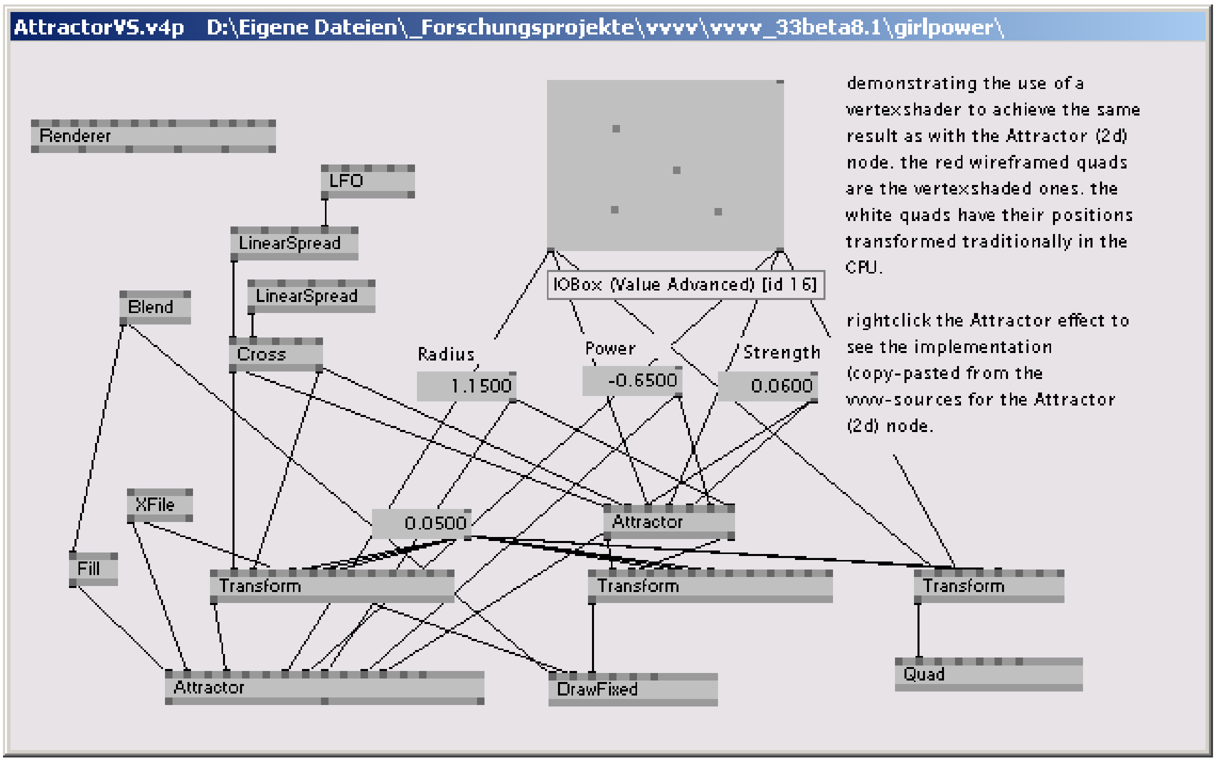
vvvv

http://vvvv.meso.net vvvv_beta8-1.zip (6.8MB) Kostenlos

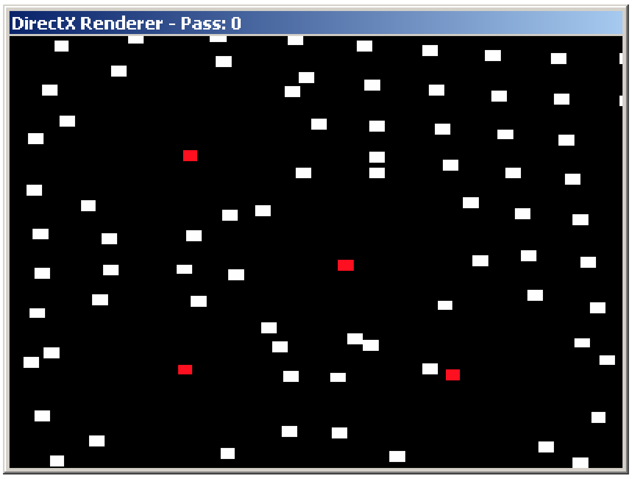
AttractorVS.v4p



AttractorVS.v4p D:\Eigene Dateien_Forschungsprojekte\vvvv\vvvv_33beta8.1\girlpower\

demonstrating the use of a vertexshader to achieve the same result as with the Attractor (2d) node. the red wireframed quads are the vertexshaded ones. the white quads have their positions transformed traditionally in the CPU.

rightclick the Attractor effect to see the implementation (copy-pasted from the vvvv-sources for the Attractor (2d) node.



http://vvvv.meso.net/tiki-index.php

vvvv is a toolkit for real time video synthesis and physical devices.

It is designed to facilitate the handling of large media environments with physical interfaces, real-time motion graphics, audio and video that can interact with many users simultaneously. vvvv is focussed on the following requirements:

- Runtime graphical programming and easy prototyping
- Effortless handling of a multitude of objects (Spreads)
- High-Speed, High-Quality, High-Resolution DirectX9.0 based rendering
- Sophisticated objects for high level animation and problem solving
- Controlling hardware devices
- Support for clustering of standard PCs to create seamless multi-projection or multi screen setups

http://vvvv.meso.net/tiki-index.php?page=Propaganda

VVVV at a Glance

- High-Speed, High-Quality, High-Resolution DirectX9.0 based rendering
- Realtime image color correction, keying, softedge and image warping features via Direct3d pixel shaders
- Control of large rendering clusters consisting of cheap standard pcs from centralized servers without the need of physically touching the clients ("Boygrouping")
- Playback of AVI, Image, MP3 and Flash formats. Import of 3d geometries.
- Proven reliability for 24/7 media installations
- Runtime graphical programming for easy prototyping and testing
- Full ability to modify and reconfigure the system while the system is running
- Simple handling of a multitude of logical objects (Spreads)
- Sophisticated objects for high level animation and problem solving
- Extensive features for connecting external devices. vvvv currently supports DMX, TCP, UDP, ArtNet, OSC, MIDI, RS232, HTTP, RSS, various USB interfaces for connecting relays, external sensors, etc. Other device drivers can be implemented very easily on demand
- Full Control over internal devices and processes, like audio mixer, keyboard, mouse, game controllers, windows, processes, etc.
- Runs on standard windows xp platforms. DirectX9.0 graphics card recommended.

Features of VVVV

vvvv is a toolkit for real time video synthesis and connecting physical devices. We use the word "synthesis" because vvvv generates or "synthesizes" video through the usage of moving graphical objects.

vvvv's architectural main features include runtime graphical programming, the effortless handling of a multitude of objects, support for a lot of exotic hardware, live integration of directx shaders and support for the clustering of standard PCs (for creating seamless multi-projection or multi screen systems).

Runtime Programming

Users of musical software already know that working in a continuous flow proves essential for all things real-time: the ability to create and modify patches during runtime, to change parameters, and also to break and make new connections. While other languages have special modes for building and for running patches, vvvv only has one mode - runtime. In other words, it is constantly performing calculations and rendering graphics.

Graphical Programming

vvvv uses graphical objects for programming instead of a textual programming interface. Programs are created within an environment that is called a patch. Individual operations or functions are represented as boxes (called "nodes"). Connections between nodes are drawn and modified with a minimal number of mouse clicks, creating a structure that sends data from one node to another.

Every node has its own attributes (a node for addition, for example, has two or more values that it adds together), which are represented by graphical elements called "pins". Input pins can receive input; outputs pins send treated output to other nodes. Node values can be set numerically in the inspector window, or by directly adjusting the value on its pins.

To create interfaces within a patch, vvvv provides flexible interface objects that allow for the exploration of intermediary forms between buttons, sliders and diagrams, while keeping a strong focus on usability.

Have a look at some screenshots to get an idea of what working with vvvv is like.

Spreads

vvvv can simultaneously handle a large count of objects, either graphical or data, without a significant effort by the user. This means that it is just as easy to control a single instance of a graphical object (a JPEG image, for example) or 100. We call this technique "spreading." Spreading is an abstraction that refers to the act of distributing different values across a set of objects (setting a table is essentially "spreading" a set of dishes out across the table, as each one has a different position on the table). vvvv contains many spread generators that make it easy to program complex behaviours for a large group of objects.

Boygrouping, Networking & Clustering

Control any amount of render computers from a single server to create multi-screen systems or seamless multi-projection setups. The network functions allow you to set up a render cluster, where you can individually select which nodes will run on the server and which nodes will run on the clients. This functionality, called "Boygrouping," allows you to replace one single object on the server with a group of objects running on the clients doing all things n'sync. The ability to control which data needs to pass the network allows for strong synchronisation of multi-screen setups over traditional stochastic tcp/ip networks.

In addition to clustering, vvvv supports standard protocol such as TCP, UDP or RS232 (for communication with other computers and software). The NetSend and NetReceive nodes can be used to exchange data between different softwares (ie. PD, Max/MSP, or AutoIllustrator) or to communicate between a number of computers in a network. Additional objects allow communuication via MIDI, DMX, ArtNet, OSC, HTTP, IRC etc.

DirectX Graphics Rendering

DirectX is a Windows technology that offers the highest performance in graphics for personal computers; it is the API (Application Programming Interface) that is usually used in the video gaming industry. Because modern graphics cards are able to outperform any available CPU in terms of raster operations and bus-speed, vvvv is designed to use the graphics card and the main processor in parallel to offer superior render performance in comparison to other graphics software that only uses the main processor (i.e. Macromedia Flash).

Shapes

vvvv offers shape generators that render graphical objects such as quads, grids, lines, circles, etc. vvvv also contains nodes for transforming these graphical elements using the classic matrix transformations (translating, scaling, rotation, aspect ratio modification, inversion, and perspective projection). Complex structures and effects (ie. warping) can be generated through structural hierarchies of node organization.

Fast Image Mapping

Dress your polygons by mapping images (standard formats and also camera input) onto graphical shapes. Because image mapping calculations are performed by the graphics card, vvvv still performs well when handling multiple textures and graphical objects. vvvv imports most standard formats for images - BMP, JPEG, PNG and others. Textures can be transformed (scaled, translated, etc) in the same fashion as graphical shapes. With its ability to read PNG image data with transparency vvvv can use file textures to create masks and alpha channels.

Vertex and Pixel Shaders

See Introduction to Shaders and the Introduction for Effect Files for details.

Ultra High Full-Screen Performance

vvvv's full-screen mode essentially takes over the machine, letting it output graphics at the best possible frame rate. Depending on your hardware, in full-screen mode there is no tearing or jittering - motion is always accurately rendered to exactly match the video frame rate at resolutions better than cinemascope[a].

Graphics Hardware

vvvv objects can access low-level features and controls to offer you greater control over your graphical output. This means that vvvv is hardware specific - certain features work better with certain graphics card hardware.

Physical Device Control

With vvvv, you can control - send input to and receive output from - external and internal physical devices. We've developed specialized nodes that exchange information with specific devices, making it easier to send relevant data and retrieve useable feedback.

A list of currently supported hardware will be collected in a How-To Database. Keep in mind that we're interested in making vvvv more compatible with more devices - and can create customized drivers upon request.

External Devices

vvvv has nodes for controlling a variety of different types of third party devices, including DVD players, industrial interfaces (for lighting and electronics), touch-screen monitors, gaming devices, dials, switches, audio devices, position & orientation sensors, weather stations, MIDI equipment, DMX interfaces, serial port devices, keyboards and (multiple simultaneous) mice.

Internal Devices

Naturally, vvvv is also capable of working with internal computer devices such as the keyboard, mouse, CD-Rom drive, the sound card's internal mixer, and audio input and outputs.

Customized Driver Development

With our long history of developing customized interfaces (visit the Meso homepage), we can also provide specific drivers for your special devices. Contact us (vvvv@meso.net) for additional details.

High level Node Library

Modelled after arbitrary cultural trends and artifacts (ie. Hollywood film, electronic music, modern industry), many nodes and node categories offer abstract approaches towards computing, visual composition and data handling. These nodes may be either used as basic tools for designing completely unrelated applications (intentionally neglecting their cultural heritage) or for further exploring their inherent structural properties. TypoSpread, for example, creates outlines of font shapes that can be used either as a pseudo-random signal generator or for typographical experiments.

Data Types

vvvv provides 5 different types of pins for specific data types: **numeric values, colours, strings, enumerations** and "**other**."

Unlike other languages, vvvv does not distinguish between different numerical value types: it treats the traditional wide variety of floating point and integer types, booleans, bangs, and bytes simply as "values". This makes patching more intuitive and allows for more flexible mapping of data between different types of nodes (i.e. one can connect a number generating node to a logic node that responds to boolean input - and the logic node will respond as if it were receiving boolean data).

Colours

From Retro to RGBA - From past & present computer graphics colour schemes, to popular digital mixing techniques, vvvv offers different methods for working with colour. vvvv also has nodes for mixing and tinting, as well as for creating complements and conversions.

String Manipulators

The string nodes can be used for text processing as well as for working with arbitrary binary data blocks. special nodes perform fundamental string operations (i.e. concatenation, text formatting, logical operations, sorting, conversion to different formats); a regular expression node and tokenizer nodes allows easy parsing of strings - so you can easily process data from external devices, websites or files and manipulate these at will

Audio

vvvv has both audio input and output functionality: it can play MP3 files, process them with various audio effects, and receive sound input from your computer's sound card. vvvv can also perform sound analysis: it can measure the frequency band and the amplitude of the music being played on your computer using FFT analysis.
DSP Disclaimer: For users looking for more powerful digital sound processing functionality, vvvv can easily be connected with PD, a powerful open source program for real-time DSP processing.

Animation

Numerical data can be generated using nodes that follow the functionality of signal generators like those used for classical synthesizers and sequencers. The LFO node, for example, outputs an ongoing sequence of values between 0 and 1. vvvv's collection of oscillators, filters, wave shapers & generator mixers offer algorithms for more advanced motion and the smoothest smoothing techniques.

Mathematical and Logical Operators

Create automata and sequencing applications by using abstractions modelled after standard electronic components. Various nodes include flip-flops, mono-flops, toggles, sample+hold, triggers, hysteresis functions, binary encoders and decoders etc. Of course all standard mathematical and boolean logic functions are also available.

As graphical programming tends to make certain algorithms rather complex, nodes are provided that allow for a compact textual representation of certain operations. The expr object allows you to evaluate mathematical formulae, while the automata node allows you to specify finite state automata in a most abstract way. This node can be used for all kind of conditional sequencing applications.

Geometric Functions

Even though it was so boring when you were forced to learn it in school, you've probably now realized that geometry is both beautiful & fascinating. Create your own geometric constructions or use the geometry nodes for specialized tasks - the Attractor node, for example, moves particles in a particle system.

Patches

```
D:.
|   4x4To3x3 (Transform) help.v4p
|   About (VVVV) help.v4p
|   Add (Color) help.v4p |   Add (Differential) help.v4p |   Add (String Spectral) help.v4p |   Add (String) help.v4p
|   Add (Value Spectral) help.v4p |   Add (Value) help.v4p
|   ADSR (Animation) help.v4p
|   AlphaTest (EX9.RenderState) help.v4p
|   AND (Boolean Spectral) help.v4p |   AND (Boolean) help.v4p
|   ApplyTransform (Transform Vector) help.v4p |   ApplyTransform (Transform) help.v4p
|   ArbitraryPoint (Transform Vector) help.v4p |   ArbitraryPoint (Transform) help.v4p
|   Args (VVVV) help.v4p
|   AspectRatio (Transform) help.v4p
|   AsString (Value) help.v4p
|   AsValue (String) help.v4p
|   Attractor (2d) help.v4p |   Attractor (Value) help.v4p |   Attractor (Value) help2.v4p
|   AttractorSpread (Spreads) help.v4p
|   AudioIn (DShow9) help.v4p
|   AudioMixer (System) help.v4p
|   AudioOut (DShow9) help.v4p
|   AudioRecordSelector (System) help.v4p
|   Automata (Animation) help.v4p
|   AviParser (File) help.v4p
|   B-Spline (Value) help.v4p
|   BarSpread (Spreads) help.v4p
|   Beep (Windows) help.v4p
|   Blend (Color) help.v4p |   Blend (EX9.RenderState Advanced) help.v4p |   Blend (EX9.RenderState) help.v4p
|   Blinker (Animation) help.v4p
|   Boost (VVVV) help.v4p
|   BoundingBox (EX9.Geometry Mesh) help.v4p
|   Bounds (Spectral) help.v4p
|   Box (EX9.Geometry) help.v4p
|   Boygroup (VVVV Client) help.v4p |   Boygroup (VVVV Server) help.v4p
|   BoygroupMeter (VVVV) help.v4p
|   Byte (String) help.v4p
|   B,zierLine (GDI) help.v4p |   B,zierLoop (GDI) help.v4p
|   B,zierSpread (Spreads) help.v4p
|   Caps (String) help.v4p
|   CAR (Spreads) help.v4p |   CAR (String) help.v4p
|   CDR (Spreads) help.v4p
|   Change (Animation) help.v4p |   Change (String) help.v4p
|   Circle (GDI) help.v4p
|   CircularSpread (Spreads) help.v4p
|   Clamp (3d) help.v4p
|   Clean (String) help.v4p
|   Clip (EX9.RenderState) help.v4p
|   ColorChannel (EX9.RenderState) help.v4p
|   compare filters.v4p
|   Complement (Color) help.v4p
|   Connect (2d) help.v4p
|   ConnectAll (2d) help.v4p
|   Cons (Spreads) help.v4p |   Cons (String) help.v4p
|   ConvexHull (2d) help.v4p
|   Copier (File Async) help.v4p |   Copier (File) help.v4p
|   Count (EX9.Geometry Mesh) help.v4p |   Count (String) help.v4p |   Count (Value) help.v4p
|   Counter (Animation) help.v4p
|   CountIndices (Spreads) help.v4p
|   CPULoad (Debug) help.v4p
|   CreateNode (VVVV) help.v4p
|   Cross (2d) help.v4p
|   Cull (EX9.RenderState) help.v4p
```

| CurrentTime (Astronomy) help.v4p
| Cursor (System) help.v4p
| Cylinder (EX9.Geometry) help.v4p
| Damper (Animation) help.v4p
| Decay (Animation) help.v4p
| Delay (Animation) help.v4p
| DeleteNode (VVVV) help.v4p
| Diff (String) help.v4p
| Differential (Spreads) help.v4p
| Dir (File) help.v4p
| Divide (Differential) help.v4p | Divide (Value) help.v4p
| DMX (Network Artnet Receiver) help.v4p | DMX (Network Artnet Sender) help.v4p
| DoQuit (VVVV) help.v4p
| DX9Texture (EX9.Texture) help.v4p
| DynamicMerge (Differential) help.v4p
| DynamicTexture (EX9.Texture Color) ..-.(EX9.Texture String) -(EX9.Texture Value)
| Effect (EX9.Effect) help.v4p | Effects help.v4p
| EQ (String) help.v4p | EQ (Value) help.v4p
| Exists (File) help.v4p
| Expr (Value) help.v4p
| Fanclub.v4p
| Fanclub_dynamic_list.v4p
| FFT (DShow9) help.v4p
| FileName (File Split) help.v4p
| FileStream (DShow9) help.v4p
| FileTexture (EX9.Texture) help.v4p
| Fill (EX9.RenderState) help.v4p
| Filter (EX9.SamplerState) help.v4p
| Finally (VVVV) help.v4p
| FlipFlop (Animation) help.v4p
| FormatValue (String) help.v4p
| Frac (Value) help.v4p
| FrameCounter (Animation) help.v4p
| FrameDelay (Animation) help.v4p | FrameDelay (String) help.v4p
| FrameDifference (Animation) help.v4p
| FrameVelocity (Animation) help.v4p
| Gamma (Value) help.v4p
| GaussianSpread (Spreads) help.v4p
| GCD (Value) help.v4p
| GDITexture (EX9.Texture) help.v4p
| GE (Value) help.v4p
| GEL (Debug) help.v4p
| GeometricMean (Spectral) help.v4p
| GetGammaRamp (Windows) help.v4p
| GetIndexBuffer (Ex9.Geometry Mesh) help.v4p
| GetMatrix (Transform) help.v4p
| GetPatchMode (VVVV) help.v4p
| GetSlice (Node) help.v4p | GetSlice (Spreads) Help.v4p | GetSlice (String) help.v4p
| GetVertexBuffer (EX9.Geometry Mesh) help.v4p
| GlobalRenderState (EX9) help.v4p
| Gregorian (Astronomy) help.v4p
| Grid (DX9) help.v4p
| GridPick (2d) help.v4p
| GridSegment (DX9) help.v4p
| GridSplit (2d) help.v4p
| GT (Value) help.v4p
| Heading (Animation) help.v4p
| help.v4p
| HLine (GDI) help.v4p
| HSL (Color Join) help.v4p | HSL (Color Split) help.v4p | HSL (Transform) help.v4p
| HSLtoHSV (Color) help.v4p
| HSV (Color Join) help.v4p | HSV (Color Split) help.v4p

| HSVtoHSL (Color) help.v4p
| HTTP (Network Get) help.v4p
| I (Spreads) help.v4p
| Icon (Windows) help.v4p
| InputMorph (Color) help.v4p | InputMorph (Value) help.v4p
| InsertSlice (Spreads) help.v4p
| Inspektor (VVVV) help.v4p
| Integral (Spreads) help.v4p
| Integrate (Differential) help.v4p
| Interval (Spreads) help.v4p
| Inverse (Transform) help.v4p
| IOBox (Color) help.v4p | IOBox (Enumerations) help.v4p | IOBox (Node) help.v4p | IOBox (String)
| IOBox (Value Advanced) help.v4p
| ioboxes more.v4p
| IP (Network) help.v4p
| IRC (Network Open) help.v4p | IRC (Network Private) help.v4p
| Join (Quaternion) help.v4p
| Keyboard (System Global) help.v4p | Keyboard (System Window) help.v4p
| KeyMatch (String) help.v4p
| LCM (Value) help.v4p
| LE (Value) help.v4p
| Levin (Spreads) help.v4p
| LFO (Animation) help.v4p | LFO (Differential) help.v4p
| Line (GDI) help.v4p
| LinearFilter (Animation) help.v4p
| LinearSpread (Spreads) help.v4p
| Log (Value) help.v4p
| LogFile (VVVV) help.v4p
| LookAt (Transform Vector) help.v4p | LookAt (Transform) help.v4p
| LT (Value) help.v4p
| M3UParser (File) help.v4p
| MainLoop (VVVV) help.v4p
| Mantissa (Value) help.v4p
| Map (Value Interval) help.v4p | Map (Value) help.v4p
| MapRange (Value) help.v4p
| Max (Value) help.v4p
| Mean (Spectral) help.v4p
| MeanSquare (Spectral) help.v4p
| Memory (Debug dx9) help.v4p | Memory (Debug) help.v4p
| Merge (Differential) help.v4p
| Mesh (EX9.Geometry Join) help.v4p
| MessageBeep (Windows) help.v4p
| MidiOut (DShow9) help.v4p
| Min (Value) help.v4p
| Mod (Value) help.v4p
| ModificationDate (File) help.v4p
| MonitorPower (System) help.v4p
| MonoFlop (Animation) help.v4p
| Mouse (System Global) help.v4p | Mouse (System Window) help.v4p
| Mover (File) help.v4p
| MP3Parser (File) help.v4p
| MultiFlipFlop (Animation) help.v4p
| Multiply (3d Vector (3d) (4d Vector) (4d (Differential ConstantScale) (Differential) (Quaternion) (String) (Transform) (Value Spectral)
| Multiply (Value) help.v4p
| MultiToggle (Animation) help.v4p
| Netreceive (Network Value) help.v4p
| NetSend (Network Value) help.v4p
| NetworkStatus (Debug) help.v4p
| Newsfeed (Network) help.v4p
| Newton (Animation) help.v4p
| Normalize (3d Vector) help.v4p | Normalize (3d) help.v4p | Normalize (Quaternion Vector) help.v4p

| Normalize (Quaternion) help.v4p
| NOT (Boolean) help.v4p
| ODE_collisions (help).v4p
| ODE_more (help).v4p
| ODE_rigid_bodies (help).v4p
| OnActivate (VVVV) help.v4p
| OnOpen (VVVV) help.v4p
| OnQuit (VVVV) help.v4p
| Optimize (EX9.Geometry Mesh) help.v4p
| OR (Boolean Spectral) help.v4p | OR (Boolean) help.v4p
| Ord (String) help.v4p
| Ord2Enum (Enumerations) help.v4p
| Ortho (Transform OffCenter) help.v4p | Ortho (Transform) help.v4p
| OSCDecoder (Network) help.v4p
| OSCEncoder (Network) help.v4p
| Oscillator (Animation Advanced) help.v4p | Oscillator (Animation) help.v4p
| Pad (String) help.v4p
| ParallelEpiped (Transform Vector) help.v4p | ParallelEpiped (Transform) help.v4p
| PeakSpread (Spreads) help.v4p
| Perspective (Transform OffCenter) help.v4p | Perspective (Transform) help.v4p
| Pillow (DX9) help.v4p
| Ping (Network) help.v4p
| Pipet (EX9.Texture) help.v4p
| Point (GDI) help.v4p
| Points2Vector (2d) help.v4p
| Pot (Value) help.v4p
| Power (Value) help.v4p
| Quad (DX9) help.v4p
| Queue (EX9.Texture) (Spreads (Spreads) | Queue (String) help.v4p
| Quote (String) help.v4p
| R (Color) help.v4p | R (Node) help.v4p | R (String) help.v4p | R (Value) help.v4p
| RadioKeyMatch (String) help.v4p
| Radix (Value) help.v4p
| Random (Value) help.v4p
| RandomSpread (Spreads) help.v4p
| Reader (File) help.v4p
| RegExpr (String) help.v4p
| Renderer (DX9 (Flash) help _ (Flash) help.v4p | Renderer (GDI)
| Renderer (GDI) morehelp.v4p
| Repeat (String) help.v4p
| Resample (Spreads) help.v4p
| RetroColors (Color) help.v4p
| Reverse (Spreads) help.v4p
| RGB (Color Join) help.v4p | RGB (Color Split) help.v4p | RGB (Transform) help.v4p
| RingBuffer (Spreads) help.v4p | RingBuffer (String) help.v4p
| RMS (DShow9) help.v4p
| Rope (DX9) help.v4p
| Rotate (Transform Vector) help.v4p | Rotate (Transform) help.v4p
| RoundRect (GDI) help.v4p
| RSh (Network) help.v4p | S (Color) help.v4p | S (Node) help.v4p | S (String) help.v4p
| S (Value) help.v4p
| S+H (Animation) help.v4p | S+H (Color) help.v4p | S+H (String) help.v4p
| Scale (Transform Vector) help.v4p | Scale (Transform) help.v4p
| ScopeSpread (DShow9) help.v4p
| Scroll (Spreads) help.v4p | Scroll (String) help.v4p
| Segment (DX9) help.v4p
| Select (Color) help.v4p
| Select (String) help.v4p | Select (Value) help.v4p
| Self (VVVV) help.v4p
| Sequencer (Animation) help.v4p
| SetAlpha (Color) help.v4p
| SetGammaRamp (Windows) help.v4p

```
|   SetMatrix (Transform) help.v4p
|   SetPatchMode (VVVV) help.v4p
|   SetSlice (Spreads) help.v4p |   SetSlice (String) help.v4p
|   SevenSegment (Spreads) help.v4p
|   Shade (EX9.RenderState) help.v4p
|   Shear (Transform) help.v4p
|   ShellExecute (Windows) help.v4p
|   Shift (Spreads) help.v4p |   Shift (String) help.v4p
|   Sift (String) help.v4p
|   Sign (Value) help.v4p
|   Sin (Differential) help.v4p
|   Slerp (Quaternion) help.v4p
|   Sort (Spreads) help.v4p |   Sort (String) help.v4p
|   SpellValue (String) help.v4p
|   Sphere (DX9) help.v4p
|   Sphere (EX9.Geometry) help.v4p
|   Split (Quaternion) help.v4p |   Split (String) help.v4p
|   Stallone (Spreads) help.v4p |   Stallone (String) help.v4p
|   Stopwatch (Animation) help.v4p
|   SubDir (File) help.v4p
|   Substitute (String) help.v4p
|   Subtract (Color) help.v4p
|   Subtract (String) help.v4p
|   Subtract (Value) help.v4p
|   SwapDim (Spreads) help.v4p
|   Switch (Node Input) (String Input) (String Output (Value Input) (Value Output)
|   SysLog (VVVV) help.v4p
|   SystemFolder (File) help.v4p
|   SystemPower (System) help.v4p
|   TCP (Network Client) help.v4p |   TCP (Network Server) help.v4p
|   Teapot (EX9.Geometry) help.v4p
|   Text (GDI) help.v4p
|   TimeToString (String) help.v4p
|   Timing (Debug) help.v4p
|   Tint (Color) help.v4p
|   TogEdge (Animation) help.v4p
|   Toggle (Animation) help.v4p
|   Tokenizer (String) help.v4p
|   Transform (Transform 2d) (Transform 3d Vector) (Transform 3d)
|   Translate (Transform Vector) help.v4p |   Translate (Transform) help.v4p
|   Trapeze (Transform) help.v4p
|   tree.txt
|   TreeView (String Advanced) help.v4p |   TreeView (String) help.v4p
|   Triangle (DX9 Indexed) help.v4p
|   TypoSpread (Spreads) help.v4p
|   UDP (Network Client) help.v4p |   UDP (Network Server) help.v4p
|   Undo (VVVV) help.v4p
|   UniformScale (Transform) help.v4p
|   UpdateWindow (Windows) help.v4p
|   Vector (3d Join) help.v4p
|   Vector (3d Split) help.v4p
|   Vector2Points (2d) help.v4p
|   VertexBuffer (EX9.Geometry Join) help.v4p |   VertexBuffer (EX9.Geometry Split) help.v4p
|   VertexDeclaration (EX9.Geometry Mesh) help.v4p
|   Vertigo (Transform Far) help.v4p |   Vertigo (Transform) help.v4p
|   VideoIn (DShow9) help.v4p
|   VideoOut (DShow9) help.v4p
|   VideoTexture (EX9.Texture DShow Deinterlace) help.v4p |   videotexture (ex9.texture dshow) help.v4p
|   ViewFrustumCulling (EX9.Geometry Mesh) help.v4p
|   VLine (GDI) help.v4p
|   VVVV (VVVV) help.v4p
|   WaveShaper (Value) help.v4p
|   Window (Windows) help.v4p
|   WordCount (String) help.v4p
|   World (ODE) help.v4p
|   Writer (EX9.Texture) help.v4p
|   Writer (File) help.v4p
|   XFile (EX9.Geometry Load) help.v4p
|   XOR (Boolean) help.v4p
|   ZWriteEnable (EX9.RenderState) help.v4p
|
+---freeframe
|       Contour help.v4p
|       Trautner help.v4p
|
+---modules
|       Netreceive (Network Value) help.v4p
|       NetSend (Network Value) help.v4p
|       Writer (EX9.Texture Grid) help.v4p |       Writer (EX9.Texture NRT) help.v4p
|
```