive down-scaling of images. If off, all operations down-scale the image during user interaction.

use fast cooking on images larger than

Sets a lower resolution limit on the images that will be down-scaled during user interaction. Images smaller than this limit will not be down-scaled. In the case that the image doesn't match the aspect ratio of this perference, the number of pixels in the images is compared and used as the limit.

interactive rendering fraction

Specifies the down-scale factor. Smaller factors give much faster updates during interaction at the expense of image quality.

cooking feedback

Allows you to set the type of feedback which you receive while a single image is cooking:

None	Nothing updates until the image is finished cooking.
Progress Bar	A progress bar at the top of the image viewer updates periodically while cooking the image.
Image Update	The viewport containing the image being cooked updates periodically, showing the actual progress of the cook. This has a fairly minimal performance hit.
Both	Both the progress bar and the viewport update while cooking. This is useful when in graph mode, where there is no image to update – the progress bar is visible to inform you of the cooking progress.

image update style

Allows you to specify if images should be updated in *Strips* or in a *Spiral* fashion.

PROXY PAGE

This page contains preferences relating to the proxy disk cache capability of the compositor. COP Tiles with the Disk Cache button depressed automatically write proxies to disk.

enable proxy generation

If enabled, any COP with the *Disk Cache* button depressed will write its images to disk. This can be used to globally disable all the Disk Cache COPs without needing to turn them all off individually.

proxy directory

The root directory where all proxies will be kept.

limit proxy disk usage to

If enabled, the disk usage of the proxy directory is kept under the limit specified.

clear all proxies

Removes all proxies and cleans up the proxy directory.

SELECTORS PAGE

This page deals with the default behaviours of the COP selector (selectors are used when adding COPs from the viewport).

never prompt for optional inputs

If enabled, once the minimum number of sources are connected, input selection is considered to be complete. If an *Always Place* option is specified, the COP will be immediately placed in its selected Viewport.

never prompt for a generator's input(s)

If enabled, inputs for generators are never prompted.

always place the new cop in the current viewport

If enabled, once selection of all inputs is complete the COP will be placed in the current viewport without prompting for the destination viewport.

always place new generators in a new viewport

If enabled, generators will be added to a new viewport, if available (otherwise the current Viewport is used). If off, you will be prompted for a destination viewport.

NAMES PAGE

This page allows you to customize the names of common plane types. The default names are the first letter of the English plane name, capitalized. These are the names that are used in the scope fields, plane lists, VEX scripts and output Houdini .pic files.

COLORS PAGE

COPs can be coloured by family type. Any new COP in that family will have its tile coloured to the family colour for easy identification.

enable cop auto coloring

Enables or disables this feature for all COP types. You can also individually disable family types that are not important to you.

generators

The colour for all generators. Because generators can take an input and generate planes 'inline' (in the middle of a network), this helps you spot where in the network planes may be created or masked.

pixel functions

The colour for all Collapsable pixel operations. Because collapsable pixel operations only work when grouped together in a chain, this helps you optimize your network for these types of operations.

timing

The colour for all timing-related operations and non-image related operations (like merge and delete). These operations normally do not cook images or use any cache memory. This helps you track down timing changes in a large network.

vex

The colour for all VEX-based operations. VEX COPs have the special characteristic that all plane names matching the VEX script's parameter names override those parameters on a per-pixel basis, so it is important to recognize these COPs.

DEFAULTS PAGE

This page defines the default values for some of the image and sequence page parmeters for Generators. This allows you to customize the compositor towards Film or Video work (by specifying the resolution and frame rate), or to a specific data type (for example, always compositing in Floating Point by default).

CINEON PAGE

This page defines the default values for all Cineon parameters in all new File COPs and Compositor output drivers. It also defines the cineon parameters that are used when a Cineon file is loaded or saved outside these two contexts (like from the Save Image... COP tile menu). Cineon files have an extension of .cin.

convert from 10 bit log

If enabled, the Cineon file is converted into linear space. Otherwise it remains in its 10 bit log space (with values from 0 - 1024).

lut file

The *LookUp Table* file containing the 10bit cineon conversion (.lut).
This file may be created manually or with the standalone *icineon* or *ilut* tools.

white point

The Cineon white point, which defines the Cineon value (0-1023) which represents 90% image intensity. This parameter is not used if a LUT file is specified.

film gamma

The Cineon film gamma, which defines the gamma to use in the conversion from Cineon space to linear image space (the space which the compositor operates in). This parameter is not used if a LUT file is specified.

image white point

The linear space value that represents 100% intensity. The default is 4095, which allows the entire cineon range to fit into a 16 bit image when the Cineon White Point is 685 and the Film Gamma is 0.6. This parameter not used if a LUT file is specified.

8.11 DESKTOPS AND TOOLBARS PREFERENCES

SAVE DESKTOPS AUTOMATICALLY

This causes changes to Panes within Desktops to be saved automatically.

SAVE TOOLBARS AUTOMATICALLY

This causes changes to Panes within Desktops to be saved automatically.

SHOW HIDDEN FILES

Shows files that are marked as hidden (starts with a '.') anyways.

SHOW ALL FILES

Shows all files, regardless of what type they are.

PATH MENU BEHAVIOUR

Prompt User Each Time	Asks the user for the path.
Copy files	Copies the files.
Move files	Moves the files.

8.12 MISCELLANEOUS PREFERENCES

SAVE METHOD

override filename

Will overwrite the existing file you opened with the present state of the hip file.

increment filename

If enabled, and the name of your file contains a number, then the number will be bumped up by one every time you save the file. This does not happen with *Save As*, only with *Save* (also see the *-i* option on the *mwrite* command in the *Scripting* section).

make numbered backup

With this setting, the filename of the .hip file will remain the same, but backup files will be created which are incrementally numbered. The default backup directory is: *./backup* and the backed up files will have a *_bak{num}* suffix.

SAVE VIEW COMMANDS FOR

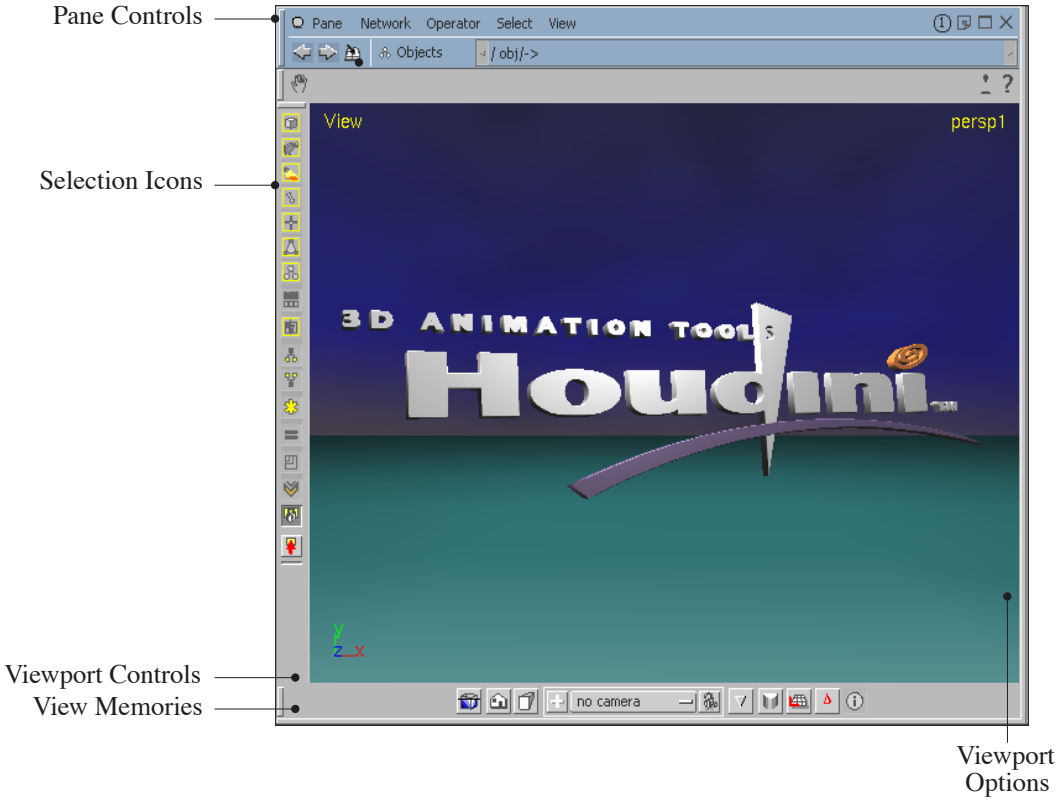
Allows you to specify if view commands are saved for: i) *All Viewers*, ii) *Existing Viewers*, or iii) *Open Viewers*.

RESERVE __ KB FOR THE UNDO SYSTEM

Specifies how much memory should be used for Houdini's Undo system – the more memory you allocate, the more Houdidi can Undo.

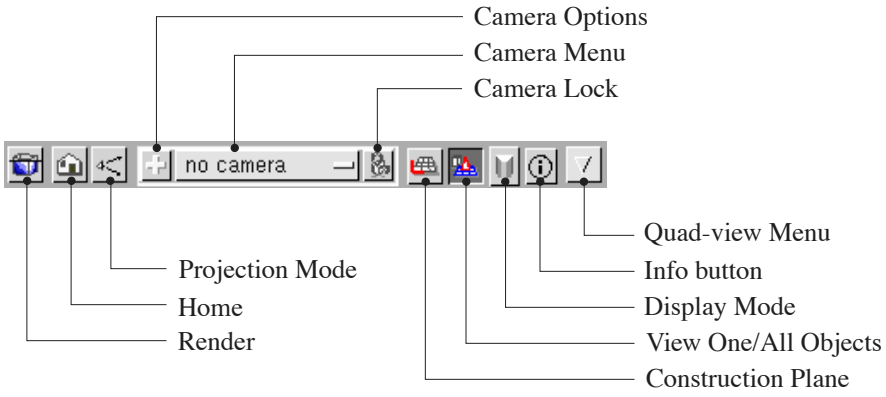
3 3D-Viewers

1 TYPICAL 3D VIEWPORT



When the Editor menu is set to Viewer, and the Type menu is set to an approriate type (Objects, SOPs, etc.), then the Viewer will display a 3D-Viewer. The 3D-Viewer is a window onto your scene allowing you to see and manipulate 3D geometry. There are both *Viewport Controls and Options* (for adjusting the display of 3D data), and OPerations – which allow you to manipulate items within the Viewport. These are discussed in the following sections.

2 VIEWPORT CONTROLS



There are two sets of Viewport controls. The first (default) set is specific to viewing and manipulating your objects in the Viewport.

The second set allows you to store and recall views. To access the second set of controls, unstow the *State Memories* at the bottom of the Viewport.

2.1 VIEW CONTROLS

TUMBLE, DOLLY, ZOOM, AND PAN

These functions are achieved directly in the Viewport with the three mouse buttons as follows:

🖱	Tumble
🖱	Dolly
🖱	Pan
Alt 🖱	Zoom
Ctrl 🖱	Rotate View

Tip: Use the Space key in conjunction with the above mouse buttons to quickly enter the View state at any time, no matter whatever else you may be doing.

tumble

Dragging with the 🖱 mouse button tumbles the view around the center of the home view. By default the home view lies in Z, so, dragging the mouse horizontally revolves the view about the Y axis, and dragging vertically revolves the view about the X axis. This changes if you've homed your view to other than the default Z axis.

The Ctrl button constrains the tumble to a rotation about your view normal (Z by default). To rotate about a different axis, home your view to that axis first.

dolly

Dragging with the mouse button causes the camera through which you view the 3D geometry to be moved closer or farther away from the geometry. This button is disabled if the *Projection Mode* is set to orthographic.

pan

Dragging with the mouse button causes the camera to pan horizontally and vertically on the 3D scene.

zoom

Dragging with the Alt causes the camera through which you view the 3D geometry to zoom in or out as with a zoom lens.

rotate

Dragging with Ctrl will rotate the view.

BOX ZOOM

Dragging with Ctrl from left to right draws a rectangle within the Viewport window. When the mouse button is released, the area within the rectangle will zoom to fill the entire Viewport.

If you drag your mouse from right to left, the Viewport box zooms out such that the entire area of the Viewport is shrunk to fill the rectangle you drew.

2.2 RENDER

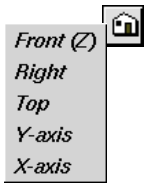
Clicking here displays a menu of output drivers, or renderers, that are capable of rendering from the Viewport. Selecting one renders whatever is currently displayed in the Viewport and shows the finished result in *mplay*. Additional output drivers can be created in the Output Editor.

- The image will always render to the screen.
- Only one image will be rendered (even if a sequence is specified in the Driver).
- For the Driver to work correctly, the camera for the driver must exist.

To setup your renders for this menu, use the *Types* menu to go to *Outputs*, and setup your render there. See the *10-Outputs* section for details.

Note: To change the default behaviour of *fast_mantra* nor *mantra1*, you must modify the HOUDINI_VIEW_MANTRA environment variable.

2.3 HOME

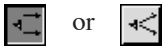


Resets the Viewport settings to the home view which, by default, looks down the negative Z-axis.

By clicking with the middle or right mouse (or), or holding down the Alt key and clicking on this icon, a menu appears with the items: *Front (Z)*, *Right*, *Top, Y-Axis,* and *X-Axis*. Selecting one of these items resets the home view to view from the selected axis or side. Subsequent clicks on the *Home* icon bring you to the home view you last selected.

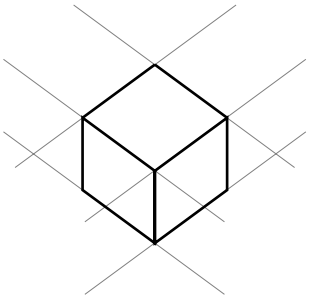
A Note about Homing: Homing brings things fully into view, but at the same time, it also rotates them so that we look at them straight down a given axis. To bring things fully into view without rotating them, you should use one of the *Viewing* menu Frame commands.

2.4 PROJECTION MODE: ORTHOGRAPHIC / PERSPECTIVE

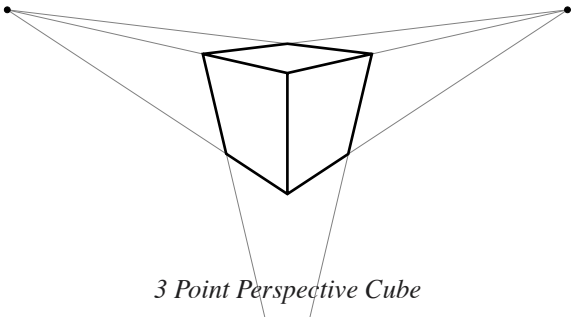


This button is a toggle – clicking once sets you into orthographic mode; clicking again puts you back into perspective mode. Switching to orthographic mode disables the Camera Dolly icon because dollying in orthographic mode has no effect on the view.

The difference between an Orthographic and a Perspective view is that an Orthographic view has no vanishing points – its lines extend infinitely into the distance and remain parallel to each other. With Perspective, objects have vanishing points, as illustrated below.



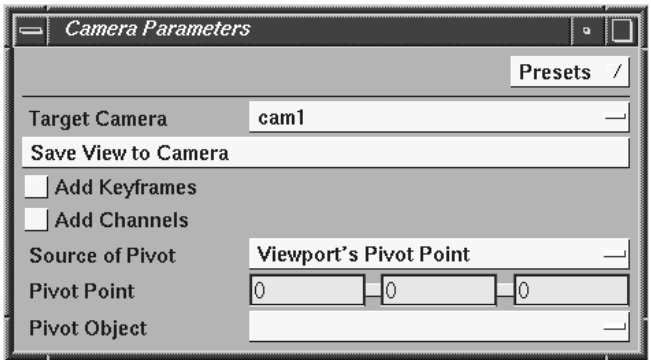
Orthographic Cube



3 Point Perspective Cube

Tip: Use the O key to toggle the projection (if you are in the View state).

2.5 CAMERA OPTIONS DIALOG



TARGET CAMERA

Select a perspective object to which the current view should be assigned. This allows you to find a perspective on a scene that is appropriate for your work, without having to manoeuvre a camera or light to that position manually.

SAVE VIEW TO CAMERA

Repositions the *Target Camera* to the current view in the Viewport.
When you save the view, the following parameters are changed accordingly:

- Translate, Rotate, and Pivot values for the X, Y, and Z axes
- Aperture
- Focal
- Projection
- Orthographic width
- Window X and Y (the centre of the camera's window)
- Window size
- Near
- Far
- Track camera window parameters

Note: You will not be able to *Save the View to Camera* if: you are using a Rotation handle; the cam / light is parented to other objects; the object being transformed has a parameter that affects its orientation (i.e. a *Path* or *Lookat* object has been set); or if its channels are being over-ridden by an expression or CHOPs.

ADD KEYFRAMES

If this option is enabled, keyframes are added automatically for the perspective object's motion. This eliminates the need to use the Playbar.

ADD CHANNELS

If the motion of the perspective object is not animated, this option adds the requisite channels for that object's motion.

Note: When *Add Channels* is employed, the following parameters are animated: *Translate*, *Rotate*, *Pivot*, *Aperture*, *Focal*, and *Orthowidth*. The */win* and */winsize* parameters are also animated if you employ the Viewport's box-zoom feature.

SOURCE OF PIVOT

Determines the pivot point of the camera. While it doesn't affect the position and orientation of the camera when saving the view, it does affect how the orientation changes when you make subsequent changes to the camera's rotation parameters. To understand this better, set the source of the pivot in the standard session to *geo1*, save your view, and then rotate the view. Now try the same with the pivot set to *light1*. Notice the difference in behaviour.

viewport's pivot point

Employs the Viewport's pivot point. If you rotate the view with the Viewport controls, it tumbles around the Viewport's pivot point. That pivot point is the center of the last *Home* operation.

camera's pivot point

Uses the camera object's pivot point, which is defined in the camera object's pivot parameters.

specified point

A point in world-space may be explicitly specified.

specified object's origin

You can specify that the pivot point be set to the origin of any object in the scene.

PIVOT POINT

Allows you to specify a position in world space about which to pivot. These x, y, and z edit fields are only available if *Specified Point* is selected for the *Source of Pivot* parameter.

PIVOT OBJECT

Pivot Object is enabled if *Specified Object's Origin* is selected in the *Source of Pivot* parameter. It allows you to specify any object in the scene as the pivot point.

CAMERA SELECTION MENU

This menu lists the available lights and cameras in your scene. Select a camera through which to view your scene here. If No Camera is selected, the scene will be viewed using the Viewport controls without reference to the setings of any particular Camera. Choosing a camera changes the Viewport to display the scene from that object's location.

The view parameters of an Object Editor's Viewport can be tied to those of a light or camera by selecting it from the Camera menu. This has the effect of making the Viewport "look through" that camera or light. You may want to look through a camera in order to preview your animation as it will be rendered, or you may want to look through a light to view its area of illumination. The following viewing parameters are used in this process:

- Transform
- Focal
- Zoom
- Near/ Far
- Orthographic
- Window X, Y
- Window Size
- Track Camera Window Channels
- Roll
- Pivot X, Y
- Aspect Ratio
- COP Net
- COP Name
- COP Frame

For discussions of these parameters, please refer to the Camera and Light Object sections below.

When viewing through a camera or light, you can still use the Viewport's viewing controls to manipulate the view temporarily. The view will snap back to that of the Light or Camera if you do one of the following:

- Change the camera or light objects from which the scene is viewed
- Change the time
- Change a parameter in any object

no camera

When *no camera* is selected, you can view the 3D scene independent of any specific camera positioning.

CAMERA LOCK

Clicking this icon locks the Viewport camera so its controls can't be indiscriminately adjusted. Clicking again allows normal adjustment of the camera view.

If enabled, locking performs a *Save View to Camera* operation (see above) using the settings defined in the View state's parameters dialog box. The only parameter that is overridden is the target camera, which is set to the view camera.

2.6 SHOW / HIDE CONSTRUCTION PLANE

Shows / hides the Construction Plane. Showing the Consruction Plane allows you to draw in reference to a Construction-plane along any plane in the Viewport. See: *Geometry > Construction Plane* p. 404 for some basics, and *Construction Plane Operation* - p. 507 for in-depth state infromation.

2.7 VIEW ONE/ALL OBJECTS

When in Model Mode (i.e. whilst viewing SOPs), this allows you to set whether you want to see your model in-context together with all scene geometry, or you can view only the geometry of the object you're working on, and eliminate the display of all the extraneous objects.

2.8 SHADING MODE

Use the W key to toggle the display shading mode between shaded and wireframe. The pop-up menu lets you set the *Shading Mode* between:

- Wireframe
- Hidden Line Invisible
- Hidden Line Ghost
- Flat Shaded
- Flat Wire Shaded
- Smooth Shaded
- Smooth Wire Shaded
- VEX Shaded
- VEX Wire Shaded

Note: When in any of the Shaded (Gouraud) modes, Z-buffering is always enabled.

While the Viewport or geometry is being moved, Gouraud mode may temporarily switch back to wireframe mode if the *Move Wireframe* button is checked in the *Viewport Display Options* dialog – this dialog is discussed in the next section.

Note: In Gouraud mode, you are limited to eight different light sources in a scene (this is an Open GL limitation, as defined in: *gl.h: #define MAXLIGHTS 8*).

Tip: If you want a Hidden Line Wireframe output, try using a Scene Render, with the *Renderer* set to *GL-Hidden Line* as in *Outputs > OpenGL Wireframe Output OP*.

2.9 INFO BUTTON

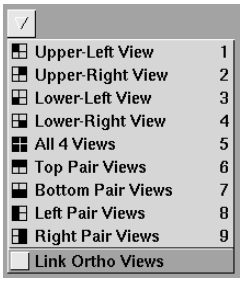
This button displays information about the geometry within the Viewport, such as how many of each type of geometry, bounding box information, and memory usage. It is disabled if there is no specific context for which to display the info.

2.10 QUAD VIEWS

Viewports can be split into quad-views. This gives you the option of viewing geometry through a single, perspective, Viewport or through three orthographic, and one perspective, Viewports. The orthographic views show the scene from the Top, Front, and Right, respectively. Though the individual Viewports are designed to function in the same way as a single, dominant, Viewport, there are some behaviours that change with the four views:

- Mouse and keyboard events are confined to the Viewport your cursor is in.
- The views share the same Construction Plane information.
- Selecting a camera while in an orthographic Viewport automatically changes the Viewport to a perspective view.
- The single toolbar at the bottom of the Viewport area will only affect the selected (yellow outline) Viewport.
- Some toolbar options are not available for orthographic views.
- View memories are global; so if you copy one view's memories into another view, the type of view will carry over as well.
- The Model mode allows you to link the orthographic views, causing changes in one orthographic view to affect the others at the same time.

SETTING QUAD VIEWS GRAPHICALLY



You can get quad-views in any 3D Viewer, just by selecting the type of quad-view you want from the pop-up menu in the bottom-right corner of the Viewport.

link ortho views

When enabled, individual quad-views will be linked together such that they will move in tandem where possible. Panning in one view will update other ortho views appropriately.

SCRIPTING ALTERNATIVES

The Quad View setup can also be changed using the *viewlayout* and *viewtype* scripting commands. These commands differ in that *viewlayout* changes the *number* of window panes, whereas *viewtype* changes the *type* of window panes.

Some of the most common setups include the following. For other options, see the *viewlayout* and *viewtype* commands in the *Scripting* section > *Interface Related Commands* p. 165.

example – split main viewport into quad views

```
viewlayout -q Build.*.world
```

This splits the the Viewport in the "Build" desktop into a quad-view.

example – adjust to a double view layout

```
viewlayout -d h 1 3 Build.pane1.world
```

Splits the the main Viewport in the "Build" desktop's pane1 to use views 1 and 3 in a vertical double-view layout.

The numbers 1 and 3 denote which of the four viewing panes to use:

1	2
3	4

See the Scripting section for more *viewlayout* and *viewtype* commands.

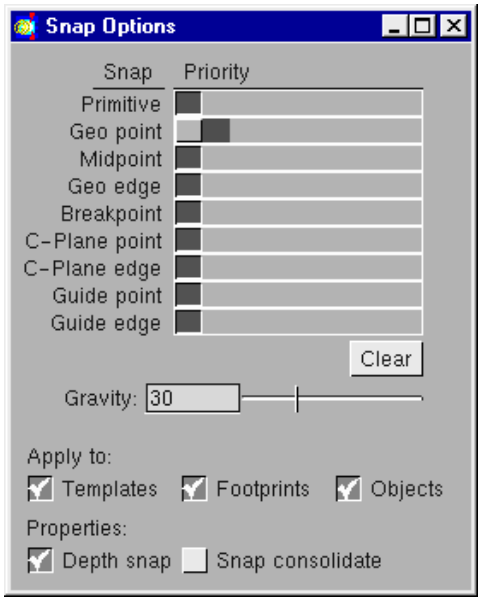
2.11 SNAP & SNAP OPTIONS...

Located at the top of the Viewport, beside the State parameter icons, this icon allows you to snap the cursor as specified in the *Snap Options* dialog. Click on the icon for a menu which allows you to enbable snapping, and access the options dialog.

Snapping allows you to precisely locate elements in relation to each other by locking the cursor onto them as you move. Use Ctrl J to toggle snapping on and off, and Ctrl K to bring up the Snap Options dialog.

SNAP OPTIONS

Brings up the *Snap Options* dialog. Move the sliders to assign a snapping priority to each type of element. The cursor will snap to elements with a higher priority before it will snap to an element of a lower priority. For information on the types of elements you can snap to, see *Geometry Types*.



When snapping at the world level (i.e. Object Space), Snap to Template, Guide Geometry, and Construction Plane will have no effect.

snap priority

Moving the thumb left decreases that element's priority, moving it right increases its priority. If the thumb is sunk into the left gutter, snapping for that element is off.

Note: Primitive snapping does not include metaballs, but works on their hulls. Also, in general, point-snapping should have a higher priority than edge snapping.

gravity

How many pixels you have to be from an element before it will attract.

apply to

Specifies if you want Snapping to snap to Templates, Footprints, and Objects.

depth snap

When checked, the cursor will snap to the XYZ location of the target geometry (point, edge, primitive, etc.).

When off, it will only match the coordinate of a snap point (as projected onto the Construction Plane), but leave the Construction Plane elevation untouched. The "off" case allows you to snap to any geometry that does not lie on the Construction Plane without actually leaving the Construction Plane surface.

snap consolidate

This option allows you to build primitives with shared points. For example, when creating a new polygon or curve, an added vertex will use an existing geometry point if you have snapped to it, and if Snap Consolidate in *on*.

clear

Resets the snap settings.

gravity

Gravity is the minimum distance you must be from an object for the cursor to snap to it. It is a radius, and is measured in pixels.

3 VIEW MEMORIES

There are nine memory buttons which allow you to store and recall the current View, Construction Plane, and Transform Jack settings.

To store the current setting into a memory, click the *M* button, then click on the numerical memory button into which you want it stored. To recall a stored Viewport setting, click the desired memory button. To return to a previous memory setting, click on the *P* button.

3.1 SELECTION MENU

The pop-up menu on the far left allows you to select the Viewport feature you want to store in the memory buttons. For example, you can set memory positions for both the Construction Plane positions and the View settings. The selection varies depending on which editor type you happen to be in.

The *view/c-plane/jack* memory buttons in the Viewports are shared between Viewports. So if you save a view in one Viewport, you can retrieve it in another.

3.2 PRESET AND NAMED SETTINGS

You are not limited to nine position memory settings. To the left of the *P* button is a menu that contains a list of named memory settings. If you click on the *M* button, the menu is replaced with an edit field where you can type a name for the setting you wish to store. Once you type enter, the setting will be stored away, and the edit field will be replaced with the menu. Named settings can be stored over and above the nine settings accessed through the preset buttons. You can access the named position memories by selecting them from the menu.

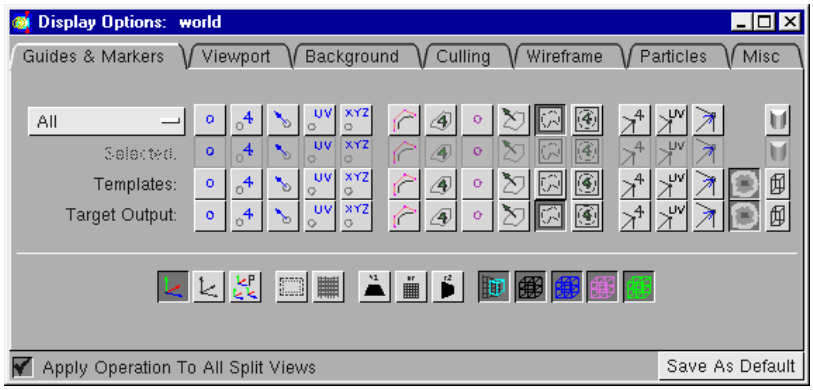
Note: If you aren't seeing the position memories of the object you want, ensure the Selection menu is set to the correct mode.

4 VIEWPORT OPTIONS DIALOG

The most commonly used icons from this dialog are displayed along the right-edge of the Viewport. Use these to quickly set the most common Viewport Options.

However, the full set of options can be explicitly called up in the Viewport Options dialog by clicking on the ✣ button located at the bottom of these icons, or by typing D within the Viewport. The Display Options dialog provides you with more options than the icons displayed along the right side of the Viewport, including points, point numbers, and normals.

4.1 GUIDES & MARKERS PAGE



The name of the Viewport under consideration is displayed in the Dialog's titlebar.

GUIDES & MARKER ICONS

These icons allow you to choose whether auxiliary display options are shown. These include those items in the Viewport which provide additional feedback about the geometry in addition to the geometry itself.

The multiple sets of similar icons let you select whether to display the icon's option for just the selected geometry, or for all geometry, on an item by item basis.

ALL / NON-SELECTED

When set to *All*, then auxiliary geometry gets displayed for all geometry, whether it is selected or not. When set to *Non-selected*, only the geometry that *isn't* selected will have auxiliary info displayed.

SELECTED

When *Selected* icons are enabled, only the geometry that is selected has auxiliary geometry info displayed in the Viewport.

TEMPLATES / TARGET OUTPUT

These icons allow you to determine if the particular auxiliary displays should be enabled for Templated objects, and for the Target Output.

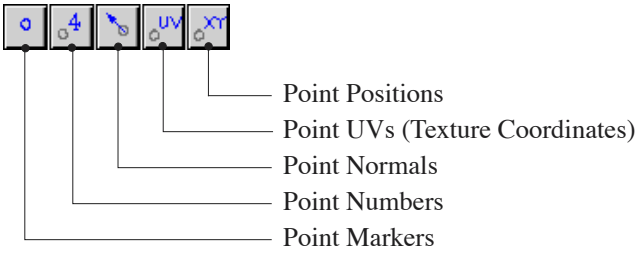
APPLY OPERATION TO ALL SPLIT VIEWS

Normally, a change in the Display Options affects only the current view. If this option is enabled, the changes will also affect the other Views in the Viewport.

SAVE AS DEFAULT

Saves the changes you make to be the default Display Options.

POINTS DISPLAY



point markers

When enabled, it displays unselected geometry points in blue, and selected geometry points in yellow. Geometry points can include: one point per primitive (e.g. ellipse, metaball), and one point per polygon node.

The reason both point and vertex displays are available, is that a point can be shared between two or more primitives, while vertices are unique. See *What is the Difference between Points and Vertices?* p. 224 in the *Geometry Types* section..

point numbers

When enabled, it displays the number of each point in blue, starting at zero. Selected points have their numbers rendered in cyan.

point normals

When enabled, if the point has a normal attribute, it displays the point normal of each geometry point in cyan (for the selected normals). A normal specifies the direction a point or surface “faces”. For more information on what a point normal is, see *Normals* p. 233 in the *Geometry Types* section. To change existing point normals, use the Point SOP.

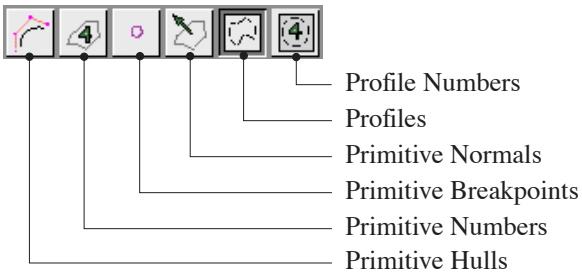
point uv's (texture coordinates)

When enabled, it displays the UV's for each point in cyan (for the selected UVs). The UV's are the texture coordinates for a point, which among other things, determine the placing of texture maps on the primitives that use that point. See the section *Point UV's (Texture Coordinates)* p. 235 for more information.

point positions (xyz)

When selected, this option displays the X, Y, and Z coordinates for each point. Using this display is not advisable for very complex geometry as the display becomes cluttered with the display of these coordinates.

PRIMITIVES DISPLAY



primitive hulls display

When enabled, this displays the hulls of NURBS, Bézier surfaces, curves, and metaballs. The hulls of any geometry that are selected are displayed in light brown. This option is useful when there are a large number of such objects that fill the screen. Turning it on reduces the visual clutter, and speeds up the display. For information on hulls, see *Splines* p. 238 in the *Geometry Types* section.

primitive numbers

When enabled, this displays the number of each primitive in violet, starting at zero. Primitives include: ellipses, metaballs, tubes, meshes, particle systems, NURBS, and polygons. See the section *The Primitive List* p. 225 in the *Geometry Types* section for a complete list of primitives.

primitive breakpoints

Displays spline breakpoints (also known as "edit points"). This allows you to build very clean skinned surfaces with very few isoparms that interpolate the given cross-sections perfectly.

primitive normals

When enabled, it displays the normal for each primitive in pink (if one exists – some primitives such as spheres, cylinders, and metaballs cannot have normals). A normal is a directional vector associated with a given primitive which is perpendicular to the point of contact with that entity. The numbers of selected primitives are shown in pink.

profiles

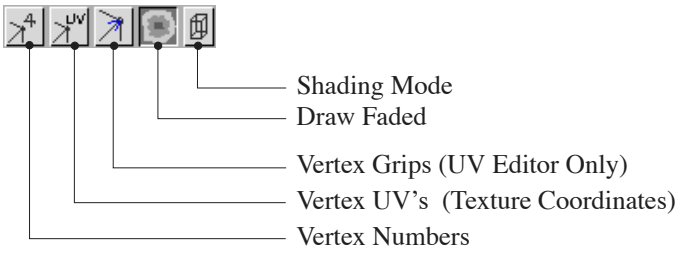
When enabled, only those profiles that are themselves selected (i.e. 1.3 0.*) will be displayed. If the surface that owns them is selected, having this option enabled does not mean the profiles themselves are selected or displayed.

It's useful to turn profile display off after boolean operations (i.e. Surfsect SOP) which tend to leave outer trim curves hanging out apparently in empty space, any time the scene looks too cluttered, and when you want to increase the display refresh rate.

profile numbers

When enabled, the Viewport displays the numbers of the profile curves. A profile number is always prefixed by the primitive number of its parent surface. For example, "0.2" is the third profile of first primitive in the geo detail.

VERTEX DISPLAY



vertex numbers

When enabled, it displays the number of each vertex in violet, and pink for each selected vertex, starting at zero.

The reason both point and vertex displays are available, is that a point can be shared between two or more primitives, while vertices are unique. See *What is the Difference between Points and Vertices?* p. 224 in the *Geometry Types* section.

vertex uv's (texture coordinates)

When enabled, this displays the UV normals for each vertex (if one exists) in violet for ones that are not selected, and in pink for those that are selected. The UV's are the texture coordinates for a vertex, which among other things, determine the placement of a texture map on the primitives containing the vertex. See section *Point UV's (Texture Coordinates)* p. 235 in the *Geometry Types* section for more information on texture coordinates.

VEXTEX GRIPS

Only applicable to the UV Editor. When enabled, displays Vertex Grips, used for UV editing (see: *UV Texture Editing* p. 137).

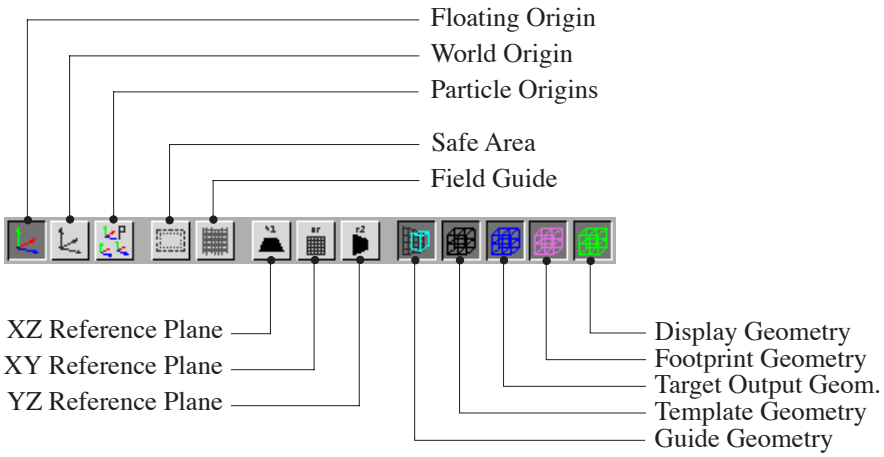
DRAW FADED

Only available for *Templates* and *Target Outputs*. When enabled, draws the specified geometry transparently such that you see only a dim outline. This allows you to see-through template geometry whilst working on your model.

SHADING MODE

Sets the Shading for the specified geometry. This is particularly useful, say if you want to wireframe the templated geometry, and shade the selected geometry.

REFERENCE DISPLAY



floating origin

The floating origin display exists in the bottom left corner of the Viewport. Enabled by default, it displays each world axis in a different colour. It is useful in cases where the world origin (below) is too obtrusive or where it is out of view altogether.

world origin

When enabled, it displays the world origin and the orientation of the XYZ axes.

Tip: To display an object's origin, set the object's "O" (origin) flag in an *Parameters > Object* type List view.

particle origins (particle axes / centre of mass)

Displays the origins of particles in a particle system.

Also displays Particle Axes (about which they rotate). If the *Display Particle Axes* optoin is turned on, and the particles have a Center of Mass ("com") attribute, then this will provide a visual representation of it.

safe area

Overlays video safe picture and safe title rectangles onto the Viewport display. When generating animations which are to be transferred to video, certain accommodations must be made for the limitations posed by the video format.

Typically, most television screens are not capable of displaying an entire video frame, and the edges of the picture get rolled-off the sides of the picture tube. The video safe area is that area of the view that can be safely displayed without worrying that it will be rolled-off at the edges.

Also, that part of the picture that most televisions do display is slightly distorted at the edges. It is therefore desirable to ensure that critical components of your image such as titles and people's faces are kept away from these edges, and within the 'safe title area'. This ensures that they won't appear distorted on a typical television.

field guide

Before computer animation, cell-animators used a grid called a *field guide* to align elements in each frame of an animation. This grid was overlaid on each frame, giving a common point of reference. When enabled, this option displays such a field guide in the Viewport.

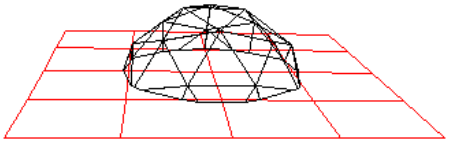
xz, xy, and yz reference planes

When enabled, these three icons display a light grey grid in the Viewport in the plane specified by the icon. It is a 20 × 20 grid with unit spacing (non-editable). The grid provides a fixed plane of reference against which to view the location of the geometry in 3D space.

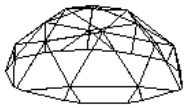
guide geometry display

When enabled, it displays additional guide geometry in the Viewport. Guide geometry is that which is displayed and affects the geometry, but is not rendered.

This helps you to view the effect of things such as clipping plane of the Clip SOP, the origin about which transforms take place of the Transform SOP, and the pivot point of the Copy SOP.



Guide Geometry On:
Clipping Plane is displayed



Guide Geometry Off:
Clipping Plane is not displayed

template geometry

Displays template geometry from SOP(s) which have their template flag enabled.

target output geometry

Displays the target output geometry.

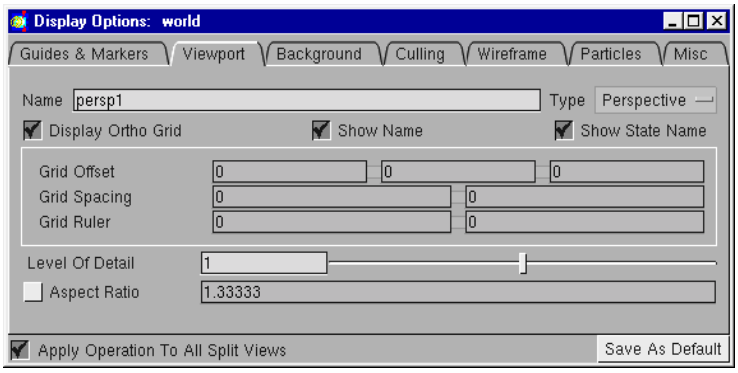
footprint geometry

Whether footprint geometry should be displayed.

display geometry

Determines whether to display your actual geometry.

4.2 VIEWPORT PAGE



NAME

Type a unique name for the Viewport in this edit field.

TYPE

Determines the display type for the selected Viewport: *Orthographic* or *Perspective*.

SHOW NAME

Displays the name of the Viewport in the top left corner of the Viewport.

SHOW CAMERA NAME

Displays the name of the camera through which you are viewing (if any).

SHOW STATE NAME

Whether to display the name of the currently selected Operation in the top-right corner of the Viewport.

DISPLAY ORTHO GRID

If enabled, a grid (whose parameters are specified below) will be displayed.

GRID OFFSET / SPACING / RULER

These parameters determine the dimensions and spacing of the grid displayed in the orthographic Viewports.

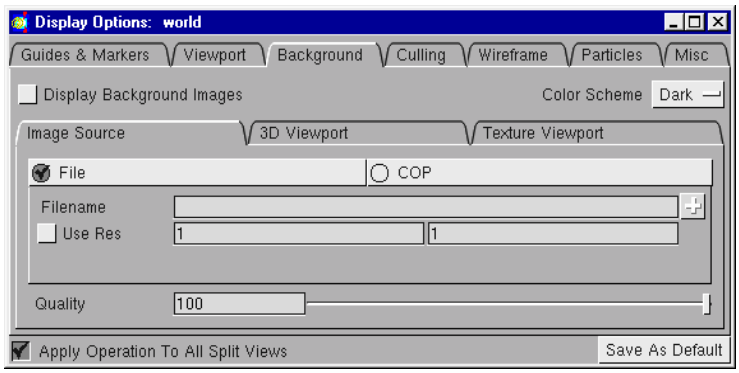
LEVEL OF DETAIL

Controls the Level of Detail (LOD) for tensor product surfaces. This is useful for increasing or decreasing the resolution of displayed Metaballs, NURBS and Bézier surfaces.

ASPECT RATIO

Allows you to set the ratio of horizontal to vertical elements in the Viewport. The aspect ratio is equal to the width/height.

4.3 BACKGROUND PAGE



DISPLAY BACKGROUND IMAGES

This parameter enables the other parameters on this page and activates the display of background images in the Viewport. For information on Rotoscoping, see: *Objects > Camera Object > Parameters – Roto Page* p. 274.

Note: Background images are specific to the Viewport and are not rendered as objects. You will have to composite your final rendering over the background images using COPs to achive that.

COLOR SCHEME

Sets a light or dark background for the Viewport.

QUALITY

The *Quality* parameter allows you to adjust the resolution of the image displayed in the background. The higher the value, the better the quality of the image. Increasing this value uses more RAM. A value of 50 would set the quality to be half that of the original.

IMAGE SOURCE > FILE

When checked, you can specify a file as the source for rotoscoped images.

filename

The image file to be rotoscoped. You can specify a range of files by using $F within the filename.

use res

When enabled, you can downsample the used images. Using 1:1 uses the full resolution of the supplied images. Entering a number like 0.5 : 0.5 will use only half the available resolution of the source images.

IMAGE SOURCE > COP

cop network

When checked, displays a list of available Composite Networks and their associated COPs. The selected COP is used to generate the images that are displayed in the background of the Viewport.

frame

If the COP holds of a sequence of images, you can specify which frame(s) should be displayed. The default ($F) is a variable which changes the background to match the current frame (be sure you have an adequate number of source images in your COP network, or else set this to 1 to obtain a static background image).

planes

These two menus specify which bitmap planes are to be used in displaying the background image. This allows you to independently specify whether to use colour or alpha (i.e. transparency) planes for your rotoscoped image.

3D VIEWPORT

texture mapped

This option specifies the method used in displaying the background image. When enabled, it will attempt to use hardware texture mapping if available. If hardware texture mapping is not available, texture-mapping will still work, but will be slower (since it will have to be executed in software).

This method has several advantages over the Frame Buffering technique that is employed if this option isn't enabled. Texture mapped backgrounds offer floating point zooming and box zooming abilities, with very little pixellation in the display of the image. Non texture-mapped backgrounds behave the same as texture-mapped backgrounds except when rotating (using the */winroll* channel).

apply zoom to background

Similar to the Camera object's *Track Camera Window Channels* parameter, this determines whether or not the Camera object's window parameters affect the size and orientation of the background.

automatically place image

If enabled, the Image Offset and Scale will be automatically calculated for you. If you want to manually control these two parameters, uncheck this option, and enter the desired values in the fields below.

image offset / scale

These two parameters allow you to offset and scale the image in the Viewport if you don't have *Automatically Place Image* turned on. An image scale of 1:1 will retain its original sizing.

TEXTURE VIEWPORT (UV VIEWPORT ONLY)

minimum / maximum uv

When you're in a UV Viewport and displaying a background image, you can adjust how much of the background image you want to see.

If greater than 1, it will tile the image.

filtered

Performs an over-resampling of the displayed background image so it doesn't appear as 'blocky' when you're zoomed in.

grid pixel spacing / offset

These two parameters determine the spacing and location of the Grid if Display Grid Over Image is turned on.

Tip: Set this to 0.5 if you want to snap to the centre of pixels.

display grid over image

Displays a grid overtop of the background image with the spacing and offset specified by the above parameters.

clamp grid to image

When enabled, it won't display the grid outside the image bounds.

display tile boundaries

If enabled, Houdini will display a grid representing the UV boundary around the UV 0-1 tiling of the image.

4.4 ROTOSCOPING

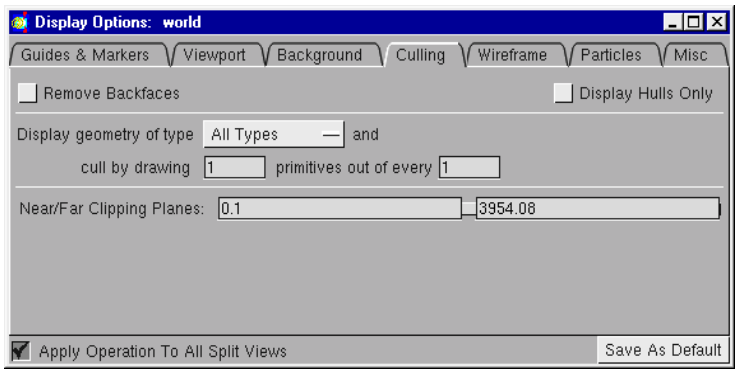
In order to rotoscope (match geometry to features in an image sequence) do:

- Call up the display options window (the ✣ button next to Wireframe/Gouraud).
- Click on the *Background* page tab.
- The option *Display Background Images*, controls whether you want to load images into the Viewport.
- Set the COP Network and COP Name in *Image Source* to the file / cop which contain the image / image sequence you want to rotoscope.
- If the Viewport is displaying the view that is seen from a Camera Object, then the *COP Network* and *COP Name* settings of the *Roto* page of the Camera Object will be used.
- This means rotoscoping is relative to the current camera that you have selected.

So, if someone has filmed different shots from different cameras, then you can assign those specific shots per camera.

- You may override these settings in the display options but the next time the scene is cooked, it will be set back to the camera's rotoscope settings.
- See *Parameters – Roto Page* p. 274 in the *Objects* section for more information on the Camera parameters associated with Rotoscoping.

4.5 CULLING PAGE



REMOVE BACKFACES

Normally, all sides of the geometry are computed and drawn, including primitives whose surface normals point away from you (the back sides you don't see). Enabling this option speeds up the display by removing the backfaces.

If your geometry is closed, then backfacing polygons are normally not visible. However open surfaces often have visible faces that face backwards.

This option is only applicable to Gouraud mode (see *Shading Mode* p. 117).

DISPLAY GEOMETRY OF TYPE

This menu allows you to specify what type of geometry is displayed in the Viewport. The types of geometry Houdini supports appear in this list and only the type you select will appear in the Viewport. This is useful for breaking down the geometry components that comprise your scene.

CULL BY DRAWING

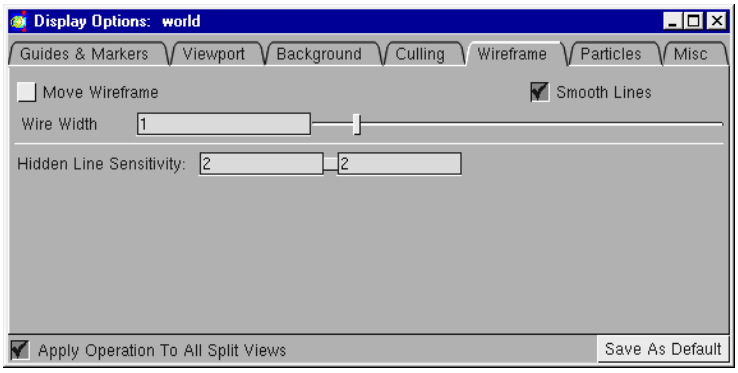
Tells Houdini to take the primitives in a scene, and skip displaying some of them. The number of primitives skipped is determined by the numbers entered in the two edit fields.

Entering *1:1* means it draws every primitive. Entering *1:2* means it draws 1 out of every 2 primitives (half of them). Entering *3:4* means it draws 3 out of every 4 primitives, and so on.

NEAR / FAR CLIPPING PLANES

These two edit fields allow you to set the near and far clipping, or visibility, planes for the Viewport. Things beyond the clipping plane are not visible.

4.6 WIREFRAME PAGE



MOVE WIREFRAME

When in Gouraud shading mode, it temporarily changes the display to a wireframe representation during movement in the Viewport.

SMOOTH LINES

Enables anti-aliasing for lines displayed in the Viewport.

WIRE WIDTH

The thickness (in pixels) of wireframe lines.

HIDDEN LINE SENSITIVITY

These two parameters are necessary because of the differences in OpenGL implementations of hidden line removal, and as such are important for getting a good hidden line display.

OpenGL handles drawing lines and polygons differently. You can see this if you enable hidden line removal, and use the *Ghost Fill* line style. Then, you will see grey areas where polygons are drawn, and the lines drawn overtop of this. The problem arises in older implementations of OpenGL (such as in some older generation SGI hardware) because the lines are drawn so that they are slightly imbedded into the polygons, causing discontinuities in the drawn lines.

The *Variable* / *Constant Sensitivity* slightly offsets the lines from the drawn polygons in terms of their distance from the viewer in the 3D depthbuffer. However, because lines may be rotated in 3D, this *Constant* amount of offset may not be enough to bring the lines from being embedded into the polygons if they are very oblique to the viewer. The *Variable* amount offsets the lines more depending on their angular rotation in relation to the viewer.

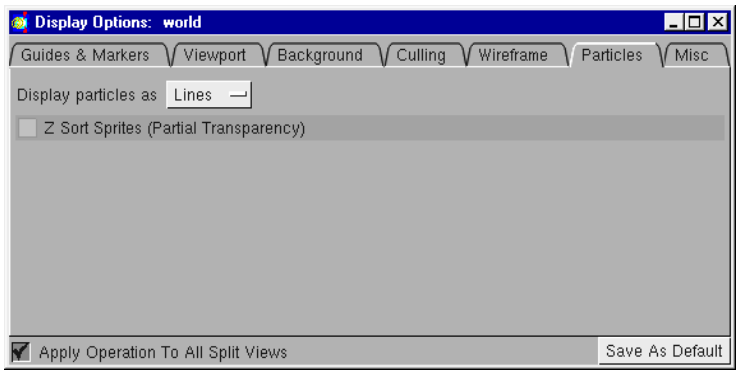
On most machines (those that have a fully implemented 1.1 version of OpenGL, such as Intergraph NT machines, and 3D Labs NT machines), this value can be set to 2:2 and left and forgotten about.

However, on certain machines (SGI Indy, NT Oxygen), this value may have to be tweaked in order to get a proper hidden line display. A value of 0:0 will always produce a hidden line display, but may contain gaps in the drawn hidden lines.

Increasing the values slightly can remove the gaps, but if the number is made too large, you will defeat the hidden line algorithm and not get a hidden line display. On an Indy for example, a setting of 0.0001 : 0.0001 will result in better hidden line display than a value of 0:0.

If a default value of 2:2 doesn't produce a good hidden line display, and 0:0 produces broken-up results, then you should experiment to find the optimal value for your machine. Try using very small (0.00001 - 1.0000) changes while experimenting. It is important to note that this setting is also affected by the current Dolly In/Out amount in the Viewport.

4.7 PARTICLES PAGE



DISPLAY PARTICLES AS

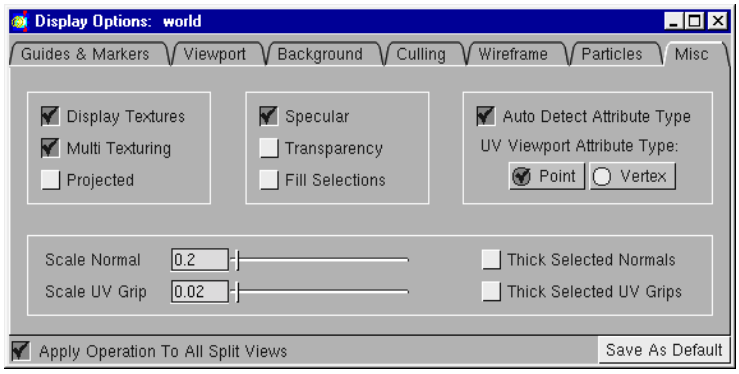
Use this menu to determine if particles should be displayed as *Lines*, *Points* or *Sprites*.

Tip: Use a Sprite POP to define your sprites.

Z SORT SPRITES (PARTIAL TRANSPARENCY)

Sorts sprites based on distance from the camera.

4.8 MISC PAGE



DISPLAY TEXTURES

This option enables the display of textures on the geometry shown in the Viewport. It is enabled by default. Disable it to speed-up animation during playback.

MULTI TEXTURING

Enables the display of all layers of multi-layered textures.

PROJECTED

With this option enabled spotlights are represented as textured maps projected from a light. If disabled, Houdini uses GL spotlights. GL spotlights are less effective because they work on a per-vertex basis, so you must have detailed geometry to get acceptable results when rendering.

Like *Tansparency*, (see below) this is a multi-pass solution with one drawing pass per spotlight in the scene. It too takes advantage of texture mapping, so those machines that do not have hardware texture-mapping support will slow noticeably.

If there is a Projector Map specified in the Light Object's Shading parameters page, it will be displayed when this option is enabled.

Note: This option does not work in conjunction with transparency or particle alpha.

SPECULAR

Determines whether specular highlights are displayed on geometry during Gouraud shading.

TRANSPARENCY

When in shaded mode anything that has an alpha of less than one will show transparently. This may be through a point attribute, a primitive attribute, a texture map or a material. Note that automatic sorting of geometry is not done. This requires sorting the objects manually in the list and also may require a Sort SOP at the end of the chain in the object.

FILL SELECTIONS

When enabled, selections are filled in wire-shaded modes. This is useful when you're manually editing something, like UVs, in realtime so you can see the change update under your selection.

Normally in Gouraud shaded mode, selections appear filled with the selection colour. Sometimes this is undesirable, say, when you would like to see the materials of the selected polygons. This option affects how selections are drawn in the wire-shaded modes. If enabled, selections are filled in the selection colour. If disabled, selections are drawn in wireframe only, leaving the underlying material unaffected.

AUTO DETECT ATTRIBUTE TYPE

Automatically detects the attribute type in a UV Viewport using the displayed geometry (it does this by following the type used in the current Display SOP).

uv viewport attribute type

Instead of having it automatically infer the attribute type from the current display SOP, you may want to manually determine whether to display point or vertex UV's – say if you're viewing Footprint geometry.

SCALE NORMAL

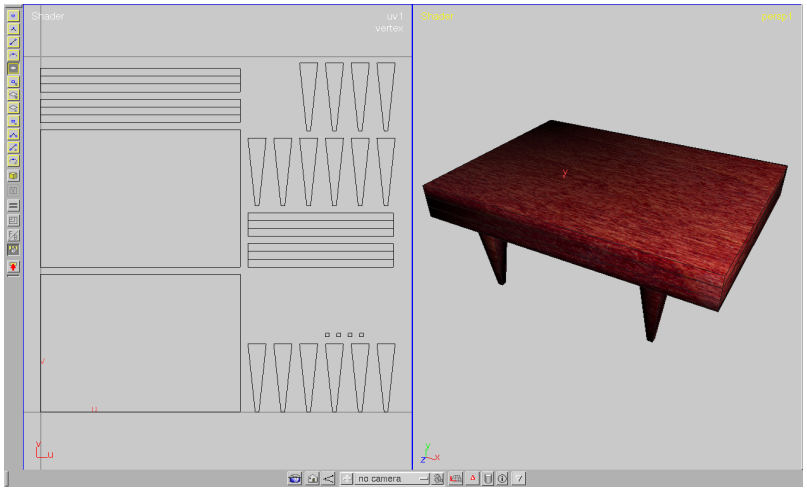
This will scale the display normal by the value entered here
(i.e. for very large objects the scale can be adjusted so that the normals can be seen).

SCALE UV GRIP

Scales the star-shaped 'grips' used in the UV Editor (enabled with *Display Options > Guides and Markers > UV Grips*).

5 UV TEXTURE EDITING

5.1 INTRODUCTION



UV Editing allows you to interactively assign UVs to textures directly on geometry.

5.2 INTERACTIVE TEXTURE POSITIONING

1. Make sure you are editing *Geometry* (and not merely Objects)!

2. In order to see both your geometry and the UV Editor at the same time, you will want to split your Viewport. Do this by selecting *Top Pair* from the *Viewport Layouts* menu on the bottom-right (or type Ctrl 6).

3. Using use the Ctrl menu from a View Operation select *UV Viewport* (or use the keyboard shortcut: Space 5) – this will invoke the UV Editor within that pane (leaving the second Viewport pane as a Perspective or Ortho view as desired).

4. Before you bring in your texture, you have to make the UV Viewport current, do this by typing: Space X.

5. You need to Bring in a Texture Map. Do this by calling up the Viewport Display Options by clicking the ✣ button to the right (or type D).

6. In the Display Options, you need to: 1) specify a Source Image* (either through File or COP); 2) enable the *Display Background Images* button; 3) turn the *Apply Operation to All Split Views* OFF); 4) in the *Texture Viewport* page, enable the *Filtered* button, and 5) in the *Guides and Markers* page, enable *Vertex Grips* or the *Points* display – depending on if you're intending to edit points or vertices).

** You will need to load a background image for the UV Viewpane – this image doesn't get applied to the geometry (since that is specified in the Shader), but it gives you a reference image to work from while tweaking the UVs.*

7. You will also need to assign a Texture Map to a portion of geometry. Do this by selecting the geometry in question (and then click to accept the selection), and append a *UV Texture* Operator (alternately, you may also want to use a UV Project OP). You may have to set the *Projection Axis*, and set the *Apply to* parameter to: *Point Texture*.

8. Then you want to append a *Shader OP* in which you specify the actual Texture Map to be placed onto your geometry (this actually happens via the SHOP – such as a *VEX Decal SHOP*).

9. To make editing of the primitive you have selected easier, you may want to use a *Visibility OP* at this point to hide all the non-selected primitives from the Viewport.

Note: Since you will be editing all the UV attributes with the *UV Edit SOP*, you will also see the points from the rest of the geometry that have zero UV values – that is why you may see a dense collection of points gathered on the bottom-left – those are not the points you are editing.

10. Finally, append a *UV Edit* Operation, and move the Points or Vertex Grips (altering the UV coordinates in the process), and watch the texture interactively update in the second pane of the Viewport (Viewport Shading must be enabled, and the *Display > Misc > Display Textures* parameter must be enabled!).

11. Once you're moving Points /Vertex Grips, edit and tweak the texture as desired.

Tip: If you're having trouble seeing the altered Texture in the Perspective View, it may be because of your Graphics Card – try rendering the view to see if the UV Edit has actually taken effect.

6 BRUSH TOOLS

6.1 INTRODUCTION

This section provides an overview of Houdini's Brush Tools. Brushes allow you to paint, comb, and sculpt surface geometry and attributes in an intuitive way.

6.2 BRUSH OPERATIONS

Brush Operations are those which use the brush painting tool in order to determine how they should act. They are:

Comb	Designed to allow you to manipulate a vector field in an intuitive manner. As the name suggests, it is usually a normal field which will then be used to grow hair.
CapturePaint	This is designed to paint capture weights. It will usually be used in the Object viewport unlike the other SOPs.
Paint	This is designed to paint arbitrary attributes. The obvious choice is to paint colour, but you could also paint any scalar field.
Sculpt	This one is designed to deform the actual surface of the geometry.

You will find these Operations all have very similar parameters. The *Brush*, *Symmetry*, and *Stroke* pages are the same for each of them. The Operation page has the operation specific parameters, but even many of these are the same.

6.3 BRUSHES QUICKSTART

DENSE GRIDS & PAINT SOP

1. Place a Grid SOP, change it to a *Mesh* type. You want the mesh to be dense, so make it 200 × 200. Depending on your graphics card, it may be faster to display in Hidden Line than in Shaded.

2. Append a Paint SOP. If you did this in the Viewport, the mesh will turn white and you are ready to go. If it was in the Network Editor, you have to enter the paint state by pressing the Enter key inside the Viewport. There will be a short initial delay while the geometry is prepared for the Brush state.

3. Move your cursor over the grid. You should see a red dotted circle with a small up arrow following your mouse over the surface – this is your 'Brush'. If this moves too slowly, you may want to use a smaller grid. The red circle marks the orientation and size of the current brush.

BRUSH RADIUS

4. Notice you have a *Radius* value in the toolbox. Altering it changes the size of your brush. The red circle will reflect this, allowing you to predict where you will be painting. Change it to *0.05* for now.

FOREGROUND / BACKGROUND COLOURS

5. In addition to the *Radius* value are two colour swatches labelled 'FG' and 'BG'. These refer to your foreground and background colours, respectively.
– Left mouse operations use the foreground colour while
– Middle mouse painting uses the background colour.
Click on the foreground swatch and change it to another colour, like blue.

6. Draw something – write your name. Click and drag with to paint on the surface. ***Note:*** If you find the colour gets washed out by the specular highlights, go to the Viewport Display options > *Misc* page and turn off the *Specular* option.

TRANSPARENCY

7. Currently the *Opacity* is 1, which means the center of the brush will completely replace the pixel's point colours below it with the new colour. Reduce the *Opacity* to 0.1 . This time drag with the , which will apply the background colour (i.e. black). With this, you have to keep painting back and forth over an area in order to darken it towards black.

OTHER TYPES OF BRUSHES

8. You can do more than just paint! Do a Ctrl in the Viewport – this displays the *Brush Options* menu. The first set of entries are those for the Left ( ) and Middle Mouse ( ) operations. Each mouse can be tied to its own operation. Note that, each mouse button is always tied to its own colour. From this menu, change the Middle Mouse button behaviour to *Eyedropper*.

9. As you along the image, you should see the BG swatch change colour to reflect the point the brush is currently over. If you switch the middle mouse button back to *Paint*, you can then paint with that colour.

ERASE CHANGES BRUSH

10. Change the *Opacity* back to 1. Change the LMB ( ) operation to *Erase Changes*. As you over the image, you will see it revert to its pre-paint SOP values. Note that this is done 'softly', as our default brush is a *Soft* brush. In other words, *Erase Changes* moves the current colour value towards the original colour value rather than just assigning it.

SMOOTHING (ANTI-ALIASING)

11. Next, from the Ctrl menu, try assigning a mouse button to *Smooth*. This smoothes and anti-aliases colours. It might not seem to be doing much, as the *Soft* brush already ensured that things are well anti-aliased. But it is an important operation to keep in mind.

SETTING HOTKEYS

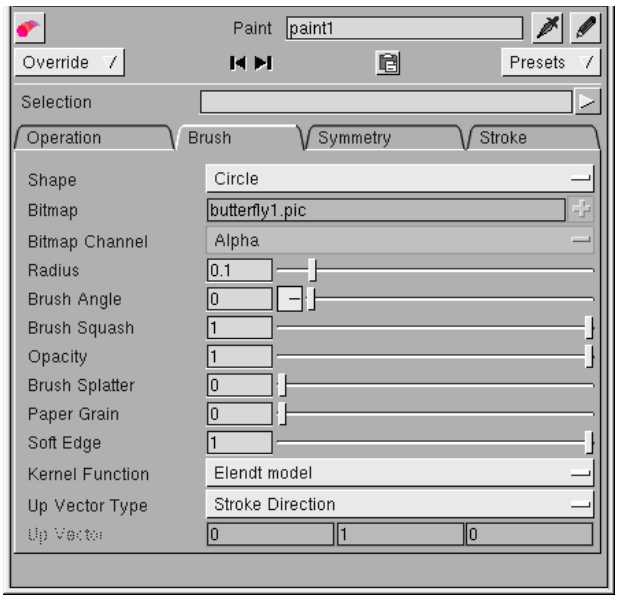
12. Houdini allows you to assign Hotkeys to anything. So, no doubt you are already wondering how to set a hotkey to change your mouse mode. To do this, do Ctrl. Move the cursor over the state you want it to hot key into, such as the *Eyedropper mode on the LMB*. Then, while still holding the RMB, type: Ctrl Alt left-Shift (look on the bright side, you won't be hitting that by accident!) – this brings up the Hotkey manager, where you can assign something like the E key to it.

DISABLING AUTO BRUSH ORIENTATION

13. Finally, in the Ctrl menu, turn off the *Orient Brush To Surface* option. You will see that the brush now doesn't follow the surface, but is rather always a perfect circle on the screen. This is most apparent if you tilt the grid so it is more edge on. While this mode doesn't look as good as having the brush always lie on the surface normal, you may find the changing of the brush orientation on non-trivial geometry is more frustrating than useful.

The above list covers all the most commonly used components of the Brush Tools. The other brushes are very similar, just with different choices of parameters under the Ctrl menu. The rest of this chapter deals with all the finer points of brushes.

6.4 SECRETS OF THE BRUSH PAGE



This page of parameters is common to all the Brush tools. Its purpose is to describe the shape of your brush-nib. With the following explanations, you can derive many different custom brush-nib effects.To test them out, use a plain 200 × 200 plane as described in the QuickStart and try playing with the different parameters to see how they feel.

SHAPE

The *Shape* menu controls the basic shape of the brush. The two built-in shapes are the circle and the square. The shape of the red brush outline will follow your change. The *Radius* parameter determines the radius of the circle, and half the diagonal size of the square brush. The custom shape is the *Bitmap* option. This choice scales the specified bitmap into the square brush, using the intensity of the bitmap to control the brush colour.

BITMAP

This is the file path of the bitmap to be used as the brushs-nib.
Use the 'op:' notation to refer to a COP.

BITMAP CHANNEL

While the default of *Alpha* usually makes sense, not all bitmaps have an alpha channel. Thus, you can interpret the bitmap as a greyscale image by either reading the red, green, or blue channels. You can also just read off the luminance of the bitmap.

RADIUS

The *Radius* of the brush, in Houdini units.

BRUSH ANGLE

The *Brush Angle* allows you to cant the brush's default orientation to create a calligraphy like effect. This obviously does nothing to the default circular brush, but is interesting if you have a square, bitmap, or squashed circular brush.

BRUSH SQUASH

This is to apply a non-uniform scale to the brush to create a brush that is thin in one direction. The default is in the Y direction. If you want to squash in the X direction, rotate the *Brush Angle* by 90 degrees.

OPACITY

The global opacity scale for the brush
(i.e. transparency: 0 = fully transparent; 1 = fully opaque).

BRUSH SPLATTER

Entirely uniform brushes are boring. Thus, to get some noise into the opacity of the brush, add *Brush Splatter*. Each point in the brush's space will have its opacity multiplied by a random number between *1* and *1-BrushSplatter*. As a point's position in the brush's space is rarely identical from stroke to stroke, this adds a form of noise that will cancel itself out over successive paintings.

PAPER GRAIN

Each point on the surface to be brushed is assigned a random number between *1* and *1-PaperGrain*. This value is multiplied by the opacity of the brush (taking into account Brush Splatter) to get the final opacity of the paint. The effect of this is to make some random points 'stickier' than other points. Because this stickiness is independent of the brush's position, you will note that some points are slower to be repainted than others. This gives a noisy grain to the object itself. With a *Paper Grain* of 1, you will find some points never seem to acquire the painted colour as they are too slippery.

SOFT EDGE

This option only applies to the Square and Circle brushes. Bitmap brushes must have their soft edge as part of the image (an easy way to do that is to filter the image in COPs with a Guassian blur).

The Soft Edge value is a number from 0 to 1 to specify how much of the brush's radius should be devoted to making a smooth roll off. By default, the entire brush is used (value 1). In this case, only the centre of the brush will have full opacity. The opacity of the remainder will be multiplied by the fall-off function applied to the distance to the center.

For example: A brush of radius 0.1 and Soft Edge 0.5 will have full opacity up to a distance of 0.05 from the center. From there to the edge (distance 0.1) the opacity will drop off according to the fall off function. For an entirely hard brush (points are either full opacity or zero opacity) set the Soft Edge to 0.

KERNEL FUNCTION

This metaball kernel determines what the fall off function is like. You can choose from any of the standard Houdini metaball kernel methods.

UP-VECTOR TYPE

When you used the non-circular brushes you likely noted that they oriented themselves in the direction you moved the mouse. This parameter is the cause of this behaviour – it defaults to *Stroke Direction*. You can lock it so the orientation is fixed by changing it to *Fixed*. For example, if you wanted to make a calligraphy brush, you can set the Shape to *Circular*, Squash to *0.2*, Angle to *45*, and Up-vector to *Fixed*.

UP-VECTOR

When the up-vector type is set to *Fixed*, there must be some global orientation which is used to determine how to align the brush. This determines that global orientation.

6.5 ACCUMULATE TO STENCIL DEMYSTIFIED

One of the more powerful modes to work with in the Brush Operations is the *Accumulate To Stencil* mode. Unfortunately, it isn't as intuitive as the normal, immediate application, mode. To understand accumulate to stencil, it is important to note that when the mode is off, you are still accumulating to a stencil. You are just automatically applying it whenever you let go of the mouse button so remain unaware of the fact. However, some operations are significantly easier with this mode on.

To see it, we will run yet another short example tutorial, as experimentation is worth a thousand lines of exposition:

EXAMPLE

1. Place a low-resolution Grid, type: *Mesh*, Size: *50* × *50*.

2. Append a Sculpt SOP. ***Tip:*** If the yellow glow of the selected geometry annoys you, you can and hold on the geometry and toggle the 'Highlight' flag. This will remove the yellow selection.

3. Paint the letter 'F' with the .

4. Why the letter F ? Well, the intuitive way to paint F is with two or more strokes. You will immediately notice that where you painted over your first stroke, the surface was sculpted twice as deep.

5. Now, reset all changes, and paint the number '8'. Do it with one stroke.

6. Notice where the stroke crossed itself in the alpha, the maximum depth was still 0.1, unlike when you had the repaint in the F. What is happening here is that during any one stroke, the maximum displacement that will be achieved is that set in

your displacement field (either FD or BD). However, each new stroke starts afresh, so successive strokes can cut further than this amount. Usually, this is the desired and intuitive behaviour. But sometimes, like when trying to sculpt some letters out of a grid, you want to have many individual strokes to count as a single stroke. As the title of this section suggests, that's done through Accumulate to Stencil.

7. Reset all the changes.

8. Turn on 'Accumulate to Stencil' from the *Operation* page.

9. Now, paint the F again with the .
Note that the crossover point is just as indented as the top corner is.

10. You will also note that after doing this, the little circle inside your brush cursor is filled in solid red. This is a reminder you currently have a stencil you are accumulating to.

11. While you are accumulating to a stencil, you can change the parameters with which to apply the stencil. For example, adjust your FD, or foreground displacement, to 0.2 and watch the depth of the cut change.

12. You an also erase parts of your stencil. Unlike normal modes where LMB means the foreground operation and MMB the background operation, in accumulate to stencil mode the LMB adds to the stencil and the MMB removes from it. MMB over the second stroke of the F to turn it into an upside down L.

13. When you are happy with the state of your stencil and want to start a new stroke, use . You could also click the 'Apply and Clear Stencil' button in the *Operation* page.

CAVEATS

While this *Apply to Stencil* mode gives you a lot of power and control over the exact brush stroke you perform, there are a few caveats that should be noted:

- The stencil is NOT saved with the SOP. So, if you have an unapplied stencil, save and quit, and reload, it will be gone. This means you can't use it as an easy way to animate a sculpt, for example. If you want a permanent stencil you can paint, you should use the Paint SOP with a custom attribute which will be used as a stencil in your operation.
- The stencil only works with one operation. You can't combine a smooth and a paint into one stencil operation, for example. If you change the operation, you'll change the way in which everything in the stencil is treated.

This is enough to get you started with Stenciling. A better understanding may be found if you also read the section on How Brushes Work.

6.6 SYMETRY PAGE – BRUSH REFLECTIONS

One of the important abilities of the brush tools is to automatically paint in a symmetrical manner. Traditionally, when one models something symmetrical, one first creates half of the object, and then mirrors the object, stitches it together, and smooths over the join. While this has many advantages, it also has some problems. The biggest of which is the inability to ensure that the seam is good while building the model.

Addressing this are the symmetry options of the brush system. These are all found in the *Symmetry* Page. Of at least as much importance is the handles associated with this. By default, they are hidden as usually they'd just get in the way of your painting. But, if you look at your handle list in a Brush SOP, you'll see two handles hidden: a Vector handle and a Plane handle. The Vector handle gives the orientation of the plane, and the axis of symmetry for rotational symmetry. The plane handle gives the plane of reflection for reflective symmetry.

BRUSH REFLECTION EXAMPLE

To demonstrate the use of reflective handles, we'll build a pumpkin – scary!

1. First, place a Sphere Operator, Type: *Polygon*, Frequency: *20*.

2. Next, place a Sculpt Operator.

3. We want to first create the ridges along the sides of the pumpkin. As our pumpkin is an exercise in symmetry, we'll have symmetrical ridges. First, enable the *Vector* handle from the handle list.

4. We want our pumpkin's 'Up' to be in the Y-axis, so the handle and chose *Align with Y-Axis* from the pop-up menu.

5. Hide the handle again as it'll just get in the way.

6. Go to the *Symmetry* page of the Sculpt Operator.

7. Turn on *Rotational Symmetry* and set the Number of Angles to: *8*.

8. Try sculpting the surface. Note how you get 8 sculpts happening at once! Play with this for a bit to get the feel of it.

9. Undo all your changes by going to the *Operation* page and clicking *Reset All Changes*. This clears out all the effects of this Operation in one go.

10. It's easiest to do this using the *Accumulate to Stencil* mode. Thus, in the *Operation* page, turn on *Accumulate To Stencil*.

11. Set your *Displacement* to: 0.05, *Opacity*: 0.5, *Radius*: 0.15.

12. Paint on the ridges. Remember that will add to your painting up to a maximum of the full 0.05, and will remove.

13. When you are happy with it, click to finish the stencil operation, and turn off *Accumulate to Stenci*l.

14. Turn off *Rotational Symmetry*, increase your *Displacement* back to 0.1, and on the top of your pumpkin repeatedly sculpt until you get a nice vine head.

15. It's now time to cut out the face. Turn on the vector and plane handles. Click with on the vector handle and align to the X-axis.

16. Turn on *Reflective Symmetry*. The plane handle turn blue – all your strokes will be mirrored about this blue plane.

17. Turn off both handles so they don't get in the way.

18. Set your *Opacity* to a full 1.0 .

19. Using , draw on an eye and half a mouth.
Watch the other half show up automagically.

20. Append a Paint Operation. Set your colour to a nice orange (e.g. 0.87, 0.62, 0.16) and set the *Brush Radius* to 1 and paint the pumpkin orange! You can save yourself some work by going to the *Operation* page and enabling *Apply To All*. This applies your current operation (be it is paint, sculpt, or smooth) uniformly to the entire geometry, usinging the current opacity of your brush.

21. Reduce the *Radius* to 0.1, and use the (which defaults to black) to paint out the eyes and mouth.

22. Set your FG to a nice vine green (e.g. 0.34, 0.6, 0.12) and paint the vine.

23. Turn off specular highlights in: Viewport Display options, and view the wonder!

6.7 THE FINER POINTS OF BRUSHES

This following parameters are available as brush-specific options in the *Operation* menu (Ctrl) of each brush. Like the Quickstart example (*Brushes Quickstart* p. 139), it is helpful to start with a dense grid to try and visualize the effects of these options.

APPLICATION MODE

Paint — Does the obvious: paints the geometry.

Eye-Dropper — This retrieves the colour of the point closest to where the brush contacts the geometry. The colour is put into the paint colour channel, ready for future painting.

Smooth — This iteratively smooths out hard edges in the colour field. This is of especial use when the painted field is an attribute which will be used to, for example, displace the geometry.

Erase Changes — This brush will move the geometry back to it's original, pre-brush state. It can be thought of as a localized 'Reset All Changes'.

MERGE MODE

This menu governs how the colour or attribute is applied to the geometry. The default behaviour is *Replace*.

Replace — The result is a blend between the original colour and the brush colour according to the brush alpha.

Add — The scaled brush colour is added to the original geometry colour.

Subtract — The scaled brush colour is subtracted from the original geometry colour.

Multiply — The result of multiplying the brush colour with the original geometry colour is then blended to the original colour by the brush opacity.

Hue — The hue of the original colour is changed to that of the brush colour. Then, the result is blended with the original colour.

Saturation — The saturation of the original colour is changed to that of the brush colour. Then, the result is blended with the original colour.

Lightness — The lightness of the original colour is changed to that of the brush colour. Then, the result is blended with the original colour.

Invert — The brush colour is unused here. Instead, the inverse of the source geometry colour is blended with the source geometry colour. Inverse in this case means to subtract from 1.0, which makes sense for colours but not necessarily other attributes.

STAMP BITMAP INSTEAD OF FG/BG COLOR

This option allows you to use the brush bitmap for more than just opacity information. This option is available if you have a bitmap brush. When engaged, it causes the brush to ignore the background or foreground colour when painting. Instead, the colour will be retrieved from the bitmap itself. For example, change the Brush > *Shape* to *Bitmap* and enable this option. With the default image, you will be painting with a butterfly brush.

APPLY TO ALL

This will apply the current operation uniformly to all of the geometry using the current brush opacity. It is useful for when you need to, for example, darken everything uniformly. You could set your brush colour to black, mergemode to Multiply, opacity to 0.1, and then hit Apply To All. With this, you can both save yourself the time of painting across the entire surface, and the risk of double-painting causing unexpected unevenness in application.

RESET ALL CHANGES

This handy button clears all the stored information associated with this brush. It's a fast way to clear out your doodling when you are about to go back to serious work.

CREATE MISSING COLOR AND ALPHA ATTRIBUTES

You may have noticed in the examples so far, we start with a plain grid and immediately start painting colour on it. This is somewhat unusual, as historically you'd have to explicitly create the colour attribute with the point sop before you began working with it. It is this flag which allows you to bypass that issue. If present, it ensures you always have the attribute before the painting begins.

WRITE ALPHA

This option is to allow you to do a "paint on glass" style effect to add paint to a clear piece of geometry. It is best explained with a mini tutorial:

1. Place a Grid as usual.

2. Add a Point Operation. Create the *Colour* and *Alpha* attributes. Set *Alpha* to 0.0. Your grid is now a transparent piece of glass.

3. Append a Paint Operation. By default, *Write Alpha* is on, so try painting on the grid. Note how when painting black, the black goes right to the edge and isn't whitened by the underlying default white colour. It, however, does add alpha as

you paint (to see black painting, you may have to switch to a light Viewport with: Space D > Background > Color scheme: Light).

4. Now, turn off Write Alpha. You can now change the colour of the areas you have painted, but the alpha is unaffected – transparent areas remain transparent.

OVERRIDE COLOR

The purpose of this option is on the surface rather obvious. Instead of painting just the Cd attribute, you can paint any floating point attribute of 1, 2, or 3 floats in size. Again, a short tutorial will give you some ideas of how this can be useful:

1. Create a low res grid, say 50 × 50.

2. Add an AttribCreate Operation. Choose to create the attribute 'height'. The other defaults should be fine.

3. Append a Paint Operation. Turn off *Create Missing*, turn off *Write Alpha*, turn on *Override Color*, and set the overriden colour from Cd to *height*.

4. When you enter the paint state, you'll see that the grid is now blue. This is because the attribute *height* is a single float in size, so false-colour visualization is used. Blue marks the coldest, or lowest, value (default 0) and red the hottest, or largest value (default 1).

5. Paint a bit of a pattern on the grid. Set the Opacity to 0.2 so you get smooth colour ramps. Or, use the Smooth tool to do some smoothing.

6. Append a VEX Mountain SOP. Note that the height of the VEX mountain has matched the intensity of your height field! This is an important feature of VEX SOPs: Any attribute with the same name as one of their parameters will override that parameter on a per point basis.

7. Now, template the VEX Mountain, and set display flag to the Paint Operation. Go into the Viewport's display options and:

- Guides & Markers > Templates: turn off draw faded
- Guides & Markers > Templates: change to Shaded
- Guides & Markers > Display Geometry: Disable (this is the green one)

Now you can paint on the *height* field while watching the deformed geometry react in real time. Make sure to reset the view options before continuing.

OVERRIDE ALPHA

This is the same as the override of the colour, but allows you to override which channel is used for the alpha attribute.

VISUALIZE ATTRIBUTE

If you are overriding the colour, this allows you to visualize the other attribute as the colour of the geometry. If the attribute has only one float, it will use false colour to do the visualization. Usually you don't have to set this as it is automatically implicitly enabled when you are in the paint state. However, if you want to keep the visual-

ization for later SOPs in the chain, it should be turned on. For example, the Paint SOP can be used as just a simple false colour applicator by just overriding the colour and setting the visualize flag.

VISUALIZE ATTRIBUTE > RANGE

This range marks what point is considered ‘blue’ or ‘black’ and what point is considered ‘red’ or ‘white’. For example, if you want to visualize a normal field as colour, you would:

- Create a polygon sphere, frequency 10.
- Append a Point Operation, add Normals.
- Append a Paint Operation.
- Turn off Create Missing Attributes.
- Override the Colour with ‘N’.
- Turn on Visualize Attribute.
- Set the range to -1 and 1.

6.8 SHARP EDGED CHISELS – THE SCULPT BRUSH

A good default geometry to try the sculpt tools with is a medium density sphere. with a Frequency set to about 10. Again, we start with the brush modes available:

DEFORM POINTS

This option will move points along their geometric normal according to the brush's displacement value. Note that this is NOT their attribute normal specified in the "N" attribute, but the geometric one from their actual orientation. Also, note that until the normals are updated (see later), it is the geometric normals of the incoming geometry and not the deformed geometry.

SMOOTH POINTS

This option will attempt to smooth out all the points under the brush to remove high frequency noise. It's often helpful to use a lower opacity to gain more control over how much smoothing occurs.

DEFORM AND SMOOTH

As the name suggests, this option will first deform the geometry, and then perform a post-deform smoothing pass to help clean up any kinks created by the deform operation. To see the difference, use an opacity 1 brush, Orient to surface Off, and the MMB to dig a hole in a the sphere using the plain deform. Then try it with the deform and smooth. The former tends to create ugly kinks at more extreme values, while the latter keeps things more under control.

ERASE CHANGES

The same as in the Paint Operation. The brush will move the geometry back to it’s original, pre-brush state. It can be thought of as a localized ‘Reset All Changes’.

AXIS & VECTOR

The deformation does not have to only be done along the normal of the surface. It can also be done according to an arbitrary axis. These fields allow you to deform in any specific direction regardless of how crinkly the geometry maybe. Like the Orient Brush to Surface option, this comes in use when you start working with geometry other than spheres.

UPDATE DISPLACEMENT NORMALS

This push button will recalculate the normals along which to displace the geometry. This might be best explained using another example:

1. Create a 50 × 50 mesh grid.
2. Append a sculpt SOP
3. Repeatedly sculpt it using deform to raise a central mountain. Note that the points always move with respect to their original normal. Thus, all the points will always move in the same direction, even through their adjoining polygons are now facing in different ways.
4. Click *Update Displacement Normals*.
5. Now sculpt the points on the side of your mountain. You should see them move outwards in addition to upwards, showing that they are no longer using their initial normals.

This updating isn't done at the end of every brush stroke because doing so would make it difficult to create the original mountain without the sides bulging outwards.

6.9 THE COMB OPERATION

The purpose of the Comb Operation is to paint vector fields on a surface. The primary reason for this would be to determine the orientation of hair, hence the name of the Operation. Its behaviour is also comb-like – the normals are moved into the direction in which you drag.

To play with the Comb Operation, take a polygonal, frequency 10, sphere, and enable the display of point normals. A brush of radius 0.3 and opacity 0.5 work well.

BRUSH MODES

Comb Normals	This mode will cause the normals to reorient in the direction of your brush movement. They are kept normalized, or of unit length, throughout this operation.
Smooth Normals	This iteratively smooths the vector field while maintaining the unit length of the vectors.
Erase Changes	This is as described elsewhere

Override Normal This lets you comb an attribute other than the default point normal. Unfortunately, there is no way to visualize that attribute easily.

6.10 CHARACTER EDITING – THE CAPTUREPAINT BRUSH

The CapturePaint brush is usually used at the Object level where most of these parameters are well hidden. However, it can be useful to delve to the SOP level and see what is happening under the hood. This discussion refers to use of the CapturePaint Brush at the SOP level.

The brush modes available are the same as those in the Paint Operation. The important difference is that all the operations are done with the capture weighting of the current capture index. That is set either at the object level, or from the Capture Index pull-down menu.

The unique parameters are:

CAPTURE INDEX

This drop down menu lets you choose which capture region's weights to paint. If it is empty, your geometry hasn't been captured so there are non available.

VISUALIZE TYPE

This lets you determine how to visualize the capture weight. The default is the standard bone colour method. Each bone's colour is used in conjunction with its weight to determine the final colour of a point. This is useful for seeing uncaptured points, or to see which bone has which point. However, it isn't effective for precision work on a specific bone. For that, change it to the False Colour mode. In this mode, you only see that one region's weight, but you can easily tell if it is being applied to the proper areas.

ZERO WEIGHT POINT COLOR

Assign this colour to unweighted points if the visualization type is *Bone Colors*.

The CapturePaint is unique in one other way. It allows two inputs. The second input is a reference input to use as the surface for the brush to use for intersection tests. By inputting the result of a deform to the second input, you can paint the deformed geometry to change the capture weights which it will be deformed with.

This creates what looks like a cyclical network which is usually a BAD thing in Houdini. However, in this special case the second input is NOT cooked. Instead, merely the last valid cooked geometry is retrieved. If there is no last valid cooked geometry, it uses the primary input as usual.

This ancillary input is automagically managed by the Capture Paint State at the Object level, so usually is not needed. However, you may find the deforming of the entire model too expensive to allow smooth painting. In this case, you can simplify things by deleting out the key part of your geometry and re-merging it back after they painting is done.

6.11 THE STROKE PAGE

This page is common to all the brushes, and often goes unused. Yet it provides some powerful feedback that can prove useful.

ORIENT BRUSH TO SURFACE

This parameter is already covered – it orients the brush perpendicular to the surface. It is generally faster to access it from the Ctrl menu before going to the parameters.

REALTIME MODE

This parameter is a bit ambiguous. After all, your normal painting seems to be going on in real time (on your 40Ghz PVII). A simple demo will demonstrate its use:

1. Put down a tube. Change to mesh, 20 × 50.

2. Append a Point Operation. Add point colours. (This will speed up the Paint Operation as it won't have to do this every cook).

3. Append a transform SOP. Set the Y rotation channel to ($F/300)*360*5. This will cause the tube to do 5 rotations every time through the play bar.

4. Append a Paint SOP.

5. Click *Play* in the Playbar and start painting. You'll note your paint strokes keep disappearing. This is because the result of your paint stroke isn't committed until the release of your mouse button for performance reasons.

6. Turn on 'Realtime Mode'. This causes the brush to always commit its changes.

7. Click *Play* in the Playbar, and paint on the lathe to your heart's content!

Before you get too excited, it is important to note that due to the technicalities of how brushes work, this isn't as useful for sculpting or normal combing as you might hope.

The last parameters on this page are read-only parameters. While it is theoretically possible that you could apply brush strokes manually by adjusting these parameters, it would be cumbersome in the extreme. Their primary use is to provide feedback as to where the brush is currently hitting your geometry. This can be used as a sort of 'geometry eyedropper' when you want to quickly investigate some troublesome geometry.

DIRECTION

This is the normal on the hit primitive at the hit position.

HIT LOCATION

This is the position in space where the brush ray hit the geometry.

HIT PRIMITIVE

This is the primitive number that was hit by the brush ray.

HIT UV

This is the parameteric coordinates where the primitive was hit. This is the UV coordinates used to evaluate the surface with a prim call, for example, and is not to confused with any "uv" texture coordinates that may be present.

EVENT

This is used internally to track the current state of the drawing mode. No-op is the default and is used to prevent any accidental writes using out of date or irrelvant hit information.

6.12 HOW BRUSHES WORK

The limitations and strengths of the brush operations can be better understood by keeping in mind how the underlying architecture behaves. The Brush SOPs are not as procedural as the traditional SOP. For those with past experience in Houdini, they can be considered fancy 'SoftLock' SOPs. For others, one should note the similarity to the Edit SOP which is also less procedural.

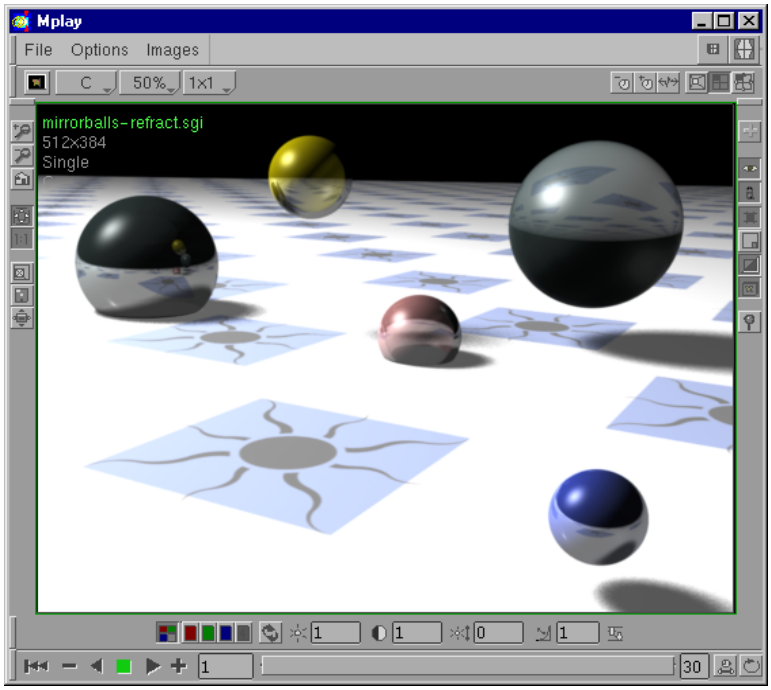
Instead of saving all of the brushstrokes you make, and what kind of brushstrokes they are, and reapplying them to cook the geometry, the Brush SOPs only record HOW you changed the geometry. They store how all the points of the input geometry should be changed to get the desired output geometry. This means if the incoming geometry is animated, the desired result is often achieved. However, the point numbers must match or the changes will be applied to the incorrect points.

The next important note is that the Brush SOPs all operate on points. Thus, if you are painting in the centre of a large polygon, you may see no change if the brush doesn't intersect one of the polygon's vertices. However, NURBs and Bezier surfaces are a special case here. They use the breakpoints rather than the actual control points, as the latter are often well removed from the surface. This can help explain some potential problems with the brushes. If you aren't sure why something isn't painting, or is painting, turn on point display to see what relevant points may be involved.

Finally, all the brush tools fall under a general design metaphor: the stencil. By painting with the brush, you are weighting the points you paint over with a value from 0 to 1. Points you colour with full opacity get a value of 1, those untouched a value of 0. These values form the stencil mask of the object. Then, the desired operation, whether it be a smooth, paint, deform, or comb, is applied to the geometry. The result of the operation and the original geometry are then blended together linearly with the stencil weight being the weighting factor.

4 2D-Viewers

1 MPLAY



mPlay replaces iPlay, see: *COPs > mPlay* ***p. 498 for complete info.***

1.1 MANIPULATING THE VIEWPORT

MMB	Pan.
Alt MMB	Zoom
Shift LMB drag	Select a region (Shift LMB outside image to deselect)
X	Make the Viewport current.
T	Toggle to occupy all Viewing Area.

1.2 CONTEXTUAL MENU (Ctrl)

The viewport context menu may be popped up by using Ctrl .
The functions of this menu are:

SAVE FRAME

Saves the current image.

SAVE SEQUENCE

Saves the current sequence.

ADD FRAME

If a frame was not initially loaded or was removed, this adds it back into the sequence.

REMOVE FRAME

Removes the current frame from the sequence. For rendered sequences and sequences loaded via Stdin, the frame is gone forever. For loaded files, it can be reloaded again using "Add Frame".

REMOVE SEQUENCE

Removes the current sequence from mPlay.

TOGGLE EXTRA UI

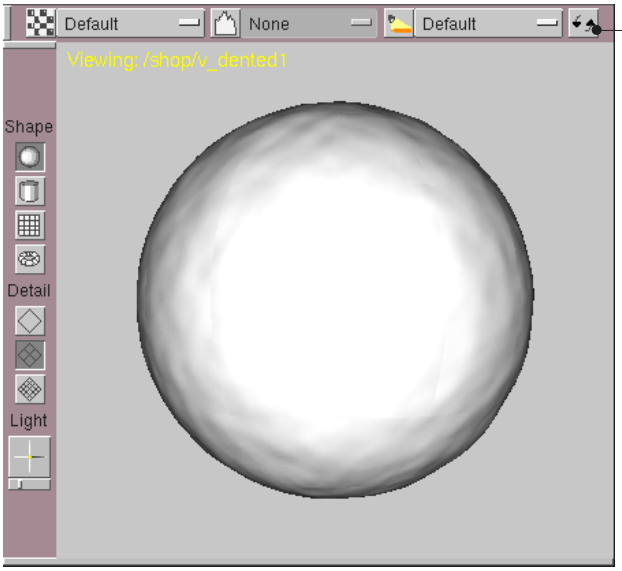
Toggles all the UI except for the viewports on or off.

THE SEQUENCE LIST

A list of all the currently loaded sequences. The currently viewed sequence is checked. Selecting another sequence will replace the current sequence. Several viewports can view the same sequence if desired.

5 Shader Viewer

1 SHADER VIEWER



This type of Viewer is available only when viewing SHOPs.

With it, you can see an image of the shader you are editing as applied to a selectable piece of generic geometry – a unit grid, tube, sphere, or torus.

If the scale of your material isn't suitable for previewing on a piece of unit geometry, you may want to apply it to the geometry in your scene in a regular 3D Viewport with the Render Preview state.

Previewing geometry in a Shader Viewer is often faster than previewing your material applied to the geometry in your scene.

1.1 VIEWPORT CONTROLS

The standard Houdini Viewport controls also apply to the Shader Viewer.
Use the following mouse-key combinations:

or	Tumble the View.
	Zoom In and Out.
Ctrl	You can pick any of the Options from the popup menu.
H	Home the View.

1.2 SHADER MENU CONTROLS

The controls along the top allows you to pick a SHOP – Surface, Displacement, or Light shader from the relevant menus. Only VEX shaders allowed.

RELOAD BUTTON Shift R

This is used when you're editing a Shader.
Click to reload VEX shader code and dialogs.

1.3 GEOMETRY AND LIGHTING CONTROLS

PREVIEW TYPES T

sphere

The Sphere preview shows your shader on a sphere set against a tiled background. The sphere radius is one (1) with its center at the origin (0, 0, 0). The UV coordinates are an orthographic projection on the surface of the sphere.

The tiled background highlights the effects of modifications in the material's reflectivity and transparency.

tube

The Cylinder preview displays your shader on a cylinder whose Y axis runs through the center of the preview window. Its radius is one, and the Y axis runs from negative one to one (-1 to 1). The UVW coordinates are an orthographic projection on the surface of the cylinder.

grid

The Ortho preview shows the shader on a flat grid (X & Y axes only). The grid coordinates are (-1, -1, 0) (1, 1, 0) and the UVW coordinates are (0, 0, 0) (1, 1, 0).

torus

The Torus preview shows the shader on a torus.

LEVEL OF DETAIL ICONS D

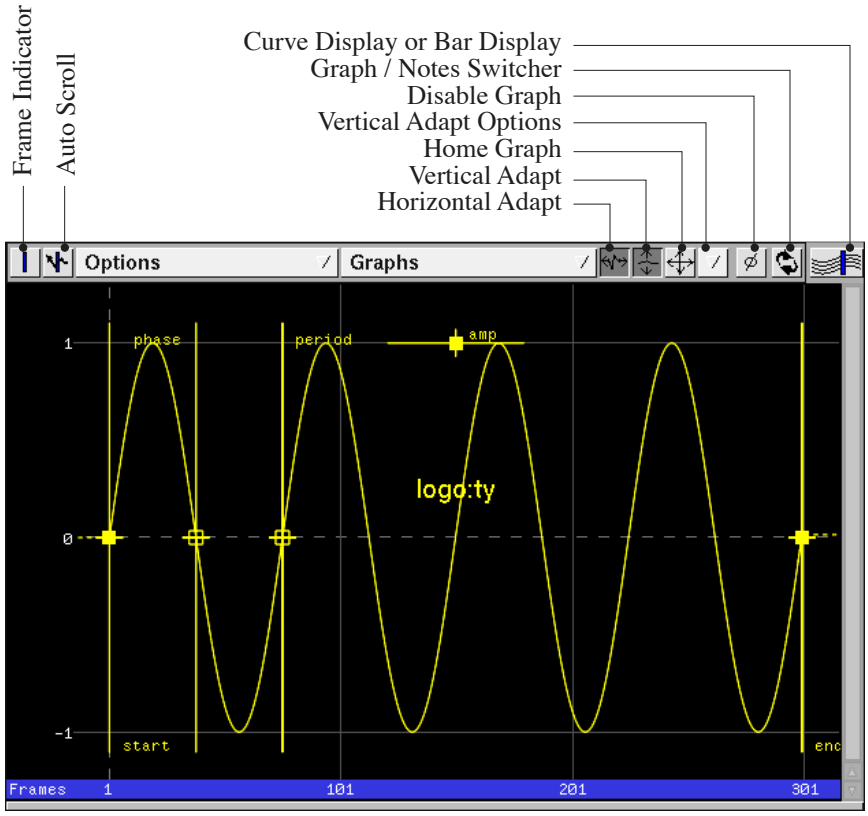
Sets the level of detail for the preview to: *Low*, *Medium*, or *High*.

LIGHT

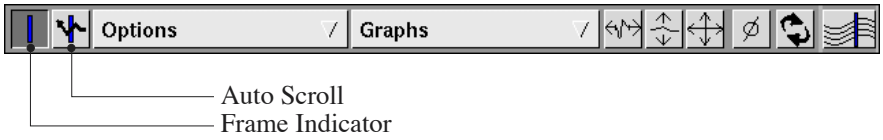
You can control the direction and intensity of the Light in the preview by manipulating this jack (see *Polar Jack* p. 216 for info).

6 CHOP Viewer

1 CHOP GRAPH DISPLAY



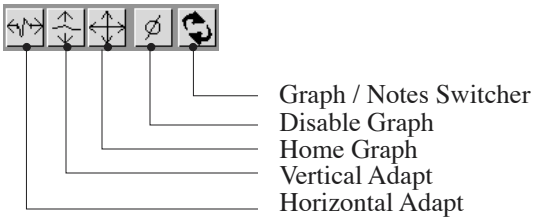
1.1 FRAME INDICATOR & AUTO SCROLL



The *Frame Indicator* button displays a blue line at the current frame. You can drag this Frame Indicator to change the current frame. If the current frame is outside the Graph when the Frame Indicator button is clicked, the Graph scrolls to make the current frame visible.

The *Auto Scroll* button causes the graph to scroll as you play the animation.

1.2 HORIZONTAL AND VERTICAL ADAPT



The Horizontal and Vertical Adapt buttons cause the graph to automatically resize, fitting all the intervals and ranges of the scoped channels in the graph area. When these buttons are enabled, the graph will always be automatically fit into the view.

VERTICAL ADAPT OPTIONS MENU

This menu specifies which vertical range to use when homing the Graph or using the Vertical Adapt button. By default it is set to *Auto*. This means when the vertical adapt is on, or the graph is homed the vertical range is determined by the minimum and maximum displayed values within each graph.

The other options list specific upper and lower values to be used. For example if it is set to 0 1 then all graphs when be located between 0 and 1 when homed or vertical adapt is turned on. A number of common presets are listed including *Other*. Select *Specify Other* to explicitly set the upper and lower graph limits.

This is useful in situations were several channels are displayed, one per graph yet they should all be displayed with the same vertical range. If you have several binary channels for example, some will not home correctly if they are completely flat. Set the menu to 0 1 then home the graph.

1.3 CURVE AND BAR DISPLAY

The Graph has two kinds of display: Curve Display and Bar Display. Switch between the two by clicking the Curve / Bar Display button.

Add to the Graph's scope by Shift-clicking on the Graph flags of CHOPs. In Bar Display, you get one bar per CHOP. The order of the CHOPs is determined by the order in the CHOPs List display – where it can also be reordered.

1.4 DISABLE GRAPH

To minimize overhead processing time absorbed by redrawing the graph, click the *Disable Graph* button. To see how frequently the graph is updating, disable *Double Buffer* in Houdini's *Preferences*. When the Frame Indicator and graph are both enabled, you will see the graph redraw every time Houdini advances a frame.

1.5 OPTIONS MENU – GRID LINES, LABELS, EXTEND CURVES

GRAPH INFORMATION L Y D X

Labels	Labels the graph with the channel names.
Handles	Displays the channel handles, which give you the ability to change CHOP parameters in the Graph directly.
Dots	Display dots at each sample of the CHOP channel. Useful for debugging sampling errors by visually examing the individual values of the channel (Shortcut: D key).
Extend Regions	Displays the values obtained when sampling the channels outside the CHOP's range.

RAW EDIT TOOLS

Displays the Raw Editing tools – see *Raw Sample Editing* p. 165.

SCOPE TOOLS

Displays Channel Scoping Tools – see *Scoping Tools* p. 163.

GRAPH UNITS U

Display the horizontal axis in *Frames*, *Samples*, and *Seconds*. This works alongside CHOPs' Units parameter in the *Common* menu, which lets you specify each CHOP's parameters in Frames, Samples, or Seconds.

GRAPH GRID DENSITY G

Select from *Low*, *Medium*, and *High* density of grid lines.

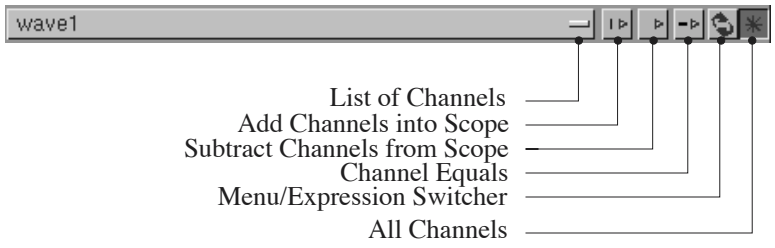
1.6 GRAPHS MENU

This lets you choose how to arrange the channels in separate graphs. It lets you set the maximum number of separate graphs visible at one time.

1.7 PARAMETER HANDLES ON THE GRAPH

You can click on the black bars of a Bar Display to modify the parameters of the CHOP.

1.8 SCOPING TOOLS



Scoped channels are often set by clicking the Graph Flag on a CHOP; or by Shift-clicking a Graph flag to add/subtract the CHOP to/from the Graph's scope. This requires a fair amount of scrolling about the CHOP network in order to do.

A faster way is to use the Scoping Tools (enabled in the *Options* menu). With them, you can add or subtract channels into/out of the current 'scope' by selecting from the *Add Channels* (+) or *Subtract Channels* (-) menus, or by setting the scope directly by selecting from the (=) menu.

The *Menu / Expression Switcher* changes the menu to an edit field where you can type in a regular expression to scope the channels such as: *t** (which scopes all the channels starting with the letter *t*).

1.9 KEYBOARD SHORT-CUTS FOR THE CHOP GRAPH

These are powerful keyboard short-cuts to help navigate through CHOP selection, parameter handle control and view control.

They work only when the cursor is within the Graph.

HORIZONTAL AND VERTICAL ADAPT

Click the Home Graph button to automatically fit the graph once. There are a number of keyboard short-cuts to fit the graph. Invoke these by placing the cursor over the graph and typing:

H	Home graph, fit channels into the graph's display area.
Shift H	Horizontal adapt – on/off.
Shift V	Vertical adapt – on/off.

LEFT MOUSE – SELECT

This selects a current CHOP, the current frame and/or CHOP parameters' handles:

	Picks the closest Frame Indicator, CHOP parameter handle or current CHOP. You can drag the Frame Indicator or parameter handle if picked.
Shift	Picks the closest CHOP, and sets it to the current CHOP.

Ctrl + LMB	The current frame set to the location of the cursor.
Alt + LMB	Finds the closest handle, sets the current CHOP to be the CHOP at that handle, and sets the current frame to the location of the handle.

MIDDLE MOUSE – ZOOM

MMB	Scales the graph horizontally and vertically, pivoting around the spot where the mouse was clicked.
Shift + MMB	If more than one graph is displayed in Curve Display, this will scale all the graphs at the same time. Otherwise only one graph will be scaled.
Ctrl + MMB	Constrains the zoom to one direction – either horizontal or vertical.

RIGHT MOUSE – SCROLL

RMB	Scrolls the graph without zooming.
Shift + RMB	If more than one graph is displayed in Curve Display, this will scroll all the graphs at the same time. Otherwise only one graph will be scrolled.
Ctrl + RMB	Constrains the scroll to one direction, horizontal or vertical.

GRAPH DETAILS

F / T	Frame Indicator – on/off.
L	Channel name labels – on/off.
E	Extend condition curves – on/off.

2 RAW SAMPLE EDITING

2.1 INTRODUCTION

You can directly edit CHOP channel data by unstowing the *Edit Tools* at the bottom of the CHOP Graph Display.

If could compare the *Raw Sample Editor* to Photoshop™, you could compare Houdini's *Channel Editor* to Adobe Illustrator™. The Raw Sample Editor is tailored to letting you edit the channel data samples directly – which is often faster and simpler than the alternative – which involves fitting the data through smooth spline approximatations, and then editing those in the Channel Editor.

All sample editing is done within the CHOP Graph. To sample edit a channel, graph it (by clicking on the Graph flag of the appropritae CHOP to turn on its display), and enable *Dots* in the Graph's *Options* menu.

(an unselected sample)	Select Samples by clicking and dragging.
Shift	Select multiple regions of samples.
(a selected sample)	Dragging one of the selected samples to move them vertically (the 'Group' editing behaviour).

2.2 SAMPLE EDITING ICONS

Pause the mouse over these icons to view the icon's name and function.

GROUP

Shift all selected samples by the same vertical amount as the selected sample.

SPAN DROP OFF

Shift neighbouring selected samples by an amount relative to the selected sample. Samples nearest the selected sample move the most.

EASE / EASE OUT/ EASE IN / CUBIC / LINEAR / CONSTANT

Interpolate the selected span with the corresponding function. The location of the currently selected sample determines the path of the interpolation.

COPY ABSOLUTE

The selected samples can be shifted horizontally to create duplicates of the selection. The shifted samples replace the original underlying samples.

COPY RELATIVE

The selected samples can be shifted horizontally to create duplicates of the selection. The shifted samples are added to the original underlying samples.

COPY UNCONSTRAINED

The selected samples can be shifted horizontally as well as vertically to create duplicates of the selection.

COL SELECT

This checkbox affects selection. Any selection made in one channel is mirrored to all other displayed channels.

2.3 EXAMPLE

1. Display three default Wave CHOPs. (One graph per CHOP). Turn on *Dots*. and choose *Cubic* icon. Turn on *Col Select*.

2. Box-select the first fifty frames of the first channel. Notice the range is resampled as a cubic spline for all three channels.

3. Pick a selected sample and drag it vertically, modifying the shape of the interpolation on all channels.

4. Pick the *Copy Relative* icon.

5. Drag the selection to frame 100.

6. Empty the selection by clicking on an empty area of the Graph. Click on the *Span Cubic* icon again.

7. Box select any area which appears to have a kink in it, and it will reinterpolate as a cubic shape.

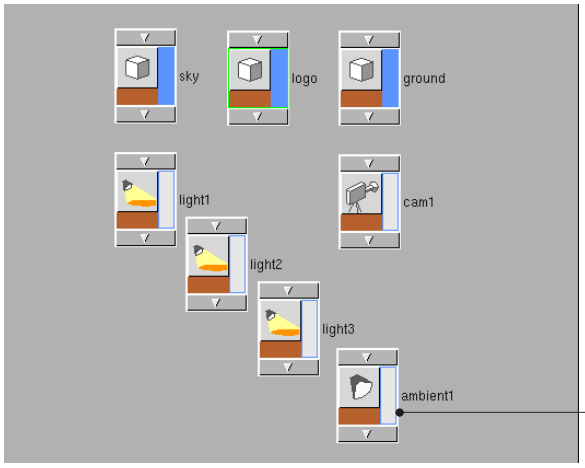
7 Network Editor

1 NETWORK EDITOR COMPONENTS

There are various types of operators in Houdini – Surface OPerators (SOPs), Composite OPerators (COPs), Particle OPerators (POPs) etc. Various Editors allow you to interconnect these operators to produce networks in which you can quickly change any parameter. The results will be transmitted through the network.

If you define your OPerators in the Viewport, Houdini will create a list of Operators for you. If you wish to manually view or edit these Operations, the Layout Area is the place to do it.

1.1 LAYOUT AREA



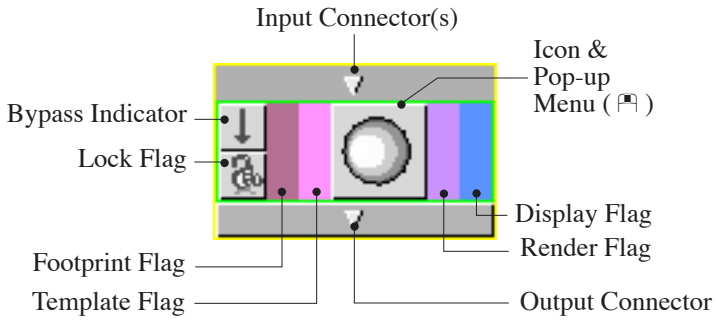
The Layout Area displays the OPs you have placed. The OPs are represented by tiles that you place in the Layout Area. For more information on how to place a tile in the Layout area and connect it to other tiles, see the sections below.

MOUSE CONTROLS

	Select individual OPs; marquee around multiple OPs.
	Zoom In / Out.
	Pan the view, scolling the displayable area.
Bksp or Tab	Add an OP tile to the Layout Area (pops-up a menu).
Ctrl	Contextual Menu.

2 OP TILES

An OP tile is a representation of an OP in the Network Editor that allows a graphic visualisation of their interconnections. There are different sorts of OP tiles, as illustrated above. The components of a typical OP tile are illustrated below:



2.1 ICON & POP-UP MENU

The icon gives a visual indication of the OP type. If the tile is outlined in green, it is selected. If it is outlined in yellow, it is current so that its parameters are available for editing. You can select multiple OPs at a time (hence you can have multiple green OPs), but you can only edit the parameters of one OP at a time (hence only one OP can be outlined in yellow).

Additionally, if the icon changes color, it provides a warning about the state of data within that OP. If the OP has turned red, the OP has not cooked properly. If yellow, the OP is only partially cooked, and may contain bad data. Chek the OPs information display ( ) click on the icon for error messages and other info.

Clicking the icon with right mouse ( ) displays an OP pop-up menu. The OP from which you selected a menu command is the only OP that is affected by the command. The typical commands for this pop-up menu follow:

TOGGLE DISPLAY / RENDER / TEMPLATE / LOCK

The *Toggle* menu commands allow you to turn on/off the various flags associated with an OP. These are equivalent to clicking on the coloured flags on the OP tile. If you are zoomed-out on the Network far enough, then the OP tiles become too small to display their flags, and they must be accessed from the menu.

PARAMETERS

Displays the OP's parameters in a floating dialog.

PREVIEW WINDOW (SOPS ONLY)

Displays an additional SOP Viewport in which you can view or edit the SOP's geometry. This is particularly useful if you want to see one SOP in a separate window while working on some other type of OP in a pane, or while manipulating some parameters higher up in the SOP chain (say from a sub-network).

SPREADSHEET (SOPS ONLY)

Displays a Spreadsheet allowing you to edit Points and attributes.
See *Geometry Spreadsheet* p. 221.

SAVE GEOMETRY... (SOPS ONLY)

Displays a dialog asking you to specify a filename for a file into which the current SOP is saved. You must select the type of file you want to save to from the pop-up menu. By choosing a suffix from the dialog's pop-up menu, you determine the format in which the geometry is saved.The options are: *.geo* and *.bgeo* (Houdini geometry files), *.poly* and *.bpoly* (*action* polygonal geometry files), *.rib* (RenderMan), *.dxf* (AutoCAD), *.iv* (Inventor), and *.obj* (Wavefront). See the *Geometry Formats Supported by Houdini* p. 253 in the *Formats* section for details.

Click on the ✣ button in the dialog box to open a standard Save dialog, and to change the directory into which the file is saved.

Once saved in these formats, the geometry can be read back into Houdini only by using a File SOP.

SAVE TEXTURE UV TO IMAGE (SOPS ONLY)

Displays a dialog from which you can render out a UV image from the SOP.

EDIT COMMENT

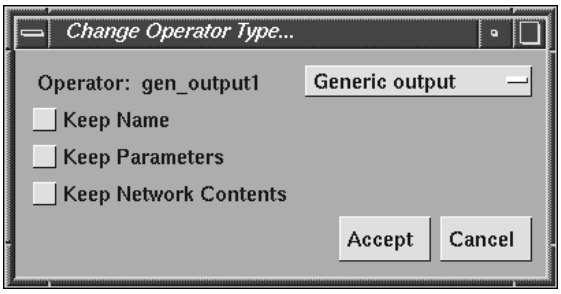
Displays a window in which you can edit the OP's comment. This is equivalent to clicking on the Comment Icon in the OP's parameters.

VIEWPORT SYNC (POPS ONLY)

Enabling Viewport Sync ensures that what is displayed in the Viewport matches the simulation at the current frame. For example, say you have changed Editors and the Playbar is set later than the last time the POP was cooked. With *Viewport Sync* enabled, all the inbetween frames will be cooked and the updated simulation shown. This ensures that the display is accurate, but takes time to accomplish. With *Viewport Sync* disabled, no frames are cooked, so you should be aware that what is shown in the Viewport may not be up-to-date. In this case, the *Reset* button in the Playbar will turn yellow to indicate that the Viewport is "stale".

Note: Changing the Render POP flag will not recook the POP to display the current status when *Viewport Sync* is disabled. To get an updated view, Reset the simulation (to frame 0) and recook.

CHANGE TYPE



Displays a dialog in which you can change the type of OP. Do this by selecting from the Operator menu to select a new OP from the standard pop-up list of OPs.

keep name / parameters / network contents

Enabling these check-boxes will keep the OP's: Name, Parameters, and Network contents as desired.

EXPORT TYPE...

This dialog allows you to specify an *.optype* file to save all the custom data associated with the named operator type. This data is the information common to all operators of this type, not just the selected operator. This includes the creation script (the initial values and the contents of subnets), the dialog script, the associated .so, .vex or .vfl, the help card, the icon, and even th elabel of the operator. To edit these values, choose *Properties...* from the operator menu.

CREATE TYPE FROM...

This command works only with subnets. It allows you to create a new scripted subnet from the existing OP node, and saves it to a file on disk – you will be prompted for a filename to save it to. You can subsequently load the *.optype* file using Houdini's *File > Install OP Type* command.

The new subnet defaults to being the selected subnet. The base name is prefixed by default by HOUDINI_DEFOPTYPEPREFIX, or the user name if that is undefined.

Tip: For power-users, command line functions: *optyperead*, *optypewrite*, *optypeinstall*, and *optypeuninstall* allow you to manage all this with scripts.

CREATE NEW VOP TYPE

This allows you to create a new VOP type. For example: One of the places where you can use use VOPs is within SHOPs (i.e. shaders). Once you place a SHOP – If the SHOP is comprised of VOPs instead of VEX code, you'll be able to select *Edit VOP Network* from the SHOP's pop-up menu.

If not – you can select *Create New VOP Type...* from the menu, and it will embed the SHOP into a new VOP which you can save to disk.

Once you've done this, then, you can place the created SHOP in a VOP network, and have access to its Inputs and Outputs from within a VOP network.

REFERENCE COPY

Makes a “referenced” copy of the selected OP(s). This is an instance, it will only reflect parameter changes on the original; changin parameters on this referenced copy will have no effect. Change the parameters on the original instead. The referenced OP will have a blue Name label and comments icon.

EDIT SUB-NETWORK (SUBNET OPS ONLY)

Selecting this menu item allows you to edit the sub-network of a Subnet OP. If there is no sub-network, this option is dimmed. To learn how to use Sub-networks, see Interface > *Sub-Networks* p. 190.

EDIT POP NETWORK (POP SOP ONLY)

If the SOP is a POP SOP, there will be an additional menu item: *Edit POP Network...* in the menu which will bring you to the POP Editor and will also set the View state parameters of the POP Editor to match those in the POP SOP.

HIDE / EXPOSE INPUTS / OUTPUTS

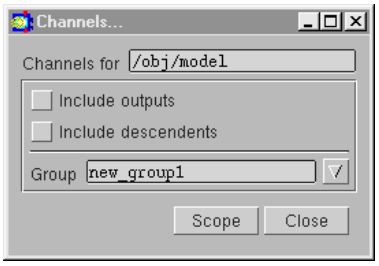
These selections will either collapse or expand a chain of connected operators. A collapsed chain is represented by a dashed connector line extending from the input connector of the node from which this command is executed.

SELECT INPUTS / OUTPUTS

Selects the entire chain of OPs that feed into / emanate from the selected OP via its input / output connector. All selected OPs become outlined in yellow. This is useful for moving, copying, or deleting them in groups.

This is useful for moving, copying, or deleting them in groups.

CHANNELS...



Brings up a dialog which allows you to scope channels of the operator and/or its descendents/outputs. You also have the option to create a group with these channels.

This dialog can also be accessed by right-clicking ( ) on an operator in the Channels List beside the channel editor.

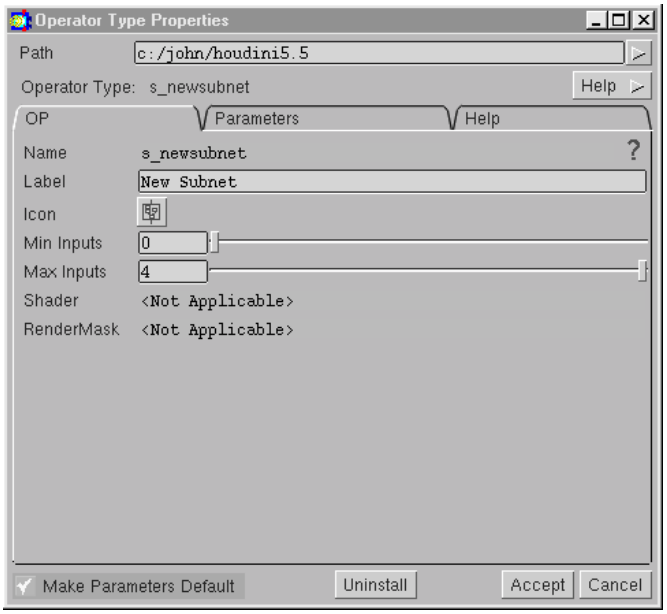
DELETE

Deletes the SOP from the Layout Area. The SOP network does its best to heal itself by reconnecting the remaining SOPs.

PROPERTIES...

Brings up a dialogue with all the Parameters and Properties for the OP.
See *Properties...* p. 172 for the description.

2.2 PROPERTIES...



This dialogue allows you to define dialogue scripts, edit Properties, and allows you to change the Help card, Label, icon, and redefine their Parameters. Get this dialog by clicking an OP tile, and selecting Properties... from the menu.

OPERATOR TYPE PROPERTIES > OP PAGE

This page controls the basic appearance and definition of the operator. It is valid for all operator types, though only some options may be available. The Options are:

Name — This is the built in name which the operator type is known by. It must be unique to its network type. It cannot be edited here. Built-in operators cannot have their names changed. Custom operators should be uninstalled and reinstalled with a new name.

Label — This is the plain text version of the name which shows up in the UI. This is also the name used in the type-selector.

Icon	This shows the current icon for the Operator. By clicking on it, a file chooser is brought up where you can replace it with a new icon. Icons are either vector (.icon or .bicon) or bitmapped (.pic). They should have a square aspect ratio for best results.
Min / Max Inputs	Only applicable to VEX and Subnet Operator Types. The Min Inputs is the minimum number of incoming nodes to prevent a red-flag warning on too few inputs; the Max Inputs governs the maximum number of incoming nodes possible.
Shader	The name of the Shader (SHOPs only).
RenderMask	The Mask to match the render type against (SHOPs only).

operator type properties > parameters page

This page allows you tod edit the dialog scripts for a scripted operator. Parameters can be added, deleted or re-ordered using the list displayed on the left. When a parameter is selected from the list, it can be edited using the controls on the right.

You can also create parameters by dragging them into the parameter list on the left. You can drag an individual channel to create a parameter to edit that one channel. Draggin an entire parameter will create a parameter to edit all the channels in that parameter. You can also drag an entire OP into the parameter list which will generate parameters to control all its visible options. This last method is the best way to recreate menus, group fields, etc.

The *callback* parameter allows you to specify an hscript command which is run every time the parameter is changed. When the callback script is executed, the parameter name for the callback will be stored in a variable *$script_parm*. For example, the callback might be something like:

```
unix xmessage ‘oppwf()‘/$script_parm
```

which would (on Unix systems at least) bring up a dialog box showing the node and parameter name every time the parameter was changed.

The *Disable When* parameter allows you to specify conditions for the parm to be disabled. The conditions are specified by a list of parm-value pairs. Each list is delimited by {} brackets. Each list evaluates to true if all the parms are equal to their values. The whole expression is true if any of the lists are true. The parm is disabled if the expression is true. For example:

```
{ foo 3 bar 4 }  {bar 2 }
```

will disable the parm if the parm named *foo* is 3 and the parm named *bar* is 4, or if the parm named *bar* is 2.

The *Auto-Link* parameter is automatically filled in when you drag and drop channels to be built. It stores where the source of the drag was.

The *Export* parameter allows you to specify in which dialogs the parameter is to appear. Choosing *None* indicates the parameter only appears in the full parameter dialog. *Dialog* indicates the parameter should also appear in the Operation Parame-

ters dialog. *All* indicates the parameter should also appear in the Operation Controls bar at the top of the Viewport.

When editing a menu, new menu entries can be added by filling in a token and label for the menu. For integer fields with a menu, the tokens *Must* be numbered consecutively from 0-N (the token is on the left, and the label is on the right).

operator type properties > help page

The text added here is displayed for the Operator when you click its *Help* button.

2.3 OTHER OP TILE COMPONENTS

INPUT CONNECTOR

Generally only available on Modifier SOPs, it accepts connections from the outputs of other SOPs. Connecting several SOPs together allows you to create and affect geometry in many ways.

Click here with the left mouse button ( ) to begin wiring a connection, click a second time on another SOP's output connector to finish the connection. If the second click doesn't land on another SOP's output, the input becomes disconnected.

Clicking here with the right mouse button ( ) provides a menu of all valid input SOPs. Select the desired SOP, and click a second time to place a new SOP which provides an input.

Clicking on the input connector with the middle mouse ( ) displays a small help status balloon which shows a brief one line description of what type of input is expected.

OUTPUT CONNECTOR

Allows you to connect the output of the SOP to the input of another SOP. Connecting several SOPs together allows you to create and affect geometry in many ways.

Click here with the left mouse button ( ) to begin wiring a connection; click a second time on another SOP's input connector to finish the connection.

Clicking here with the middle or right mouse buttons (or) provides a menu of all applicable modifier SOPs. Select the desired SOP, and click a second time to place a new SOP to receive the output.

If you click with the middle mouse ( ) the new SOP will be branched off. If you click with the right mouse ( ) the new SOP's output will inherit all the outputs of this SOP. Thus, it is possible to insert a new SOP in the middle of a SOP chain without losing any existing connections.

DISPLAY FLAG (SOPS & OTHERS)

Turns blue if it is the SOP to be displayed in the Viewport.

Clicking here makes this the Display SOP. The Display SOP is the one that is used for display when you view the object containing these SOPs in the Object Editor.

RENDER FLAG (SOPS ONLY)

When selected, this flag turns violet indicating that this is the SOP which will be rendered when selecting from the *Render* icon below the viewport. Only one SOP per object can be rendered.

TEMPLATE FLAG (SOPS AND POPS)

Turns pink if it is a template SOP for use in the SOP or Model Editor.

Any SOP's geometry can be used as template geometry for another SOP in the Model Editor or viewport. Clicking here makes this SOP the source of template geometry, and turns off all other templates.

Template geometry is not editable in the Model Editor; it serves only as a visual cue.

You can Shift - click to select multiple template sources – all of which will show up as a composite template. To turn off all template displays, click on any template indicator that is on.

COOK FLAG (POPS ONLY)

When selected, this flag turns violet indicating that this is made the Cook POP.

The Cook POP determines which POP will actually be cooked. By cooking the POP, it also ensures that all required ancestors are cooked.

The cook flag has a different meaning in subnetworks. If the POP is in a sub-network, the cook flag represents which POP is the output of the sub-network. Note that the cook POP of the topmost level of the POP network always represents which POP to cook, or the output of the entire network.

VIEW MODE AND THE POP FLAGS (POPS ONLY)

The *View* mode (in the *Options* menu) works closely with the POP flags:

view cook pop

In this mode, the Viewport shows the state of the particle system at the POP indicated by the cook flag. In this mode, changing the display flag will not affect what is shown in the Viewport. Use this mode to get the most efficient performance in the Viewport.

view display pop

In this mode, the Viewport shows the state of the particle system at the POP indicated by the display flag. In this mode, copies of the particle system are made to be shown in the Viewport. Because of this, playback will be slower than in the *View: Cook POP mode*.

LOCK FLAG

Indicates whether an OP is locked. For SOPs, this may mean soft-locked (so you can still procedurally change aspects of your model), or hard-locked (meaning the geometry is static and will not change).

A SOP also becomes locked automatically when the geometry it contains has been modified in Model Mode (i.e. changed with the Select state). You can manually lock or unlock a SOP by clicking on the lock icon. Manually hard-locking a SOP prevents the SOP from cooking.

When an OP is locked, it is shown in the Network Overview in yellow.

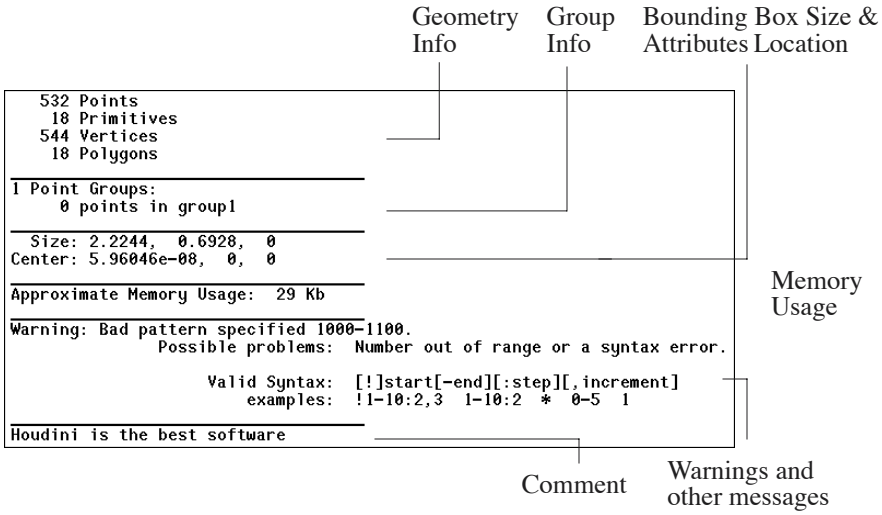
BY-PASS FLAG

Indicates whether the OP affects the network. When an OP is bypassed, it acts as if it were not there. Regardless of how many inputs an OP has, the first input is passed through. Another way to think about it – when bypassed, the network will cook the same as if this OP were deleted. This flag is useful for testing to see what the network would do with an OP removed without having to do any rewiring.

When an OP is bypassed, it is displayed in the Navigator in dark grey.

INFORMATION POP-UP (ON ICON)

Clicking on an OP with middle mouse ( ) displays information about the data contained by that OP at this point.



The information window contains either a list of geometry contents, possible warning messages, or error messages.

warnings and other messages

Sometimes the Information pop-up will turn yellow or red. This is displayed both on the SOP itself and in its miniature representation in the SOP Navigator. These warn you of problems with the way you have set up your SOP(s).

Yellow: If the Information pop-up has turned yellow, it means that the OP was able to cook (i.e. process all pertinent parameters and data), but has probably produced strange or unexpected results. Check the information pop-up to view warning messages.

Note: If an OP is yellow due to an interupted cook, recook by entering a Textport:

```
opcook [-f frameRange] /pathToObject
```

This can be assigned to an F-Key using the *Dialogs* > *Aliases/Variables* menu.

Red: If the Information pop-up has turned red, it means that the SOP was unable to cook, and there will be no geometry created. This is usually due to a lack of sufficient inputs to the SOP; bad primitive types; invalid parameters, or any number of other causes. Check the information balloon to view the error message.

If the SOP was unable to cook completely (i.e. process all pertinent parameters and data in order to make the desired geometry), the bottom-most area will display warning messages which provide some idea of what might cause the problem.

2.4 DASHED CONNECTORS (INTERCONNECTING POP TILES)

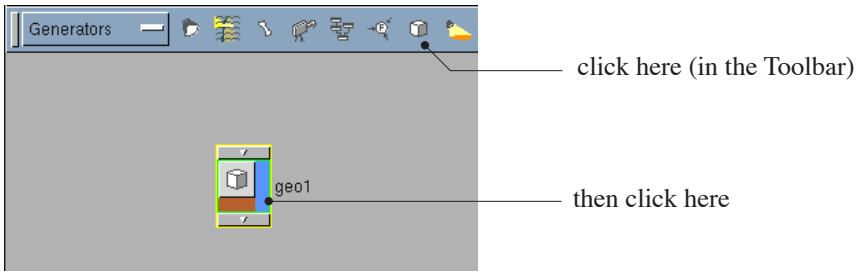
A dashed connector means that the input particles are used as a reference but they are not altered by any descendent POPs. An example is a connector to the Split POP. The Split POP uses the positions of the input particles to determine where to birth new particles. In this way it references the input particles. However, any POPs that follow the Split POP only affect the newly split particles and will not alter the original particles.

3 MANIPULATING OPS

3.1 PLACING OPS INTO LAYOUT AREA

FROM THE TOOLBAR

To place an operator into the Layout area, click on a operator's icon – located in the row above the Layout area – and the operator's icon becomes indented. Move the cursor into the Layout area, and click once to place the OP.



The OP then becomes current and selected.

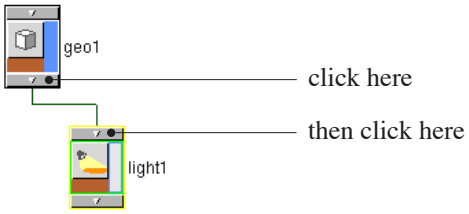
FROM A POPUP MENU

Typing Tab or Bksp pops-up a menu in the Layout Area from which you can select an OP. The items from the toolbar are presented in a hierachical pop-up menu from which you can select. After selecting an OP from the menu, it is placed in the Layout Area.

3.2 CONNECTING OPS

You connect OPs by wiring their input and output connectors together.
This can be done several ways:

- Click on an output connector to begin wiring, and click on another OP's input connector to complete the wiring.



- To create a branch (i.e. one output goes to more than one place), click an output which is already connected, and then click on the additional input to which you want it to connect to.

- To insert an additional OP between two existing OPs, click on the *input* of the second OP with the right mouse ( ), select an OP type, and click a second time to locate the OP in the Layout Area somewhere between the two.
- You can click the right mouse ( ) over the *output* of an OP. This links the OP's first input to its parent and the OP's output to all the children of its parent. The parent will end up with only one child – the newly inserted OP.
- To stop wiring a connection after you've already got a connection line started, click anywhere in the Layout Area except on a OP, or type Esc.
- To break a connection that already exists, click its input, and then click anywhere in the Layout Area except on an OP.
- At low network LOD (level of detail), a tile can be Ctrl-clicked to connect the tile. The connections currently work from first output to first input.

3.3 SELECTING AN OP

To select a OP, click on it in the OP Layout Area. It becomes the current OP (green outline), and its parameters are made available in the Parameter Area for editing.

Hold down the Shift key and click on several OPs to select more than one OP at a time. This allows you to perform operations on them as a group. Valid group operations include moving, copying, pasting and deleting.

To quickly select a group of operators while in Network view, click and drag with . This creates a selection box. Any operators inside of the box become selected.

THE DIFFERENCE BETWEEN SELECTED AND CURRENT

You can only have one Current OP (green outline) at a time – it is the one whose parameters are displayed in the Parameter area. On the other hand, you can Select more than one OP at a time by Shift clicking (yellow outlines). You can select more than one OP at a time in order to do things such as cut, paste, delete, or move them together in the OP Layout Area.

3.4 MOVING OPS

Do this by clicking and dragging an OP to move it about in the Layout Area. If you click and start dragging with your very first click, the OP is moved, but not selected. Clicking once to select, and then clicking a second time to move both moves and selects the OP.

You can also scroll the Layout area by clicking on any part of it that does not contain an OP with or . The cursor changes into a small hand, indicating that your mouse drag will scroll the Layout area, or alternately, use the Navigator. The left mouse button ( ) does not scroll the Layout Area as it is used for box selecting operators.

3.5 DELETING OPS

DELETING OPS

To delete OPs from the Layout Area, select the OPs (using Shift if necessary), and type the Del key, or select *Delete OPs* from the *Edit* menu.

If you want to delete all OPs from the Layout Area, select *Edit > Select All* (Alt A), and then type Del.

SMART RECOVERY

When deleting an OP from the network, the network tries to regenerate itself as much as possible by connecting all the descendants of that OP to the OP's first ancestor (i.e. the OP's first input). Together with the forward insertion operation described above, regenerative deletion is a reasonably fast alternative to a dedicated "replace" operation. Regeneration works equally well when several OPs are deleted at the same time.

3.6 COPYING AND PASTING OPS

To copy and then paste one or more OPs, select the operator(s) by clicking (or Shift clicking) on them. Select the *Copy* command from the *pop-up* menu (Ctrl), or type Alt C to copy the operator to the clipboard.

To paste the OP, select the *Paste* command from the *pop-up* menu (Ctrl)menu, or type Alt V. The copies are then placed into the Layout area. Pasted OPs are offset from their original position, and are displayed as selected in the Layout Area.

Copy and paste can be used between two or more concurrent Houdini sessions to transfer OPs and networks. This makes it easier to selectively merge components between jobs. If instead you want to make the clipboard files unique, use the environment variable, SESI_COPY_SUFFIX.

3.7 POP-UP MENUS AVAILABLE ON OP TILES

You have several pop-up menus available from an OP Tile. They are as follows:

- The right mouse button pops up the OP menu from the OP's icon.
- Click on an output connector with the middle or right mouse (or) to get a pop-up menu of OPs; after selecting one, click again to place it.
- Click the input connector of a modifier OP with the left mouse ( ) to start wiring for connection to an existing output.
- Click the input connector of an OP with the right mouse ( ) to display a pop-up of all OPs. Click again to place.
- Click the input connector of a modifier SOP with the middle mouse ( ) to display a brief one-line description of its allowable input.

4 NETWORK EDITOR MENU

Call up this menu by Ctrl clicking within the Network Editor.

SWITCH TO LIST T

Changes the display from a Layout Area full of OP tiles to a list.
See *Tables and Lists* p. 187.

LAYOUT ALL / SELECTED L / Shift L

Takes the existing tiles in the Layout Area and rearranges them in rows and columns starting in the bottom left corner. This operation can be undone by selecting *Undo* from the *Edit* menu.

Layout Selected rearranges only the selected Operators.

HOME ALL / SELECTED H / Shift H

Homes the display of OP tiles within the Layout Area. *Home Selected* homes only the selected OPs. Homing changes the amount of Zoom such that it will fill the Layout Area with the OPs.

FRAME ALL / SELECTED F / Shift F

Frames the display of OP tiles within the Layout Area. *Frame Selected* frames only the selected OPs. As opposed to Homing, Framing leaves the Zoom amount alone.

ALIGN ALL / SELECTED TO GRID

Aligns all / selected OP tiles to the Grid (settable in Display Options).

ADD OPERATOR Tab / Bksp

Adds an OPerator to the Layout Area or List. A menu pops-up allowing you to choose an operator to place.

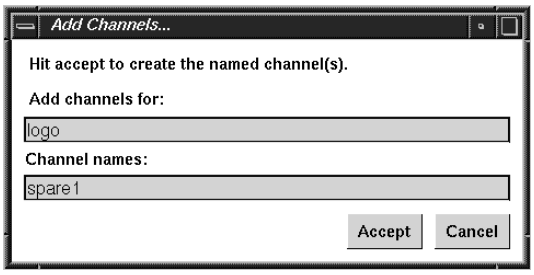
ADD GROUP Shift G

Adds the curently selected OPs to the group in the Group List. See *Open Group List* X p. 178. Groups allow you to associate a number of OPs together, so you can enable, disable, select, deselect, and otherwise operate on them together.

ADD NEW NETWORK Shift N

Allows you to add a new Network within a branch of a particular type of OPs. This is one level higher in the Houdini directory structure than the individual OPs. For example, you might add a new network (e.g. */ch2*) which contains a number of CHOPs. Individual OPs are added within the network.

ADD SPARE CHANNELS...



Displays a dialog which enables you to create additional channels for an operator or group of operators. You can view and edit your channels from in a Channel Editor pane.

example of a use for spare channels

Spare channels are useful within expressions. For example, you can multiply a channel with a spare channel to scale the values within it. Say you have a *sin()* expression controlling the */ty* parameter of an object, but you wanted to scale that *sin()* expression over time. You could change the expression for */ty* to something like:

*ch("spare1") * abs(sin($F*12))*

and then graphically modify the envelope of the *sin()* expression in a Channel Editor pane by tweaking the spare channel. Bracketing the *sin()* function with the *abs()* function makes the whole thing bounce. The spare channel can then control the height of the bouncing over time. Nifty, eh?

renaming channels

In order to rename channels that you've already added, you have to edit the .hip file that contains them. For information on how to do this, please see, *Editing hip Scripts* p. 195 of the *Scripting* section.

deleting a spare channel

Delete any channel using the *chrm* command in a Textport. For example:

```
chrm /obj/geo1/spare1
```

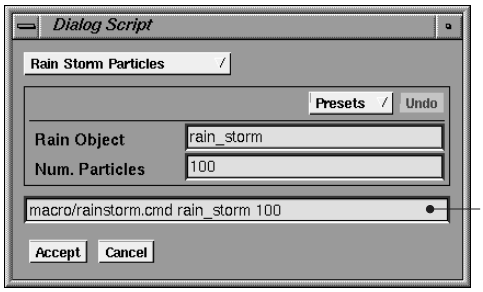
SCOPE CHANNELS

Takes the channels of the selected OP, and scopes them in a Channel Editor, allowing you to see them graphically.

RENAME SELECTED NODES...

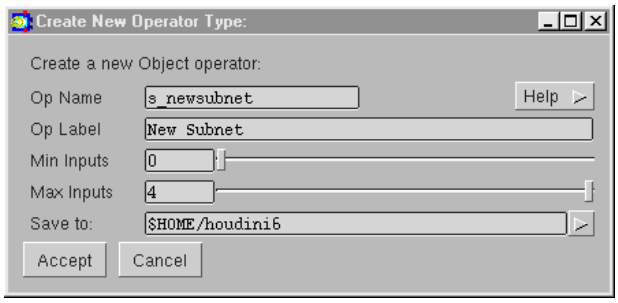
Allows you to rename the nodes currently selected in the Network Editor.

EXECUTE OPERATOR SCRIPTS...



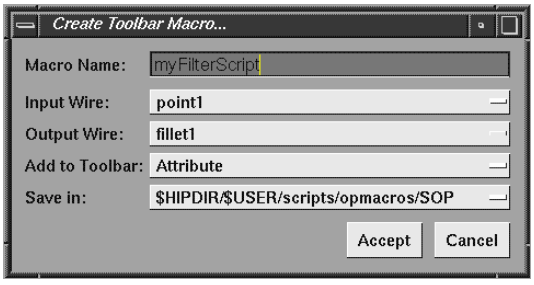
Operator Scripts are macros which provide a graphical interface to command scripts. The macros allow you to change parameters for the script. These scripts can do a variety of different tasks to simplify common procedures or processes. For more information on Dialog Scripts, which include Macros, see *Dialog Scripts* p. 219.

CREATE SUBNET OPERATOR TYPE...



Scripted subnets are defined in a text file located in HOUDINI_PATH. This dialog will create a new table entry in the location of your choosing. Following this, you can add parameters to the scripted sub-network. To complete the process, choose *Make Permanent Defaults* from the Preset menu of the OP. Then when a scripted sub-network is created in the future, it will have the proper setup.

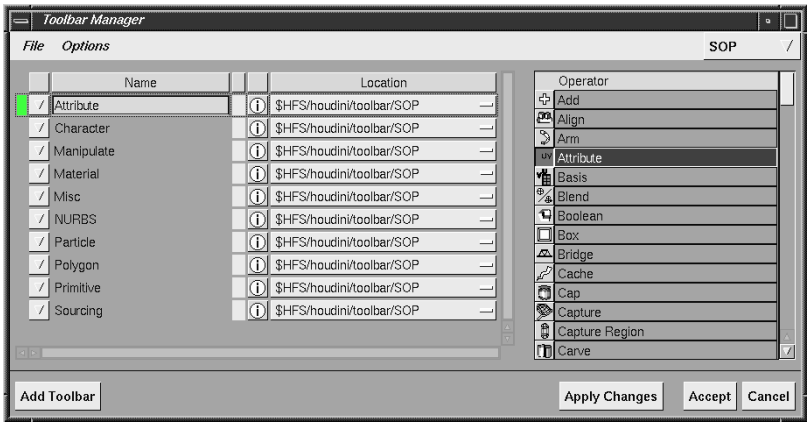
CREATE TOOLBAR MACRO...



Allows you to to create a macro out of the selected OPs. Your macro will show up as if it were just another OP. To do this, simply select the OPs you want in your macro, then select this command. In the dialog that appears, supply a name (how you want it to show up in the toolbar), specify the input and output to your “blackbox”, and click Accept. You will also need to specify a Toolbar into which your macro should

be placed. Toolbar macros are saved as *opscripts* which are saved in your Houdini directory.

EDIT TOOLBARS... Shift T



Allows you to customise the Toolbars found in the Network Editor for each OP type. This also defines the sub-menus you see when you type Tab to place a new OP in the Layout Area.

With this, you can create new toolbars containing sub-sets of the OPs. For example, you can create a toolbar of Generator SOPs, and another one for Modifier SOPs.

creating a custom toolbar

To create a custom toolbar, select the type of OP from the OP-Type menu on the top-right. Then click on the *Add Toolbar* button. The list on the right will display all available OPs that you can add into the toolbar. Toggle the items you want included in that toolbar by clicking on items in the list. When you are done, click on the pop-up menu beside the name of the Toolbar you've added, and select *Save*. If you don't want a toolbar any longer, select *Delete*.

JUMP UP / DOWN U / I OR Enter

Navigates one level up or down into the OP hierarchy.

SET CURRENT ABOVE / BELOW / LEFT / RIGHT K / J / < / >

Moves to the next OP in the specified direction, making it the current OP.

CUT / COPY Alt C

Cuts or Copies all selected objects to the Editor's clipboard.

PASTE Alt V

Makes a copy of the clipboard's contents and adds it to your scene. If the names of the copied objects are already taken, the new objects are named with a different sequence number appended.

Tip: If you are having problems Copy/Pasting between two .hip files, you might need to use the Scripting commands *opscript* or *opwrite* to save out the OPs, and the *source* command to get your OPs loaded back into another .hip file. See *OPscript* p. 189, *opwrite* p. 191, and *source* p. 119 in the *Scripting* section for details.

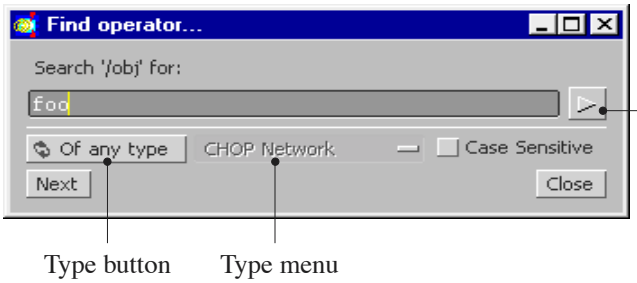
DELETE Del

Deletes all the currently selected operators.

SELECT ALL... Alt A

Selects all operators that exist in the Editor.

FIND OPERATOR... Alt F



Displays a dialog that lets you select an operator by name and scrolls the Layout Area or List view to the given operator making it current and selected in the Layout Area, the Object List (in the Object Editor), and in the Parameter Editing area.

If an operator with the exact name specified isn't found, then operators merely containing the specified text are found.

Tip: Typing / in a network editor acts like Alt F. However, it is a fast OP find which allows you to type in your OP name and then Enter without using the mouse or explicitly closing any dialogs.

wildcards in search

For further search flexibility, the edit field accepts standard wildcard characters as part of the search pattern. For example, searching for "left*bone" will find operators whose name starts with "left" and ends with "bone".

type button / menu

By clicking on the *Type* button and specifying a type from the Type pop-up menu, searching takes place by operator type. i.e. if you have selected an Add SOP and the name field has: *t** in it, the search will find all Add SOPs that start with *t* .

When you switch to *Of any Type* the edit field automatically changes to contain * (match all names) to avoid the confusion that can be caused when it fails to find anything when no name is supplied for the search.

pop-up menu

Clicking on the pop-up menu button displays a list of operators available in the current editor. This makes it easy to pick an operator, and have *Find* locate and centre the object in the Layout Area.

next button

The *Next* button continues the search using the search string specified in the edit field. This is useful for finding objects that have similar names.

close button

Closes the Find dialog.

EXPOSE ALL Shift E

Exposes all hidden operators previously hidden with the *Hide Selected* command.

HIDE SELECTED Shift D

Hides the selected operators while in network or list modes, reducing screen clutter.

COLLAPSE SELECTED INTO SUBNET Shift C

Collapses the selected OPs down into a Sub-network as described in *Sub-Networks* p. 190.

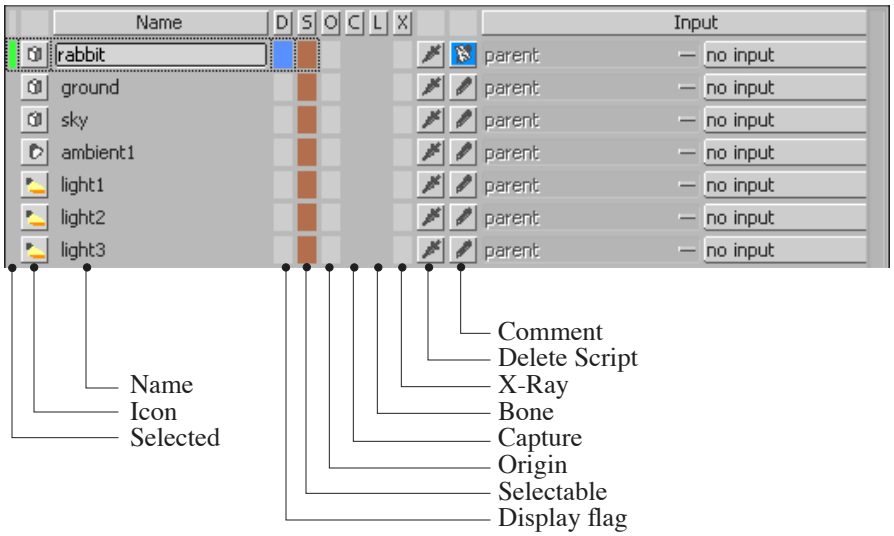
PREVIOUS ZOOM LEVEL B

Goes back to the previous zoom level (after using to zoom the Layout).

DISPLAY OPTIONS... D

Allows you to configure the behaviour of the Layout Area, including setting the grid snap, and what OP tile information is displayed. See *Network Editor Display Options* p. 62.

5 TABLES AND LISTS



All the flags and attributes normally displayed about an OP on a tile can also be displayed in a list view. For a complete listing of the individual flags pertinent to each type of OP, see that OP section's Tile description (e.g. POP Tiles, CHOP Tiles, etc.).

5.1 LIST INTERACTION

MOUSE-DRIVEN ACTIONS

	Selects list items that haven't already been selected, and controls the operation of menus and buttons within a cell.
	Always selects itemss no matter where in the list you click with the cursor.
	Allows you to perform drag and drop operations with cells, though this function is only available if the *Sort Order* of the list has been set to *User Defined*.
Ctrl	Pops up the *List* menu.

SELECTION STATUS

- *Green*..... OP is the *Current* OP – it's parameters are being displayed and edited.
- *Yellow*..... Network is *Selected*.
  A range can be selected with Shift,
  Independent additions can be made with Alt.

5.2 LIST MENU – LIST ORDER

The menu for List view is the same as for the Network Editor's Layout Area (see *Network Editor Menu* p. 181), with the exception of some additional commands to control list ordering. In this menu, *List Order* allows you to select how the objects in your list are organized:

user defined

Selecting this type of ordering presents the operators in a user-defined order. When the List is in *User-Defined* mode, you can change the order of the nodes by dragging them with . If you drag on a node that is not selected, only that node will be reordered. However, if you drag on a selected node, then all selected nodes will be reordered to go to the point specified when you've finished dragging. The cells to be dragged will have a white hash pattern over them.

alphabetic

Sorts the operators alphabetically by name.

operator type

Sorts the operators by their type.

hierarchical

In the Object Editor only, operators are arranged according to their parenting relationship with other objects in the scene. When this type of ordering is selected, objects that are parented to another object are indented from the object they are parented to. If the indenting would cause it to be displayed off-screen, the indented part is truncated, and a "*+number*" is added before the name to show how much further it would be indented.

MODIFYING INPUTS

The right-most column in a list entry controls an operator's inputs. Depending on the Editor Type you happen to be working with, inputs to an operator can define parenting relationships (Object Editor), or the information (SOP, COP, TOP) being fed into the operator. The list of available input operators is displayed in pop-up menu format.

If an operator, a Particle SOP for example, requires more than one input, two pop-up menus are displayed. The first specifies the type of input while the second allows you to specify which operator supplies information to that input. Some operators like the Merge SOP, can handle many inputs. The first menu always lists the first *n+1* inputs, where *n* is the number of inputs that are actually in use.

Operators, like generator SOPs, that require no input have nothing in this field.

OPERATOR INFO

To see the information pop-up window, click on the *Information* button in the list entry. Editor-specific differences between information displayed is discussed in that editor's section of the manual.

5.3 TABLE INTERACTION

MAKING A CONTIGUOUS SELECTION

Click on a cell that doesn't contain a control such as a button, menu, or edit field. This clears any previous selection. If you drag your cursor, you will extend the selection area. If you drag past the bottom or top edges of the list, it will automatically scroll. If you click in the grey area below the end of the list, it will select the last cell.

Note: Your anchor point, where the selection area begins, is denoted by a white highlight while the selection area itself is bounded by a black border.

CHANGING THE SELECTION AREA

You can extend or shrink the selection relative to the anchor point (once the drag has been completed) by shift-clicking Shift .

MAKING AN NON-CONTIGUOUS SELECTION

To make a non-contiguous selection, hold down the Alt key as you drag your cursor across the cells you wish to select. This action creates a new anchor point, and you can then drag your cursor or Shift click on the new block.

caveats

- There is no way to deselect a cell in the middle of a selection block.
- If the cells you select contain text, only one of the cells becomes editable. Clicking on it allows you to modify the contents of the edit field. It does not allow you to make a new selection. To make a new selection, you must first deselect the current field. Note: You can always use to make a selection, even starting from an already selected cell.

6 SUB-NETWORKS

6.1 INTRODUCTION

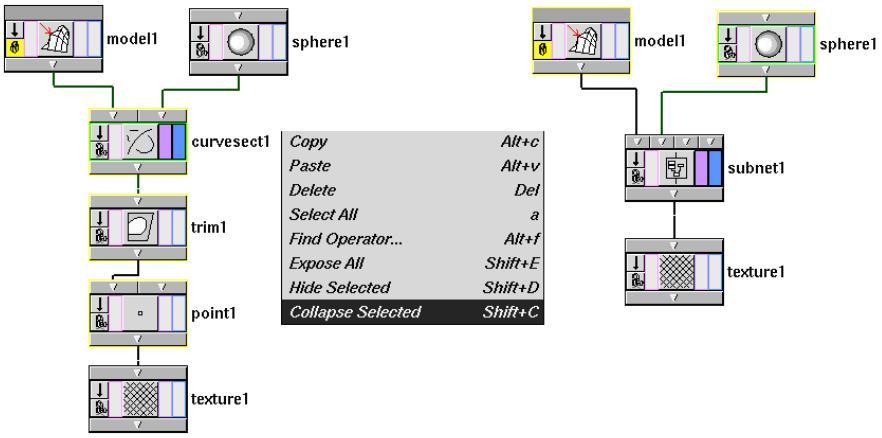
Sub-net(work)s are essentially a way of collapsing several OPs down into a single OP in the Layout Area. A Sub-net SOP / COP / POP can contain an entire OP Network within it, stream-lining and simplifying your network both visually and conceptually.

A Sub-network OP supports up to four inputs. Once you've collapsed a set of OPs placed a Sub-net Operator you can edit it by choosing the *Edit Sub-Network* from the OP's menu or by typing the Enter key when the sub-net is the current OP.

SEE ALSO

- *Properties...* p. 172
- *Create Subnet Operator Type...* p. 183
- *Export Type...* p. 170
- *Scripted Subnets p. 192*

6.2 CREATING A SUB-NET



Select the OPs you want to include in the sub-net, click on an OP, and select *Collapse Selected* from the Network Editor's pop-up menu (with Ctrl).

6.3 OTHER THINGS TO KNOW

SETTING THE OUTPUT

You determine which OP will be wired back to the output connector of the Sub-net OP by setting it with the display flag. The OP on which you set the display flag will be output to the Sub-net OP's output connector in your original network.

ENTERING AND GETTING BACK TO THE ORIGINAL NET

You can enter the Subnet, to change it, by Alt clicking its tile. Navigate back up the hierarchy in the usual manner – select the object above the subnet in the path displayed at the top of the Layout Area.

ADDING LABELS TO THE INPUT CONNECTORS

When you click on an input connector, a label is displayed describing the purpose of the input. You can set these in the Subnet OP's parameters. This makes it much easier for you to remember what each input is used for.

7 SCRIPTED SUBNETS

7.1 WHAT ARE SUBNETS?

Subnets are not to be confused with the OP tiles you place in your network. Rather, they are closer related to the little OP tiles in your toolbar that you never use now that you can type in names. An operator tile is an instance of an operator type, just as my car is an instance of a Ford Contour.

The purpose of Subnets is thus not just to create an interface panel on a subnet. Rather, it is to create an entirely new classification of operator tiles which you can reuse in the future. The primary use is to take a clever series of OPs, collapse them into a single OP with a simpler interface, and then be able to reuse them without having to be concerned with the insides anymore.

Scripted Subnets, are one example of Operator Types. However, so are custom HDK OPs. Even a Houdini built-in operator like 'Cookie' is an operator type.

7.2 CUSTOMIZING THE FILE SOP

Subnets go beyond what was previously possible with Scripted Subnets, our first task will be to customize the File SOP for our own purposes.

1. Start Houdini, enter your favourite SOPs desktop, and place a File SOP.

2. Click with on the File SOP's OP tile and select *Properties...* from the menu. This is where you alter your installation's version of this operator type.

3. In the Properties bar, you have a *Help* button. Also, 3 of 4 of the tabs have a help button. Click on them and read carefully the instructions in there.

As you may have ignored the last step, I'll now give an example of fun stuff to do:

4. In the OP tab, click on the Label and change the text from File to "What me worry?"

5. Click on the icon showing the file picture. In the Icon Browser, jump to *$HFS/houdini/config/Icons/Sample* . Go to the Pics directory and choose the "paint" picture.

6. Switch to the Help tab. You can see a text editor with the current help. Insert as the first line: "File SOPs are for wimps. I make my geometry with Add SOPs."

7. Note the "Path" option is set to: *$HOME/houdini6* .
This is where these changes will go. Leave it as is.

8. Click *Accept*.

9. Now, put down a new File SOP. Note that you have to type "What me worry?" rather than "File" as you have changed the name. Also note the new SOP has a picture as its icon. Click on the help for the SOP and see your admonishment for wimps reflected in its glory!

UNDOING YOUR CHANGES

Well, that's all fine and dandy. But how do you get back to the File SOP you know and love? Fortunately, there is an answer. Go back to the *Properties* page and choose *Uninstall*. Brashly click *Yes* without reading the warning. Restart Houdini, and you'll find your file SOP back to normal.

The uninstall option removes all custom files associated with that Operator Type from the specified directory and higher on the search path. So, if you for example had a custom File SOP operator type in: *$HSITE/houdini* . Uninstalling from *$HOME/houdini* will uncover the $HSITE version.

7.3 EXAMPLE – CREATING YOUR OWN OP-TYPE

This section runs you over a process of building a subnet and creating a new operator type out of it.

1. Start Houdini in a new object.

2. Put down a box, size .5, 1, 0.5

3. Append a Twist, Operation Linear Taper, Strength 1.4

4. Append a PolyExtrude. Group 5, Translate .6

5. On a seperate path, put down a circle. Type polygon, divisions 2, arc type Closed Arc, Arc Angles 0-10.

6. Append a copy, number 5, Z rotation of 72.

7. Append a transform, turn OFF Pivot about Group, translate 0, .8, 0.26.

8. Append a merge and bring it together into a windmill!

9. Now we have a cool windmill. But, as we know, we are always wanting to make windmills! So lets make a new windmill SOP!

10. Select all the OPs you've made.

11. Ctrl-RMB on the network. Choose "Collapse Selected into Subnet".

12. Rename your subnet "Windmill".

13. Now for the cool stuff. This subnet is alright, and you could always cut and paste it into other .hip files. But we want to be able to add controls. And, we want it to show up on the generator list with other key primitives like Platonic Solids.

14. RMB on the windmill SOP tile. Choose "Create Type From..."

15. Read carefully the Help card in the dialog

16. The Op Name is the global name for this new optype. Note that it should be prefixed by your user name. For other prefixes, you can set HOUDINI_DEFOPTYPEPREFIX. This name should be unique across the network type. After I make my jlait_windmill, no one else can install one in my copy of Houdini without causing confusion.

17. The Op Label is how it shows up in the Tab menu and elsewhere.

18. Min and Max inputs shouldn't need to be adjusted. They default to the number your subnet requires, which is 0 and 0 in this case.

19. The Save To directory is where in Houdini to install the new Op. The *$HOME/houdini6* directory is a good default.

20. Click *Accept...*

21. ...and you are back in the Properties Page! Note the OP in your network window has changed its appearance to a generator.

22. Click on the icon again and change it to CHOP/CHOP_fan.bicon.

23. Go to the Parameters tab. Here is lots of cool stuff for making your own UI for this new operator type. Click on Create New and choose Float. Change the name to "bladerot" and label to "Blade Rotation".

24. Click on Create New again and make another Float. Make this guy "bladesize" and "Blade Size". Make the default 1.

25. In the Help tab, describe how to use this Op. Mention that the channels are /bladerot and /bladesize respectively.

26. Click *Accept*.

27. Tab menu and type in "windmill". You will see your new Sop and note it creates the right type. Yay! But still, it is not linked up properly..

28. RMB on your new windmill and choose Parameters...

29. Dive into your new windmill.

30. Go to the Circle SOP. Click on the "Blade Size" word and drag it onto the Radius word. Choose "Relative References".

31. Go to the Transform SOP. Click on the "Blade Rotation" and drag it to the third parameter of the Rotate field. Choose "Relative References".

32. Close the Parameters dialog and go back up to your top level. Play with the Rotation and Size parameters and watch the windmill react!

33. You still need to save these changes. You have only altered your own instance of the Windmill SOP, not the Operator Type. To change the operator type, go back to Properties. Make sure you do it on the SOP you just finished linking up. Note that "Make Parameters Default" is on. This will cause the OP to save out its current values as the default, allowing your changes to become the operator type. Click *Accept*.

34. Put down a new Windmill SOP. Note it can rotate too! Note, however, that the original Windmill CANNOT spin, as it hasn't been upgraded with the new changes. There is no way to do that now.

SAVING YOUR OPERATOR TYPE

The Windmill you just created is actually stored in a lot of different magical files throughout your Houdini installation. Thus, there is no easy way to back up your current version, or hand it to someone else. To solve these problems, the .optype file was developed. Go to a Windmill op and RMB menu it. From the menu, choose "Export Type...". Save it to $HOME/SOP_yourname_windmill.optype.

You can now store this for archiving or CVSing. You can also give it to others so they can enjoy your windmill!

This saves more than just scripted subnets. For HDK Operators, it will save the .so into the proper dso path (eg: dso_nt_intel for Windows). For VEX operators it will save the .vfl and .vex files.

7.4 INSTALLING NEW .OPTYPES

Saving is useless without installing:

1. Go to the properties of your windmill and choose "Uninstall".

2. Quit and restart Houdini.

3. Try and put down a windmill. It is gone! Alas, alack!

4. Go to File::Install Op Type...

5. Find your .optype. The install directory is where to put it. $HOME/houdini5 is fine.

6. Click *Install...*

7. Go to your SOP network and put down a Windmill! Back as good as new.

If you want, you can also install my version of the windmill .optype.

7.5 AUTOMATING INSTALLATION

The described process is fine and dandy when you are installing one or two new .optypes on your own local machine. But what if you need to install a large number? Or do so on a network server without graphics? Here, you can use the commands optyperead, optypewrite, and optypeuninstall. You can also make a .cmd file to batch a bunch of optypereads together to install your sites library of .optypes all at once.

7.6 ADVANCED KNOWLEDGE OF .OPTYPE

The *.optype* files are a very generic file format. They are CPIO archives like Houdini. You can thus run hexpand and hcollapse to open, edit, and close the files. Inside, you will find a number of files in relative directories to where they will be installed. There is also a optype.commandlist file. This file tells the installer how to treat each file.

POSSIBLE COMMANDS

replace	Overwrite any existing file of that name with the one in the archive.
append	Append the file in the archive to the existing file of that name. If none exists, merely copy it.
optypeuninstall	Uninstall the specified optype from the system from the install directory and up.

It is possible to create a .optype file which installs more than one operator type. Merely merge the directory structures of multiple .optype files and append together any files that are duplicated.

8 OTLS – OPERATOR TYPE LIBRARIES

8.1 INTRODUCTION

Files with the extension .otl are known as an "Operator Type Library" (aka 'Digital Assett'). This file contains Custom Operators as they were known in previous versions of Houdini, but with many improvements.

There can be many OTL archives in your file system that Houdini can reference. To manage this, there is a new Operator Type Manager located in the main *Tools* menu.

In addition, your Hip file can contain an embedded OTL file in addition to or instead of an OTL file on disk. This embedded OTL file contains definitions for every operator that is actually used in the Hip file. This OTL appears in the operator type manager in the "Current Hip File" branch under the heading "Embedded". To reduce hip file size and reduce confusion when using externally-referenced OTL files, you can disable the saving of this embedded OTL from the operator type manager configuration page.

Given this, there is the dilemma of having multiple Custom Operators with the same or similar definitions and names contained in several OTL archives as well as internal to Houdini. This means that T.D.s and Systems people will need to devise a plan in larger facilities, but the payoff in time saved due to asset sharing is well worth the initial up-front planning.

USES FOR OTLS

The main use for OTLs is to let a user share content among several hip files, without having to do a lot of copying and pasting. It also lets a TD limit the level of access and control that an animator has on a particular piece of a hip file. This is an incredibly useful piece of functionality. It would be incredibly useful even if we didn't allow any sort of networks within networks (though it's usefulness is greatly magnified by NWN).

8.2 WHAT IS AN OTL EXACTLY?

An Operator Type Library is an archive of other files. The "OTL" archive contains an archive of the actual operator definitions. For those familiar with Houdini 5.5, this means the Index files (like SHOPsurface or SUBobj), the dialog scripts (.ds files) and other related files like the .cmd creation scripts, the .vex/.vfl files (in the case of VEX operators) and so forth. In fact, an OTL can contain anything you want, including arbitrary scripts, even geometry and image files like .rat textures. Basically, anything that a Houdini OP would normally reference from disk can be embedded into an OTL file. Note that adding images and geometry can make your OTL files very large.

To manage the OTL archive loading in houdini we use a file called OPlibraries which points at OTL archives. The HOUDINI_PATH is searched in order and all the OPlibraries files are used. No need for one to reference the other in a cascading

fashion as was the case with SHOP dialog script files. The default OPlibraries file can be found in: *$HFS/houdini* .

In the Configuration menu of the Operator Type Manager, you can set preferences as to which OTL archive to use. It is quite obvious as to what the different options for "Operator definition priorities" does. The third folder is for "Embedded operator type library:" and this manages how Operator Definitions are used in the current Houdini session. It should also be obvious to how these work. There is one more option on the configuration page that let's you specify whether you want to receive a warning dialog when you open a hip file that is using internal operator definitions when more recent definitions are available on disk.

SUMMARY

- OTL archives contain Custom Operators and can be saved to disk or in a single internal OTL archive in Houdini.
- They are picked up by Houdini using files called OPlibraries in the HOUDINI_PATH.
- There can be several identical Custom Operators in respective OTL archive including an internal version to Houdini.

8.3 ONE PROPOSED WORKFLOW

A TD is preparing a character, say a Bird for use in a production by several animators. The TD cobbles together a tube/sphere/box bird with all the animation hooks in place (he is a good TD who maps out the channels on paper first). He packages this up as a Custom Operator in his own OTL archive reserved for development. He also saves the same OTL to a shared OTL archive set up for the animators to see.

The animators get notified that in the shared project OTL archive, the bird is ready to start animating with through e-mail or some other form. They place the bird in the scene and start to animate.

The TD continues to develop the Bird and saves it to his development OTL archive. He also adds some other neat tools to streamline the animation at the animators requests. The TD spends a couple days on this and then updates the Bird Operator to the shared OTL archive plus the other tools.

The animators are then notified that the TD has pushed up a new version of the bird in the shared OTL archive. This will be picked up automatically by Houdini on the next load of the file. Those animators that have "Give Preference to Definitions with Latest Date" as the only preference will see the new Bird immediately. Those animators with the default "Give preference to Definitions Saved in Hip File" will have to update the internal Bird reference to pick up the latest Bird object, which they do using the Operation Type Manager. If the animator is using the internal definitions, but there is a more recent definition available on disk, they will be notified when they open the hip file that they are using an out-of-date definition.

Alternatively, they can use the *File > Refresh Operator Type Libraries* and the new Bird will magically appear in the currently open Houdini session, assuming their preferences are set to "Give Preference to Definitions with Latest Date."

As for the custom developed tools, once they are "Installed" (added to the Oplibraries file), the shared OTL archive tools will become available. Once used, then they too will be saved internal to the hip file if the animator actually has one of the tools in the hip file, assuming they have their preferences set to "Save Operator Definitions to Hip File" which is the default behaviour.

The TD can continue updating the Bird but he makes some errors. He removes some parameters and renames some others. Houdini will be upset about this and complain. If the animators pick up the latest version from the shared OTL archive and it breaks the scene, they can still revert to the internally saved reference in the hip file and notify the TD of the errors.

The TD can have several OTL archives in his path for revision control and have his own HOUDINI_PATH to manage this separate from the animators.

8.4 MORE DETAILS ON DIFFERENT AREAS

OTL ARCHIVE MANAGEMENT

OTL archives are a new file format that saves custom operators and any locked operators (mainly locked SOPs) as .hip pieces in an archive that is somewhat difficult to manage. To make managing the internals of an OTL file easier, there are now two commands that can be run from the Houdini textport (**otexpand otcollapse**) or a single command from a Unix/DOS shell: **hotl**

If you are familiar with hexpand and hcollapse for .hip files, then these commands behave similarly and can be used on OTL archives on your file system. They take an existing OTL archive on disk and save it to a directory that contains the contents plus all of the individual operator definitions for easy management. See the Help for otexpand and otcollapse in the Houdini textport for more information.

If you still want to manage Mantra shaders with your existing tools, you can port this over to support OTL archives to manage your shader groups. You just have to lead with otlexpand, do your stuff, then otlcollapse to complete.

You can still use the old method of managing operators, using index files and .vex files for shaders. For scripted subnets, all the internal Houdini tools now only generate OTL's, but if you create or edit the scripts by hand, you can still use the old-style scripted subnets. This gives you time to ease in to the new way while in production.

Scripted subnets created in OTL archives do not need to use the "referencing" mechanism. If you turn off the "Save Contents as Locked" option in the Type Properties dialog, the subnet will be created the way it used to be, not using the new referencing mechanism.

Warning: Turning off "Save Contents as Locked" will NOT allow you to save Custom Operators with any locked (i.e. locked SOPs, locked CHOPs) operators.

Basically, if you have "Save Contents as Locked" turned on, you can use the ability to update the node if it changes on disk and you can have locked SOPs and CHOPs inside the node that is defined by the OTL. However, you cannot dynamically modify the contents of the OTL defined node (add new Lights for example) and remain “in sync” with the disk file.

If you have "Save Contents as Locked" turned off, you can dynically modify the contents of the OTL defined operator but you cannot save locked SOPs and CHOPs, and if the OTL definition on disk changes and you re-sync, you will lose the dynamically modified nodes internal to the OTL defined operator.

REFERENCING OTL ARCHIVES AND THEIR INTERNAL CUSTOM OPERATORS

When you add a Custom Operator into a Houdini session and the OTL is defined on disk, a reference is made in the Houdini .hip file that points to this OTL and tells Houdini that this OTL is where to find the Custom Operator that you are using.

Only those definitions that are in use are stored in the hip file when you save. All unused definitions are discarded. For example, if you delete the last object called "my_widget" from your scene, save the file, the internal reference to "my_widget" is subsequently lost. You can always use the external version of this tool.

You have control over which definition is used based on:

- definitions from index files
- time stamp
- internal vs. external

A"reference" is a pointer to an external OTL file. A "reference" is set up when you install an OTL. It is "definitions" that get saved to the embedded OTL in the Hip file (the hip file also stores "references", but these appear along side the Embedded OTL in the optype manager, and must be exlicitly deleted to get rid of them).

8.5 INTERFACE CONTROLS FOR OTL

OTL CONTROLS

The control for OTLs is found in:

Menubar > Tools > Operator Type Manager

The Operators folder is the file browser for both external OTL archives and those in the hip file. This will list both the OTL files and their contents, the actual operators.

Use the RMB to get info on each operator.

OPERATOR ENTRIES

Using the RMB on operator entries gives you a menu with these options:

type properties...

Allows you to edit the Custom Operator with a GUI. This contains all the options to change, augment or redefine the Custom Operator in that particular OTL archive

file. This option is only available on the "current" definition of a given operator. If you have two versions of operator "my_operator" available, only one of them will give you the "Type Properties" option.

copy...

Allows you to copy the Custom Operator from one OTL archive to another OTL archive on the hard drive

update time stamp...

Set time to now. This is good if you have the usage based on Time and you want to rearrange the order.

jump to next definition

If you have an operator defined multiple times (i.e. Within the .hip files and on disk or several times on disk in different places) you can use this to jump to the next definition so that you can quickly find all the definitions for a single operator.

edit contents...

This editor allows you to embed any file you want into an OTL. See the Help-> popup in here for more information.

delete

Note: This only works on actual operators in OTLs, not on OTLs themselves!

Delete the Custom Operator from disk. Can't an operator located in Houdini if it is in use. A warning will be given before deletion. File permissions may not allow deleting of the operator, in which case a warning to that effect will pop up.

OTL FILE ENTRIES

Use on the .otl file entry and you get these options:

uninstall (only on otl files)

Uninstall (remove from the Oplibraries file) an OTL file. Does not delete or modify the actual .otl file itself, just makes Houdini unaware of it.

expand/collapse / expand all

Quickly opens and displays or hides the contents of the .otl file.

sorting options

These should be self-evident, sort alphabetically the entries.

There are some UI operations underneath the tree view that allows you to search for any operator given it's name: Find current, Find latest, or Find next definition. This takes Unix style wildcards, so to find the current definition of an operator named "MassiveBrainExplosion" you could type in "*Brain*" and then click Current Definition.

Check the Help > Popup menu for more information.

In the Configuration tab is where you set the priorities for the Custom Operators and also set a couple of preferences.

give preference to definitions from index files

With this on, any operators defined with Index Files (SHOPsurface, SUBobj) will have priority over operators defined in an OTL file.

give preference to definitions saved in hip file

This means that the operator's definition that is in the .hip file has precedence.

give preference to definitions with latest date

This means the newest operator will be used, so if this has changed on disk, the newer one will be loaded and used in the .hip file when the .hip file is opened or when "otrefresh" is typed in the textport.

Note: If all or some of the above are on together, then the priority is the order they are listed. If you have all three turned on, for example, and you have a customer operator named "buddy" that is embedded in the .hip file, and is also defined on disk in an .otl file, the one in the .hip file will be used. However, if the one on disk is newer, a warning will be given telling you that a newer one is available. You can ignore this and still use the one in the .hip file or update to the newer one. This warning requires that the Display Warning for Out of Date Operators option (below) be turned on.

save operator definitions to hip file

All OTL definitions used in the current .hip file will be saved into the .hip file. This can make the .hip file larger but means there is a "backup" of the Custom Operator's definition that can be used to fall back on if the one on disk is rendered unusable for some reason.

display warning for out of date operators

If you load a .hip file (or use "otrefresh") and you have operators that are out of date, but you do not have "Give Preference to Definitions with Latest Date" turned on, then a warning will come up telling you that you might want to update the definitions.

8.6 A LOOK AT THE OPLIBRARIES FILE

- It controls which OTL archives to load in to current session
- HOUDINI_PATH only controls OPlibraries so the .otl files themselves do not need to be in HOUDINI_PATH only the OPlibraries
- cd $HH; cat OPlibraries for an example of the default ones built into Houdini
- All VEX Operators and SHOPs from old versions of Houdini (i.e. 5.5) are converted to OTL archives referenced by the OPlibraries file in $HH for Houdini 6.0.

8.7 FILE MENU OPTIONS

CREATE NEW OPERATOR TYPE

Creates a new operator type and adds it to an OTL archive. This results in an "empty" operator, i.e. A Subnet type would just be an empty OP.

INSTALL OPERATOR TYPE LIBRARY

Provides a UI-based way of adding OTL's to the various OPlibraries files. This assumes that an .otl already exists on disk somewhere and you want to add it to the OPlibraries index file. You can also install old .optype files (from v5) this way, though they will be not be put into .otl files, just expanded like they were in 5.5.

REFRESH OPERATOR TYPE LIBRARIES

This is the same as using "otrefresh" in the textport. It will re-scan any installed OTL files and if there are newer ones it will use them (if that is the preference) or at least warn that newer ones are available. This is what an artist would use when a TD tells them a model or tool has been updated.

8.8 NOTES

INSTALL LIBRARY COMMAND

The menu item **Install Library To >** appears in all dialogs where you can create or edit an operator definition, or otherwise modify an OTL. This allows you to save a custom operator to an existing OTL archive or to a new OTL archive of your preference. If you specify an existing .otl file then the new operator will be installed into that existing file, essentially adding a new operator to that single .otl file. If you type a new name then a new file will be created containing just that single operator.

The special case of Current Hip File Only will only add the OTL internally in the current .hip file.

NESTING CUSTOM OPS

If you are planning on nesting custom ops within other custom ops, then you have to make sure that ALL references are valid. For example, if you have a character that contains a job-wide skeleton set-up operator and a character geo operator all inside a custom object subnet that has the animation interface, then all three operator definitions MUST be present at all times or you will have the missing operators show up as being empty. To set things back to normal, you can install the required OTL's and the operators will magically fill up with the proper contents without even having to restart Houdini.

DEREFERENCING SKELETONS

You can always dereference the skeleton and the object at any time then save the new definition to an OTL archive. This will cause the existing definition for the skeleton and the object not to be used. You can always re-establish the reference for the skeleton and the object at any time.

DIFFERENCE BETWEEN SAVING IN HIP AND INSTALLING

The difference between saving an operator definition in the Hip file, and Installing an OTL into the "Current Hip File Only". The former involves actually saving the operator definition into the OTL called "Embedded", which is the single OTL that is saved into Hip files. There is some danger to doing this, because definitions are only saved to that embedded OTL if the hip file actually contains an operator of that type. If you accidentally save your hip file without an operator of that type, you've lost your operator definition. The other option is saving your operator definition to an external OTL file, but installing that OTL to the Current Hip File only. That way any changes you make will always be saved into that external file, so you don't have to worry about losing them. The disadvantage there is that you don't get the neat capability of using your old Hip files as a backup mechanism, where each version of the hip file contains a slightly newer version of the operator definition.

DEALING WITH VEX OPERATORS AND OTLS

vcc has a load of new options for creating OTLs directly from .vfl files. They are all described (albeit briefly) in the vcc usage message. To simplify, I'll just give some examples:

```
vcc –m vops.otl shader.vfl
```

This creates a new VOP operator and put it in vops.otl. This is exactly what Houdini does now when you choose "Create New VOP Type" from the RMB menu of a SHOP or VOPNET operator.

```
vcc –l shaders.otl shader.vfl
```

This will compile shader.vfl, and put the generated dialog script file, .vex file, and the uncompiled .vfl file into shaders.otl. This saves you from having to worry about editing the index file, putting the .ds and .vex files in the right places, etc. You can also specify on the vcc command line an icon, name, and label for your shader. Why is the .vfl code put in the OTL as well? That leads me to my second point.

In the Type Properties dialog (to open this dialog, use the RMB menu of any operator tile, or the RMB menu of an operator definition in the Operator Type Manager dialog). You can now use this dialog to do almost anything related to editing an operator. For VEX operators you can edit the .vfl code. For VOP operators you can edit the VFL code and the input/output signatures. You can edit handle bindings. Please check out this dialog. Try editing your favorite SHOP (it has to be a SHOP defined in an OTL file for you to be able to edit the VEX code), or your favorite VOP.

Although not directly related to OTLs, this feature was a direct result of changes required to support OTLs: you can now access a variety of file types from web servers. For example, in the file SOP you can put "http://server/file.bgeo". You can also

keep your OTLs on a web server. Just add the line "http://server/shaders.otl" to the OPlibraries file. In most cases (COPs2 being the main exception) you can also access image and texture files this way. In a Decal SHOP you can use "http://server/texture.rat". When you do a render, mantra will download the texture map (i.e. Houdini doesn't inline the texture in the ifd). I'm not sure how useful this feature would be in production, but I thought I'd mention it.

And finally, there is a command line tool called *hotl* which takes the name of an OTL file and outputs the vital statistics for all the operators defined in that OTL. The same data is available in the Operator Type Manager dialog by MMB clicking on an operator type definition.

8.9 EXAMPLE OF USING OTL LIBRARIES

ONE WAY OF USING OTL LIBRARIES...

Start up Houdini, and from the File menu choose "Install Operator Type Library" and install "demochar1.otl" into the current hip file only.

Now create a Demo Character object. I know, it's a hideous box-man, but this is just a proof of concept. Feel free to do some animating. There are handles bound to the IK end effectors for the arms and legs. You can also attach a Pose handle to the spine bone chain (remember to turn off the new "select entire subnet" option on the toolbar to the left of the Viewport).

Notice that you can't edit anything inside the demochar1 object. All parameters are disabled, you can't change display flags, etc. Basically everything is locked. However, in the RMB menu of the demochar, you can choose "Allow Editing of Contents". This will unlock the contents so you can make changes, which you could then save out to the OTL, or to a new OTL. If you make some changes, and then decide you want to throw out your changes, you can choose "Match Current Definition" from the RMB menu, and the contents will be re-synced with the definition. So although it is possible for any animator to mess with the contents of the demochar, they have to make a conscious effort to do so. This provides protection against someone accidentally ruining parts of the hip file they aren't supposed to touch.

And if they do force things and make changes, it's easy to back out by re-syncing the object to the definition. In this particular case I have used this feature to protect the capture frame pose of the demochar. There is absolutely no way that anyone could ruin the capture pose of this object without explicitly doing the "allow edit contents".

Also notice that if you create one demochar and save the hip file, the file size will be about 300K. Now add a bunch more demochars and save the hip file again. The file size will barely have changed at all. This is because we don't actually have to save the contents of each demochar object (because we know they're all the same). This can be a big advantage for large projects where it could save reams of disk space.

So now go to the file menu again and install demochar2.otl, again into the current hip file only. Notice that suddenly all your demochar objects have been updated. The demochar2 definition for Demo Character is the same as the demochar1 version except there are some SHOPs embedded in the skin object, and a shader SOP to

apply the shaders. I hope the advantages here are obvious. Animators will no longer have to wait until a character is completely modelled, textured, captured, etc. before they can start working. All they need is a skeleton embedded in an OTL, and the skin and materials can be added later (whenever they are ready). And it doesn't matter how many copies of the character are in the hip file, or how many hip files contain the character. The skin just needs to be added in one place (the OTL) and suddenly all instances of the character will be skinned and textured.

Now go to the Operator Type Manager from the tools menu. If you expand "Operator Type Libraries"->"Current Hip File"->demochar1.otl, you will see that there are definitions for the Demo Character and for bone. This is because in the demochar object, there are a bunch of bone objects, but on my machine I have bones set up in OTLs. And even though the bones are inside the demochar OTL, they are still updateable in the same way as the entire demochar object. Try putting down a bone object (make sure you still have some demochars in your scene), and choose "Allow Editing of Contents" from the RMB menu. Now dive into the bone, and in the link_and_capture SOP, plug in an extra default sphere SOP. Now choose "Type Properties..." from the bone RMB menu, and hit Apply. All of the bones in all of your demochar objects will suddenly have the extra sphere geometry. The only thing this demonstration shows above and beyond what was in the previous paragraph is that you can build a hierarchy of OTLs. The demochar OTL uses the bone OTL. But the bones inside the demochar are still references, just like the whole demochar is just a reference.

QUESTIONS

First, if you are asking why the user has to do the "Install Operator Type Library" for every hip file that wants to use the demochar, the answer is that they do not. In the Install OTL dialog (and the Type Properties dialog, and every other dialog where you can create or edit an OTL) you have a menu for "Install Library To". This menu lets you decide if you want to "install" the library for the current hip file only, or to your HOME directory, your JOB directory, or anywhere else in your HOUDINI_PATH. This does not mean that the library actually get copied or moved to the directory you specify (or into the hip file). The OTL file stays exactly where it is. When you start Houdini, we scan the HOUDINI_PATH for files called "OPlibraries". This file is simply a text file with a list of OTL file names. What that "Install Library To" option does is specify in which directory you want to modify the OPlibraries file. If you specify the "Current Hip file Only" option, it creates a section in your hip file called .OPlibraries that gets treated like any other OPlibraries file when you load the hip file.

Second, you may be asking about the fact that all operator definitions used in a hip file are saved to a special embedded OTL inside the hip file. When you reload the hip file, by default the definition saved in the hip file is used, not the one in the OTL file on disk. However, if you look at the Configuration page of the operator type manager, there are several options that let you achieve a variety of different behaviors. First, you can turn off the feature that saves the embedded OTL to the hip file ("Save Operator Definitions to hip file"). If you turn off this option, your hip file will be smaller, and Houdini will be forced to use the definitions from the OTL files on disk. The disadvantage is that if you send the hip file to someone else who doesn't have the same OTLs on their machine, your hip file may not function properly.

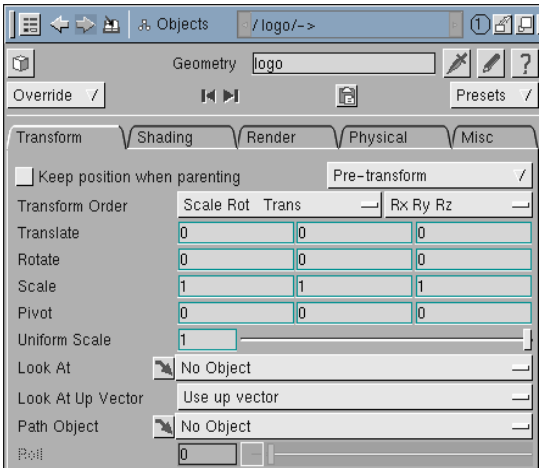
Another option is to turn off the "Give preference to definitions saved in hip file" option. If this option is turned off, then the definition saved in the HIP file will only be used if no definition can be found in an external OTL file. Basically, this says "use the embedded hip file definition only as a last resort". If this option is on, and a definition for an operator is found in the hip file, Houdini will use that definition even if a newer definition is found in an OTL on disk. However, if the "Display warning for out of date operators" option is also turned on, whenever you load a hip file, Houdini checks all the embedded operator definitions, and looks for more recent versions of those definitions in external OTL files (it only checks those files listed in the various OPlibraries files. Under no circumstances do we ever scan your disk or your HOUDINI_PATH for OTL files).

If it finds a more recent version on disk, it brings up a dialog telling you which embedded operator definitions are out of date and lets you ignore the warning, use the external definitions instead of the embedded definitions, or open the OTL manager so you can configure things manually.

8 Parameters

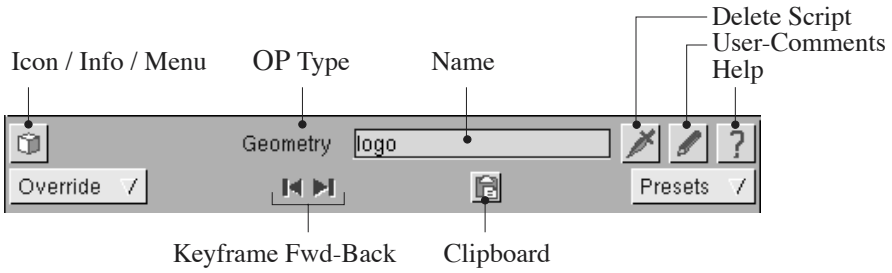
1 MODIFYING PARAMETERS

When you click on an OPerator in the Layout Area to make it the current operator, its parameters are displayed in the Parameter Area.



Parameters give definition to what an OP will ultimately do or look like. In the example above, a Sphere SOP, parameters include: the radius, the center of the sphere, and the number of rows and columns in the sphere's mesh. If a parameter is not valid or applicable, it is greyed-out.

1.1 THE PARAMETER HEADER



The Parameter Header is always present at the top of any Parameters Area. It provides you with access to the following controls and indicators which are applicable for every Operator.

Tip: If you enter a value you don't like in one of the parameter fields, Click with the right mouse ( ) in an edit field to undo the last parameter change.

ICON / INFO / MENU

Displays the icon for the OPerator. Clicking the icon with displays OP-specific help with a listing of all the parameters and their channels (if any).

Clicking with gives you info on the contents of the OP node (see *Information Pop-up (on Icon)* p. 176).

And clicking on the icon gives you the same menu you get from clicking on the OP's tile in the Network Editor.

OPTYPE / NAME

This edit field displays the name of the operator. By default, it is the operator type and a unique number (e.g. the first Sphere OP is initially named "sphere1").

Change the name by selecting the text in the field, and by typing a new name. Use only alphanumeric characters. Alphanumeric characters include upper and lowercase letters from "A" through "Z", and the digits "1" to "9"; spaces are not allowed.

DELETE SCRIPT

Click this icon to view the Delete script. This script will be executed when the OP is deleted.

COMMENTS

Clicking the Pencil icon pops up a display with comments. The button changes to show a piece of paper on top of a pencil when there is a comment.

If you select *Edit Comment...* a comment dialog appears into which you can type information you want to associate with the operator.

The comments you attach to a particular operator stay with it. Activating another operator changes the Comment window's display to those of that of operator. The operator comments are also displayed when you click an OP's Info button (click with on the OP tile).

HELP

Provides OP-specific help with a listing of all the parameters and their channels.

OVERRIDE

When CHOPs are exported, they override the individual parameters values.

KEYFRAME BACK / FORWARD

Moves forward/backward to the next/previous key of the OP whose parameters are displayed. Keys are the defining points in time at which a new expression takes control of a parameter's value. Initially there are only two keys: at the first and last frames of the animation.

CLIPBOARD

Displays the values, expressions, and references of parameters that have been Copied to the Clipboard (use on a Parameter, and select *Copy Parameter* to do so).

PRESETS POP-UP MENU

A *Preset* file contains values for the parameters of an operator.

load

Loads a presets file previously saved with *Presets > Save As...*. This restores the values entered in the parameter edit area to those at the time of the save.

save as...

Presents a *Save* dialog box in which you are prompted for a filename. Once a filename is entered, the values entered into the edit fields of the Parameter Area are stored into the file, including the values of animated channels.

defaults

Resets the values in the Parameter Edit Area back to their defaults.

Tip: The presets file is a text file which can be edited manually with a text editor. You can delete some parameters so that only some parameters are over-written during a load.

2 CONTROLS

Parameters can be changed via the various controls in the Parameter Area. These controls include the following:

2.1 PAGE TABS

When there are more parameters than can fit into the parameter area, Page Tabs allow you to switch between pages of parameters. Click on a page tab to bring its page of parameters to the foreground.

click to activate the “Render” page

2.2 PARAMETER POP-UP MENU (ON PARAMETER NAME)

TOGGLE EXPRESSION

Keyframe Jumper

If you have entered an expression such as: *sin($F*10)* in an edit field, you may either want to see the actual expression, or the value computed by the expression at a given frame. Clicking on the name of the parameter ( ) toggles between the expression and the value the expression produces. When viewing the expression, Houdini will highlight the name of the parameter.

entering expressions as parameters

Entering mathematical expressions as parameters within edit fields allows you to animate the geometry definitions in time according to the formulae you enter.

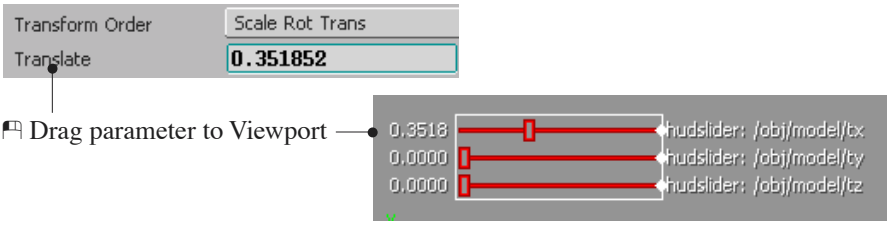
For example, to make the radius of a tube oscillate open and closed over time, you could enter the formula below into a Tube SOP’s *Radius* field:

```
abs( sin($T*180) )
```

You can see the effect of the formula by clicking *Play* button in the Playbar, and watch the result in the Viewport. To see the actual animation curve generated by the formula select *Scope Channels* in from the parameter’s pop-up menu using .

For more information on expressions, see the *Expression Language* section.

HUD SLIDERS



HUD Sliders (Heads-up Display) allow you to tweak regular parameters directly within the Viewport. To use them, simply drag-and-drop a parameter's name into the Viewport. Select *HUD Slider* from the prompt, and a HUD Slider for that parameter will be created. When keyframes have been set, the slider handle becomes red to indicate the presence of a channel.

Click with on a slider to get a menu where you can *Lock* and *Unlock* the slider; *Dock*; and set *Handle Parameters*. You can also edit and delete sliders using the *Persistent Handle Editor* (from the Tools menu).

STEP TO PREV / NEXT KEY

The value of a parameter may change over time. To hold the changing values of parameters over time, Houdini uses a "channel". Within a channel, are keyframes – certain defining values at certain frames, and the values imbetween these frames are computed (i.e. "interpolated") using segment functions. To move back and forth between these keyframe values, you can *Step to Prev Key*, or *Step to Next Key*.

You can also do this my clicking on the *Keframe Jumper* to go back a keyframe, or clicking on the keyframe jumper to go forward a keyframe.

SET / REMOVE KEYFRAME Alt / Ctrl

To add a new keyframe in the channel of values defined for a parameter, move the playbar to the desired frame, enter a value, and select *Set Keyframe*. The value will be added into the channel as a unique new defining value (i.e. "keyframe") at that frame for that parameter.

To remove the keyframe value, jump back or forward to it, and then select *Remove*.

YANK PARAMETER

Copies the select parameter value to the clipboard.

PUT YANKED VALUES / EXPRESSIONS

Pastes the yanked values (the computed result of an expression), or the actual expression into a parameter field.

PUT YANKED REFERENCES

If a parameter was determined by a channel reference to a parameter elsewhere in Houdini (e.g. *ch('/obj/geo5/tz')*), then this will paste the reference into the parameter field.

SCOPE APPEND CHANNELS Shift

Displays the the channel values for this parameter in addition to the ones already being displayed.

SCOPE CHANNELS Shift

Displays the channel values for this parameter, no longer displaying whatever else may have been displayed in the Channel Editor previously. Pops up a Channel Editor pane in a floating window if no pane has been specified as a Channel Editor.

DELETE CHANNELS

Removes time-based changes to the parameter altogether, making it a static, constant value for the entire length of the animation.

2.3 EDIT FIELDS

To change a value in an edit field, select the text within it, and type a new number from the keyboard. The new number replaces the selected text. Double-clicking selects all the text within the field.

SPECIAL CONTROLS

- Undo last entry and revert to previous value.
- Pops up a value grid with which to adjust numbers.
- Alt E within an edit field to edit field text in an editor (*vi*).

MEANING OF COLOURED FIELDS

Sometimes edit fields are coloured. Here's why:

Light Green — The parameter is controlled by an expression (e.g. $F), and is on a keyframe.

Light Blue — The parameter is controlled by an expression (e.g. $F), and is within an interpolated frame, or the channel value at current time is disabled from the dopesheet.

Red — The enetered parameter has been changed, but not yet registered as a keyframe.

Orange — The parameter is currently controlled by a CHOP. Go to CHOPs, and change the export path to change.

CUT AND PASTE

- Alt X cuts the selected text from the field.
- Alt C copies selected text in the edit field.
- Alt V pastes text into the edit field.

cut and paste notes

The cut and paste commands also work with the X11 clipboard mechanism (click with middle mouse to paste) allowing you to move text between Houdini and other applications.

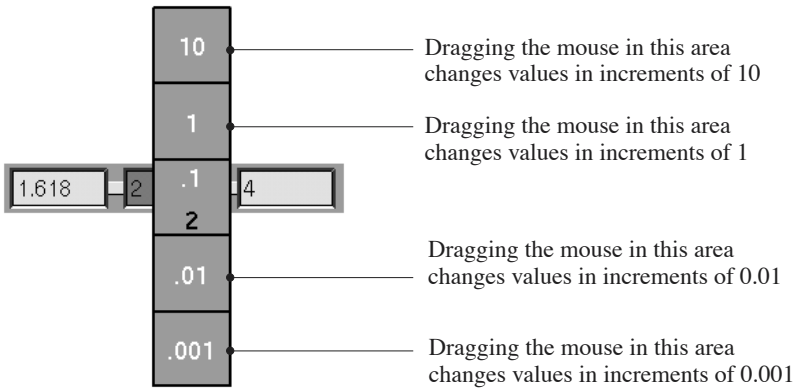
Tip: You can change the editor invoked with Alt E by setting the *$EDITOR* environment variable. For example, you could add the following line to your *.login* file:

```
setenv EDITOR vim
```

Because a text-based editor is assumed, a textport is always started; therefore you must use a text-based editor such as *vi* or *vim*. If you want to use *jot*, you need to use: *setenv EDITOR 'jot -f -p 4,816,95,195'* which will get *jot* to run in the foreground in a sized window – although the textport will still be present.

2.4 VALUE GRID

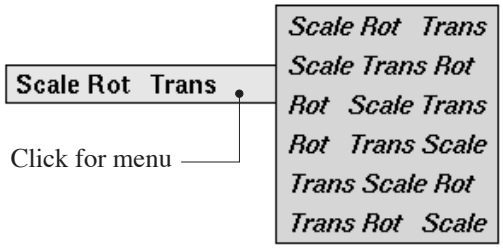
Clicking within an edit field with the middle mouse ( ) in an edit field pops-up a value grid similar to the one below (Some fields may appear with different values).



- Dragging and releasing the mouse through this grid allows you to dynamically change the value of an edit field.
- If you pop-up the value grid by clicking on the name of the parameter itself, you can change the values of all three fields simultaneously.
- While the mouse is within the value grid itself, it will be in a "dead zone" – in which you can pick the range to change by. Once outside the dead zone, mouse movement will affect the value in increments of the range selected within the dead zone.
- You can re-enter the dead zone at any time to select a new increment range.
- Using a Mouse wheel on XCF field affects the numeric portion.

2.5 POP-UP MENUS

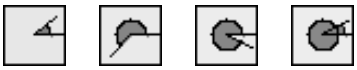
When several options are possible, they are often listed in a pop-up menu. Click and hold the mouse button on the menu, then drag the mouse to the desired item – so it becomes highlighted – and release the mouse button.



If a menu contains more entries than will fit on the screen, it will display a message in the menu that says, “Menu too large...”. Choosing this entry presents you with a dialog which contains a scrolling field of the menu entries and a *Cancel* button. Selecting an entry from the scrolling field is equivalent to choosing the entry from the menu.

2.6 ANGLE CONTROL ICON

Edit fields which accept angular measurements have an angle control icon next to them. You can either type the number of degrees in the edit field, or click and drag on the angle control icon to quickly set the number of degrees graphically.



Tip: Normally, angles are constrained to 5° increments. Holding Ctrl while dragging removes this constraint.

2.7 VALUE SLIDERS

If an edit field accepts values only within a certain pre-defined range, then a value slider is often provided next to the edit field to provide rapid input of values. Change the value of a slider by grabbing its thumb and dragging along its length.

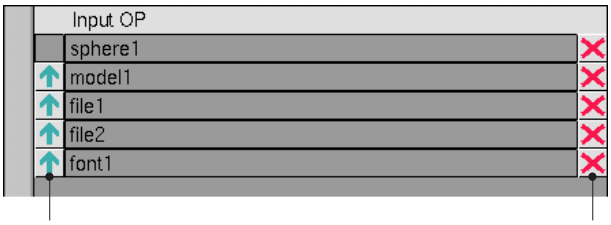
Typing a value in the edit-field can override the predefined range.

2.8 POLAR JACK

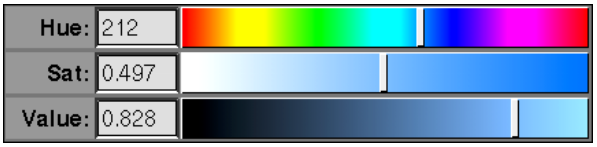
The Polar Jack is somewhat like a wee joystick which allows you to interactively modify a directional vector by dragging the small point at the intersection of the two rings. The resulting X, Y, Z components of this vector are shown in the fields beside the jack. You can also edit these fields directly by typing values into them.

2.9 INPUT SWITCHER

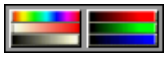
Some OPs, such as the Merge SOP, Switch SOP, and the Sequence TOP, allow multiple inputs to their single input connector. In such a case, an Input Switcher appears in the Parameter Area allowing you to reorder or disconnect any of the inputs.



2.10 COLOR CONTROLS



HSV / RGB MODE



Click one of these two buttons to select between HSV and RGB color systems. The color sliders update between RGB and HSV accordingly.

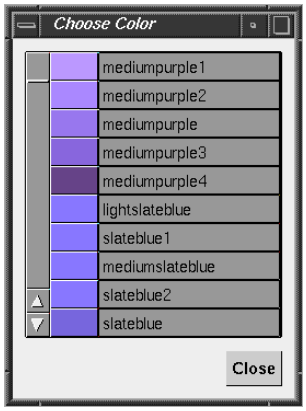
HSV BEHAVIOUR

When specifying colours with the HSV system – the three sliders determine amount of Hue, Saturation, and Color Value. This makes it easy to set colors that are the same shade or brightness.

RGB BEHAVIOUR

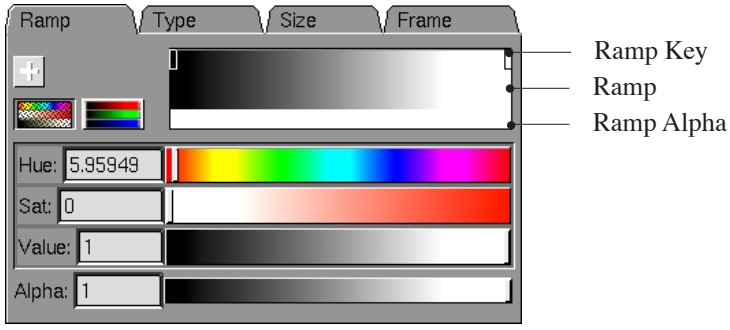
When specifying colours with the RGB system – each of the three sliders determines the amount Red, Green, and Blue contribution to the colour respectively. Sliding one of the bars immediately shows the effect on the color ranges available on the other two bars. On each color bar, every horizontal position shows the color that will result if you move the slider thumb to that position. Thus, if you see the desired color on any of the RGB bars, simply click on that location to get that color.

COLOR LIST



Click the ✣ button beside the *HSV* button to display a dialog from which you can choose from a range of pre-defined colors.

2.11 COLOR RAMP



Color ramps are used to edit continuous gradations of color. With it, you interactively choose and modify the colors (and their gradations) on a Ramp. You can also modify the alpha (transparency) of the ramp.

RAMP OPERATIONS

- Change the color of an existing Key by clicking on it – the Color controls below the Ramp update to reflect the color of that Key. Change the color and alpha of a key by using the Color Controls.
- To add a new key, click on the ramp between two existing keys.
- To delete a key, click and drag it outside the Ramp area.

  Ramps whose starting colour match the ending colour can be created by moving the last key in the ramp away from the right-most end of the ramp edit area. Alternatively this last key can be deleted by dragging it outside the ramp area and releasing it, followed by the insertion of new colour keys in the interior of the ramp edit area.

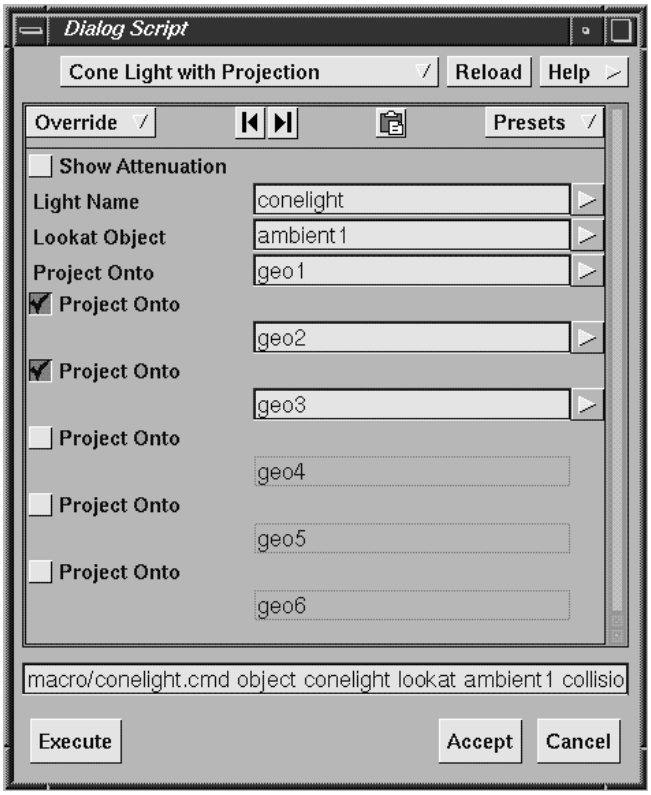
RAMP ALPHA

The *Ramp* area contains a strip along its lower edge which specifies the alpha for the ramp. For this reason, the strip is usually entirely white – denoting no transparency for the entire ramp range. You can add keys into this area to set varying levels of alpha (transparency) along the ramp.

3 DIALOG SCRIPTS

3.1 GENERAL DIALOG SCRIPT FEATURES

Dialog Scripts are found throughout Houdini. They can be accessed by clicking on the ✣ button in the parameters area. The *Macro* option in the *Edit* menu also invokes a Dialog Script. Common parameters are outlined below. For information on how to edit Dialog Scripts, see *Dialog Scripts* p. 198 of the *Scripting* section.



SELECTION MENU

Clicking on this pull-down menu displays a list of available scripts. The scripts can be added to and modified by changing the parameters below.

RELOAD

The *Reload* command is used when you have edited a dialog script in *$HH/config/scripts* and wish to see the changes reflected immediately in the dialog. This is particularly important if you have been generating dialog scripts with *rmands* (see: *Scripting > rmands* p. 205), as it avoids having to restart Houdini.

HELP

Displays the contextual help available for this dialog.

PRESETS

The *Presets* put-down menu allows you to *load*, *save*, or *reset* the parameters for this dialog.

UNDO

This option undoes the last action you performed.

PARAMETERS

Depending upon the script that you select, this area will display different parameters that are available to be edited. Parameters can be added to this area by editing the Dialog Script in: *$HH/config/scripts* .

SCRIPT EDIT FIELD (SHADERS)

This button is only visible in the RenderMan shaders dialogs. It allows you to edit the actual RenderMan shader script (and options) to be executed.

ACCEPT / CANCEL

Accept implements the macro and closes the dialog, while *Cancel* simply closes the dialog.

EXECUTE (MACROS)

Execute implements the script while keeping the dialog open for further modifications to the script's parameters.

RENDER (SHADERS)

This allows you to render using the selected output driver without having to close the dialog.

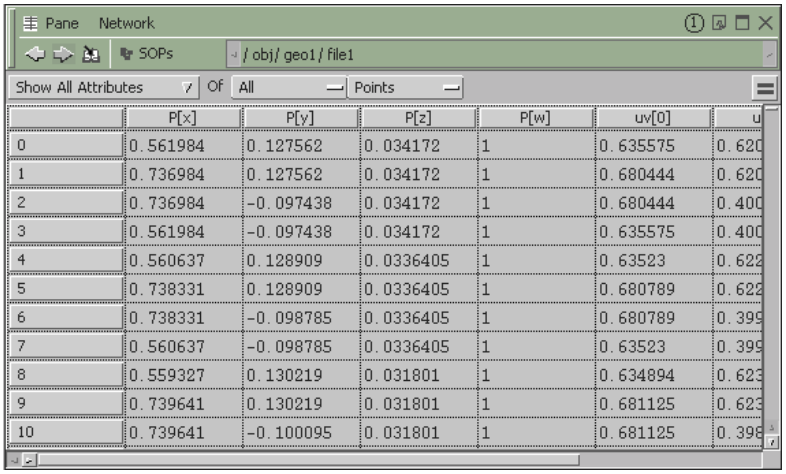
3.2 EXAMPLE – CONE LIGHT MACRO

In the Cone Light macro, the angle is used to control the Field of View. This macro sets up a cone light where its field of view is based on the cone angle. This is accomplished with an expression for the light's *Aperture*. The formula used is:

```
aperature=ch(focal)*(2*tan(ch(coneangle)/2+ch(conedelta)))
```

9 Geometry Spreadsheet

I INTRODUCTION



	P[x]	P[y]	P[z]	P[w]	uv[0]	u
0	0.561984	0.127562	0.034172	1	0.635575	0.620
1	0.736984	0.127562	0.034172	1	0.680444	0.620
2	0.736984	-0.097438	0.034172	1	0.680444	0.400
3	0.561984	-0.097438	0.034172	1	0.635575	0.400
4	0.560637	0.128909	0.0336405	1	0.63523	0.622
5	0.738331	0.128909	0.0336405	1	0.680789	0.622
6	0.738331	-0.098785	0.0336405	1	0.680789	0.399
7	0.560637	-0.098785	0.0336405	1	0.63523	0.399
8	0.559327	0.130219	0.031801	1	0.634894	0.623
9	0.739641	0.130219	0.031801	1	0.681125	0.623
10	0.739641	-0.100095	0.031801	1	0.681125	0.398

The geometry spreadsheet allows you to edit the coordinates of any point or set of points in the Model Editor in a spreadsheet.

The behaviour of the cells in the Spreadsheet is the same as the Object Editor's Object List. For a comprehensive discussion of how to select and modify the information contained in a cell, see *Tables and Lists* p. 187 in the *Interface* section.

SELECTED

Allows you to view all or only selected geometry.

POINT / PRIMITIVE ATTRIBUTES

This menu allows you to select either point or primitive attributes for editing.

ATTRIBUTES

This menu's three options allow you to show or hide all attributes or switch between existing attributes.

1.1 OTHER CONTROLS

GEO SELECT BUTTON

For both SOP and POP viewing, a *Geo-Selection* button is provided. Clicking the right mouse ( ) brings up a menu where the Selection Rule can be specified (just like the Selection Rules in the regular viewers' Select state). This menu also includes a checkbox labelled *Auto*. To use it, select regions in the spreadsheet the usual way (dragging the mouse to create a rectangular region). Only the rows are relevant to the geometry selection because they represent the points/primitives. If *Auto* is enabled, the geometry selection will be transferred immediately after you release the mouse button for the rectangular region. (The points/primitives are added/removed/toggled/replaced depending on the Selection Rule). If *Auto* is not enabled, you have to press the Geo Selection button with the *left* mouse ( ) button to apply the selection.

VIEWING POPNETS

When viewing a POP network in the spreadsheet, the path will always only display "/popnet1" or whatever the name of the POPNET node is. Basically, you're viewing the POPNET, and implicitly you're seeing the output of the cook POP. The browser won't allow you to select a specific POP node to view.

POP RESET BUTTON

This button, only available when viewing a POP spreadsheet. Use it to quickly jump to the first frame of the range (as opposed to dragging the Thumb of the Playbar there manually).

This is necessary when working with POPs, and the network topology or parameters of the POP network have changed. When this happens, the Viewport becomes "stale", and only a reset of the simulation will provide accurate results.

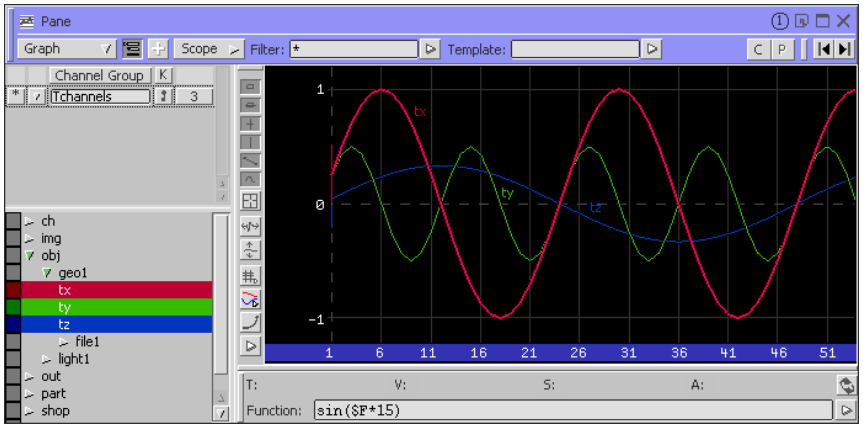
10 Channel Editor

1 CHANNEL EDITOR

Channel Editor panes provide you with an immediate means of creating and editing animation channels. A floating Channel Editor window can be invoked from just about anywhere in Houdini by selecting *Scope Channels* from any parameter's pop-up menu (click with on the name of the parameter in a Parameter pane). This lets you animate the parameter. To delete a parameter's channel, select *Delete Channels* from the pop-up menu.

Note: For greatly increased amount of flexibility in how you can deal with channel data, you should use the CHOP Editor, discussed in the *Channels* section.

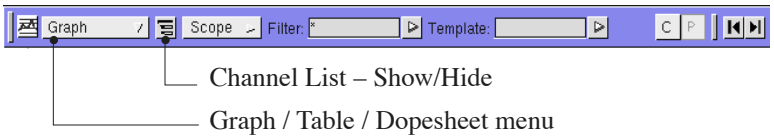
1.1 INTERFACE



You can edit the channels themselves using a Graph view, Table view, or Dope Sheet by selecting from the pop-up menu in the Toolbar.

2 GRAPH TOOLBARS

2.1 TOOLBAR CONTROLS



GRAPH / TABLE / DOPESHEET MENU

Switches between Graph, Table, and Dopesheet editing. For specifics, see:

- *Interactive Editing Procedures* p. 236
- *Using The Table Editor* p. 239
- *Dope Sheet* p. 241

CHANNEL LIST – SHOW / HIDE

Clicking this icon shows / hides the *Channel List* p. 244.

FILTER / TEMPLATE FIELDS ("SCOPE")

The *Filter* and *Template* fields allow you to quickly *scope* items listed from among all available channels (i.e. operators that have one or more channels associated with them). This menu will scope the entire collection or add the collection to the scope if the Shift key is held down.

using the scope

By typing in a string in this field, you can "scope" channel names to specify which channels are to be modified. Any channels not matching the pattern (default '*') are dotted and their keyframes hidden.

This string is pattern-matched, so you can use shortcuts like: *t** to select all channels starting with *t*. The ▷ pop-up menu to the right of the edit field provides some useful preset entries for pattern matching.

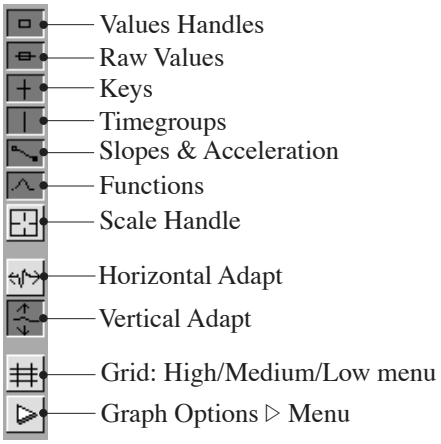
C - COPY / P - PASTE BUTTONS

The *Yank* button copies the values of all the active channels at the current time.

The *Put* button is subsequently enabled and will paste the yanked values to the current time – creating keyframes if required. The yanked values are displayed as tick marks on the current time bar.

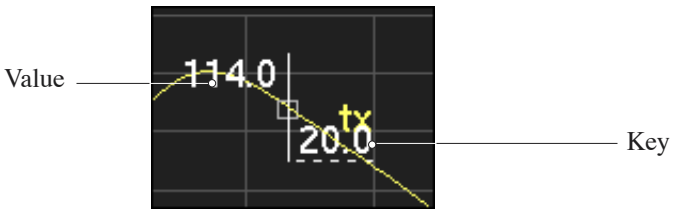
2.2 TOOLBAR ICONS

Along the side of the Graph, a series of icons are provided that perform various functions on the selected channels. The buttons are as follows:



VALUES HANDLES

This command toggles the display of value handles on or off. The Value Handle is displayed as a box if the values at that key are tied together, or as two independent triangles if they are not tied together.

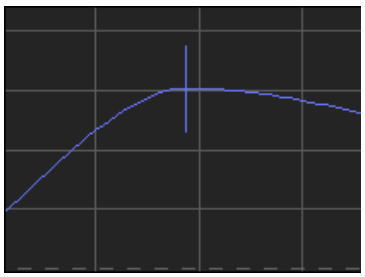


If you select one of the values by clicking on it the value of that key will be displayed in the *Value* text field. You can enter exact values in this text string and the selected value will change. If you have picked more than one key for editing then the Value text string will display "---" as each selected key has a different value. You can change them all to the same value by entering a new value into the text string.

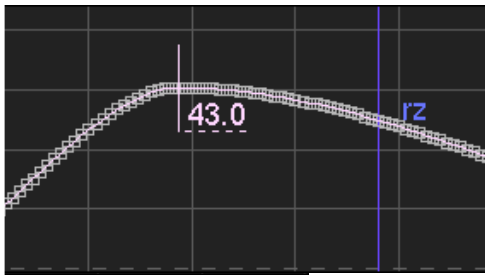
RAW VALUE HANDLES

The Channel Editor supports *raw()* segments and allows editing of raw values by providing a value handle for each raw sample.

To display raw value samples at each frame, type *raw()* in the *Function* edit field after selecting a segment.



Without raw values displayed



With raw values displayed

The segment you selected should now have value box indicators along its length. You can edit these values individually or in groups. To change an individual frame's value, grab the box with your mouse and reposition it vertically. To alter the values of a group of frames, click and drag your mouse across the range creating a marquee box around the group of frames whose values you wish to edit. Grabbing any of the values in the selection moves the entire selection.

KEY HANDLES

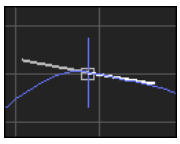
This option turns the display of the keys on or off. The key is displayed as a vertical line. While displayed, keys can be edited by being dragged horizontally. A key that is positioned between two other keys can only be moved within the bounding area defined by the keys on either side of it.

Note: Time values are displayed on the graph with a dashed line under them. When the *Snap to Frame* option is enabled, keys will move to the nearest frame time once you release the mouse button. You can select several keys with the Shift or box pick to move them together.

TIME GROUPS HANDLES

This option turns the display of time groups on or off. The time group is displayed as a vertical yellow line. Time marks that belong to the time group are indicated by a horizontal dashed yellow line connecting the two. While displayed, time groups can be edited by being dragged horizontally. Like keys, when the *Snap to Frame* option is enabled, keys will move to the nearest frame time when you release the mouse button. You may also select several time groups at a time using the Shift or box selection methods to move them together.

SLOPES AND ACCELERATION HANDLES



Slope Handle

With the slopes turned on you can modify the slopes at the end points of individual segments. Slope handles appear as straight white lines originating from the segment end points.

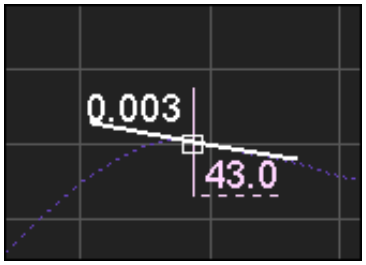
Only certain functions, such as the *cubic()*, *bezier()*, and *quintic()*, have slope handles. Functions, like *bezier()* and *quintic()*, also use acceleration. If you are using this type you can change the acceleration of the function by clicking on the diamond

shaped box located at the end of the slope handle and changing the length of the handle.

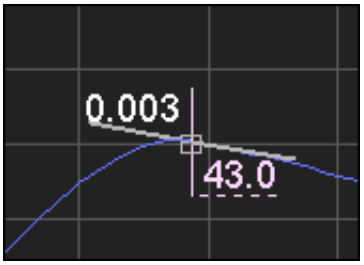
The scale of the handles can be adjusted for greater or lesser control over the curve using the toolbar's handle-length buttons.

FUNCTIONS

The *Segments* option enables you to select segments (the portions of the curve that lie on either side of a key) of your timing curve for editing. For further discussion of segments see *Segments* p. 236 above.



Segments (Off)
Dashed Curve Line



Segments (On)
Solid Curve Line

To edit a segment's function, click on the segment curve. The functions edit field, located at the bottom center of the Channel Editor, becomes active when you select a segment, allowing you to change the function by typing in the field or by using the menu of segment functions to its right.

SCALE HANDLE

Allows you to scale and transform blocks of keys within the graph.

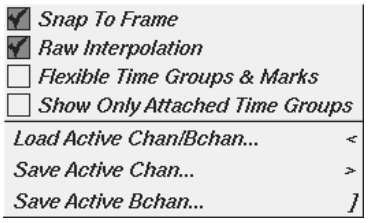
HORIZONTAL / VERTICAL ADAPT

Automatically adjusts the graph's horizontal and vertical scales to automatically fit the values currently displayed.

GRID DISPLAY MENU

Set this to display a grid on the graph: high, medium, or low density of lines.

2.3 GRAPH OPTIONS ▷ MENU



SNAP TO FRAME

When the *Snap to Frame* option is turned on keys will move to the nearest exact frame time after the mouse button is released.

RAW INTERPOLATION

Raw Interpolation is on by default and means that raw arrays are stretched to the corresponding segment length and resampled with linear interpolation between samples at each frame. When *Raw Interpolation* is off, raw arrays are cropped or extended to the new sample length. When extending raw values, the first or last value of the original array is replicated. Cropping does not delete values until the channel is saved, at which time samples outside the crop area are discarded. The state of the *Raw Interpolation* flag is not saved and restored between sessions.

FLEXIBLE TIME GROUPS & MARKS

If enabled, this option will treat the distance between the key and the time group it belongs to as flexible. For example, if you are adjusting the position of the time group indicator by dragging it horizontally across the frame range and move the indicator close to an already existing key, the time group's keys will move closer to the time group indicator, in effect squeezing themselves against it. If this option is disabled, the distance between the time group indicator and its keys remains fixed, allowing you to move no closer to the preexisting key.

SHOW ONLY ATTACHED TIME GROUPS

This option only displays those time groups that have keys as members of the channels that are currently shown in the channel list.

LOAD ACTIVE CHAN / BCHAN... <

The Channel Editor loads and saves files containing per-frame channel values (referred to as "raw" channel data). The default filename is *untitled.chan* . The raw channel loading works in the same fashion as the saving in that it affects the visible channels across the visible time range.

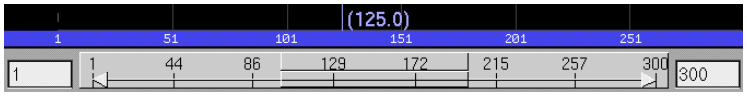
Only segments whose function is *raw()* are affected by the load operations. Columns in the file are assigned to channels in the order that they appear in the list. Samples in the file are assumed to be one per frame. If the channel is using a different sample rate then sampling errors will result. Each file value is assigned to the closest channel value.

SAVE ACTIVE CHAN / BCHAN... > /]

The visible channels can be evaluated and saved over the currently visible time range (one sample per frame per channel) into an ASCII.chan file.

3 FRAME SLIDER AND FUNCTION FIELDS

3.1 FRAME RANGE SLIDER

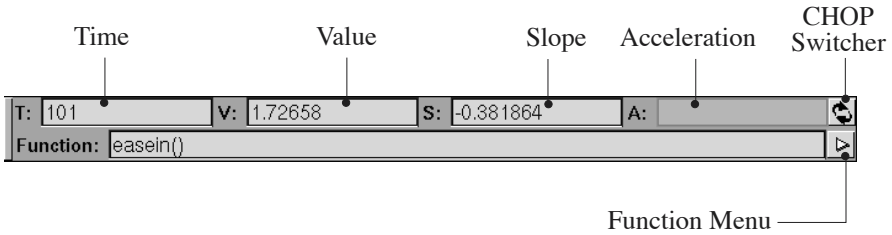


The horizontal slider at the bottom of the Graph allows you to adjust the horizontal display scale of the channels being edited. By moving the arrows found at either end of the time scale (frame numbers) you can expand or compress the amount of the channel that is displayed in the Graph.

With a smaller section of the graph selected you can click and move the slider bar along the time scale to view specific sections of the selected channels. This feature is helpful for editing complex sets of channels or for editing long time scales. The numeric value of the displayed start and finish frame numbers are displayed at the left and right side of the horizontal slider. This range also selects the frame range used when saving .chan and .bchan files.

Clicking within the blue area positions the Playbar to where you clicked. Dragging allows you to more precisely position the new current Playbar location.

3.2 FUNCTION EDIT INPUT FIELDS



The input fields allow you to enter precise values, including your own function expressions, for each of the channel's variables.

CHOP SWITCHER

A switcher changes the channel editor (T:V:S:A:) options, to reveal options interactive modification of keyframe data from a CHOP source. To use this feature, you first specify a CHOP node to update from. The export name of the track must match the scoped channel exactly. Then, the *Live* button must be enabled. After that, whenever the CHOP data of the matching track changes, the selected keys will be relatively moved. In between CHOP track data changes, different keys may be selected.

TIME (T:)

The Time field displays the numerical frame value of the selected key or time group. If more than one key or time group has been selected, this field displays "---". Editing the value in this field affects all selected keys and time groups.

VALUE (V:)

This field displays the numeric representation of a selected segment's in or out value. With the *Values on/off* option turned on you can click on any value handle and the precise value appears in the text string. You can edit this value by entering a new one in the edit field.

SLOPE (S:)

This field displays the value of the slope for the selected curve segment.

ACCELERATION (A:)

This field is active only with bezier and quintic function types.

FUNCTION:

When you are editing functions the *Function:* edit field displays the interpolation or the mathematical function that governs the selected segment. You can adjust the mathematical expression of any one or more selected channel segments by typing in the text string in the edit field.

These segment functions will most often describe the behaviour of your object while in motion. The examples that follow are based on translating an object, in this case a cube, two units along the positive X axis. That is, moving it to the right across the Viewport.

scripting language support

There are several scripting language commands that deal specifically with channels and can help simplify the process of editing channels. For more information about channel specific scripting language commands see the *Scripting* section.

expression language support

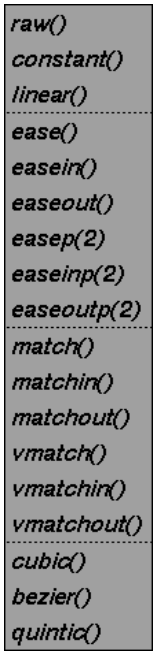
Houdini's expression language contains several variables that are specific to channel editing, for more information see *Channel Interpolation Functions* p. 18 in the *Expression Language* section.

FUNCTION MENU

Select an interpolation function from this pop-up menu.
See the next section – *Function Menu* p. 232 for a complete description.

Note: Channel Editor Preferences are now in: *Animation Preferences* p. 100.

4 FUNCTION MENU



The pop up menu to the right of the *Function:* edit field contains several short-cuts for inputting the most commonly used interpolation functions. The number of possible functions is not limited to the ones given here – you can write your own custom functions by typing them into the *Function:* edit field.

Some of the most common (but not all) these functions are illustrated below. For a complete list of segment interpolation functions and descriptions, refer to the *Expressions* section *Channel Interpolation Functions* p. 18.

FUNCTION MENU PRESETS

RAW()

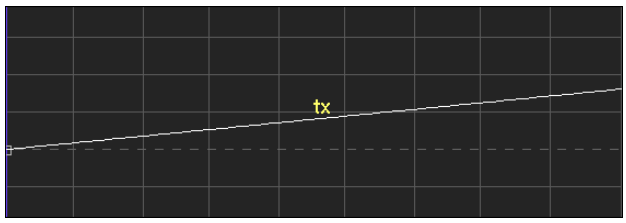
The *raw()* function is special in that it converts a segment into a series of discrete values – one for each frame. For example, instead of having a single mathematical function like *cubic()* that interpolates and determines the values along the segment, those individual values become fixed as actual numbers and are no longer calculated by a function.

Note: There is no easy way to convert back from a *raw()* segment to a mathematical function. Avoid doing so unless you are sure you can live without being able to go back to a regular segment function.

constant()

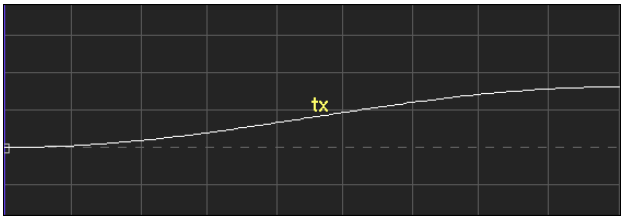
The *constant()* interpolation function generates a flat line. The value at the beginning of the segment is used throughout the whole segment.

LINEAR()



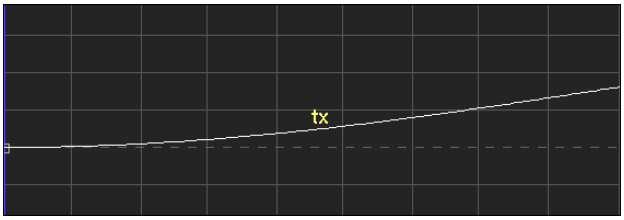
The *linear()* interpolation function generates the appearance of uniform speed across the frame range. Your object won't appear to gain or lose speed as you progress through each frame.

EASE()



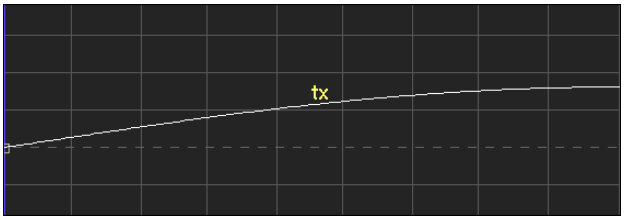
The *ease()* interpolation option generates motion that gently accelerates the object at the beginning of the frame range and decelerates it at the end of the range. The object starts the animation at rest and ends it at rest.

EASEIN()



The *easein()* interpolation option generates motion that gently accelerates the object from rest at the beginning of the frame range and continues accelerating until the end of the range.

EASEOUT()



The *easeout()* interpolation function generates motion that gently decelerates the object to rest as it approaches the end of the frame range.

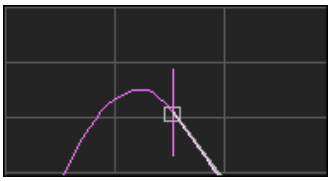
EASEP(2) / EASEINP(2) / EASEOUTP(2)

These are fundamentally the same as an *ease()* function, but instead of being raised to the power of 1, they are raised to the power 2. This steepens the slope at the beginning and end of the curve. Change the power to a higher number (e.g. 3, 4, 5) to further steepen the curve. Refer to the *Expressions* section *Channel Interpolation Functions* p. 18 for more information.

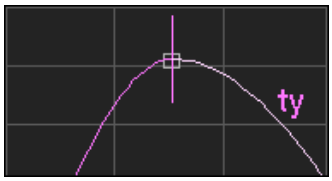
MATCH()

The *match()* function makes the slope of the curve segment on one side of a key equal to the slope of the incoming curve segment.

Incoming Segment



Before Match



After Match

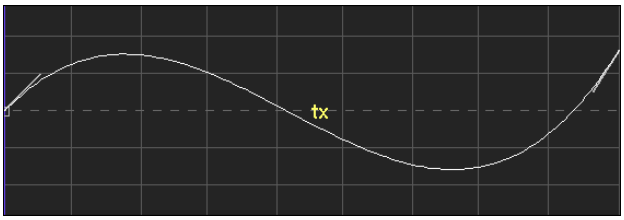
In the example above the segment to the right of the key used a *Cubic* interpolation function. After selecting it and altering its interpolation type to *Match*, its slope adjusted to match the slope of the curve segment to the left of the key.

MATCHIN() / MATCHOUT() / VMATCH()...

Refer to the Expressions section *Channel Interpolation Functions* p. 18 for more information on these functions.

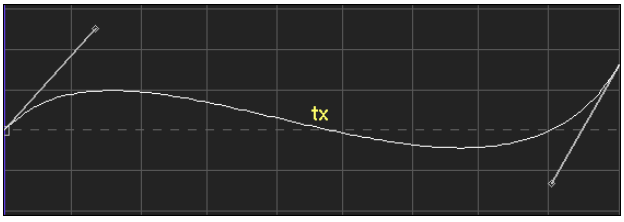
CUBIC()

Handle



The *cubic()* interpolation function generates more complex motion in that you have tangency control. Keys are equipped with handles that allow you to adjust the slope and acceleration of a curve segment. To make adjustments, grab the handle with your mouse by positioning your cursor over the handle and clicking on it ( ). Then, move your cursor up or down and the curve segment adjusts accordingly. The Slope edit field below the sheet editor displays the new slope value.

BEZIER()



The *bezier()* interpolation function offers the same functionality as the *cubic()* but adds interactive modification of the acceleration property of the curve segment. The acceleration handles appear as boxes on the end of the slope handles. To adjust the acceleration, grab the box with your mouse and pull it in the direction of the handle. this lengthens the handle and alters the shape of the curve to conform to your adjust-

ments. The acceleration edit field below the sheet editor updates to display the curve's new acceleration value.

QUINTIC()

The *quintic()* interpolation function offers the same level of interactivity as the Bezier in adjusting both the slope and acceleration variables, but with a higher order curve.

SPLINE()

The *spline()* function takes each span of the neighbouring spline segments, and creates one smooth section.

QLINEAR()

The *qlinear()* function performs a spherical linear interpolation in quaternion space between two keyframes. It is intended to be used on rotation channels and expects that all three rotation channels will also be *qlinear()* and have keys that are all at the same frames.

To perform a quaternion based interpolation between two key frames, the *qlinear()* function needs to know the three euler angles. It does this by figuring out what the neighbouring channels are. This is accomplishes this through the heuristic of looking for channels ending with x, y, or z, or by looking for channels ending with a number. Thus, "rx ry rz" will be one group, and "test0 test1 test2" would be another.

Segments with quaternion interpolation will show up as straight lines in the graph editor rather than their actual intermediate values, as their intermediate values expressed as euler angles is mostly meaningless. The curious can use the expression "qlinear() + 0" to fool the graph editor into showing the actual values.

qlinear() is the default interpolation type for rotation channels.

5 INTERACTIVE EDITING PROCEDURES

The Channel Editor has many built-in short-cuts for interactively editing channels. They allow you to bypass using menus and typing values in edit fields and directly manipulate the curve itself using your mouse and the keyboard.

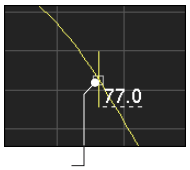
5.1 SEGMENTS

A channel contains a series of segments. Each segment has two end points. Each endpoint is at a *Key* and has the following properties: value, slope, and acceleration. The segment as a whole also has a segment function associated with it and, depending on what it is, will use any number of the properties at the end points. For example, the *constant()* segment function uses the "value" of the first end point only, while the *bezier()* segment function uses all three (slope, acceleration, and value) at each end.

SELECTING A SEGMENT

There are two ways to select, or activate, a curve segment. You can select a segment by positioning your cursor over the segment and clicking on it with your mouse. You can also click with your left mouse ( ) and drag your cursor in any direction over the segment. This action generates a marquee box whose size is determined by the amount you drag your cursor. Anything inside the box is considered selected. A selected segment becomes white to indicate that it is active. The *Functions* edit field below the graph becomes active when a segment has been selected.

5.2 ADDING / DELETING KEYS



ADDING A KEY

You can add a key to a curve by positioning your cursor on the curve and Alt clicking at the location you want to add the key. A key indicator then appears on the curve to display your addition. The value underscored by a dashed line indicates the frame number at which the mark has been added.

If more than one channel is active then a key is added to each channel.

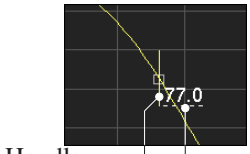
During the insertion of new keys or timegroups, a guide will appear, deferring the insertion until the relase of the mouse button. This allows you to slide the keyframe or timegroup into correct position without modifying the channels.

If a key already exists for any of the active channels, a duplicate key will not be created. Existing keys at that time will be selected and can be subsequently dragged.

DELETING A KEY

Delete a key by box-selecting it (click in an area outside of the key, and then drag a rectangle around it) in order to select it, and then type the Del key.

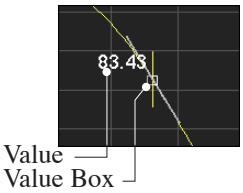
5.3 MOVING KEYS (LEFT / RIGHT)



To reposition a key, grab the vertical bar with the mouse and move your cursor left or right along the curve. The frame number will update as you drag the key to its new location. This action automatically adjusts the interpolation values between the moved key, and the adjacent keys.

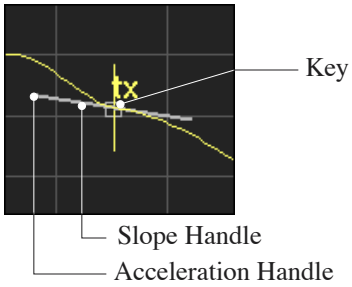
When multiple keys are selected – if you modify the values of the selected key using the Value Grid ( ) within the *FUNCTION Edit Input Fields* p. 230, the values will move *relative* to their existing values.

5.4 CHANGING KEY VALUES



To change a function value at a key, grab the key's value box and move it up or down until you reach the desired value. As you move the marker, the value display will update at the cursor's location.

SLOPE AND ACCELERATION HANDLES



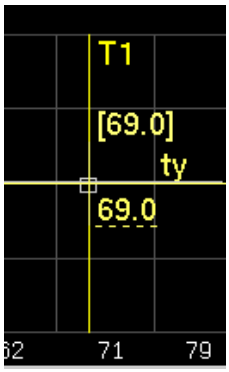
Cubic, Bezier, and Quintic interpolation functions have handles on the ends of their curve segments that allow you to interactively adjust their slope and acceleration values. To adjust a curve segment's slope, grab the handle with your mouse and drag the cursor up or down. The handle sweeps in an arc originating at the key. The curve deforms as you manipulate the handle.

The Bezier and Quintic interpolations have an acceleration handle attached to them. To interactively adjust the acceleration of a segment grab the handle with your cursor and slide it along the slope handle away from, or towards, the key.

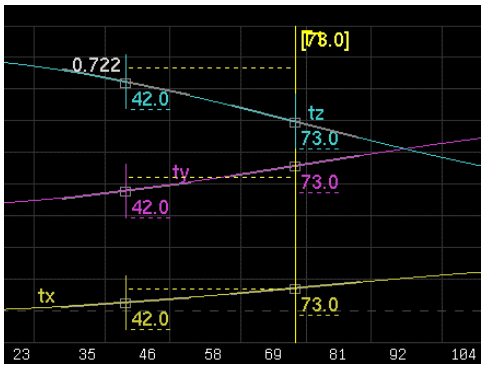
After you've altered the acceleration or slope values for the curve segment, the slope and acceleration edit fields update to show the new values.

Slope handles are available in three different scale factors. The radio button to the right of the handle toolbar selects between short, medium and long handles with acceleration. Depending on the scale and shape of the curve these provide coarse, medium, and fine control over the acceleration values.

5.5 ADDING TIME GROUPS



A Time Group



A Time Group after Keys Added
(note dashed lines)

A Time Group is a collection of Keys.

Add a time group by positioning the cursor over the frame number where you wish to add the group and then clicking Ctrl in combination and selecting *Attach/Detach Timegroups*. A yellow time group indicator then appears. It contains the name of the time group shown, for example: T1. The value in brackets indicates the frame number at which the group has been added.

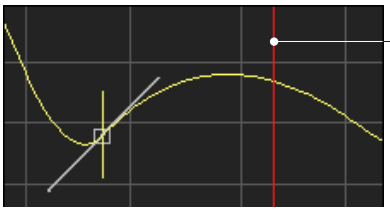
To add keys while creating the group, use the combination Ctrl Alt and click at the desired frame number.

To delete a Time Group, on the group indicator and type Del .

To add keys to an existing time group, select the keys, then Shift select the timegroup line, then select *Attach/Detach* from the ▷ pop-up menu along the top of the graph. You will see yellow dashed lines connecting the keys to the time group to confirm the addition.

Use to drag the time group independently of the time marks.

5.6 FRAME INDICATOR



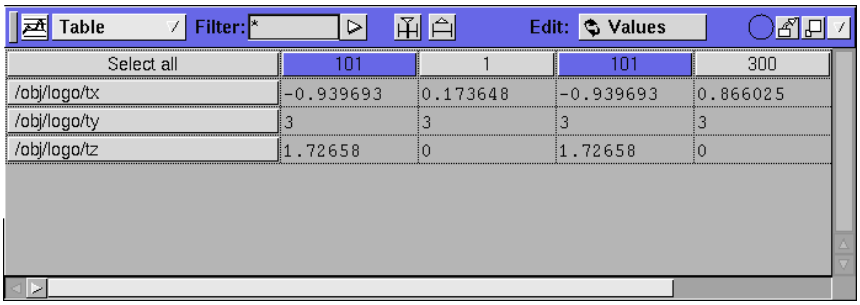
Frame Indicator

You can 'scrub' the playback by dragging the frame indicator left or right across the frame range. As you drag, the frame is displayed in parenthesis next to it.

CHANNEL HOLD

Drag with on the timebar will Hold the values of all currently scoped channels.

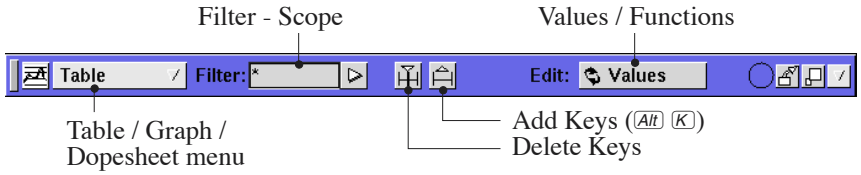
6 USING THE TABLE EDITOR



To see a Table view of your channels, select *Table* from the popup menu in the Toolbar. The Table Editor allows you to create keys and to edit values at already existing keys. If you add channels to objects elsewhere in Houdini, the Table Editor updates to display that addition.

- Each column corresponds to a frame which contains a key in the selected channels.
- Each row corresponds to one of the selected channels.
- The header contains the full path name for that channel.
- Columns with a purple header list the values of the displayed channels at the current time/frame.

6.1 TABLE COMMANDS AND CONTROLS



VALUES / FUNCTIONS SWITCHER

segment functions

Functions displays the type of interpolation function (*ease()*, *bezier()* etc.) associated with a curve segment. When the table is in *Functions* mode, a ▷ pop-up menu becomes active. If a function is chosen from this menu, it is placed in the selected segments in the table.

key values

Displays the numerical value associated with each key. Once you have selected a cell or cells in the table view, you can also add or remove keys associated with the cell(s).

6.2 TABLE OPERATIONS

SELECTING AND MODIFYING

The behaviour of the Table in the Channel Editor is the same as the Object Editor's Object List. For a comprehensive discussion of how to select and modify the information contained in a cell, see *Tables and Lists* p. 187 in the *Interface* section.

ADDING KEYS

To add a key using the table editor, select a cell, choose *Add Keys* from the *Key* menu.

COPYING AND PASTING CELLS

copy / paste

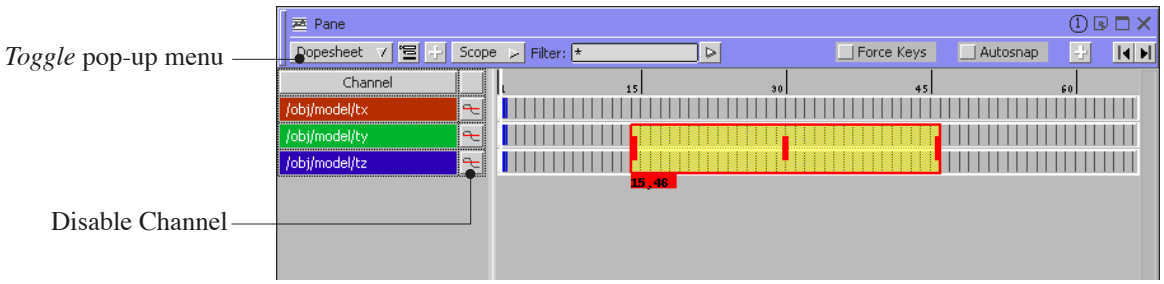
To copy the information from a cells, simply select the contents of the cells and use the Alt C keyboard combination to copy the information. To paste that information into a different cells, select the new cell and type Alt V .

6.3 TABLE EDITOR NOTES

- The current frame in the spreadsheet is always displayed in column zero. If the current frame is also a keyframe it will appear again in one of the columns to the right.
- Pasting in the spreadsheet will not delete keys if the pasted value is not a key. Pasting values always creates a key at the target cell if one is not there already.

7 DOPE SHEET

To get a Dope Sheet, select it from from the *Toggle* pop-up menu in the top-left of the Channel Editor pane controls.



7.1 BASIC OPERATION

The Dope Sheet allows you to move blocks of keys:

- Select a range of values.
- Zoom range of frames.
- Pan the frame range.

MOVING TIME

You can slide a selection forwards/backwards in time by typing Tab to get the handle up on the selection. Then use the middle mouse ( ) on the middle handle to translate the keys. This makes it unecessary to have to cut-delete and paste keys.

DRAG AND DROP KEYS

You can drag and drop keys by using the C and V keys to do the copy and paste. There is also a 'visual' version of these which is invoked with: Shift C and Shift V – then you can see what your doing. Use Ctrl to see the pop-up menu options.

7.2 SUB-ICONS

FILTER

Allows you to filter which channels are displayed in the dopesheet.

CROP BUFFER BUTTON

Displays a floating dialog which allows you to crop the selected frames for moving and copying. Controls are the same as for the Frame Range Slider in the Playbar.

FORCE KEYS

Adds keyframes at the beginning and end of the pasted selection, even if they weren't there originally.

AUTOSNAP

Auto-snaps the selection handle to fit around to the closest keys within the range of the selection handle.

DISABLE CHANNEL

Allows you to enable / disable channels.

7.3 DOPE SHEET MENU

ADD / DELETE KEYS K / Del

Adds or deletes a key.

CUT / COPY / PASTE X / C / V

Removes the selection and places it on the clipboard.

STRETCH PASTE Z

Stretches the pasted range of frames to fill the selection.

REPLACE R

Replaces the selection with the copied frames.

VISUAL PASTE Shift V

Allows you to interactively paste the range of frames. The frames are displayed with a red outline which you can drag into position as desired.

VISUAL STRETCH PASTE Shift Z

Allows you to interactively paste a range of frames to fill the selection.

VISUAL REPLACE Shift R

Allows you to interactively paste the range of frames. Drag the red outline into position as desired. The pasted frames will be stretched to fill the area you are replacing.

DISABLE / ENABLE KEYS D / E

Allows you to selectively enable or disable keys.

SNAP SELECTION TO KEYS S

Automatically moves the selection bounds such that it fits around to the nearest key-frames within the selection.

SELECT COLUMNS / ROWS A / W

Allows you to select columns or rows of values within the dopesheet.

TOGGLE HANDLE Y

Turns the scale handle on / off. The Scale Handle allows you to move and manipulate blocks of keys after you've selected them.

HOME SELECTION / ALL Shift H / H

Fits all / currently selected frames into the displayed range of frames.

SWITCH TO GRAPH / TABLE / DOPESHEET 1 / 2 / 3

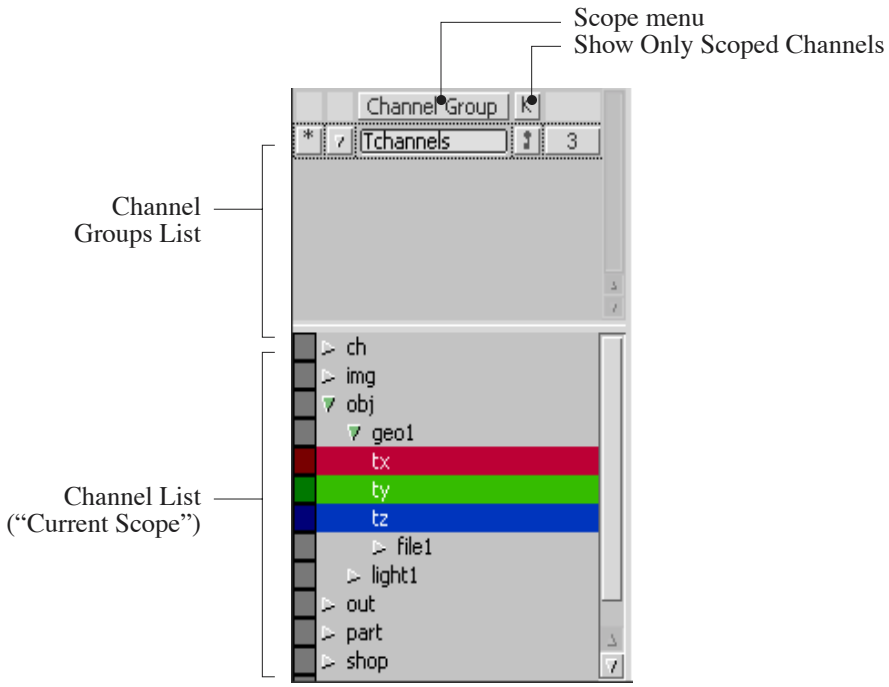
Switches to the specified Channel Editor type.

11 Channel List

1 CHANNEL LIST

The Channel List works together with the Channel Editor to allow you to specify which channels to 'scope' – these will be made active and displayed in the Channel Editor. Since you will typically want to work on a group of channels at a time, you can optionally display a Grouped Channel List.

1.1 CONTROLS



SCOPE MENU / SHOW ONLY SCOPED CHANNELS

The *Scope* menu allows you to quickly scope items listed from among all available channels (i.e. operators that have one or more channels associated with them).

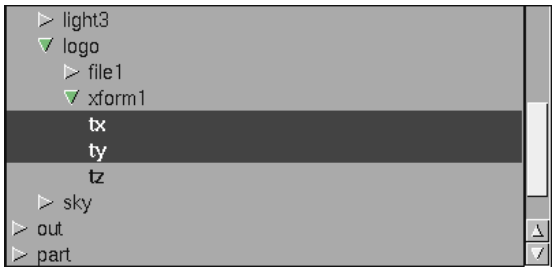
The *Show Only Scoped Channels* eliminates all non-scoped channels from being displayed in the list.

CHANNEL GROUPS LIST

Channel Groups provide you with quick access to a set of related channels in the Graph / Table Editor / Dope Sheet. When you click on a channel group, the List updates to display the channels contained in that group.

The Channel Groups List displays all available channel groups. Channel Groups are collections of channels that you've associated by placing them into a Channel Group (by selecting the channels in the Channels List, and using the Ctrl Channel List pop-up menu).

1.2 CHANNEL LIST ('CURRENT SCOPE')



The Channel List displays all the parameters which have channels in a hierarchical list. When channels are selected here, they are said to be "scoped". By clicking on the list triangles to the left of the listed names, you reveal the channels available. Clicking channels selects them and makes them available for editing in the Channel Editor.

To select more than one channel at a time, Shift click on the channels you wish to activate in succession.

Scoped channels will affect the behaviour of the Key Back/Forward buttons in the Playbar. The Key Back/Forward buttons will jump to the next keyframe within the currently scoped channels.

1.3 CHANNEL GROUPS MENU (USE Ctrl)

ADD GROUP Shift G

Creates a new Group, and adds the groups seleceted in the Channel List to it.

SCOPE SELECTED GROUP(S) V

Makes the channels included in the selected group active in the Channel List.

ADD SELECTION TO GROUP(S) +

Channels selected in the Channel List are added to the currently selected group.

REMOVE SELECTION FROM GROUP(S) -

Selected channels are removed from the currently selected group.

SET GROUP(S) TO SELECTION =

Makes the currently selected group contain those channels which are selected.

SELECT ALL Ctrl A

Selects all channels in the list.

SET KEY ON SELECTED GROUPS K

For the groups that are selected, sets a key on them.

REMOVE KEY ON SELECTED GROUPS Shift K

Removes a key from groups.

COLLAPSE SELECTED GROUPS IN DOPESHEET C

Collapses selected groups.

EXPAND SELECTED GROUPS IN DOPESHEET E

Expands the selected groups.

DELETE SELECTED GROUPS Del

Removes currently selected groups from the Groups List.

SHOW KEY STATUS

Shows the status of the selected keys.

OPEN GROUP LIST X

Alternately displays / hides the Channel Groups Display to the top of the Channels List – *Channel Groups List X* p. 244.

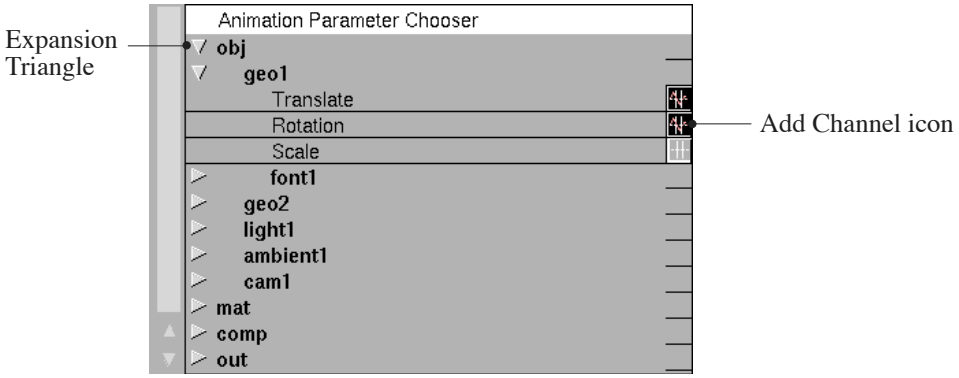
OPEN PARAMETER CHOOSER Z

Choosing *Open Parameter Chooser* displays a listing of *all* operators (geometry, COPs, SOPs, TOPs, etc.) and the animatable parameters associated with them available across Houdini, and allows you to add channels to their parameters. See *Parameter Chooser* p. 247.

OPEN CHANNEL CHOOSER Q

Opens a Channel Chooser, so you can select individual channels.

1.4 PARAMETER CHOOSER



Opening this listing (from the Ctrl menu > *Open Parameter Chooser*) provides a listing of *all* operators (geometry, COPs, SOPs, TOPs, etc.) avilable in Houdini, and the animatable parameters associated with them.

Clicking on the *Add Channel* icon allows you to add channels to their parameters; Alt clicking on the *Add Channel* icon deletes channels from the selected parameter.

To navigate down to the operator whose parameters you would like to activate, click on the expansion triangle located on the left side of the dialog box. Each of Houdini's editors form the top layer of the hierarchy. The available operators are displayed below this header. To create a channel, click on the Channel Icon to the right of the name of the parameter. The Channel Chooser updates to display the parameters you've created.

12 Custom Panel

1 CUSTOM PANEL

1.1 INTRODUCTION

The Custom Panel allows you to setup a pane with custom wired controls linked to any parameter within Houdini. You can layout your own buttons, edit fields, popup menus, and completely create your own parameters dialogue. Once this is done, you can wire your controls to any specific Houdini parameter in an OP.

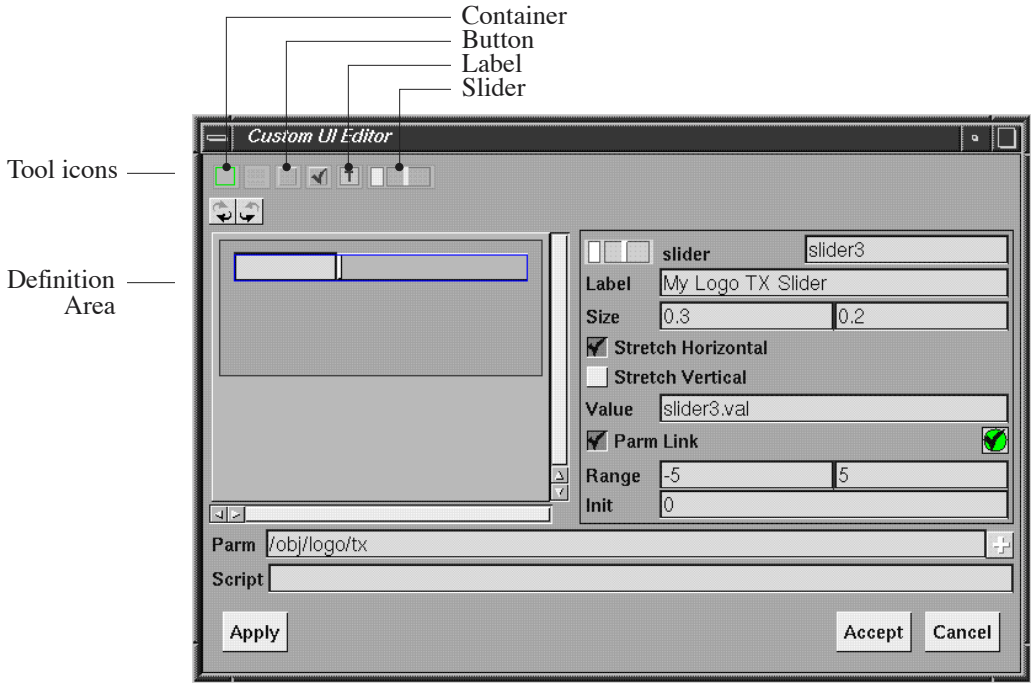
> ***Tip:*** You can switch between multiple custom panels by selecting from the menu of panels (beside the *Options* menu) in the pane's toolbar.

1.2 EXAMPLE

To remote-control the Translate-X parameter of the default *logo* object:

1. Switch one of the panes to Custom Panel, and select Create New Panel, and the Edit Panel from the Options menu in the Custom Panel pane's Toolbar.

2. You see a "Custom UI Editor" dialog appear.



3. If one doesn't exist already (would be a yellow outline in the definition area), create a new "container" by clicking on the Container icon located in the tool icons.

4. Create a new Slider within this container by clicking on the Slider icon (also located in the Tool icons).

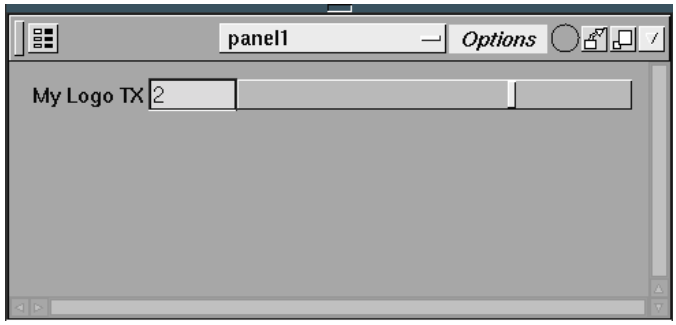
5. In order for this slider to do anything, you have to wire it up to an existing paramter which it will control. Do this by clicking once on it so it turns blue, and you the parameters change so you can edit the definition for the slider.

6. Click on the *Parm Link* button, and then click on the ✣ button beside the *Parm* field, and select the parameter you want to link to from the dialog. In this case, you should navigate down to: */obj/logo/Translate/tx* . If you wanted to avoid the tedium of going through the dialog, you could just enter this directly into the Parm field: */obj/logo/tx* . A green checkmark icon confirms a valid link.

7. To let the user know what your slider does, you should give it a meaningful label. To do this, enter something like "My Logo TX Slider" in the Label edit field.

8. Also, you want to define the limits for the range of the slider, and an initial value that should be entered if the user has not yet entered anything. Do this by entering values of: *-5* and *5* in the *Range* fields, and *0* in the *Init* field.

9. Click *Accept* to show that you are done defining your custom panel. You should now have a custom panel that controls the X-Translation of the *logo* object:

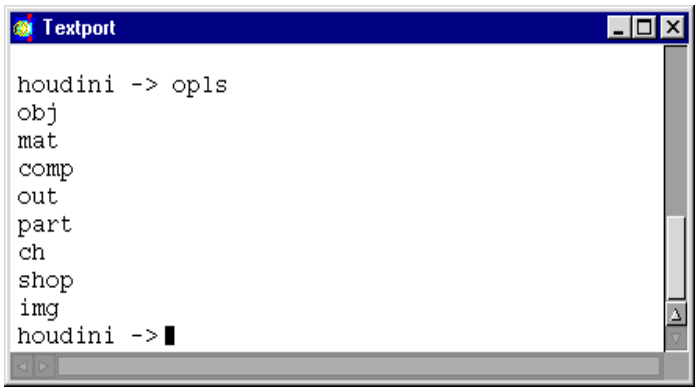


10. Move the *My Logo TX* slider, and you should see the logo move back and forth in the Viewport. If you have a Parameters pane up displaying the parameters for the *logo* object, you should see the values in the *Translate* field update accordingly.

Add your own controls as desired in the same manner.

13 Textport

TEXTPORT

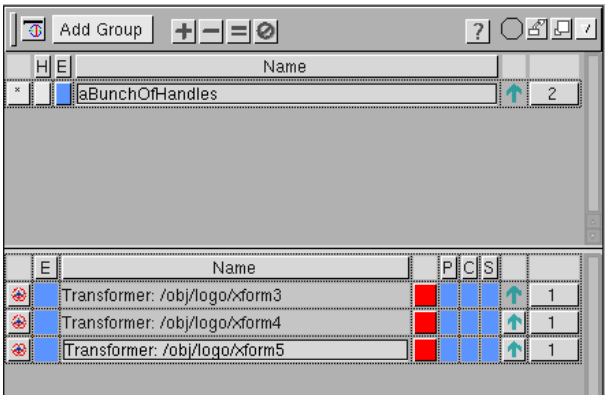


Selecting the *Textport* displays a pane which allows you to enter text-based commands. For more information on the commands available through the Textport, type *help* at the command prompt to see a list of options.

For complete information on each of these commands, see the *12-Scripting* section.

14 Handle List

1 HANDLE LIST



1.1 DESCRIPTION

Displays a list of Handles which have been made persistent in the Viewport. A Handle is a device which allows you to tweak and edit geometry during a Modelling Operation in the Viewport.

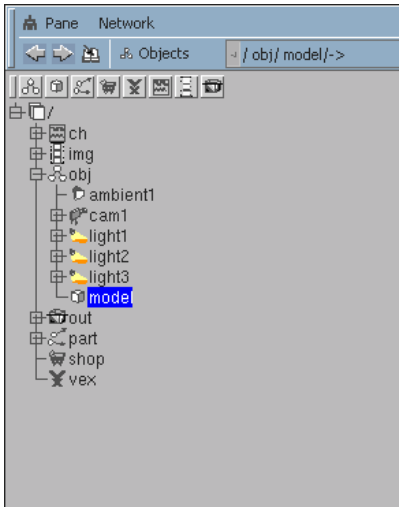
MAKING HANDLES PERSISTENT

While you are modelling (say, with an Edit or Transform OP), you can use the pop-up menu, to select an option called *Persistent*. This makes the Handle stay there, even while you go on to do other things. The *Manipulator List* allows you to view and manage a list of these persistent handles.

See *Geometry* > *Handles* p. 408 for more info on Handles.

15 Tree View

1 TREE VIEW



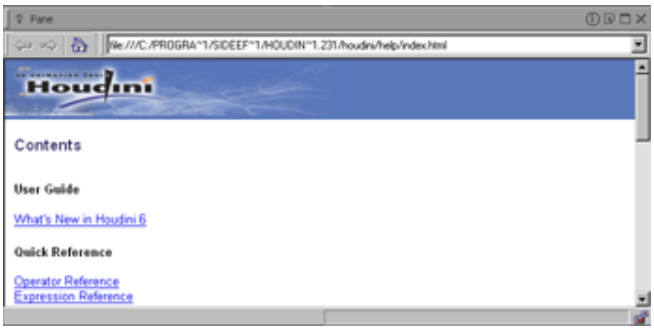
1.1 DESCRIPTION

Provides a basic hierarchical tree navigator to navigate your Houdini project.

- Click the + buttons to open up a branch of the hierarchy.
- Click the - buttons to close-up a branch of the hierarchy.
- Use the arrow keys if you like.

16 Help Browser

1 HELP BROWSER



For online HTML help, select a *Help Browser*. It allows you to explore Houdini's wide range of online built-in help. Click on a link to find out more...