Documentation & Help For VP9

Contents:

Introduction

Current version of Visual Pinball referred to in this guide:	9.09
Current version of this guide:	1.0
Date:	26 October 2010

This document is designed to provide documentation for Visual Pinball v9.09 and above. It is intended to familiarize the reader with the VP editor and all its' options, and although reference is made to what is scriptable and what isn't (see Appendix III), it is unfortunately beyond the scope of this document to give assistance with how to script a visual pinball table. For help with scripting a table, please visit vpforums.org.

Due to the history of the development of Visual Pinball, and the decision by the creator, Randy Davis, to make Visual Pinball open source in February 2010 there has been no definitive help documentation for users to reference – this guide is intended to fill that gap.

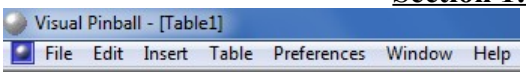
This guide is a work in progress, and is intended to be updated whenever there is a new official release of VP. Please feel free to copy, distribute, or edit this document. At time of initial release the author intends the most up to date version of this document be made available via vpforums.org.

If you have any comments, or suggestions for improvement, either contact me through vpforums (user: Wizards_Hat), or via email wizardshat@hotmail.com

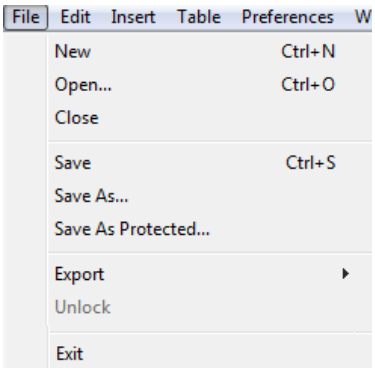
I hope this guide is of use to you,

Dan Hardy (Wizards_Hat).

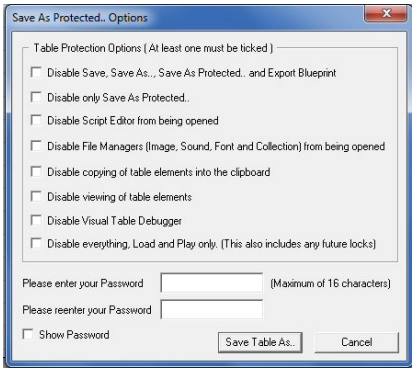
Section 1: Editor Menus



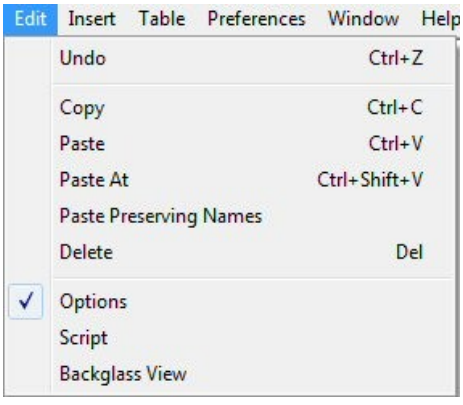
In the following pages all screenshots show the default options unless a specific example is demonstrated.



New – Creates a new basic table
Open – select a table you want to open
Close – Closes the current table that's open
Save – Saves the current table that's open
Save As… – saves your table with the ability to change the file name
Save As Protected.. – opens up a dialog box with options to password protect various features of the table (see below)
Export – Exports a blue print of your table in bmp format.
Unlock – only available if a locked table is open, allows you to enter the password to unlock the locked features
Exit – Exits Visual Pinball



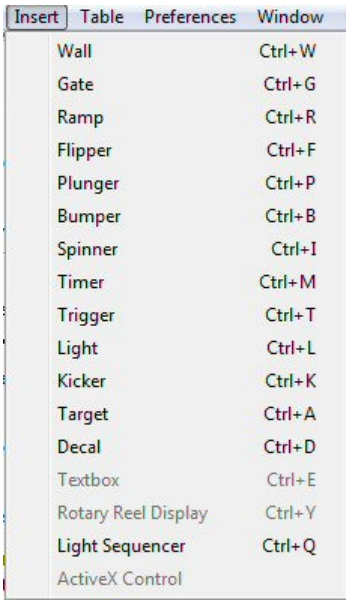
The Save As Protected dialog box has several options allowing you to protect specific areas of the table.
The definitions given in the dialog box here are deemed to be sufficient enough to not require any further description here.
However, it should be noted that a table can only be unlocked with the version of VP that was used to lock it. So if, for example, a table was locked (saved as protected) with VP8, it would not unlock in VP9 even if the correct password was supplied.



Undo – undoes the last action
Copy – copies selected item(s)
Paste – Pastes selected item(s)
Paste At – Pastes selected item(s) at the position of the mouse cursor (better to use the keyboard shortcut!)
Paste Preserving Names – Pastes selected item(s) from another table while keeping the same object name(s) as on the source table. If there is a conflict of names (i.e. if the name to be pasted already exists on the target table) then another dialog box appears:
- *Use default names* will give the object(s) to be pasted the same name as would be assigned if it was made as a new object(s).

- after a new name is entered press *Change name* to execute the paste with the name entered.
- clicking *cancel* will mean that no object is pasted.

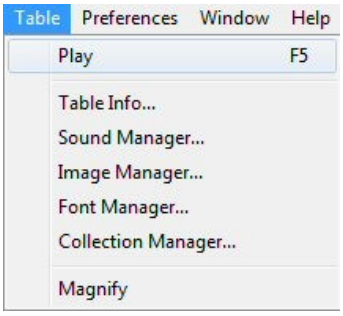
Delete – Deletes the selected item(s)
Options – Toggles the options dialog in the editor (also has a button in the toolbar)
Script – Opens the script editor (also has a button in the toolbar)
Backglass View – Toggles the Backglass view in the editor window (also has a button in the toolbar)



The insert menu has the same function as the objects toolbar (see below for details), with the exception that the objects "Rotary Reel Display" (menu item) and "EMReel" (toolbar item) are the same thing.

Textboxes and Rotary Reel Displays are only available whilst using the Backglass View in the editor.

Currently "ActiveX Control" is not available in any editor view.



Play – Starts the VP player for the currently selected table

The additional menu screens are described below.

Magnify – selects the magnify tool (see Editor toolbar below)

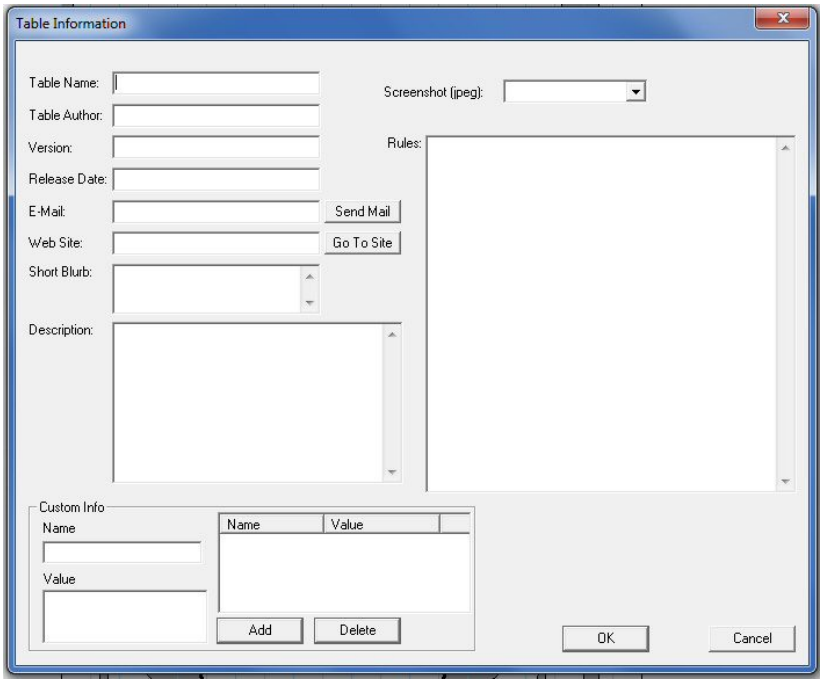
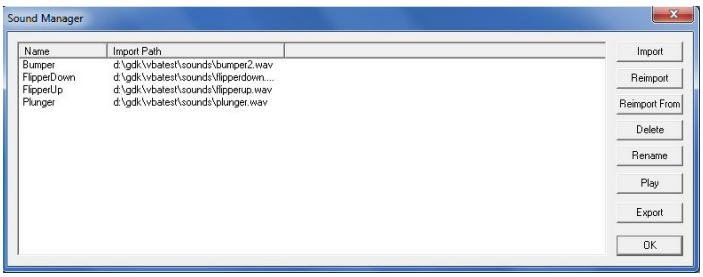


Table information – allows you to enter text information about the table. The information stored here does not interact with any other part of the editor.

To include a screenshot in the dropdown menu the image must first be imported via the image manager.



Sound Manager – controls the importing and exporting of .wav files into, and out of, the editor.
Reimport (Reimport From) can be used if the original files from which the sound was created has changed (and changed location) to refresh the sound file within VP. If the original file has changed but has not been reimported, VP will use the file as it was when it was first imported into the editor.
Play can be used to listen to the selected imported sound file.

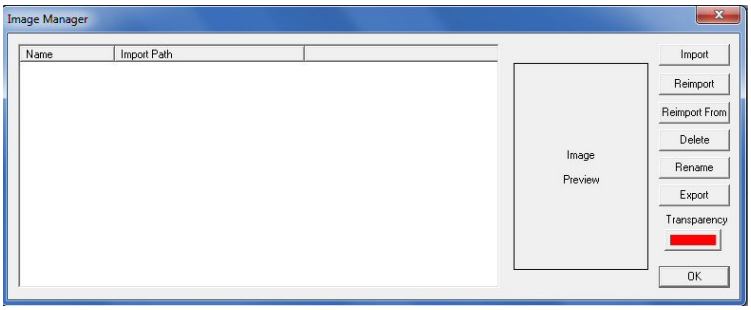
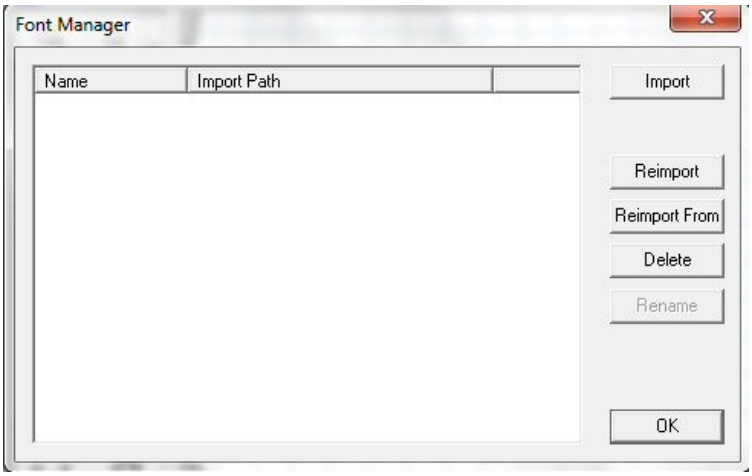
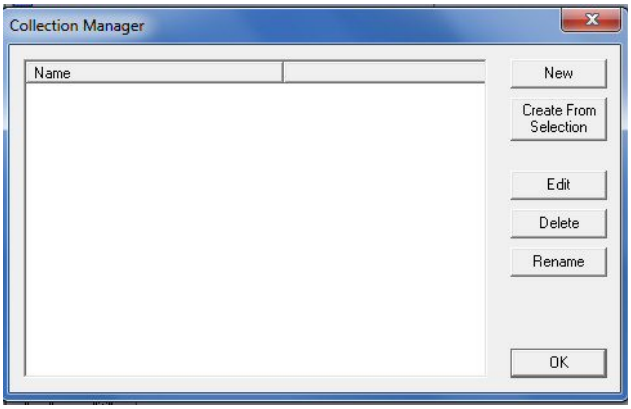


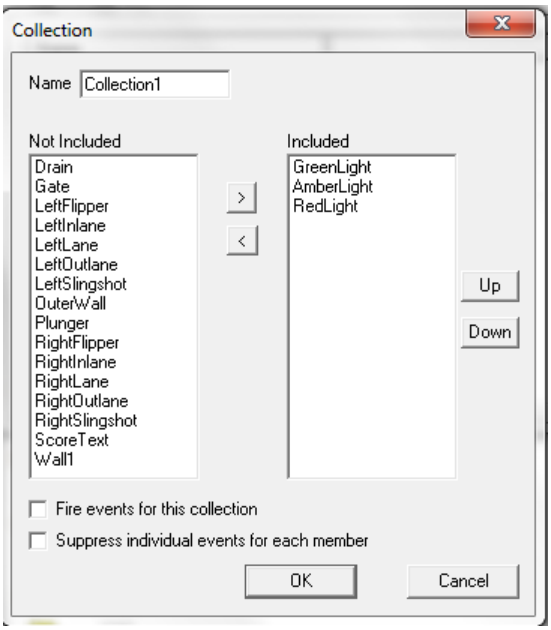
Image Manager – controls the importing and exporting of .bmp and .jpg file types into, and out of, the editor.
Reimport (Reimport From) can be used if the original files from which the image was created has changed (and changed location) to refresh the image file within VP. If the original file has changed but has not been reimported, VP will use the file as it was when it was first imported into the editor.
Transparency – click to select the color you wish to appear transparent for the selected image. The selected color will only appear transparent in the VP player, it will still be visible in the editor.



Font Manager – controls the importing of .ttf file types into the editor. This allows fonts to be used on tables when loaded onto machines that do not have a particular font natively installed.
Reimport (Reimport From) can be used if the original files from which the font was created has changed (and changed location) to refresh the font file within VP. If the original file has changed but has not been reimported, VP will use the file as it was when it was first imported into the editor.



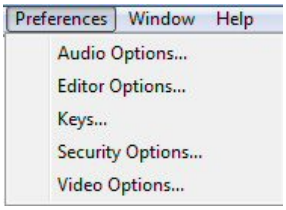
Collection Manager – playfield objects can be grouped together in "collections" that can be referenced in the script, or selected by using the drop down menu on the select button on the editor toolbar.
Edit – allows you to place objects in, or remove from, a collection that has been created.



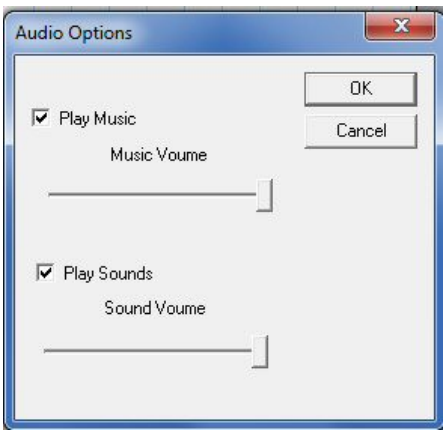
To include objects in a collection select them and press the ">" button.
Arrange objects in the collection by selecting them in the "included" window, and use "Up" and "Down". The collection makes an array of objects in order as they are shown in this edit box, with the first item in a collection being assigned 0.
For example, in the collection shown opposite:

Collection1(0) = GreenLight
Collection1(1) = AmberLight
Collection1(2) = RedLight

In order for events to occur with this collection, the "Fire events for this collection" must be ticked. (Individual events are shown below in each objects settings).
If suppress individual events for each member is ticked, the script will not execute events for an item in that collection, but will only execute commands written for the collection.



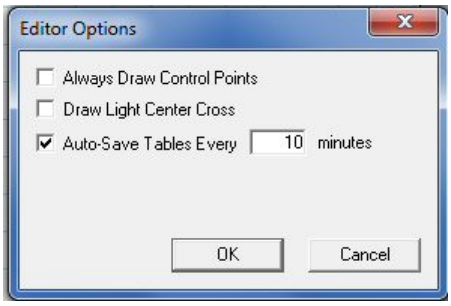
Preferences allows access to further submenus:



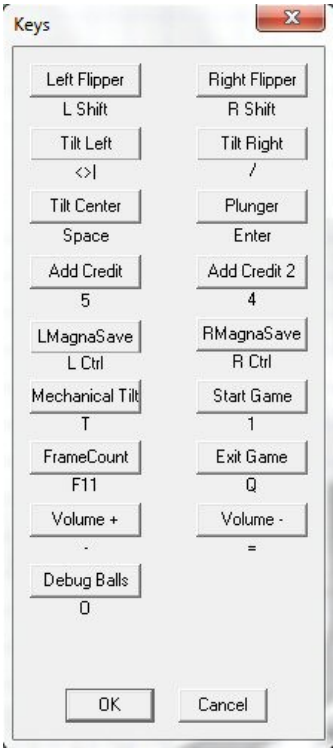
"Play Music" allows VP to access mp3, wma, and wav files placed the "Music" folder.

"Play Sounds" enables .wav files imported via the sound manager, and executed through the script to be played.

The slider bars can be used to set the volume of .mp3 and .wav files relative to the pc's system volume.



Always Draw Control Points – when ticked this will display the control points of every wall and ramp, as well as those for every custom shaped light and trigger, object in the editor. When not ticked only the control points for the selected object will be displayed.
Draw Light Center Cross – when ticked this will display the light center cross for every light object in the editor. When not ticked only when custom shaped lights are selected will a light center cross be displayed.
Auto-Save Tables Every X minutes – saves a backup copy of the current table to the directory folder where the VP exe is being run from.

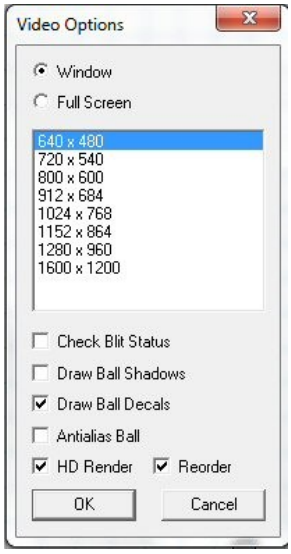


Clicking on any of these boxes allows you to assign a different key to that function. However the default table only utilizes the top 3 of each column in this dialog, unless scripted otherwise these keys will perform no function in VP.
These keys can be referred to by their keynames in the script:
LeftFlipperKey
RightFlipperKey
LeftTiltKey
RightTiltKey
CenterTiltKey
PlungerKey
AddCreditKey
AddCreditKey2
LeftMagnaSave
RightMagnaSave
MechanicalTilt
StartGameKey
ExitGame

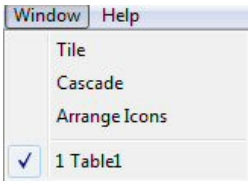
Security Options: ActiveX Controls – 5 options are available to choose from:
Always create controls – create anything and everything asked for if possible.
Warn on each type of control not marked as safe - if a control isn't explicitly stated to be safe, then VP will warn about it.
Warn on each type of any control – the first instance of a control is warned, subsequent instances are not warned - so if you run two copies of the same control in the same script/table you only get one warning (for

example, it's possible to have 3 copies of the DMC2 Sound player control, and it is necessary to do so, to have it instantly playback three different MOD files without a delay for preloading - as they are all preloaded when the control initializes).
Warn on each creation of any control – whenever VP creates a control then it warns you about it (for example, when running VPMame)
Never create controls - self explanatory

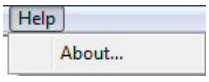
Detect and error on hanging script – can be used to help debug script, and give errors before the table loads in the VP player.



Use the controls to change video options of the player – Window/Full Screen and also choose the resolution (depends on capabilities of graphics card).
Check Blit Status – tells the program when it is ready to draw the next frame for the ball - most all newer computers/video cards don't need it. It can improve performance/smoothness of animation on old hardware (2002 and prior)
Draw Ball Shadows – draws ball shadows during play, assumes playfield is lit from left hand side. May slow down the frame rate.
Draw Ball Decals – if ball decals are selected in the table options they will be rendered if this is ticked.
Antialias Ball – can make the ball look smoother, but may slow down the frame rate.
HD Render – Toggles hardware rendering; tells VP to use your video card's built-in ram for storing textures, or if unchecked, to use slow system memory to store textures. On some computers it will be much faster to use HDRender, on others it will be faster to use System memory.
Reorder – forces VP player to draw playfield objects in a certain order (the final rendering order for objects is: Lights, Flippers, Spinners, Gates, Bumpers, Walls, Ramps, Decals, EMReels, Textboxes), overwriting any preferences used by the objects' right-click options "draw in front" and "draw in back", except that you retain individual draw in front/draw in back control over Ramps, Decals, EMReels, and Textboxes. This option was included to try and correct most, if not all rendering issues with VP9 on relatively new ATI cards like the 4650.



The window menu allows you to see which tables are currently open in the editor, and by clicking on the desired table, bring to the front for editing.
The options "Tile", "Cascade", and "Arrange Icons" currently have no function in VP.



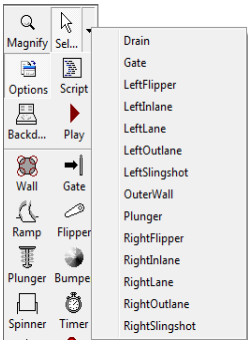
The only currently available function of the help menu is to bring up the about screen.

The Web Site button directs you to vpforums.org

Section 2: Toolbars

2.1: Editor Toolbar

Magnify – when this button is selected, the mouse cursor displays a magnifying glass; left-clicking on the editor screen will zoom in, right-clicking will zoom out. (The same function can be achieved without magnify being selected by holding down the CTRL key).



Select – when selected the mouse cursor displays an arrow. Left clicking an object will highlight and select that object, allowing its' properties to be shown in the options tab. More than one playfield object can be selected by either holding down the left mouse button and dragging over an area, or by holding down the CTRL key and clicking each item separately. Right-clicking an object will display the appropriate right-click menu for that object.

Clicking the down arrow just to the side of the select button will display a drop-down menu showing every playfield object on the current table, clicking a particular object to highlight and select that object.

Options –this button toggles the display of the options tab at the right hand side of the editor window. The options tab displays different properties for the table, backdrop and individual objects – these properties are explained in detail in each relevant section below.

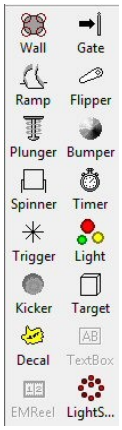
Script – selecting this button will open the table script in a new window (or switch to the script window if it is already open). It is beyond the scope of this document to explain the script, except to say that VP uses a slightly modified version of Visual Basic scripting. Some information useful for scripting can be found in Appendix III, but for much more detailed information visit vpforums.org.

Backdrop – this button toggles the display of the editor between the table view and the backdrop view.

Play – this will open the current table in the VP player.

2.2: Objects Toolbar

Playfield view

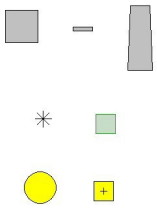
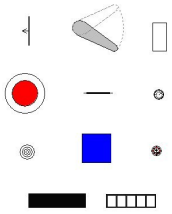


Backdrop view

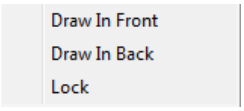
Selecting a button on the objects toolbar changes the cursor, when moved over the editing area of the window, to display the type of object selected. Clicking on the edit area will place that object at the point clicked.
Dependant on the type of view selected (playfield, or backdrop) a different selection of objects is available.
If an object has been selected by mistake, either click on the desired object, or click on the select, or magnify, buttons on the editor toolbar.

Section 3: Right-Click Options

Playfield (& backdrop) objects fall into 2 main categories for right-click menu options;

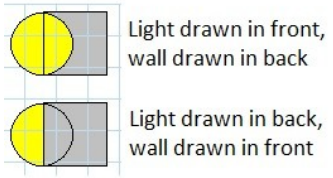
Orientation   on-orientation

The non-orientation group includes: gates, flippers, plungers, bumpers, spinners, timers, kickers, decals, light-sequencers, textboxes, and emreels.



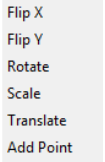
All objects have the options shown on here, but for the non-orientation group only these 3 options are available:

Draw In Front – If 2 objects are placed in the editor at the same place, or overlap, draw in front ensures that this object is placed above the other in drawing.
Draw In Back – If 2 objects are placed in the editor at the same place, or overlap, draw in front ensures that this object is placed below the other in drawing.

Example: 

Lock – this option means that the object is locked in place on the table, and cannot be moved, or deleted (unless unlocked).

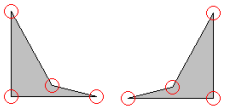
The orientation group includes: walls, targets, ramps, triggers, lights, custom triggers, and custom lights.



For these objects there are also the additional options shown here.
(For non-custom shaped triggers and lights, only Translate is selectable from this menu)

Flip X – Reverses the horizontal orientation of each control point of an object:

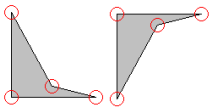
Example



These walls are copies of each other with Flip X applied to one of them.

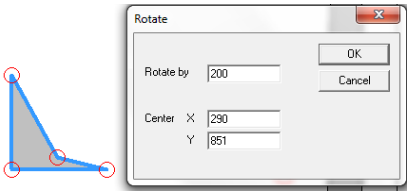
Flip Y - Reverses the vertical orientation of each control point of an object:

Example

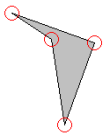


These walls are copies of each other with Flip Y applied to one of them.

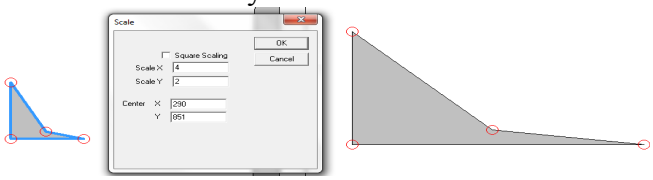
Rotate – Displays a dialog box with options to rotate by a chosen angle, and by what co-ordinates to rotate it around (default co-ordinates are the geometric center of the object).



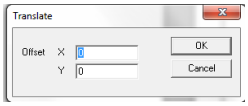
Would look like this



Scale – Displays a dialog box with options to change the size of an object by scaling the horizontal and vertical axis. This can be done independently, or they can be forced to be scaled by the same amount by having "Square Scaling" selected (square scaling is selected by default). The center point about which the scaling takes place can be set with the center x & y co-ordinates.

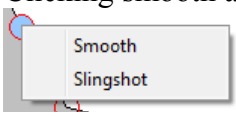


Translate – Displays a simple dialog box with "offset" for x & y co-ordinates – the co-ordinates entered will move the object by that amount horizontally (x) and vertically (y).

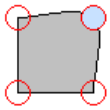


Add Point – adds a control point to the object at the point that it clicked.

Control-point right-click: (control-points also have their own options)
Clicking smooth and slingshot toggles these options (indicated with a tick)



Smooth causes the lines around a control point to curve towards and away from that point.

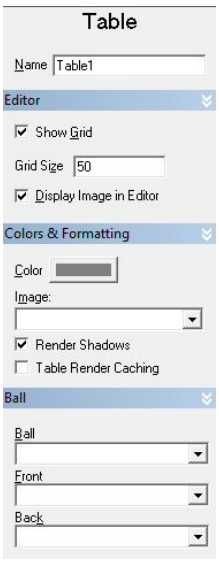


Selecting slingshot causes the part of the object between that point and the next point to act like a slingshot – indicated by a bold line.

The 2 control points that connect the slingshot cannot be smooth points, and selecting slingshot will deselect smooth for either of the points if it was selected.

Section 4.1: Table Options

Selecting the "options"  button on the editor toolbar toggles the options tab on the right hand side of the editor. The tab displays different options depending on what is currently selected; this section deals with the options available when the playfield is selected (this can be done by left clicking on any part of the editor other than an object, object button, or menu, and only when backdrop view is not selected).



The default view of the table options has all the sections expanded except for physics. Options can be expanded or contracted by clicking the double arrow on the right-hand side of the blue bar to which they pertain.

Editor:

Show Grid – toggles a grid overlay on the playfield for editing assistance in lining up, or sizing, objects. Grid does not appear in VP player view.

Grid Size – changes the space between lines on the grid if it is enabled

Display Image in Editor – show the playfield image selected in colors & formatting during editing (may slow down the performance of VP during edit only).

Colors & Formatting:

Color – click to open up a color dialog box, select the color for the surface of the playfield (only applies if no image is selected in the box below).

Image (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will be stretched to fit the entire playfield area (as defined in the physics options)

Render Shadows – defines whether VP player will render (draw) shadows of playfield objects. Assumes the playfield is lit from the bottom left-hand side so shadows appear on the right-hand side of objects. Rendering shadows will use more graphics system resources.

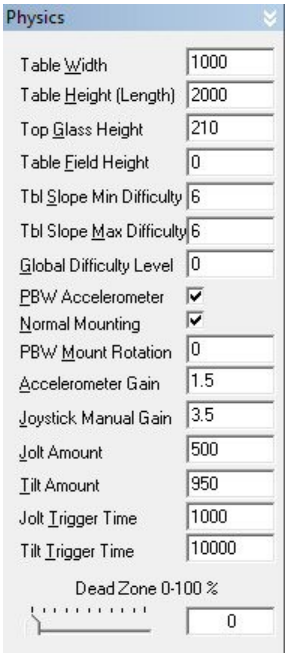
Table Render Caching – creates a file in the directory the table was loaded from with the name - table name.vpcache. When you have this selected, VP renders the table and saves a copy of the rendered data into the vpcache file. On subsequent loads if this is checked and a vpcache file already exists it will read in the file and accelerate the rendering process.

Ball:

Ball (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will be scaled down to 50 x 50 units and appear instead of the ball.

Front (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will appear as an image stuck on the "front" of the ball, and will follow round the roll of the ball.

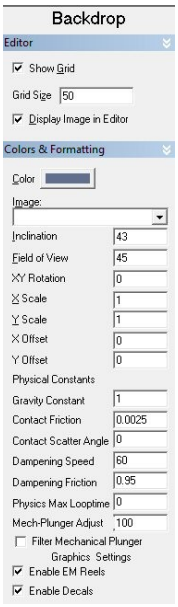
Back (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will appear as an image stuck on the "back" of the ball, and will follow round the roll of the ball.



Physics:
Table Width – sets the horizontal width of the playfield in VP units.
Table Height (Length) – sets the vertical length of the playfield in VP units.
Top Glass Height – sets the maximum height of the ball movement. Although glass does not appear in the VP player, if a ramp, for example, goes above the glass height the ball will be unable to travel to the top of that ramp.
Table Field Height – sets the height of the bottom playfield surface inside the cabinet; e.g. if you want to add 50 units of depth to an existing table without changing the height of all the objects you have already placed, set this to -50 and add a wall at 0 for the ball to sit on.
Tbl Slope Min Difficulty – sets the minimum slope of the table. The range of possible slopes is modified by the Global Difficulty Level (see below).
Tbl Slope Max Difficulty – sets the maximum slope of the table. The range of possible slopes is modified by the Global Difficulty Level (see below).
Global Difficulty Level – sets a modifying value to the table for table slope, scatter angle and scatter velocity of playfield objects. Altering the Global Difficulty Level alters the general difficulty of the game. The default value means that nothing is changed from individually specified editor settings of each object. Range of values 0.0000001 to 1.0000000. Table slope is calculated using: Slope = SlopeMin + ((SlopeMax – SlopeMin) x GlobalDifficulty)
PBW Accelerometer – enables polling the PinballWizard, MotionIO Device, Ushock Board, and the Microsoft Freestyle USB Game Controller to determine table sloping and react to nudges.
Normal Mounting – assumes the default direction/orientation of the above named PBW controllers.
PBW Mount Rotation – changes the X and Y axis of the controllers - uncheck normal mounting to use alternate options here.
Accelerometer Gain – decreases the sensitivity of the controller so you don't have to tilt as much to accomplish the same change on screen.
Joystick Manual Gain – unknown at this time.
Jolt Amount – increases (multiplies) the nudge force applied to the ball with an accelerometer.
Tilt Amount – unknown at this time.
Jolt Trigger Time – unknown at this time.
Tilt Trigger Time – unknown at this time.
Dead Zone 0-100% – (slider &/or number entry) sets the percentage area of movement around the center point of a joystick, or other game controller, before the control recognizes the movement.

Section 4.2: Backdrop Options

Selecting the "options"  button on the editor toolbar toggles the options tab on the right hand side of the editor. The tab displays different options depending on what is currently selected; this section deals with the options available when the backdrop is selected (this can be done by left clicking on any part of the editor other than an object, object button, or menu, and only when backdrop view is selected).



Editor:

Show Grid – toggles a grid overlay on the playfield for editing assistance in lining up, or sizing, objects. Grid does not appear in VP player view.

Grid Size – changes the space between lines on the grid if it is enabled

Display Image in Editor – show the backdrop image selected in colors & formatting during editing (may slow down the performance of VP during edit only).

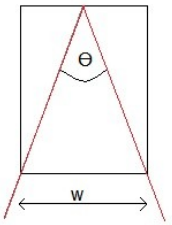
Colors & Formatting:

Color – click to open up a color dialog box, select the color for the backdrop (only applies if no image is selected in the box below).

Image (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will be stretched to fit the entire backdrop area.

Inclination – sets the angle at which the player is apparently viewing the playfield from – 0 represents looking straight down on the table from above, 90 would represent looking directly at the edge of the playfield surface (horizontally).

Field of View – sets the width of apparent angle of view using the bottom of the table as the maximum observable width.



Θ = Field of view angle
w = width of display

XY Rotation – sets an angle of rotation of the playfield relative to the backdrop, by rotating the playfield about the vertical bottom, and horizontal center of the backdrop. This setting does not display any changes in the editor, and only displays the rotation in the VP player.

X Scale – changes the rendered size of the table in full-screen mode with a 0º inclination and 45º field of view.

Y Scale – changes the rendered size of the table in full-screen mode with a 0º inclination and 45º field of view.

X Offset – allows horizontal centering of the table in cabinet-style view

Y Offset – allows vertical centering of the table in cabinet-style view

Physical Constants:

Gravity Constant – sets the strength of gravity (& therefore acceleration) applied to the ball.

Contact Friction – sets the friction applied to all objects, including the playfield. Some individual objects can have friction set to more, but not less, than this value in their own object settings.

Contact Scatter Angle – adds a random element for ball trajectory after it has hit an object

Dampening Speed – Sets the maxballspeed, which is the top speed the ball is allowed to travel at.

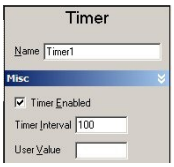
Dampening Friction – unknown at this time.

Physics Max Looptime – unknown at this time.
Mech-Plunger Adjust – unknown at this time.
Filter Mechanical Plunger – unknown at this time.
Graphics Settings
Enable EM Reels – toggles whether EM reels can be used on the backdrop or not
Enable Decals – toggles whether decals can be used on the backdrop or not

Section 5: Object Settings

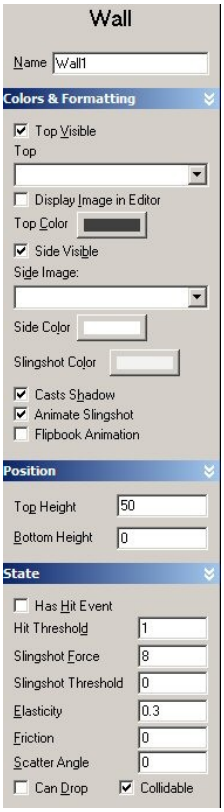
With the exception of decals, all playfield objects have a "Misc" section to their options. This section sets details of the object's timer. This section is exactly the same for each object, and the only section in the options for the "timer" object. Therefore the timer will be dealt with first, and the "Misc" section of each object will not be mentioned in each object's options description.
Object names: all object names in VP must be unique (they are case insensitive) and must not contain any spaces.

5.1 – Timer



Timer enabled – sets whether the timer will be running at initialization of the table. Can be changed in script.
Timer Interval – time period of the timer (how long between timer cycles), measured in milliseconds.
User Value – acts as a pre-defined variable in the script. Setting this here sets a value for the variable at the initialization of the table.

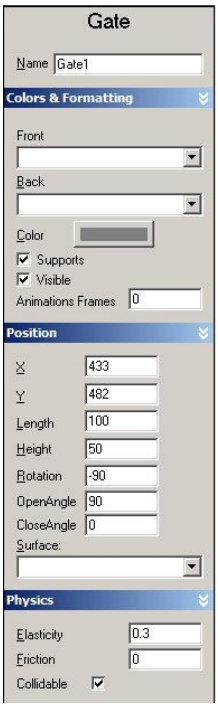
5.2 Wall & Target



Walls and targets are together here as they are effectively the same object, with 1 major difference in display and 3 minor differences in their default options. Major display difference is in how each object displays the "side image". The minor differences in default options are; the initial shape, the top and side colors, and whether "Has Hit Event" is selected or not (for wall default is not selected, for target it is selected by default). Every other option, including the default naming, is the same for both objects.
Colors & Formatting:
Top Visible – toggles whether the top of the wall is rendered in the VP player; overwrites any image or color setting. Shadows are cast regardless of this setting if selected to be cast (see below).
Top – (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will be stretched to fit the entire playfield area and so only the portion of the image where the wall appears will be seen on the wall.
Display Image in Editor – show the top image selected above during editing (may slow down the performance of VP during edit only).
Top Color – click to open up a color dialog box, select the color for the top surface of the wall.

Side Visible – toggles whether the side of the wall is rendered in the VP player; overwrites any image or color setting. Shadows are cast regardless of this setting if selected to be cast (see below).
Side – (drop-down selection) – choose an image from any loaded into VP via the image manager. For a *Wall* object the image will be wrapped around all of the sides of the wall, whereas for a *Target* object the image will be wrapped onto the front side of the target (and then around the other 3 sides of the object, but on the default target these other sides are unseen). This is due to the way the Texture Coordinates are set on the control points of the default wall or target object.
Side Color – click to open up a color dialog box, select the color for the side surface of the wall.
Slingshot Color – click to open up a color dialog box, select the color for the slingshot effect when a ball strikes a wall that has control points selected to produce a slingshot side to it.
Casts Shadow – toggles whether object casts a shadow (shadow will only be rendered if selected when "render shadows" is also selected in Table Options).
Animate Slingshot – toggles whether slingshot animation is shown (using slingshot color selected above).
Flipbook Animation – toggles whether the top of the wall is rendered when dropped. If selected the top of the wall is not rendered when wall is dropped.
Position:
Top Height – sets the top height of the wall in VP units
Bottom Height – sets the bottom height of the wall in VP units
State:
Has Hit Event – toggles whether the wall can be scripted to fire events when hit (*Wall* is default off, *Target* is default on). Hit will not be counted if the side of the wall hit has a slingshot set by the control points.
Hit Threshold – determines how hard the wall must be hit for the hit event to fire (no idea what this is measured in though…it's the same as kicker strength though ?)
Slingshot Force – sets the strength of the slingshot effect if one is available due to the setting of control points.
Slingshot Threshold – determines how hard the wall must be hit for the slingshot (if selected) to fire (no idea what this is measured in though…it's the same as kicker strength though ?)
Elasticity – sets how elastic a collision with the ball will be. Setting of 1 represents a perfectly elastic collision (no loss of kinetic energy during collision). Settings above 1 cause the ball to gain energy in a collision with the wall.
Friction – sets the friction of the object compared to the global "contact friction" setting in the backdrop options. Can be set between 0 and 1.
Scatter Angle – adds a random element for ball trajectory after the ball hits the wall.
Can Drop – toggles whether the wall can be set to drop or not via the script.
Collidable – toggles whether the wall interacts with the ball or not. If set to not collidable then the wall will be rendered in the VP player but will have no influence on the ball, and no hit events or slingshots will be able to fire.

5.3 Gate:



Colors & Formatting:

Front – (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will appear of the side of the gate by which the ball can enter.

Back – (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will appear of the side of the gate by which the ball leaves.

Color – click to open up a color dialog box, select the color for the surfaces of the gate if no images are chosen above.

Supports – toggles whether support legs are attached & visible

Visible – toggles whether the gate is rendered or not. If visible is not selected the supports will still be rendered if selected.

Animation Frames – sets the number of frames in which the action of the gate is shown, minimum 2 (0 sets the default number of frames).

Position:

X – sets the horizontal position of the center of the gate on the playfield.

Y – sets the vertical position of the center of the gate on the playfield.

Length – sets the width (length) of the gate not including supports, in VP units.

Height – sets the top height of the gate, in VP units.

Rotation – sets the angle of the gate. 0° allows the ball to travel vertically up through the gate, as indicated by the arrow on the playfield icon. -90° is the default angle.

Open Angle – sets the angle to which the gate flap is at to the horizontal when fully open.

Close Angle – sets the angle to which the gate flap is at to the horizontal when fully closed.

Surface – (drop-down selection) sets the surface, or rather bottom height, on which the gate sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.

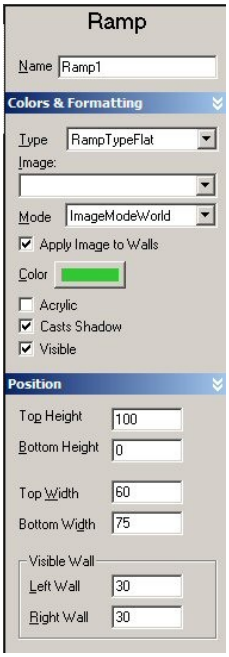
Physics:

Elasticity – sets how elastic a collision with the ball will be. Setting of 1 represents a perfectly elastic collision (no loss of kinetic energy during collision). Settings above 1 cause the ball to gain energy in a collision with the gate.

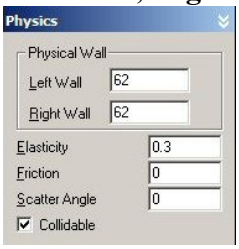
Friction – sets the friction of the object compared to the global "contact friction" setting in the backdrop options. Can be set between 0 and 1.

Collidable – toggles whether the gate interacts with the ball or not. If set to not collidable then the gate will be rendered in the VP player but will have no influence on the ball, and no hit events will be able to fire.

5.4 Ramp



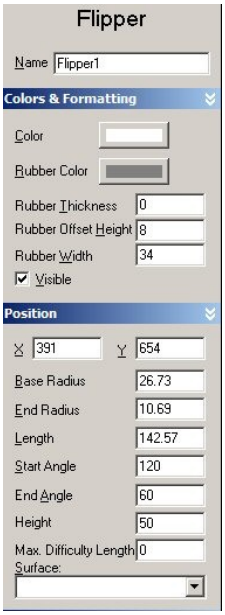
Colors & Formatting:
Type – (drop-down selection) determines the type of ramp from a choice of 5; 2-wire, 3-wire left, 3-wire right, 4-wire, or flat.
Image – (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will only be displayed on ramp type flat.
Mode – (drop-down selection) determines how the image chosen above will be displayed (only applies if ramp type flat), from a choice of 2; ImageModeWorld – the image chosen will be stretched to fit the entire playfield area and so only the portion of the image where the wall appears will be seen on the ramp; and ImageModeWrap the image chose will be stretched to fit the entire area of the ramp surface excluding the walls.
Apply Image to Walls – toggles whether the selected image will be drawn on the walls.
Color – click to open up a color dialog box, select the color for the ramp.
Acrylic – toggles whether a flat ramp is rendered with an acrylic texture/dithered transparency, or not. Requires hardware rendering to be on.
Casts Shadow – toggles whether object casts a shadow (shadow will only be rendered if selected when "render shadows" is also selected Table Options).
Visible – toggles whether the ramp is rendered in the VP player; overwrites any image or color setting.
Position:
Top Height – sets the height for the top of the ramp; the ramp will have a smooth gradient along its' entire length from bottom height to top height.
Bottom Height – sets the height for the bottom of the ramp; the ramp will have a smooth gradient along its' entire length from bottom height to top height.
Top Width – sets the width for the top of the ramp; the ramp will have a smooth curvature along its' entire length from the bottom width to the top width.
Bottom Width – sets the width for the bottom of the ramp; the ramp will have a smooth curvature along its' entire length from the bottom width to the top width.
Visible Wall:
Left Wall; Right Wall – sets the heights of the visibility of the walls on a flat ramp.



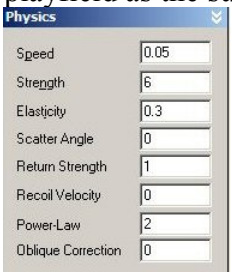
Physics:
Physical Wall:
Left Wall; Right Wall – sets the heights of the "physical" walls at either side of a flat ramp.
Elasticity – sets how elastic a collision with the ball will be. Setting of 1 represents a perfectly elastic collision (no loss of kinetic energy during collision). Settings above 1 cause the ball to gain energy in a collision with the ramp.
Friction – sets the friction of the object compared to the global "contact friction" setting in the backdrop options. Can be set between 0 and 1 (*assume this is a percentage of the global value? – though 0 doesn't give 0 friction).
Scatter Angle – adds a random element for ball trajectory after the ball hits the ramp.

Collidable – toggles whether the ramp interacts with the ball or not. If set to not collidable then the ramp will be rendered in the VP player but will have no influence on the ball.

5.5 Flipper



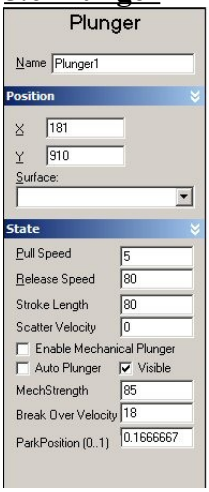
Colors & Formatting:
Color – click to open up a color dialog box, select the color for the flipper bat.
Rubber Color – click to open up a color dialog box, select the color for the flipper rubber.
Rubber Thickness – sets the thickness of the rubber in VP units
Rubber Offset Height – sets the height above the flipper surface at which the rubber begins to be rendered
Rubber Width – sets the height of the flipper rubber from the *Rubber Offset Height* set above.
Visible – toggles whether the flipper is rendered in the VP player.
Position:
X – sets the horizontal position of the center of the flipper base on the playfield.
Y – sets the vertical position of the center of the flipper base on the playfield.
Base Radius – sets the radius of the flipper base in VP units, the flipper will follow a smooth taper between the base radius and end radius along its' entire length.
End Radius – sets the radius of the flipper end in VP units, the flipper will follow a smooth taper between the base radius and end radius along its' entire length.
Length – sets the length of the flipper in VP units.
Start Angle – sets the angle with respect to the horizontal of the flipper at rest.
End Angle – sets the angle with respect to the horizontal of the flipper at full extension.
Height – sets the height of the top of the flipper from the flipper surface in VP units.
Max. Difficulty Length – sets a minimum shorter length for the flipper depending on the Global Difficulty Level. When set the default arc and the minimum length arc are shown in the editor, along with the actual arc that will be used as determined by any Global Difficulty Level currently set.
Surface – (drop-down selection) sets the surface, or rather bottom height, on which the flipper sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.



Physics:
Speed – sets the speed at which the flipper moves through its' swing.
Strength – sets the amount of energy imparted to the ball.
Elasticity – sets how elastic a collision with the ball will be. Setting of 1 represents a perfectly elastic collision (no loss of kinetic energy during collision). Settings above 1 cause the ball to gain energy in a collision with the flipper.
Scatter Angle – adds a random element for ball trajectory after the ball hits the flipper.
Return Strength – sets the strength/speed at which the flipper will rotate to start (values between 0 & 1).

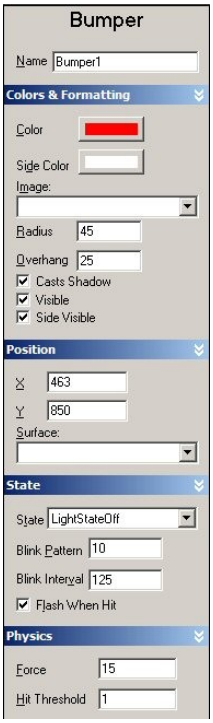
Recoil Velocity – sets the way the flipper responds to the ball velocity…effectively how the ball "springs" off the flipper, and how the flipper itself can recoil when struck by a ball.
Power-Law – sets a multiplier against the rest of the flipper settings. Values between 0 & 4. [A better description would be welcomed here]
Oblique Correction – sets how the ball moves off the flipper, with a higher setting meaning the ball has a greater tendency to travel up towards the center of the table. [A better description would be welcomed here]

5.6 Plunger



Position:
X – sets the horizontal position of the center of the plunger on the playfield.
Y – sets the vertical position of the furthest point of pull-back of the plunger on the playfield.
Surface – (drop-down selection) sets the surface on which the plunger sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.
State:
Pull Speed – sets the speed at which the plunger moves into the pull-back position.
Release Speed – sets the speed at which the plunger moves into its rest position.
Stroke Length – sets the length of the plunger, and its' pull-back. Minimum length is 16.5 VP units.
Scatter Velocity – sets a variation in the plunger velocity.
Enable Mechanical Plunger – allows the use of an analogue plunger control.
Auto Plunger – unknown at this time.
Visible – toggles whether the plunger is rendered in VP player.
MechStrength – sets the strength/speed of an analog plunger if enabled above.
Break Over Velocity – sets the maximum speed of an analog plunger (if enabled above) regardless of the actual speed of motion if faster.
Park Position (0..1) – where the analog plunger is at rest (if enabled above).

5.7 Bumper



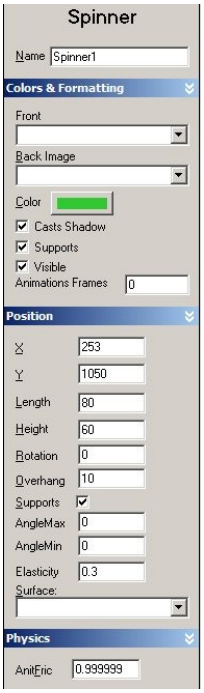
Colors & Formatting:
Color – click to open up a color dialog box, select the color for the top surface (cap) of the bumper.
Side Color – click to open up a color dialog box, select the color for the side (shaft) of the bumper.
Image – (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will be stretched to fit the top of the bumper cap (as defined by the *radius* below).
Casts Shadow – toggles whether object casts a shadow (shadow will only be rendered if selected when "render shadows" is also selected in Table Options).
Radius – sets the width of the central column (shaft) of the bumper.
Overhang – sets the width of the bumper cap beyond that of the central column (shaft) of the bumper.
Visible – toggles whether the bumper is rendered in the VP player; overwrites any image or color setting. This setting causes the entire bumper to be rendered or not regardless of the *Side Visible* setting below. Shadows are cast regardless of this setting if selected to be cast (see above).
Side Visible – toggles whether the bumper side (shaft) is rendered in the VP player; overwrites any image or color.

Position:
X – sets the horizontal position of the center of the bumper on the playfield.
Y – sets the vertical position of the center of the bumper on the playfield.
Surface – (drop-down selection) sets the surface on which the bumper sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.

State:
State – (drop-down selection) sets the light state of the bumper from a choice of 3; LightStateOff, LightStateOn, LightStateBlinking.
Blink Pattern – sets a binary sequence of the light state being on (1), or off (0), which the LightStateBlinking will follow and repeat until the light state is set differently (via the script).
Blink Interval – sets the time in milliseconds between each state in the blink pattern of a light with LightStateBlinking.
Flash When Hit – toggles whether the bumper will flash once when hit

Physics:
Force – sets the strength of the bumper and determines how hard the bumper will strike the ball.
Hit Threshold – sets the impact force required for the bumper to fire.

5.8 Spinner

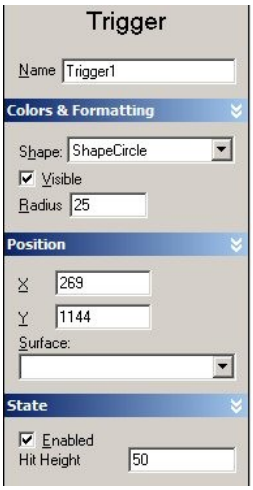


Colors & Formatting:
Front – (drop-down selection) – choose an image from any loaded into VP via the image manager.
Back Image – (drop-down selection) – choose an image from any loaded into VP via the image manager.
Color – click to open up a color dialog box, select the color for the front & back of the spinner if no image selected.
Casts Shadow – toggles whether a shadow of the spinner at rest is shown (i.e. shadow does not follow movement of the spinner); provided Render Shadows is selected in table options.
Supports – toggles whether the side supports of the spinner are rendered.
Visible – toggles whether the spinner front and back are rendered in the VP player; overwrites any image or color setting. Shadows are cast regardless of this setting if selected to be cast (see above).
Animations Frames – as the spinner rotates through 360º, the angles shown are determined by (360º)/(number of animation frames). The default value of 0 shows the maximum number of frames.

Position:
X – sets the horizontal position of the center of the spinner on the playfield.
Y – sets the vertical position of the center of the spinner on the playfield.
Length – sets the length of the spinner faces (does not include supports).
Height – sets the height of the spinner faces (does not include supports).
Rotation – sets the angle of the spinner in relation to the x-axis of the playfield.
Overhang – sets the distance of the support legs from the edges of the spinner face.
Supports – toggles whether the side supports of the spinner are rendered.
Angle Max – sets the maximum angle the spinner will rotate through
Angle Min – sets the minimum angle the spinner will rotate through, although this only has an apparent effect if the Angle Min is larger than the Angle Max.
Elasticity – sets how elastic a collision with the ball will be. Setting of 1 represents a perfectly elastic collision (no loss of kinetic energy during collision). Settings above 1 cause the ball to gain energy in a collision with the spinner.
Surface – (drop-down selection) sets the surface on which the spinner sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.

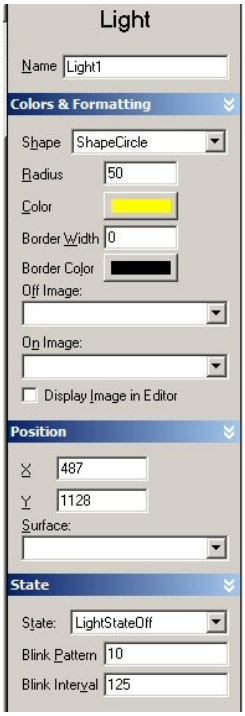
Physics:
AntiFric – determines the friction between the spinner and its' supports. Value setting between 0 & 1. A setting of 0 indicates no friction and the spinner will spin indefinitely.

5.9 Trigger



Colors & Formatting:
Shape – (drop down selection) choose between "ShapeCircle" and "ShapeCustom". The custom option gives the trigger control points, and these can be used in the same way as for a wall object. When the custom trigger is selected changing the following options has no effect; visible (custom trigger is never rendered/visible), radius, X and Y position.
Visible – toggles whether a circle trigger is rendered in the VP player
Radius - sets the size of a circle trigger.
Position:
X – sets the horizontal position of the center of a circle trigger on the playfield.
Y – sets the vertical position of the center of a circle trigger on the playfield.
Surface – (drop-down selection) sets the surface on which the trigger sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.
State:
Enabled – toggles whether the trigger routines are activated in script when the trigger is hit
Hit Height – maximum height at which the trigger recognizes the ball. If the center of a ball travels over a trigger at or below this value the trigger hit event is fired.

5.10 Light



Colors & Formatting:
Shape – (drop down selection) choose between "ShapeCircle" and "ShapeCustom". The custom option gives the light control points, and these can be used in the same way as for a wall object.
Radius – sets the size of ShapeCircle light only
Color – click to open up a color dialog box, select the color for the light. The color does not apply if "off" or "on" images are selected (see below).
Border Width – sets the size of border to be drawn around a light. (Applies to both shapes of lights, but is only shown in the editor with ShapeCircle).
Border Color – click to open up a color dialog box, select the color for the light border.
Off Image – (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will be stretched to fit the entire playfield area and so only the portion of the image where the light appears will be seen on the light when the light state is set to off (including the "off" cycle in a blinking sequence).
On Image – (drop-down selection) – choose an image from any loaded into VP via the image manager. The image chosen will be stretched to fit the entire playfield area and so only the portion of the image where the light appears will be seen on the light when the light state is set to on (including the "on" cycle in a blinking sequence).

Display Image in Editor – toggles whether the image chosen will be displayed in the editor. Will only display an image in the editor if there is an image chosen for both on and off states. Will display the on image or off image depending on the on/off state set in the state option (see below), if LightStateBlinking is selected, no image will be shown in the editor.

Position:

X – sets the horizontal position of the center of a ShapeCircle light, or the light center cross of a ShapeCustom light, on the playfield.

Y – sets the vertical position of the center of a ShapeCircle light, or the light center cross of a ShapeCustom light, on the playfield.

Surface – (drop-down selection) sets the surface on which the light sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.

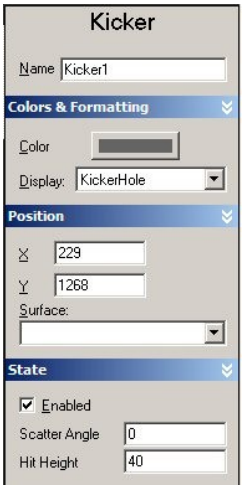
State:

State – (drop-down selection) choose the initial light state as on, off or blinking

Blink Pattern – sets the on/off blinking pattern of a light with LightStateBlinking. "1" represents light state on, "0" represents light state off. For example, "110010" represents a light that will blink with the pattern "on, on, off, off, on, off".

Blink Interval – sets the time in milliseconds between each state in the blink pattern of a light with LightStateBlinking.

5.11 Kicker



Colors & Formatting:

Color – click to open up a color dialog box, select the color for the kicker. This only applies to KickerHole, and KickerCup (see below).

Display – (drop-down selection) choose from 4 types of kickers; KickerCup – displays a cup shaped kicker in the VP player, KickerHidden – renders and then hides a kicker in the VP player, KickerHole – displays a hole shape in the playfield in the VP player, KickerInvisible – does not render the kicker in the VP player.

Position:

X – sets the horizontal position of the center of the kicker.

Y – sets the vertical position of the center of the kicker.

Surface – (drop-down selection) sets the surface on which the kicker sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.

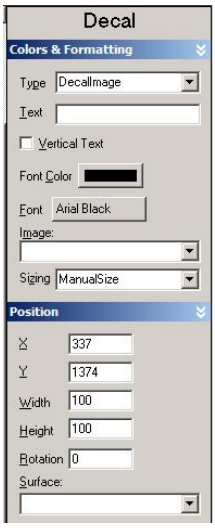
State:

Enabled – toggles whether the kicker has any influence on the ball.

Scatter Angle – adds a random element for ball trajectory after the ball hits the kicker.

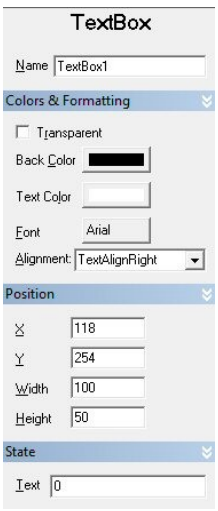
Hit Height – maximum height at which the kicker recognizes the ball. If the center of a ball travels over a kicker at or below this value the kicker hit event is fired.

5.12 Decal



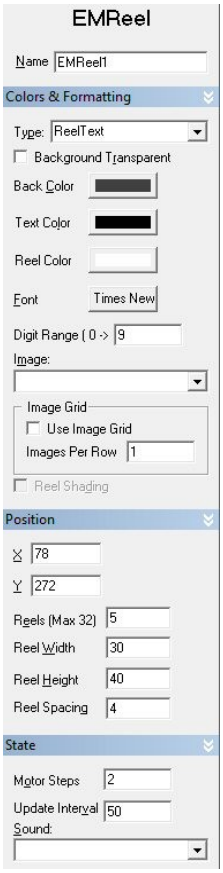
Colors & Formatting:
Type – (drop-down selection) choose between DecalImage, or DecalText.
Text – the text displayed if type chosen is DecalText
Vertical Text – toggles vertical or horizontal text if type chosen is DecalText
Font Color – click to open up a color dialog box, select the color for the font if type chosen is DecalText.
Font – click to open up a font dialog box, select the font, font style, size, and script (available options dependant on the fonts installed on your version of Windows).
Image – (drop-down selection) – choose an image from any loaded into VP via the image manager.
Sizing – (drop-down selection) choose from AutoSize, AutoWidth, and ManualSize. AutoSize only applies to DecalText and will alter the size of the decal to fit the text appropriately. AutoWidth will re-adjust the width of the decal to set the correct aspect ratio of the image as loaded via the image manager, with regards to the height set in the position options (see below). ManualSize sets the size of the decal as defined by the position options (see below).
Position:
X – sets the horizontal position of the center of the decal.
Y – sets the vertical position of the center of the decal.
Width – sets the horizontal width of the decal.
Height – sets the vertical height of the decal.
Rotation – sets the angle of rotation for the decal.
Surface – (drop-down selection) sets the surface on which the kicker sits. Selections include wall and ramp objects, no selection sets the playfield as the surface.

5.13 TextBox



Colors & Formatting:
Transparent – toggles whether the back color is displayed.
Back Color – click to open up a color dialog box, select the back (background) color for the textbox.
Text Color – click to open up a color dialog box, select the color for the text.
Font – click to open up a font dialog box, select the font, font style, size, and script (available options dependant on the fonts installed on your version of Windows).
Alignment – (drop-down selection) choose the alignment of the text in the textbox from; center, left, or right.
Position:
X – sets the horizontal position of the left hand side of the textbox.
Y – sets the vertical position of the top of the textbox.
Width – sets the horizontal width of the textbox.
Height – sets the vertical height of the textbox.
State:
Text – sets the initial text to be displayed in the textbox.

5.14 EMReel



Colors & Formatting:

Type – (drop-down selection) gives options for ReelText, or ReelImage.

Background Transparent – toggles whether the back (border) is displayed.

Back Color – click to open up a color dialog box, select the back (border) color for the reel, if background transparent is unticked.

Text Color – click to open up a color dialog box, select the color for the text.

Reel Color – click to open up a color dialog box, select the color for the reels, if the type is ReelText.

Font – click to open up a font dialog box, select the font, font style, size, and script (available options dependant on the fonts installed on your version of Windows), if the type is ReelText.

Digit Range – if the type is ReelImage then this defines the number of digit images on a reel up to 199. If no image grid is being used (see below) then the width of the image being used will be divided into this number of pieces and each section displayed as a separate reel image.

Image – (drop-down selection) – choose an image from any loaded into VP via the image manager.

Image Grid:

Use Image Grid – toggle to indicate to the editor whether to use the image selected as a grid of images or a single row of images, if type is ReelImage.

Images Per Row – if a grid of images is being used then the image loaded will be separated in width by the images per Row and in height by the Digit Range divided by the Images Per Row; i.e. the VP player will calculate the image as a grid containing a "Digit Range" of cells that is "Images Per Row" wide, and therefore make the height of each cell "Digit Range"/"Images Per Row".

Reel Shading – this is always unavailable as it has not yet been implemented into VP.

Position:

X – sets the horizontal position of the left hand side of the reel.

Y – sets the vertical position of the top of the reel.

Reels (Max 32) – sets the number of reel digits that make up the reel, and each digit will display 0, or the first image, by default. For ReelText Reels indicates the maximum number that can be displayed by this reel (e.g. if Reels=4 then the highest number possible on the reel is 9999, adding 1 to this will reset the entire reel to 0000).

Reel Width – sets the width of each reel digit

Reel Height – sets the height of the reel

Reel Spacing – sets the size of the spacing (size of border if Background Transparent is not ticked) between each digit in the reel.

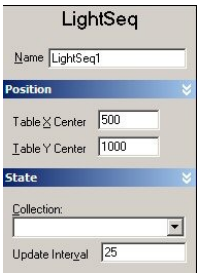
State:

Motor Steps – sets the number of animation steps when moving between images or numbers on a reel.

Update Interval – sets the time in milliseconds between each motor step (minimum value 10)

Sound – (drop-down selection) – choose a sound from any loaded into VP via the sound manager. The reel will play this sound each time a reel moves.

5.15 LightSeq



Position:

Table X Center – sets the horizontal reference point which the LightSeq uses as a center for its' effects.

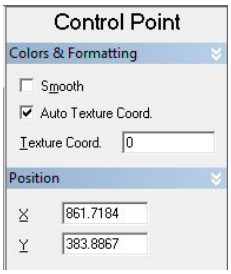
Table Y Center – sets the vertical reference point which the LightSeq uses as a center for its' effects.

State:

Collection – (drop-down selection) select the collection of lights or bumpers (or combination of both) you want the light sequencer to control. The drop-down selection shows all collections set up through the collection manager, regardless of whether they are suitable for use with a light sequencer or not.

Update Interval – sets the speed, or time, in milliseconds at which the light sequencer will run.

5.16 Control Points



Colors & Formatting:

Smooth – toggles whether the lines around a control point to curve towards and away from that point, or are sharp angles.

Auto Texture Coord [*Walls & Targets only*] – determines whether the Texture Coord entered (below) has any influence on the way images are wrapped around the sides of the object. If all points have Auto Texture Coord selected, any image applied will be wrapped around the entire object.

Texture Coord [*Walls & Targets only*] – determines the reference points for applying an image across the side of the object, provided Auto Texture Coord (above) isn't ticked. The selected image will be wrapped between coordinates 0 and 1. To produce the view of the image loaded via the image manager 0 must be to the left of 1, otherwise a mirror image will be shown. The image can be stretched along any part by introducing control points with coordinates between 0 and 1 – e.g. if a point has a Texture Coord of 0.9, 90% of the image will be rendered between 0 and that point, and the remaining 10% will be rendered between that point and Texture Coord 1. If no point is selected as Texture Coord 1, then the image will wrap from 0 to the next 0. If there are no 0 or 1 points, the image will wrap around the whole object as if all points had Auto Texture Coord selected.

Position:

X – sets the horizontal position of the control point.

Y – sets the vertical position of the control point.

Section 6: Escape Key

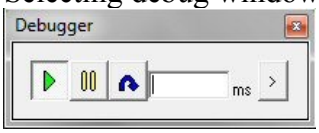
Whilst in VP player, pressing the escape key pauses all aspects of the player (with the exception of sounds that are currently playing), and brings up a 3-option window.

Quit to Editor – stops & closes the VP player and returns to the editor (or the application that launched the VP player).

Resume Game – releases the pause mode, and immediately returns to the VP player, resuming from wherever the game was when the escape key was pressed.

Debug Window – (not available if the game has been saved as protected). [Please note: most of this description is taken from vpforums postings by Randy Davies (creator of VP) in Oct 2004]

Selecting debug window brings up this window:



The window has 4 buttons, 3 grouped together and a "step window". When the debug window is active, the game is paused. The status of the three grouped buttons indicates how the game will play when the game window is active.

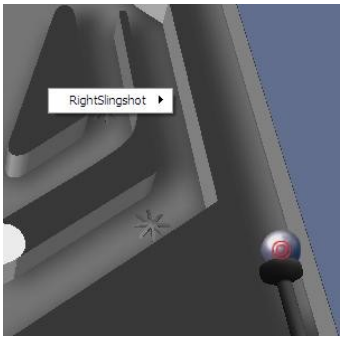

When the play button is on (as shown above), the game will play normally.

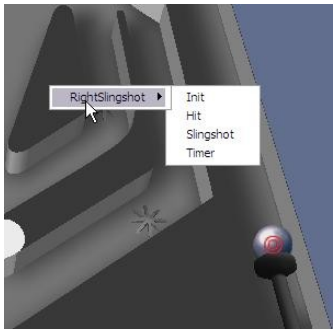

When the pause button is on, the game will be paused.


When the step button in on, the game will play for the number of milliseconds specified in the step window, and then return to the paused state.


The expand button is explained below.

In the game window, while the debug window is showing, you can directly manipulate the table.





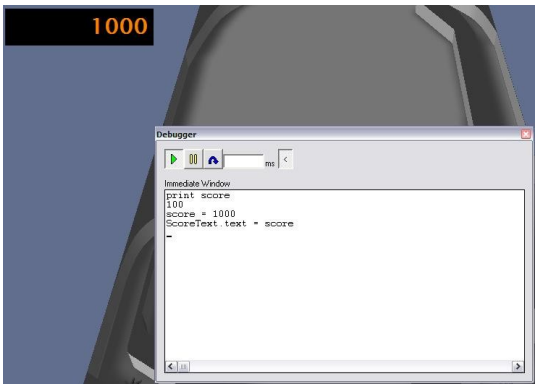
You can right-click on any object and bring up a menu of all the events that object has. Then you can select an event and it will fire. This simulates the ball hitting that object so you can try things out in the table much faster. In some cases script might differentiate between which ball is doing the action. The 'active' ball has a red target drawn on it. You can right click a ball to make it the active ball. When script checks what ball preformed the action in question, it will be the active ball.

Note that this can be used to get the game into a state it was not expecting, like hitting the ball drain before the game is started.

The expand button expands the immediate window. The immediate window works much like the window of the same name in VB (except you can't use `?` as shorthand for `PRINT`.) There is a ton of stuff you can do with this window so I will attempt to point out the highlights.

If you have a global variable called 'score' you can display its value with `PRINT`. (Note that the first command in your VB Script must be `Option Explicit`, or at the very least the variable you are attempting to print must be defined, e.g. Dim Score)

```
print score
```



You can set the value like this:

```
score = 10000000
```

If you have a subroutine in your script called `DoSomeStuff`, you can call it directly by just typing:

```
DoSomeStuff
```

If you have an element on the table, say a light named Light1, you can edit it directly:

```
Light1.State = LightStateBlinking
```

Or you can get the value of course:

```
Print Light1.State
```

If you want your script to notify you of things, like when a timer fires, you can do this in the table script:

```
Sub Thing_Timer()
      Debug.Print "Timer Fired"
End Sub
```

`Debug.Print` will only do something if the debug window is up, so if you are expecting messages then open it right away. The immediate window does have to be expanded.

Appendix I – Keycodes

Keycode 1 (Escape key) is reserved by VP as a necessity to be able to exit the program, and therefore cannot be scripted.

Reserved key names: (can be set using “keys” in the “preferences menu”)
LeftFlipperKey
RightFlipperKey
LeftTiltKey
RightTiltKey
CenterTiltKey
PlungerKey
AddCreditKey
AddCreditKey2
LeftMagnaSave
RightMagnaSave
MechanicalTilt
StartGameKey
ExitGame

Key	Keycode		
1	2	E	18
2	3	END	207
3	4	EQUALS	13
4	5	ESCAPE	1
5	6	F	33
6	7	F1	59
7	8	F2	60
8	9	F3	61
9	10	F4	62
0	11	F5	63
A	30	F6	64
ADD	78	F7	65
APOSTROPHE	40	F8	66
AT	145	F9	67
AX	150	F10	68
B	48	F11	87
BACK	14	F12	88
BACKSLASH	43	F13	100
BACKSPACE	14	F14	101
C	46	F15	102
CALCULATOR	161	G	34
CAPSLOCK	58	H	35
COLON	146	HOME	199
COMMA	51	I	23
D	32	INSERT	210
DECIMAL	83	J	36
DELETE	211	K	37
DIVIDE	181	L	38
DOWNARROW	208	LALT	56

LBRACKET	26	PAUSE	197
LCONTROL	29	PERIOD	52
LEFTARROW	203	PGDN	209
LSHIFT	42	PGUP	201
LWIN	219	Q	16
M	50	R	19
MINUS	12	RALT	184
MULTIPLY	55	RBRACKET	27
N	49	RCONTROL	157
NUMLOCK	69	RETURN	28
NUMPAD0	82	RIGHTARROW	205
NUMPAD1	79	RMENU	184
NUMPAD2	80	RSHIFT	54
NUMPAD3	81	RWIN	220
NUMPAD4	75	S	31
NUMPAD5	76	SCROLL	70
NUMPAD6	77	SEMICOLON	39
NUMPAD7	71	SLASH	53
NUMPAD8	72	SPACE	57
NUMPAD9	73	STOP	149
NUMPADCOMMA	179	T	20
NUMPADENTER	156	TAB	15
NUMPADEQUALS	141	U	22
NUMPADMINUS	74	UNDERLINE	147
NUMPADPERIOD	83	UPARROW	200
NUMPADPLUS	78	V	47
NUMPADSLASH	181	W	17
NUMPADSTAR	55	X	45
O	24	Y	21
P	25	Z	44

Keycode	Key		
2	1	20	T
3	2	21	Y
4	3	22	U
5	4	23	I
6	5	24	O
7	6	25	P
8	7	26	[
9	8	27	]
10	9	28	Enter
11	0	29	Left Ctrl
12	-	30	A
13	+	31	S
14	Backspace	32	D
15	Tab	33	F
16	Q	34	G
17	W	35	H
18	E	36	J
19	R	37	K

38	L	71	Numpad 7
39	;	72	Numpad 8
40	'	73	Numpad 9
41	`	74	Numpad -
42	Left Shift	75	Numpad 4
43	#	76	Numpad 5
44	Z	77	Numpad 6
45	X	78	Numpad +
46	C	79	Numpad 1
47	V	80	Numpad 2
48	B	81	Numpad 3
49	N	82	Numpad 0
50	M	83	Numpad .
51	,	86	\
52	.	87	F11
53	/	88	F12
54	Right Shift	157	Right Ctrl
55	Numpad *	181	Numpad /
56	Alt	183	Prt Scrn
57	Space	184	Alt Gr
58	Caps Lock	197	Pause
59	F1	199	Home
60	F2	200	Up
61	F3	201	Pg Up
62	F4	203	Left
63	F5	205	Right
64	F6	207	End
65	F7	208	Down
66	F8	209	Pg Dn
67	F9	210	Ins
68	F10	211	Del
69	Num Lock	219	Windows Key
70	Scr Lock	221	Menu Key

Appendix II – VP Units

[Article copied from vpforums.org]

1000 VP units = 21.25" or 539.75 mm
therefore...
1 vp unit = .53975 mm
47 VP units = 1 inch

Early Bally Wide body table playfield i.e. future spa is....
1167.2 VP units Wide x 1908.3 VP units long.
==========

EMs

Bally 70s EM - standard = 20.25" x 41.00" = 952 x 1927 vpus = 506 x 1024 bestpic size
G 70s EM - standard = 20.25" x 41.00" = 952 x 1927 vpus = 506 x 1024 bestpic size

Early SSs

*G System 1 - standard = 20.25" x 42.00" = 952 x 1974 vpus = 494 x 1024 bestpic size
Early SS Stern - widebody = 23.875" x 45.00" = 1122 x 2115 vpus = 543 x 1024 bestpic size
Zaccaria - standard SS - 20.25" x 42.00" = 952 x 1974 vpus = 494 x 1024 bestpic size
*Pre WMS Bally - standard = 20.25" x 42.00" = 952 x 1974 vpus = 494 x 1024 bestpic size
*Pre WMS Bally - widebody = 26.75" x 40.50" = 1257 x 1904 vpus = 676 x 1024 bestpic size
*WMS System 1-11 - standard = 20.25" x 42.00" = 952 x 1974 vpus = 494 x 1024 bestpic size
WMS System 1-11 - widebody = 27.00" x 42.00" = 1269 x 1974 vpus = 658 x 1024 bestpic size
*Atari - widebody = 27.00" x 45.00" = 1269 x 2115 vpus = 614 x 1024 bestpic size
*G System 80 - standard = 23.75" x 46.50" = 1116 x 2186 vpus = 524 x 1024 bestpic size
*G System 80 - widebody = 26.75" x 46.50" = 1258 x 2186 vpus = 590 x 1024 bestpic size

Modern SSs

Capcom tables = 20.25" x 46.00" = 952 x 2162 vpus = 451 x 1024 bestpic size
*Data East - standard = 20.25" x 46.00" = 952 x 2162 vpus = 451 x 1024 bestpic size
Data East - widebody = 25.00" x 51.75" = 1175 x 2432 vpus = 494 x 1024 bestpic size
*Safecracker = 16.50" x 41.50" = 776 x 1950 vpus = 408 x 1024 bestpic size
*WPC (through 1987) - standard = 20.50" x 42.00" = 964 x 1974 vpus = 500 x 1024 bestpic size
*WPC (1987 on) - standard = 20.50" x 46.00" = 964 x 2162 vpus = 457 x 1024 bestpic size
**Apparently at least one of these is not like the others.. I measured the Drac PF and it was/is 20.25" x 45.00" = 952 x 2115 vpus = 461 x 1024 bestpic size
*WPC - superpin = 23.25" x 46.00" = 1093 x 2162 vpus = 518 x 1024 bestpic size
*Pin2K tables = 20.50" x 43.00" = 964 x 2021 vpus = 488 x 1024 bestpic size

---
From WMS site
"Williams' standard playfield length increased from 42" to 46" in 1988"
==========
just remember the ball=50 units=1.0625"=.54054mm and most any dimension should come out fine
==========
Williams Varkon (1982)
24" across x 21" deep!
(Yes, it is wider than it is long. Plays like a cocktail table.)
==========
According to the press info and documentation, the Hercules playfield took up 18 square feet, or 36 by 72 inches.
Actual cabinet size was 39" by 93", and 83" tall.
==========
A few examples from Plumb's size conversion sheet:

Pre WMS Standard size Bally: 952 x 1974
Standard Williams through 1987: 964 x 1974
Standard Williams after 1987: 964 x 2162
Super Wide WPC: 1093 x 2162
==========

APPENDIX III – Object Properties, Events & Methods

BALL

Properties

BackDecal (Editor)/Script As String) in editor only the default ball can be changed
Color (Script As OLE_Color)
FrontDecal (Editor/Script As String) in editor only the default ball can be changed
Image (Editor/Script As String) in editor only the default ball can be changed
UserValue (Script As Variant)
VelX (Script As Double)
VelY (Script As Double)
VelZ (Script As Double)
X (Script As Double)
Y (Script As Double)
Z (Script As Double)

BUMPER

Properties

BlinkInterval (Editor/Script as Integer)
BlinkPattern (Editor/Script as String) "0"=off, "1"=on
CastsShadow (Editor As Boolean)
Center X (Editor as Single)
Center Y (Editor as Single)
Color (Editor As OLE_Color)
Disabled (Script As Boolean)
FlashWhenHit (Editor/Script as Boolean)
Force (Editor/Script As Single)
Image (Editor As String)
Overhang (Editor as Integer)
Radius (Editor as Integer)
SideColor (Editor as OLE_Color)
SideVisible (Editor as Boolean)
State (Editor/Script as LightState) 0=off, 1=on, 2=blink
Surface(Editor As String)
Threshold (Editor/Script As Single)

TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UserValue (Editor/Script as Variable)
Visible (Editor as Boolean)

Events

Sub _Hit()(Script) when Bumper is hit, procedures are carried out
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

COLLECTION

Properties

_NewEnum (Script As UNKNOWN)
Count (Script As Integer)
Item (Script As IDispatch)

Events

Sub _Hit(idx)(Script: idx As Integer) when collection item is hit, procedures are carried out
Sub _Init(idx)(Script: idx As Integer) when table is loaded, procedures are carried out
Sub _Slingshot(idx) (Script: idx As Integer) when slingshot is hit, procedures are carried out
Sub _Spin(idx) (Script: idx As Integer) when spinner spins, procedures are carried out
Sub _Timer(idx) (Script: idx As Integer) time interval will cycle, procedures are carried out
Sub _Unhit(idx) (Script: idx As Integer) when collection item is no longer hit, procedures are carried out

DECAL (this is an interface, a decal has no name, therefore it can't be accessed in the script)

Properties

Font (Editor As IFontDisp)
FontColor (Editor As OLE_Color)
HasVerticalText (Editor As Boolean)
Height (Editor As Single)
Image (Editor As String)
Rotation (Editor As Single)
SizingType (Editor As SizingType) 0=AutoSize, 1=AutoWidth, 2=ManualSize
Surface (Editor As String)
Text (Editor As String)
Type (Editor As DecalType) 0=DecalText, 1=DecalImage
Width (Editor As Single)
X (Editor As Single)
Y (Editor As Single)

DRAGPOINT (CONTROL POINT)

Properties

IsAutoTextureCoordinate (Editor As Boolean)
Smooth (Editor As Boolean)
TextureCoordinateU (Editor As Single)
X (Editor As Single)
Y (Editor As Single)

EMREEL (DISPREEL)

Properties

BackColor (Editor/Script As OLE_Color)
Font (Editor As IFontDisp)
FontColor (Editor As OLE_Color)
Height (Editor As Single)
Image (Editor As String)
ImagesPerRow (Editor As Integer)
IsShading (Editor As Boolean)
IsTransparent (Editor/Script As Boolean)
Name (Editor As String)
Range (Editor As Single)
ReelColor (Editor As OLE_Color)
Reels (Editor As Single)
Sound (Editor/Script As String)
Spacing (Editor As Single)
Steps (Editor/Script As Single)
TimerEnabled (Editor/Script As Boolean)
TimerInterval (Editor/Script As Boolean)
Type (Editor As ReelType (0=Text, 1=Image))
UpdateInterval (Editor/Script As Integer)
UseImageGrid (Editor As Boolean)
UserValue (Editor/Script As Variant)
Width (Editor As Single)
X (Editor As Single)
Y (Editor As Single)

Methods

AddValue (Script: Value As Integer)
ResetToZero (Script)
SetValue (Script: Value As Integer)
SpinReel (Script: ReelNumber As Integer, PulseCount As Integer)

Events

Sub _Init() (Script) procedures are carried out when table is loaded
Sub _Timer() (Script) procedures are carried out after timer interval

FLIPPER

Properties

BaseRadius (Editor as Single)
Color (Editor as OLE_Color)
CurrentAngle (Script (read-only!) As Single)
Elasticity (Editor/Script as Single)
EndAngle (Editor as Single)
EndRadius (Editor as Single)
FlipperRadiusMin(Script As Single)
Height (Editor as Single)
Length (Editor as Single)
MaxDifficultyLength (Editor/Script as Single)
Name (Editor as String)

ObliqueCorrection (Editor as Single)
PowerLaw (Editor as Single)
Recoil (Editor as Single)
Return (Editor as Single)
RubberColor (Editor as OLE_Color)
RubberHeight(Editor as Single)
RubberWidth (Editor as Integer)
ScatterAngle (Editor as Single)
Speed (Editor/Script as Single)
StartAngle (Editor as Single)
Strength (Editor/Script as Single)
Surface (Editor as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UserValue (Editor/Script as Variable)
Visible (Script as Boolean)
X (Editor as Single)
Y (Editor as Single)

Methods

RotatetoEnd (Script) rotates Flipper to flip position
RotatetoStart (Script) rotates Flipper to Start position

Events

Sub _Collide()(Script) when ball collides with flipper, procedures are carried out
Sub _Hit()(Script) when Flipper is hit, procedures are carried out
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

GATE

Properties

Animations (Editor As Integer)
CloseAngle (Editor as Single)
Collidable (Editor/Script as Boolean)
Color (Editor As OLE_Color)
Elasticity (Editor as Single)
Friction (Editor as Single)
Height (Editor As Single)
ImageBack (Editor As String)
ImageFront (Editor As String)
Length (Editor As Single)
Name (Editor As String)
Open (Script As Boolean)
OpenAngle (Editor as Single)
Rotation(Editor As Single)
Supports (Editor as Boolean)
Surface(Editor As String)
TimerEnabled (Script As Boolean)
TimerInterval (Script AS Integer)
UserValue(Editor As String)
Visible (Editor As Boolean)
X (Editor As Single)

Y (Editor As Single)

Events

Sub _Hit() (Script) when Gate is hit, procedures are carried out
Sub _Init() (Script) procedures are carried out when table is loaded
Sub _Timer() (Script) procedures are carried out after timer interval

ITABLE (another interface, use the following properties without any object name – applies Globally across the script)

Properties

ActiveBall (Script as Object)
AddCreditKey (Editor/Script As Integer)
AddCreditKey2 (Editor/Script As Integer)
CenterTiltKey (Editor/Script As Integer)
ExitGame (Editor/Script As Integer)
GameTime (Script As Integer) elapsed time in milliseconds
GetPlayerHWnd (Script As Integer) window handle of main table player window
LeftFlipperKey (Editor/Script As Integer)
LeftMagnaSave (Editor/Script As Integer)
LeftTiltKey (Editor/Script As Integer)
MechanicalTilt (Editor/Script As Integer)
PlungerKey (Editor/Script As Integer)
RightFlipperKey (Editor/Script As Integer)
RightMagnaSave (Editor/Script As Integer)
RightTiltKey (Editor/Script As Integer)
StartGameKey (Editor/Script As Integer)
UserDirectory (Script As String)
VPBuildVersion (Script: Read-only As Single)

Methods

BeginModal (Script)
EndModal (Script)
EndMusic (Script: Not Filename Specific)
FireKnocker (Script: Count As Integer)
GetTextFile (Script: FileName As String)
LoadValue (Script: TableName As String, ValueName As String) loads from VPReg.stg
Nudge (Script: Angle As Single, Force As Single)
PlayMusic (Script: String As String.mp3)
PlaySound (Script: Sound As String (wav or wma), [LoopCount As Integer], [Volume As Single])
QuitPlayer (Script: CloseType As Integer)
SaveValue (Script: TableName As String, ValueName As String, Value As Variant) saves to VPReg.stg
StartShake (Script)
StopShake (Script)
StopSound (Script: Sound As String (wav or wma), Filename Specific)

Events

Sub _Exit() (Script) when table exits, procedures are carried out
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _KeyDown (Script) keycode As Integer: when a key is pressed, procedures are carried out

Sub _KeyUp (Script) keycode As Integer: when a key is released, procedures are carried out
Sub _MusicDone()(Script) when mp3 file is finished, procedures are carried out
Sub _Paused()(Script) when table is paused, procedures are carried out
Sub _UnPaused()(Script) when table is resumed, procedures are carried out

KICKER

Properties

Color (Editor/Script As OLE_Color)
DrawStyle (Editor As KickerStyle) 0=KickerHidden, 1=KickerHole, 2=KickerCup
Enabled (Editor/Script as Boolean)
HitHeight (Editor as Single)
Name (Editor as String)
Scatter (Editor as Single)
Surface (Editor as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UserValue (Editor as Variable)
X (Editor As Single)
Y (Editor As Single)

Methods

BallCntOver (Script)
CreateBall (Script)
CreateBall.Color (Script as OLE_Color)
CreateBall.Image (Script as String)
DestroyBall (Script)
Kick (Script: Angle As Single, Speed As Single, [Inclination As Single])

Events

Sub _Hit()(Script) when Kicker is hit, procedures are carried out
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

LIGHT

Properties

BlinkInterval (Editor/Script as Integer)
BlinkPattern (Editor/Script as string) "0"=off, "1"=on
BorderColor (Editor As OLE_Color)
BorderWidth (Editor As OLE_Color)
Color (Editor As OLE_Color)
Name (Editor As String)
OffImage (Editor As String)
OnImage (Editor As String)
DisplayImage (Editor As Boolean)
Radius (Editor As Single)
Shape (Editor As ShapeType) 0=ShapeCircle, 1=ShapeCustom
State (Editor/Script as LightState) 0=off, 1=on, 2=blink
Surface (Editor as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UserValue (Editor as Variable)

X (Editor as Integer)
Y (Editor as Integer)
Events
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

LIGHTSEQ
Properties
CenterX (Editor as Single)
CenterY (Editor as Single)
Collection (Editor as String)
Name (Editor as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UpdateInterval (Editor/Script as Integer)
UserValue (Editor/Script as Variable)
Methods
Play (Script: Animation As SequencerState, [TailLength As Integer], [Repeat As Integer], [Pause As Integer])
StopPlay (Script)
Events
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Playdone() (Script) procedures are carried out when Light Sequence Animation ends
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

PLUNGER
Properties
AutoPlunger (Editor As Boolean)
BreakOverVelocity (Editor As Boolean)
FireSpeed (Editor/Script As Single)
MechPlunger (Editor As Boolean)
MechStrength (Editor As Single)
Name (Editor as String)
ParkPosition (Editor As Single)
PullSpeed (Editor/Script As Single)
ScatterVelocity (Editor As Single)
Stroke (Editor as Single)
Surface (Editor as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UserValue (Editor as Variable)
Visible (Editor as Boolean)
X (Editor as Single)
Y (Editor as Single)
Methods
CreateBall (Script)
Fire (Script)
MotionDevice (Script)

Position (Script)
PullBack (Script)

Events

Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

RAMP

Properties

Acrylic (Editor as Boolean)
CastsShadow (Editor As Boolean)
Collidable (Editor/Script As Boolean)
Color (Editor as OLE_Color)
Elasticity (Editor As Single)
Friction (Editor As Single)
HasWallImage (Editor as Boolean)
HeightBottom (Editor As Single)
HeightTop (Editor as Single)
Image (Editor as String)
ImageAlignment (Editor as RampImageAlignment) 0=world, 1=wrap
IsVisible (Editor As Boolean)
LeftWallHeight (Editor As Single)
Name (Editor as String)
RightWallHeight (Editor As Single)
Scatter (Editor As String)
Type (Editor as RampType) 0=Flat, 1=2-wire, 2=4-wire, 3=3-wire left, 4=3-wire right
UserValue (Editor as Variable)
VisibleLeftWallHeight (Editor As Single)
VisibleRightWallHeight (Editor As Single)
WallLeft (Editor As Single)
WallRight (Editor As Single)
WidthBottom (Editor as Single)
WidthTop (Editor as Single)

Events

Sub _Init()(Script) when table is loaded, procedures are carried out

SPINNER

Properties

Animations (Editor As Integer)
AngleMax (Editor As Single)
AngleMin (Editor As Single)
CastsShadow (Editor as Boolean)
Color (Editor as OLE_Color)
Elasticity (Editor As Single)
Friction (Editor/Script as Single)
Height (Editor as Single)
ImageBack (Editor as String)
ImageFront (Editor as String)
Length (Editor as Single)
Name (Editor as String)

Overhang (Editor As Single)
Rotation (Editor as Single)
Supports (Editor as Boolean)
Surface (Editor as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
Visible (Editor As Boolean)
X (Editor as Single)
Y (Editor as Single)

Events

Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Spin() (Script) when spinner spins, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

TABLE

Properties

AccelerManualAmp (Editor As Single)
Accelerometer (Editor As Boolean)
AccelerometerAngle (Editor As Single)
AccelerometerAmp (Editor As Single)
AccelNormalMount (Editor As Boolean)
AlternateRender (Editor As Boolean)
BackdropColor (Editor as OLE_Color)
BackdropImage (Editor as String)
BallBackDecal (Editor as String)
BallFontDecal (Editor as String)
BallImage (Editor as String)
DampeningFriction (Editor As Single)
DeadSlider (Editor As Integer)
DeadZone (Editor As Integer)
DisplayBackdrop (Editor as Boolean)
DisplayGrid (Editor as Boolean)
FieldOfView (Editor as Single)
GlassHeight (Editor as Single)
GlobalDifficulty (Editor As Single)
Gravity (Editor As Single)
GridSize (Editor as Single)
HardFriction (Editor As Single)
HardScatter (Editor As Single)
HardwareRender (Editor As Boolean)
Height (Editor as Single)
Image (Editor as String)
Inclination (Editor as Single)
JoltAmount (Editor As Single)
JoltTriggerTime (Editor As Single)
MaxBallSpeed (Editor As Single)
Name (Editor as String)
PhysicsLooptime (Editor As Single)
PlayfieldColor (Editor as OLE_Color)

PlungerFliter (Editor As Boolean)
PlungerNormalize (Editor As Single)
RenderDecals (Editor As Boolean)
RenderEMReels (Editor As Boolean)
RenderShadows (Editor as Boolean)
Rotation (Editor As Single)
ScaleX (Editor As Single)
ScaleY (Editor As Single)
SlopeMax (Editor As Single)
SlopeMin (Editor As Single)
TableCaching (Editor as Boolean)
TableHeight (Editor As Single)
TiltAmount (Editor As Single)
TiltTriggerTime (Editor As Single)
Width (Editor as Single)
UseD3DBlit (Editor As Boolean)
Xlatex (Editor As Single)
Xlatey (Editor As Single)
YieldTime (Script as Variable)

Methods

Nudge (Script: Angle As Single, Force As Single)
StartShake (Script)
StopShake (Script)

Events

Sub _Exit() (Script) when table exits, procedures are carried out
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _KeyDown (Script) keycode As Integer: when a key is pressed, procedures are carried out
Sub _KeyUp (Script) keycode As Integer: when a key is released, procedures are carried out
Sub _MusicEnded()(Script) when music is ended, procedures are carried out
Sub _Paused()(Script) when table is paused, procedures are carried out
Sub _UnPaused()(Script) when table is resumed, procedures are carried out

TEXTBOX

Properties

Alignment (Editor/Script as TextAlignment) script must refresh Text property
BackColor (Editor/Script as OLE_Color) script must refresh Text property
Font (Editor as IFontDisp) select from Font settings
FontColor (Editor/Script OLE_Color) script must refresh Text property
Height (Editor as Single)
IsTransparent (Editor/Script Boolean) script must refresh Text property
Name (Editor as String)
Text (Editor/Script as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UserValue (Editor/Script as Variable)
Width (Editor as Single)
X (Editor as Single)
Y (Editor as Single)

Events
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

TIMER
Properties
Enabled (Editor/Script as Boolean)
Interval (Editor/Script as Integer)
Name (Editor as String)
UserValue (Editor/Script as Variant)
Events
Sub _Init() (Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out

TRIGGER
Properties
Enabled (Editor/Script as Boolean)
HitHeight (Editor As Single)
Name (Editor as String)
Radius (Editor as Single)
Shape (Editor as Shape)
Surface (Editor as String)
TimerEnabled (Editor/Script as Boolean)
TimerInterval (Editor/Script as Integer)
UserValue (Editor as Variable)
Visible (Editor as Boolean)
X (Editor as Single)
Y (Editor as Single)
Methods
BallCntOver (Script)
DestroyBall (Script)
Events
Sub _Hit() (Script) when trigger is hit, procedures are carried out
Sub _Init() (Script) when table is loaded, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out
Sub _Unhit() (Script) when trigger is no longer hit, procedures are carried out

WALL (TARGET)
Properties
CanDrop (Editor as Boolean)
CastsShadow (Editor as Boolean)
Collidable (Editor/Script as Boolean)
Disabled(Script as Boolean)
DisplayTexture (Editor as Boolean)
Elasticity (Editor as Single)
FaceColor (Editor as OLE_Color)
FlipbookAnimation (Editor As Boolean)
Friction (Editor As Single)
HasHitEvent (Editor as Boolean)

HeightBottom (Editor as Single)
HeightTop (Editor as Single)
Image (Editor as String)
IsDropped (Script as Boolean) "CanDrop" must be checked in Options
Name (Editor as String)
Scatter (Editor As Single)
SideColor (Editor as OLE_Color)
SideImage (Editor as String)
SideVisible (Editor as Boolean)
SlingshotAnimation (Editor As Boolean)
SlingshotColor (Editor as OLE_Color)
SlingshotStrength (Editor as Single)
SlingshotThreshold (Editor as Single)
Threshold (Editor as Single)
TimerEnabled (Script as Boolean)
TimerInterval (Script as Integer)
UserValue (Editor as Variable)
Visible (Editor as Boolean)

Events

Sub _Hit()(Script) when trigger is hit, procedures are carried out
Sub _Init()(Script) when table is loaded, procedures are carried out
Sub _Slingshot() (Script) when slingshot is hit, procedures are carried out
Sub _Timer() (Script) time interval will cycle, then procedures are carried out